30th Winter School, Srni 2010

An Introduction to DifferentialGeometry with Maple

Preliminary Remarks

History

I began working on computer software for differential geometry and its applications to mathematical physics and differential equations in1989. Initial implementations were done by students at Utah State University.

At that time there were many software packages available which could only be used to solve very specialized problems. In addition, it was practically impossible to pass results from one program to another.

Reliability and documentation was also a major problem.

The first publicly released version of the software was in 2001 -- called "Vessiot".

In 2005 Maple agreed to include my software as part of the Maple distributed library.

A massive amount of work was required to meet Maple standards for code, testing and documentation.

The first version of DG appeared in Maple, Release 11, 2007.

Goals

At the outset my goals were:

- to create a very flexible set of programs with which specialized applications can be easily constructed;
- to adopt the conventions and language of modern differential geometry;
- to create user-friendly input and output;
- to take advantage of Maple's superb ODE and PDE solvers;
- to provide detailed, mathematically correct, documentation;
- to provide introductory lessons and advanced special topic tutorials.

DifferentialGeometry In Maple

The *DG* package in Maple consists of the following components:

- Calculus on Manifold: Vector fields and differential forms, transformations;

- Tensors: Tensors, connections, curvature, spinors, NP-formalism;

- LieAlgebras: Basic operations with Lie algebras, Lie algebra cohomology, representations, structure theory;

- GroupActions: Lie groups, symmetries, invariant geometric objects, moving frames;

- JetCalculus: Jet spaces, variational bicomplexes, calculus of variations;

- Library: Tables of Lie algebras, differential equations, solutions of the Einstein equations;

We shall demonstrate some of the capabilities of the DG software by performing some basic computations with Lie groups and homogeneous spaces.

A Short Review of Homogeneous Spaces

1. A Lie group G is a group which is also a manifold and for which the operation of multiplication is smooth.

2. Let G be a Lie group and M a manifold. An action of G on M is a mapping

$$\mu : G \times M \to M$$

such that

[i] $\mu(e, x) = x$ or $e * x = x$, $x \in M$

[ii] $\mu(g, \mu(h, x)) = \mu(g \cdot h, x)$ or $g * (h * x) = (g \cdot h) * x$, $g, h \in G, x \in M$

3. Let G act on M. The isotropy sub-group at the point $p \in M$ is

$$G_p = \{g \in G \mid g \cdot p = p\} .$$

4. Let G act on M. Then M is called a G-homogeneous space if G acts transitively on M, that is, for any two points $p, q \in M$, there is a $g \in G$ such that $g * p = q$.

5. Let H be a closed sub-group of a Lie group G. Then the coset space $G / H = \{gH \mid g \in G\}$ can be given a smooth manifold structure such that:
- the canonical projection map $\pi : G \to G / H$ is smooth. and
- smooth local cross-sections $\sigma : G / H \to G$ can be defined in the neighorhood of each point.

6. There is a natural smooth action of G on G / H

7. If M is a G-homogeneous space, then the map $G \to M, g \to g * p$ induces a equivariant diffeomorphism $G / H \to M$.

8. H invariant quadratic forms on TG drop to G invariant quadratic forms on $T(G / H)$.

Worksheet Overview

In this worksheet we shall demonstrate the capabilities of DG software package with the following compuations.

- For a given Lie algebra g and subalgebra h, we construct the associated Lie group G and the homogeneous space $\pi : G \rightarrow G / H$.
- We calculate the h invariant quadratic forms on g. We find the corresponding invariant symmetric tensors on G and push these down to invariant symmetric tensors on $M = G / H$.
- We show that the invariant metrics on M so constructed are solutions to the Einstein equations.
- We identify the invariant metric in the mathematical physics literature.

These computations illustrute some of the commands in the following packages.

```
> restart:with(DifferentialGeometry):
> with(LieAlgebras):
> with(Tensor):
> with(GroupActions):
> with(Library):
```

Part A. Algebraic Steps

Our goal is to construct a 4 dimensional homogeneous space for a 5-dimensional Lie group.

We begin by looking at the 5 dimensional Lie algebras available for our use.

All 5-dimensional Lie algebras have been classified and the results of these classifications are contained in the DifferentialGeometry Library.

The References command gives us a list of the articles and books whose results are in the DifferentialGeometry Library.

```
> References(verbose);

Doubrov, 1
           Classification of Subalgebras in the Exceptional
Lie Algebra of Type G_2
           Proc. of the Natl. Academy of Sciences of Belarus,
Ser. Phys.-Math. Sci., 2008, No.3



Gong, 1
           Classification of Nilpotent Lie Algebras of
Dimension 7( Over Algebraically Closed Fields and R)
```

PhD. Thesis, University of Waterloo (1998)

Gonzalez-Lopez, 1
Lie algebras of vector fields in the real plane (with Kamran and Olver)
Proc. London Math Soc. Vol 64 (1992), 339--368

Kamke, 1
Differentialgleichungen
Chelsa Publ. Co. (1947)

Mubarakzyanov, 1
Lie algebras of dimmensions 3, 4
Izv. Vyssh. Uchebn. Zaved. Math 34(1963) 99

Mubarakzyanov, 2
Lie algebras of dimension 5
Izv. Vyssh. Uchebn. Zaved. Math 34(1963) 99

Mubarakzyanov, 3
Lie algebras of dimension 6
Izv. Vyssh. Uchebn. Zaved. Math 35(1963) 104

Olver, 1:
Equivalence, Invariants and Symmetry, 472--473

Petrov, 1:
Einstein Spaces

Stephani, 1:
Exact Solutions to Einstein's Field Equations, 2nd Edition (with Kramer, Maccallum, Hoenselaers, Herlt)

Turkowski, 1:
Low dimensional real Lie algebras
JMP(29), 1990, 2139--2144

Turkowski, 2
Solvable Lie Algebras of dimension six
JMP(31), 1990, 1344--1350

```
Winternitz, 1:
           Invariants of real low dimensional Lie algebras,
(with Patera, Sharp and Zassenhaus)
           JMP vol 17, No 6, June 1976, 966--994
```

```
[["Doubrov", 1], ["Gong", 1], ["Gonzalez-Lopez", 1], ["Kamke", 1], ["Morozov", 1],        (5.1)
   ["Mubarakyzanov", 1], ["Mubarakyzanov", 2], ["Mubarakyzanov", 3], ["Olver", 1],
   ["Petrov", 1], ["Stephani", 1], ["Turkowski", 1], ["Turkowski", 2], ["USU", 2],
   ["USU", "2D"], ["Winternitz", 1]]
```

The paper by Winternitz *Invariants of real low dimensional algebras* contains a convenient list of all Lie algebras of dimension <= 5 which we will shall use here.

The indices by which these Lie algebras are labeled in the paper can be obtained using the Browse command.

```
> Browse("Winternitz", 1);
[[3, 0], [3, 1], [3, 2], [3, 3], [3, 4], [3, 5], [3, 6], [3, 7], [3, 8], [3, 9], [4, 0], [4, 1], [4,        (5.2)
   2], [4, 3], [4, 4], [4, 5], [4, 6], [4, 7], [4, 8], [4, 9], [4, 10], [4, 11], [4, 12], [5, 0],
   [5, 1], [5, 2], [5, 3], [5, 4], [5, 5], [5, 6], [5, 7], [5, 8], [5, 9], [5, 10], [5, 11], [5,
   12], [5, 13], [5, 14], [5, 15], [5, 16], [5, 17], [5, 18], [5, 19], [5, 20], [5, 21], [5,
   22], [5, 23], [5, 24], [5, 25], [5, 26], [5, 27], [5, 28], [5, 29], [5, 30], [5, 31], [5,
   32], [5, 33], [5, 34], [5, 35], [5, 36], [5, 37], [5, 38], [5, 39], [5, 40], [6, 1], [6, 2],
   [6, 3], [6, 4], [6, 5], [6, 6], [6, 7], [6, 8], [6, 9], [6, 10], [6, 11], [6, 12], [6, 13], [6,
   14], [6, 15], [6, 16], [6, 17], [6, 18], [6, 19], [6, 20], [6, 21], [6, 22]]
```

Let us look at some of these Lie algebras in more detail:

```
> Browse("Winternitz", 1,  [[5, 28], [5, 29], [5, 30], [5, 31]]
  );
```

"Winternitz", 1, [5, 28]

$$[[e1, e5] = a\, e1, [e2, e3] = e1, [e2, e5] = (a-1)\, e2, [e3, e5] = e3 + e4, [e4, e5] = e4]$$

"Winternitz", 1, [5, 29]

$$[[e1, e5] = e1, [e2, e4] = e1, [e2, e5] = e2, [e4, e5] = e3]$$

"Winternitz", 1, [5, 30]

$$[[e1, e5] = (a+1)\, e1, [e2, e4] = e1, [e2, e5] = a\, e2, [e3, e4] = e2, [e3, e5] = (a-1)\, e3, [e4, e5] = e4]$$

"Winternitz", 1, [5, 31]

$$[[e1, e5] = 3\, e1, [e2, e4] = e1, [e2, e5] = 2\, e2, [e3, e4] = e2, [e3, e5] = e3, [e4, e5] = e3 + e4]$$

(5.3)

We choose to work with the Lie algebra [5, 30].

We retrieve the Lie algebra structure equations (with the command Retrieve) for this algebra and pass these structure equations to the DGsetup program to initialize the Lie algebra. This Lie algebra has a parameter *a*. For simplicity, we shall set the parameter *a* = 2.

```
> L, P := Retrieve("Winternitz", 1, [5, 30], Alg, parameters =
  "yes");
```

$$L, P := [[e1, e5] = (a+1)\ e1, [e2, e4] = e1, [e2, e5] = a\ e2, [e3, e4] = e2, [e3, e5] = (a-1)\ e3, [e4, e5] = e4], [a] \quad \textbf{(5.4)}$$

```
Alg > L := eval(L, P[1] = 2);
```

$$L := [[e1, e5] = 3\ e1, [e2, e4] = e1, [e2, e5] = 2\ e2, [e3, e4] = e2, [e3, e5] = e3, [e4, e5] = e4] \quad \textbf{(5.5)}$$

All calculation with the DG software begin with the DGsetup command. This is used to initialize manifolds, Lie algebras, representations, etc.

```
Alg > DGsetup(L, verbose);
```

The following vector fields have been defined and protected:

$$[e1, e2, e3, e4, e5]$$

The following differential 1-forms have been defined and protected:

$$[\theta 1, \theta 2, \theta 3, \theta 4, \theta 5]$$

Lie algebra: Alg **(5.6)**

```
Alg > MultiplicationTable("LieTable");
```

```
        `|`   e1     e2     e3    e4    e5
        ---   ----   ----   ----  ----  ----
   e1   `|`   0      0      0     0     3 e1
   e2   `|`   0      0      0     e1    2 e2      (5.7)
   e3   `|`   0      0      0     e2    e3
   e4   `|`   0      -e1    -e2   0     e4
   e5   `|`   -3 e1  -2 e2  -e3   -e4   0
```

Once the algebra is initialized, we can do all sorts of calculations and tests. For example, the Query command can be used to check many properties of Lie algebras

```
Alg > Query("Solvable");
```

$$true \quad \textbf{(5.8)}$$

```
Alg > Query([e1], "Ideal");
```

$$true \quad \textbf{(5.9)}$$

Now in order to construct our 4 dimensional homogeneous space we need to choose a 1-

dimensional subalgebra -- it will be the isotropy subalgebra at the preferred point.

The inequivalent 1 dim. subspaces of g are:

$$\langle e_1\rangle, \langle e_2\rangle, \langle e_3 + r{\cdot}e_1\rangle, \langle e_4\rangle, \langle e_5\rangle.$$

The subalgebra $\langle e_1\rangle$ is an ideal and will not lead to an effective action of G on G/H.

We want our homogeneous space to admit a G invariant metric so let's calculate the h invariant quadratics forms on g.

Only $h = \langle e_4\rangle$ works -- that is, only the vector space of e_4 invariant quadratics forms contains nondegenerate elements.

```
Alg > Qalg := evalDG([theta2 &t theta2 - theta1 &s theta3,
      theta3 &t theta3, theta3 &s theta5, theta5 &t theta5])
      ;
```

$$Qalg := [-\theta 1\ \theta 3 + \theta 2\ \theta 2 - \theta 3\ \theta 1, \theta 3\ \theta 3, \theta 3\ \theta 5 + \theta 5\ \theta 3, \theta 5\ \theta 5] \quad \textbf{(5.10)}$$

```
Alg > nops(Qalg);
```

$$4 \quad \textbf{(5.11)}$$

```
Alg > LieDerivative(e4, Qalg);
```

$$[0\ \theta 1\ \theta 1, 0\ \theta 1\ \theta 1, 0\ \theta 1\ \theta 1, 0\ \theta 1\ \theta 1] \quad \textbf{(5.12)}$$

The details are in the following subsection.

h Invariant Quadratic forms

The command InvariantGeometricObjectFields is a powerful geneal purpose command for calculating all kinds of invariant tensors, connections, etc.

Here we are going to use it to find h invariant quadratic forms.

Case 1: $h = e_2$: Only degenerate forms

```
Alg > Q2 := GenerateSymmetricTensors([theta1, theta3,
      theta4, theta5], 2);
```

$$Q2 := \left[\theta 1\ \theta 1, \frac{1}{2}\ \theta 1\ \theta 3 + \frac{1}{2}\ \theta 3\ \theta 1, \frac{1}{2}\ \theta 1\ \theta 4 + \frac{1}{2}\ \theta 4\ \theta 1, \frac{1}{2}\ \theta 1\ \theta 5 + \frac{1}{2}\ \theta 5\ \theta 1, \theta 3\ \theta 3, \frac{1}{2}\ \theta 3\ \theta 4 + \frac{1}{2}\ \theta 4\ \theta 3, \frac{1}{2}\ \theta 3\ \theta 5 + \frac{1}{2}\ \theta 5\ \theta 3, \theta 4\ \theta 4, \frac{1}{2}\ \theta 4\ \theta 5 + \frac{1}{2}\ \theta 5\ \theta 4, \theta 5\ \theta 5\right] \quad \textbf{(5.1.1)}$$

```
Alg > convert(InvariantGeometricObjectFields([e2], Q2),
      DGArray);
```

$$\begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \_A5 & \frac{1}{2}\,\_A6 & \frac{1}{2}\,\_A7 \\ 0 & 0 & \frac{1}{2}\,\_A6 & \_A8 & \frac{1}{2}\,\_A9 \\ 0 & 0 & \frac{1}{2}\,\_A7 & \frac{1}{2}\,\_A9 & \_A10 \end{bmatrix} \tag{5.1.2}$$

Case 2: $h = e_3$ [Only degenerate forms]

```
Alg > Q3 := GenerateSymmetricTensors([theta1, theta2,
      theta4, theta5], 2):
Alg > convert(InvariantGeometricObjectFields([e3], Q3),
      DGArray);
```

$$\begin{bmatrix} \_A1 & 0 & 0 & \frac{1}{2}\,\_A3 & \frac{1}{2}\,\_A4 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ \frac{1}{2}\,\_A3 & 0 & 0 & \_A8 & \frac{1}{2}\,\_A9 \\ \frac{1}{2}\,\_A4 & 0 & 0 & \frac{1}{2}\,\_A9 & \_A10 \end{bmatrix} \tag{5.1.3}$$

Case 3: $h = e_4$

```
Alg > S4 := GenerateSymmetricTensors([theta1, theta2,
      theta3, theta5], 2):
Alg > convert(InvariantGeometricObjectFields([e4], S4),
      DGArray);
```

$$\begin{bmatrix} 0 & 0 & -\_A5 & 0 & 0 \\ 0 & \_A5 & 0 & 0 & 0 \\ -\_A5 & 0 & \_A8 & 0 & \frac{1}{2}\,\_A9 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{1}{2}\,\_A9 & 0 & \_A10 \end{bmatrix} \tag{5.1.4}$$

Case 4: $h = e_5$ [Only degenerate forms]:

```
Alg > Q5 := GenerateSymmetricTensors([theta1, theta2,
       theta3, theta4], 2):
Alg > convert(InvariantGeometricObjectFields([e5], Q5),
       DGArray);
```

$$\begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \qquad \mathbf{(5.1.5)}$$

Part B. Constructing the homogeneous space

Next we construct the Lie group G for our Lie algebra Alg.

First we define local coordinates for the group G.

The command LieGroup in the GroupAction package implements Lie's 2nd and 3rd theorems (so far for solvable groups) and directly constructs a global Lie group whose Lie algebra is the given algebra Alg.

The LieGroup command results a Maple structure called a module. The module LG has various exports which provide the information about the Lie group.

```
Alg > DGsetup([x1, x2, x3, x4, x5], G);
```

frame name: G **(6.1)**

```
G > LG := LieGroup(Alg, G);
```

LG := **module**( ) **(6.2)**

  local *i*, *j*;

  export *Frame*, *Identity*, *LeftMultiplication*, *RightMultiplication*, *Inverse*;

end module

Here is the explicit formula for left multiplication in G of the group element with coordinates [x1, x2, x3, x4, x5] by the group element with coordinates [a1, a2, a3, a4, a5].

```
G > dotLeft := LG:-LeftMultiplication([a1, a2, a3, a4, a5]);
```

$$dotLeft := \Big[x1 = a1 - \frac{1}{3}\, a2\, x4\, e^{a5} - \frac{1}{6}\, a3\, a4\, x4\, e^{a5} + \frac{1}{6}\, a4^{2}\, x3\, e^{a5} + \frac{2}{3}\, a4\, x2\, e^{2\, a5} + x1\, e^{3\, a5}, x2 = a2 - \frac{1}{2}\, a3\, x4\, e^{a5} + \frac{1}{2}\, a4\, x3\, e^{a5} + x2\, e^{2\, a5}, x3 = a3 + x3\, e^{a5}, x4 = a4 + x4\, e^{a5}, x5 = a5 + x5\Big] \qquad \mathbf{(6.3)}$$

Let's look at the left and right invariant vector fields and forms for this Lie group. These are calculated with the InvariantVectorsAndForms command.

```
G > XL, OmegaL, XR, OmegaR := InvariantVectorsAndForms(LG):
```

Here are the right invariant vector fields.

```
G > XR;
```

$$\left[D\_x1, -\frac{1}{3}\, x4\, D\_x1 + D\_x2, -\frac{1}{2}\, x4\, D\_x2 + D\_x3, \frac{2}{3}\, x2\, D\_x1 + \frac{1}{2}\, x3\, D\_x2 + D\_x4, 3\, x1\, D\_x1 + 2\, x2\, D\_x2 + x3\, D\_x3 + x4\, D\_x4 + D\_x5\right] \tag{6.4}$$

Here are the left invariant vector fields.

```
G > XL;
```

$$\left[e^{3\,x5}\, D\_x1, \frac{2}{3}\, x4\, e^{2\,x5}\, D\_x1 + e^{2\,x5}\, D\_x2, \frac{1}{6}\, x4^2\, e^{x5}\, D\_x1 + \frac{1}{2}\, x4\, e^{x5}\, D\_x2 + e^{x5}\, D\_x3, -\frac{1}{6}\, (x3\, x4 + 2\, x2)\, e^{x5}\, D\_x1 - \frac{1}{2}\, x3\, e^{x5}\, D\_x2 + e^{x5}\, D\_x4, D\_x5\right] \tag{6.5}$$

The command LieAlgebraData computes the structure equations for the right invariant vector fields XR. These structure equations coincide with the structure equations *L* for the Lie algebra which we started with.

```
G > LieAlgebraData(XR);
```

$$[[e1, e5] = 3\, e1, [e2, e4] = e1, [e2, e5] = 2\, e2, [e3, e4] = e2, [e3, e5] = e3, [e4, e5] = e4] \tag{6.6}$$

```
G > L;
```

$$[[e1, e5] = 3\, e1, [e2, e4] = e1, [e2, e5] = 2\, e2, [e3, e4] = e2, [e3, e5] = e3, [e4, e5] = e4] \tag{6.7}$$

The structure equations for the left invariant vector fields XR differ by just a sign. Because the coefficients of the vector fields XR contain transcendental functions we use an alternative calling sequence for the LieAlgebraData program.

```
G > LieAlgebraData(XL,initialpointlist = [[x1=0,x2=1,x3=0, x4
    =0, x5=0]]);
```

$$[[e1, e5] = -3\, e1, [e2, e4] = -e1, [e2, e5] = -2\, e2, [e3, e4] = -e2, [e3, e5] = -e3, [e4, e5] = -e4] \tag{6.8}$$

The left and right invariant vector fields for any Lie group commute.

```
G > Matrix(4,4, (i,j)->LieBracket(XL[i], XR[j]));
```

$$\begin{bmatrix} 0\, D\_x1 & 0\, D\_x1 & 0\, D\_x1 & 0\, D\_x1 \\ 0\, D\_x1 & 0\, D\_x1 & 0\, D\_x1 & 0\, D\_x1 \\ 0\, D\_x1 & 0\, D\_x1 & 0\, D\_x1 & 0\, D\_x1 \\ 0\, D\_x1 & 0\, D\_x1 & 0\, D\_x1 & 0\, D\_x1 \end{bmatrix} \tag{6.9}$$

The quotient of G by the 1 dimensional subgroup H generated by the vector field XL[4] is a 4 dimensional manifold.

We call this manifold M and use coordinates [x, y, z, w] on M.

The key step now is to calculate the coordinate formula for the projection map $\pi : G \rightarrow G/H$

This projection map sends the group element g to the coset gH, that is, $\pi(g) = gH$.

Consequently, for any h in H we have that $\pi(gh) = ghH = gH = \pi(g)$ and therefore the projection map π is invariant with respect to the right action of H on G.

The infinitesimal generation of this right action is the left invariant vector field Z = XL[4]. Thus, if we set

$$\pi(x_1, x_2\,, x_3, x_4, x_5) = \left[F_1(x_1, x_2\,, x_3, x_4, x_5), ..\, F_4(x_1, x_2\,, x_3, x_4, x_5)\right]$$

it follows that component functions $F_1, F_2\,,...\, F_4$ are all invariants of the vector field Z.

We can use the LieDerivative and pdsolve commands to find these invariants.

```
G > InvariantPDE := LieDerivative(XL[4], F(x1, x2, x3, x4, x5)
    );
```

$$InvariantPDE := -\frac{1}{6}\,(x3\,x4 + 2\,x2)\,\mathrm{e}^{x5}\left(\frac{\partial}{\partial x1}\,F(x1, x2, x3, x4, x5)\right) - \frac{1}{2}\,x3\,\mathrm{e}^{x5}\left(\frac{\partial}{\partial x2}\,F(x1, x2, x3, x4, x5)\right) + \mathrm{e}^{x5}\left(\frac{\partial}{\partial x4}\,F(x1, x2, x3, x4, x5)\right) \tag{6.10}$$

```
G > pdsolve(InvariantPDE);
```

$$F(x1, x2, x3, x4, x5) = \_F1\left(x3, \frac{x3\,x4 + 2\,x2}{x3}, x5, -\frac{1}{3}\,\frac{(x3\,x4 + 2\,x2)\,x2}{x3} + x1\right) \tag{6.11}$$

Now we define the manifold M and the projection map π. Mappings are constructed with the Transformation command.

```
G > DGsetup([x, y, z, w], M);
```

$$frame\ name: M \tag{6.12}$$

```
M > pi := Transformation(G, M,  [x = x3, y = x3*x4+2*x2, z =
    exp(x5), w = x4*x2+3*x1+(1/2)*x3*x4^2]);
```

$$\pi := \left[x = x3, y = x3\,x4 + 2\,x2, z = \mathrm{e}^{x5}, w = x4\,x2 + 3\,x1 + \frac{1}{2}\,x4^2\,x3\right] \tag{6.13}$$

Our next task to calculate the action of the Lie group G on the manifold M. For this we shall need a "cross-section" to the projection map π, that is, a map

$$\sigma : M \rightarrow G \ \text{ such that } \ \pi\,(\sigma(x)) = x$$

Such a map is easily calculated using the InverseTransformation command.

```
M > sigma :=  eval(InverseTransformation(pi), _C1 =0);
```

$$\sigma := \left[x1 = \frac{1}{3}\, w,\, x2 = \frac{1}{2}\, y,\, x3 = x,\, x4 = 0,\, x5 = \ln(z) \right] \qquad \textbf{(6.14)}$$

```
M > simplify(ComposeTransformations(pi, sigma));
```

$$[x = x,\, y = y,\, z = z,\, w = w] \qquad \textbf{(6.15)}$$

The action of G on M is now computed as the composition of the projection map π, the left multiplication map *dotLeft* of *G* to *G*, and the cross-section σ.

```
M > mu := map(simplify,ComposeTransformations( pi, dotLeft,
    sigma));
```

$$\mu := \left[x = a3 + x\, e^{a5},\, y = a3\, a4 + 2\, a4\, x\, e^{a5} + 2\, a2 + y\, e^{2\, a5},\, z = e^{a5}\, z,\, w = a4\, a2 + \frac{3}{2}\, a4^{2}\, x\, e^{a5} + \frac{3}{2}\, a4\, y\, e^{2\, a5} + 3\, a1 + w\, e^{3\, a5} + \frac{1}{2}\, a4^{2}\, a3 \right] \qquad \textbf{(6.16)}$$

This is our homogeneous space which we wanted to construct.

The infinitesimal generators Γ for the action of *G* on *M* are calculated using the InfinitesimalTransformation command.

```
M > Gamma := InfinitesimalTransformation(mu, [a1, a2, a3, a4,
    a5]);
```

$$\Gamma := \left[3\, D\_w,\, 2\, D\_y,\, D\_x,\, 2\, x\, D\_y + \frac{3}{2}\, y\, D\_w,\, x\, D\_x + 2\, y\, D\_y + z\, D\_z + 3\, w\, D\_w \right] \qquad \textbf{(6.17)}$$

Finally, we can check our answer. The structure equations for the Lie algebra of vectors fields Γ coincides with the Lie algebra we started with:

```
M > LieAlgebraData(Gamma);
```

$$[[e1, e5] = 3\, e1,\, [e2, e4] = e1,\, [e2, e5] = 2\, e2,\, [e3, e4] = e2,\, [e3, e5] = e3,\, [e4, e5] = e4] \qquad \textbf{(6.18)}$$

The isotropy subalgebra at the preferred point is exactly $\Gamma[4]$

```
M > pt := ApplyTransformation(pi, [x1 = 0, x2 = 0, x3 = 0,x4 =
    0, x5 =0]);
```

$$pt := [x = 0,\, y = 0,\, z = 1,\, w = 0] \qquad \textbf{(6.19)}$$

```
M > Iso := IsotropySubalgebra(Gamma, pt);
```

$$Iso := \left[2\, x\, D\_y + \frac{3}{2}\, y\, D\_w \right] \qquad \textbf{(6.20)}$$

```
M > Gamma[4];
```

$$2\, x\, D\_y + \frac{3}{2}\, y\, D\_w \qquad \textbf{(6.21)}$$

Part C. An invariant metric on the homogeneous space M

We return to the invariant symmetric rank 2 tensors on the Lie algebra which we calculated in Part A.

```
M > Qalg;
```

$$\left[-\theta1\ \theta3 + \theta2\ \theta2 - \theta3\ \theta1,\ \theta3\ \theta3,\ \theta3\ \theta5 + \theta5\ \theta3,\ \theta5\ \theta5 \right] \tag{7.1}$$

We express these tensors in terms of the left invariant Maurer-Cartan forms $[o_1, o_2, .. o_5]$ to arrive at symmetric tensors on the Lie group *G*.

```
Alg > o1, o2, o3, o4, o5 := op(OmegaL);
```

$$o1, o2, o3, o4, o5 := \mathrm{e}^{-3\,x5}\,dx1 - \frac{2}{3}\,\mathrm{e}^{-3\,x5}\,x4\,dx2 + \frac{1}{6}\,\mathrm{e}^{-3\,x5}\,x4^2\,dx3 + \frac{1}{6}\,\mathrm{e}^{-3\,x5}\,(2\,x2 - x3\,x4)\,dx4,\ \mathrm{e}^{-2\,x5}\,dx2 - \frac{1}{2}\,\mathrm{e}^{-2\,x5}\,x4\,dx3 + \frac{1}{2}\,\mathrm{e}^{-2\,x5}\,x3\,dx4,\ \mathrm{e}^{-x5}\,dx3,\ \mathrm{e}^{-x5}\,dx4,\ dx5 \tag{7.2}$$

```
G > Qgroup := evalDG([-o1 &s o3 + o2 &t o2, o3 &t o3, o3 &s
    o5, o5 &t o5]);
```

$$Qgroup := \left[-\mathrm{e}^{-4\,x5}\,dx1\,dx3 + \mathrm{e}^{-4\,x5}\,dx2\,dx2 + \frac{1}{6}\,x4\,\mathrm{e}^{-4\,x5}\,dx2\,dx3 + \frac{1}{2}\,\mathrm{e}^{-4\,x5}\,x3\,dx2\,dx4 - \mathrm{e}^{-4\,x5}\,dx3\,dx1 + \frac{1}{6}\,x4\,\mathrm{e}^{-4\,x5}\,dx3\,dx2 - \frac{1}{12}\,\mathrm{e}^{-4\,x5}\,x4^2\,dx3\,dx3 - \frac{1}{12}\,\mathrm{e}^{-4\,x5}\,(4\,x2 + x3\,x4)\,dx3\,dx4 + \frac{1}{2}\,\mathrm{e}^{-4\,x5}\,x3\,dx4\,dx2 - \frac{1}{12}\,\mathrm{e}^{-4\,x5}\,(4\,x2 + x3\,x4)\,dx4\,dx3 + \frac{1}{4}\,\mathrm{e}^{-4\,x5}\,x3^2\,dx4\,dx4,\ \mathrm{e}^{-2\,x5}\,dx3\,dx3,\ \mathrm{e}^{-x5}\,dx3\,dx5 + \mathrm{e}^{-x5}\,dx5\,dx3,\ dx5\,dx5 \right] \tag{7.3}$$

Now we push these tensors on the Lie group *G* down to tensors on the manifold *M* using PushPullTensor

```
G > Qmanifold := PushPullTensor(pi, sigma, Qgroup);
```

$$Qmanifold := \left[-\frac{1}{3}\,\frac{dx\,dw}{z^4} + \frac{1}{4}\,\frac{dy\,dy}{z^4} - \frac{1}{3}\,\frac{dw\,dx}{z^4},\ \frac{dx\,dx}{z^2},\ \frac{dx\,dz}{z^2} + \frac{dz\,dx}{z^2},\ \frac{dz\,dz}{z^2} \right] \tag{7.4}$$

Here is the result we have been after -- a metric g on a 4 dimensional manifold *M* with the prescribed Lie algebra of Killing vectors!

```
M > g := DGzip([c1, c2, c3, c4],Qmanifold, "plus");
```

$$g := \frac{c2\,dx\,dx}{z^2} + \frac{c3\,dx\,dz}{z^2} - \frac{1}{3}\,\frac{c1\,dx\,dw}{z^4} + \frac{1}{4}\,\frac{c1\,dy\,dy}{z^4} + \frac{c3\,dz\,dx}{z^2} + \frac{c4\,dz\,dz}{z^2} - \frac{1}{3}\,\frac{c1\,dw\,dx}{z^4} \tag{7.5}$$

We check that the vectors fields Γ, calculated in Part B, are Killing vectors for g.

```
G > Gamma;
```

$$\left[3\, D\_w,\, 2\, D\_y,\, D\_x,\, 2\, x\, D\_y + \frac{3}{2}\, y\, D\_w,\, x\, D\_x + 2\, y\, D\_y + z\, D\_z + 3\, w\, D\_w\right] \tag{7.6}$$

```
M > LieDerivative(Gamma, g);
```

$$[0\, dx\, dx,\, 0\, dx\, dx,\, 0\, dx\, dx,\, 0\, dx\, dx,\, 0\, dx\, dx] \tag{7.7}$$

In fact, we can use the KillingVectors program in the Killing package to calculate the Lie algebra of Killing vectors for the metric g.

The result is a 5 dimensional algebra thereby proving that Γ is the full infinitesimal isometry algebra of the metric g.

```
M > KV := Killing:-KillingVectors(g);
```

$$KV := \left[-\frac{3\, x\, D\_x}{c1} - \frac{6\, y\, D\_y}{c1} - \frac{3\, z\, D\_z}{c1} - \frac{9\, w\, D\_w}{c1},\, -\frac{3\, D\_x}{c1},\, \frac{4\, x\, D\_y}{c1} + \frac{3\, y\, D\_w}{c1},\, \frac{4\, D\_y}{c1},\, -\frac{3\, D\_w}{c1}\right] \tag{7.8}$$

```
M > GetComponents(KV, Gamma, method = "real", trueorfalse =
    "on");
```

$$true \tag{7.9}$$

Part D. Physical properties of the invariant metrics

We calculate the Einstein tensor of our metric. By subtracting an appropriate cosmological term, we conclude that our metric is a pure radiation solution.

```
M > Ein := EinsteinTensor(g);
```

$$Ein := -\frac{36\, z^4\, D\_x\, D\_w}{c4\, c1} + \frac{48\, z^4\, D\_y\, D\_y}{c4\, c1} + \frac{12\, z^2\, D\_z\, D\_z}{c4^2} + \frac{36\, c3\, z^4\, D\_z\, D\_w}{c4^2\, c1} - \frac{36\, z^4\, D\_w\, D\_x}{c4\, c1} + \frac{36\, c3\, z^4\, D\_w\, D\_z}{c4^2\, c1} - \frac{36\, z^6\, (2\, c4\, c2 - 3\, c3^2)\, D\_w\, D\_w}{c4^2\, c1^2} \tag{8.1}$$

```
M > evalDG(Ein - Lambda*InverseMetric(g));
```

$$\frac{3\, z^4\, (\Lambda\, c4 - 12)\, D\_x\, D\_w}{c1\, c4} - \frac{4\, z^4\, (\Lambda\, c4 - 12)\, D\_y\, D\_y}{c1\, c4} - \frac{z^2\, (\Lambda\, c4 - 12)\, D\_z\, D\_z}{c4^2} - \frac{3\, c3\, z^4\, (\Lambda\, c4 - 12)\, D\_z\, D\_w}{c1\, c4^2} + \frac{3\, z^4\, (\Lambda\, c4 - 12)\, D\_w\, D\_x}{c1\, c4} - \frac{3\, c3\, z^4\, (\Lambda\, c4 - 12)\, D\_w\, D\_z}{c1\, c4^2} + \frac{9\, z^6\, (\Lambda\, c4^2\, c2 - \Lambda\, c4\, c3^2 - 8\, c4\, c2 + 12\, c3^2)\, D\_w\, D\_w}{c4^2\, c1^2} \tag{8.2}$$

```
M > evalDG(Ein - 12/c4*InverseMetric(g));
```

$$\frac{36\, z^6\, c2\, D\_w\, D\_w}{c1^2\, c4} \tag{8.3}$$

Note that D_w is a null vector. This means that g is a *pure radiation solution*.

```
M > TensorInnerProduct(g, D_w, D_w);
```

$$0 \tag{8.4}$$

We find that the Petrov type of our metric is "O".

```
M > PetrovType(g);
```

$$\text{"O"} \tag{8.5}$$

Part E. Classification of the invariant metric.

Is our pure radiation solution a new solution to the Einstein equations or does it exist in the literature?

We are constructing a very detailed and accurate data base of known solutions and a Maplet to search this database.

```
N > Library:-GRExactSolutionsSearch();
```

So the one candidate we found is in the Exact Solutions Books, Chapter 12, equation 38. Here is the metric.

```
M > g38 := Retrieve("Stephani", 1, [12, 38, 2], output = [N,
    "Fields"])[1];
```

$$g38 := -\frac{3}{2}\,\frac{du\,dv}{y^2\,|\_\Lambda|} - \frac{3}{2}\,\frac{dv\,du}{y^2\,|\_\Lambda|} - \frac{3\,\_\varepsilon\,dv\,dv}{|\_\Lambda|\,y} + \frac{3\,dy\,dy}{y^2\,|\_\Lambda|} + \frac{3\,dz\,dz}{y^2\,|\_\Lambda|} \tag{9.1}$$

And finally, let us find an explicit diffeomorphism relating these two metrics. We use the EquivalenceOfMetrics command (still under development) and set the infolevel for this command to 2 so we can see what the program is doing.

```
N > infolevel[GroupActions:-EquivalenceOfMetrics]:=2;
```

$$infolevel_{DifferentialGeometry:-GroupActions:-EquivalenceOfMetrics} := 2 \tag{9.2}$$

```
M > Phi, SideConditions := GroupActions:-EquivalenceOfMetrics
    (g, g38, parameters = {c1, c2, c3, c4});
Finding Killing vectors  for  metric 1.
Finding Killing vectors  for  metric 2.
Finding isometry Lie algebra  for  metric 1.
Finding isometry Lie algebra  for  metric 2.
Finding isomorphism between isometry algebras.
Finding diffeomorphisms  which mapKilling vectors for metric 1
to Killing vectors for metric 2.
Finding diffeomorphism which pulls metric 2 back to metric 1.
```

$$\Phi, SideConditions := \left[u = \frac{\_C6^2\, x}{\_C2} + \frac{8}{3}\,\frac{\_C4^2\, w}{\_C2} + \frac{2\,\_C6\,\_C4\, y}{\_C2} + \_C7 + \_C8\, z^3, v = \_C2\, x + \_C3, y = \_C1\, z^2, z = \_C6\, x + \_C4\, y + \_C5\right], \left[c1 = \frac{12\,\_C4^2}{\_C1^2\, |\_\Lambda|}, c2 = -\frac{3\,\_\varepsilon\,\_C2^2}{|\_\Lambda|\,\_C1}, c3 = -\frac{9}{2}\,\frac{\_C8\,\_C2}{\_C1^2\,|\_\Lambda|}, c4 = \frac{12}{|\_\Lambda|}\right] \tag{9.3}$$

```
N > _EnvExplicit := true:
```

```
M > g1 := PushPullTensor(InverseTransformation(Phi), g);
```

$$g1 := -\frac{1}{8}\,\frac{c1\,\_C1^2\, du\, dv}{y^2\,\_C4^2} - \frac{1}{8}\,\frac{c1\,\_C1^2\, dv\, du}{y^2\,\_C4^2} + \frac{c2\,\_C1\, dv\, dv}{y\,\_C2^2} + \frac{1}{16}\,\frac{\_C1\,\left(3\, c1\,\_C8\,\_C2 + 8\, c3\,\_C4^2\right)\, dv\, dy}{y\,\sqrt{\_C1\, y}\,\_C4^2\,\_C2} + \frac{1}{16}\,\frac{\_C1\,\left(3\, c1\,\_C8\,\_C2 + 8\, c3\,\_C4^2\right)\, dy\, dv}{y\,\sqrt{\_C1\, y}\,\_C4^2\,\_C2} + \frac{1}{4}\,\frac{c4\, dy\, dy}{y^2} + \frac{1}{4}\,\frac{c1\,\_C1^2\, dz\, dz}{y^2\,\_C4^2} \tag{9.4}$$

```
N > subs(SideConditions, g1);
```

$$-\frac{3}{2}\,\frac{du\, dv}{y^2\,|\_\Lambda|} - \frac{3}{2}\,\frac{dv\, du}{y^2\,|\_\Lambda|} - \frac{3\,\_\varepsilon\, dv\, dv}{|\_\Lambda|\, y} + 0\, dv\, dy + 0\, dy\, dv + \frac{3\, dy\, dy}{y^2\,|\_\Lambda|} + \frac{3\, dz\, dz}{y^2\,|\_\Lambda|} \tag{9.5}$$

```
N > g38;
```

$$-\frac{3}{2}\,\frac{du\, dv}{y^2\,|\_\Lambda|} - \frac{3}{2}\,\frac{dv\, du}{y^2\,|\_\Lambda|} - \frac{3\,\_\varepsilon\, dv\, dv}{|\_\Lambda|\, y} + \frac{3\, dy\, dy}{y^2\,|\_\Lambda|} + \frac{3\, dz\, dz}{y^2\,|\_\Lambda|} \tag{9.6}$$

Perfect.

Note: The Exact Solutions book contains a small error with regards to the metric (12.38). The assertion is made that all such metrics are of type N.

Summary

In this worksheet we

- picked a Lie algebra from a database;
- chose a subalgebra of this Lie algebra and constructed the corresponding homogeneous space;
- found the invariant metrics on this homogeneous space
- showed that these metrics solve the Einstein equations
- located this solution is the literature.

Lie's Theorem for Solvable Lie Algebras

Fundamental Computational Problems in Lie Theory and Transformation Groups

Ten years ago all of the following problems were beyond the reach of every computer algebra system.

1. Let $\mathfrak{g}$ be a (finite dim.) Lie algebra. Find a (local) Lie group G whose Lie algebra is $\mathfrak{g}$. (*Lie's 2nd and 3rd Theorems).*

2. Let Γ be a (finite dim.) Lie algebra of vector fields on a manifold M. Find a Lie group G and a group action of G on M whose infinitesimal generators are Γ.

3. Let G act on M. Find the G invariant functions, tensors, connections, distributions , etc.

4. Let G act on a fiber-bundle E. Calculate the differential invariants on $J^k(E)$.

5. Let $\pi : P \rightarrow M$ be a principal G bundle. Calculate the horizontal lifts of curves on M. (*Lie equations*)

6. Use Lie group methods to robustly integrate ODE and other finite-type differential systems (*Example follows).*

7. Let $\mathfrak{g}_1$ and $\mathfrak{g}_2$ be two Lie algebras. Determine if they are isomorphic and, if so, find an explicit isomorphism $\phi : \mathfrak{g}_1 \rightarrow \mathfrak{g}_2$.

8. Let Γ_1 and Γ_2 be two Lie algebras of vector fields on manifolds M_1 and M_2. Determine if they are equivalent, that is, if there is a diffeomorphism $\Phi : M_1 \rightarrow M_2$ such that $\Phi_*(\Gamma_1) = \Gamma_2$ (*equivalence problem for infinitesimal group actions*).

9. Let g_1 and g_2 be metrics on manifolds M_1 and M_2. Determine if they are equivalent, that is, if there is a diffeomorphism $\Phi : M_1 \rightarrow M_2$ such that $\Phi^*(g_2) = g_1$ (*equivalence problem for metrics*).

10. Classify the subalgebras of a given Lie algebra.

*But now good progress is being made,
thanks to a combination of advances in both theory and software.*

Lie's Theorem: Every representation

$\rho : \mathfrak{g} \rightarrow gl(V)$ of a solvable Lie algebra admits a common eigenvector, that is, there is a $v \in V$ such that

$$\rho(x)(v) = \lambda_x v \text{ for all } x \in \mathfrak{g}.$$

In this talk, I'll talk about the surprising role that Lie's theorem in these fundamental problems.

Load packages

```
> restart: with(DifferentialGeometry):
> with(Library):
> with(LieAlgebras):
> with(GroupActions):
> with(LieAlgebraRepresentations):
```

Motivational Example 1

Maple uses several different approaches to integrate scalar ODE.

One approach is based upon the symmetries of the ODE and repeated reduction of order.

This approach is used to integrate the following ODE.

```
> PDEtools[declare]([y(x), u(t)]);
```

u(t) will now be displayed as u

y(x) will now be displayed as y **(3.1)**

```
> Retrieve("Kamke", 1, [7, 16])[1];
```

$$3\, y_{x,x} y_{x,x,x,x} - 5\, y_{x,x,x}^2 \qquad \textbf{(3.2)}$$

```
> DE1 := 3*(diff(y(x), x, x))*(diff(y(x), x, x, x, x))-5*(diff
  (y(x), x, x, x))^2;
```

$$DE1 := 3\, y_{x,x} y_{x,x,x,x} - 5\, y_{x,x,x}^2 \qquad \textbf{(3.3)}$$

```
> ans1 := dsolve(DE1);
```

$$ans1 := y = 3\,(x + \_C2)\,\sqrt{6}\,\_C1^2 \sqrt{-\frac{\_C1}{x + \_C2}} + \_C3\, x + \_C4 \qquad \textbf{(3.4)}$$

Let us make a simple change of variables y = exp(u).

```
> DE2 := simplify(PDETools:-dchange({x = t, y(x) = exp(u(t))},
  DE1, [t, u(t)]));
```

$$DE2 := -e^{2u} \left(-3\, u_{t,t} u_{t,t,t,t} + 18\, u_{t,t,t} u_{t,t} u_t - 9\, u_{t,t}^3 + 18\, u_{t,t}^2 u_t^2 + 9\, u_{t,t} u_t^4 - 3\, u_t^2 u_{t,t,t,t} \right. \qquad \textbf{(3.5)}$$

$$-2\, u_{t,\,t,\,t}\, u_t^3 + 2\, u_t^6 + 5\, u_{t,\,t,\,t}^2\big)$$

```
P > ans2 := dsolve(DE2):
```

Just by looking to the length of *ans2* we see that the result must be very complicated.

```
P > length(ans1), length(ans2);
```

$$95,\ 34034 \qquad (3.6)$$

```
> ans2:
> PDEtools[undeclare]([y(x), u(t)], quiet):
```

Can we use some geometric methods to make the ODE integrator less sensitive to coordinate transformations?

Motivational Example 2

Let $\mathfrak{g}$ be an *n*-dimensional Lie algebra with structure constants *C*. Then Lie's Third (Second?) Theorem asserts that there is, at least locally, a Lie algebra of *n* pointwise independent vector fields Γ on an *n*-dimensional manifold *M* with structure constants *C*.

Cartan gave 2 different proofs of this theorem. A elegant proof appears late (1946, 1951) in Cartan's career and is given in Flanders, page 108.

Here is an example of the implementation of this proof by Cartan.

First we initialize a 4 dimensional Lie algebra.

```
> LD:= Retrieve("Winternitz", 1, [4, 8], Alg1);
```

$$LD := [[e2, e3] = e1,\ [e2, e4] = e2,\ [e3, e4] = -e3] \qquad (4.1)$$

```
> DGsetup(LD);
```

$$\textit{Lie algebra: Alg1} \qquad (4.2)$$

Here is the manifold upon which we shall define our vector fields.

```
M > DGsetup([x1, x2, x3, x4], M);
```

$$\textit{frame name: M} \qquad (4.3)$$

Here is the realization of this Lie algebra as a Lie algebra of vector fields, using the algorithm of Cartan (1946, 1951).

```
M > Gamma1 := LiesThirdTheorem(Alg1, M, method = "Cartan",
    output = "vectors");
```

$$\Gamma 1 := \left[D\_x1,\ -\frac{x3\left(-x4 + e^{x4} - 1\right) D\_x1}{x4\left(-1 + e^{x4}\right)} + \frac{x4\, D\_x2}{-1 + e^{x4}},\ -\frac{x2\left(x4 + e^{-x4} - 1\right) D\_x1}{x4\left(-1 + e^{-x4}\right)} - \frac{x4\, D\_x3}{-1 + e^{-x4}},\ \frac{x2\, x3\left(4 - 2\, e^{x4} - x4\, e^{-x4} - 2\, e^{-x4} + x4\, e^{x4}\right) D\_x1}{x4^2\left(-1 + e^{x4}\right)\left(-1 + e^{-x4}\right)}\right. \qquad (4.4)$$

$$+\frac{x2\left(-x4+\mathrm{e}^{x4}-1\right)D\_x2}{x4\left(-1+\mathrm{e}^{x4}\right)}+\frac{x3\left(x4+\mathrm{e}^{-x4}-1\right)D\_x3}{x4\left(-1+\mathrm{e}^{-x4}\right)}+D\_x4\Big]$$

Check it.

```
M > LD;
```

$$[[e2, e3] = e1, [e2, e4] = e2, [e3, e4] = -e3] \tag{4.5}$$

```
M > LieAlgebraData(Gamma1);
```

$$[[e2, e3] = e1, [e2, e4] = e2, [e3, e4] = -e3] \tag{4.6}$$

Are the vector fields in Γ_1 smooth at the origin? -- yes -- but very complicated.

Much earlier, in 1906, Cartan gave another proof of this theorem, using the structure theory for Lie algebras. Here is the results of this implementation.

```
M > Gamma2 := LiesThirdTheorem(Alg1, M, method = "Solvable",
    output = "vectors");
```

$$\Gamma2 := [D\_x1, (-x4\,x3 - x3)\,D\_x1 + D\_x2, -x2\,x4\,D\_x1 + D\_x3, -x2\,x3\,D\_x1 + x2\,D\_x2 - x3\,D\_x3 + D\_x4] \tag{4.7}$$

Check it.

```
M > LieAlgebraData(Gamma2);
```

$$[[e2, e3] = e1, [e2, e4] = e2, [e3, e4] = -e3] \tag{4.8}$$

Here is a second example.

```
> LD2 := Retrieve("Winternitz", 1, [5, 21], Alg2);
```

$$LD2 := [[e1, e5] = 2\,e1, [e2, e3] = e1, [e2, e5] = e2 + e3, [e3, e5] = e3 + e4, [e4, e5] = e4] \tag{4.9}$$

```
> DGsetup(LD2);
```

$$\textit{Lie algebra: Alg2} \tag{4.10}$$

```
M > DGsetup([x1, x2, x3, x4, x5], N);
```

$$\textit{frame name: N} \tag{4.11}$$

The results are not useful.

```
M > Gamma3 := LiesThirdTheorem(Alg2, N, method = "Cartan",
    output = "vectors"):
M > length(Gamma3);
```

$$10401 \tag{4.12}$$

```
M > Gamma3:
M > Gamma4 := LiesThirdTheorem(Alg2, N, method = "Solvable",
    output = "vectors");
```

$$\Gamma4 := \Big[D\_x1, -\frac{1}{2}\,x3\,D\_x1 + D\_x3 + D\_x4, \frac{1}{2}\,x4\,D\_x1 + D\_x2 + D\_x3, D\_x2, 2\,x1\,D\_x1 \tag{4.13}$$

$$+ (x2 + x3)\, D\_x2 + (x3 + x4)\, D\_x3 + x4\, D\_x4 + D\_x5 \Big]$$

Motivational Example 3

Here we consider the isomorphism problem for Lie algebras.
We start with a simple example of a 4 dimensional Lie algebra.

```
> LD1 := Retrieve("Mubarakyzanov",1, [4,7], Alg1);
```

$$LD1 := [[e1, e4] = 2\, e1, [e2, e3] = e1, [e2, e4] = e2, [e3, e4] = e2 + e3] \quad (5.1)$$

```
> DGsetup(LD1);
```

$$\text{Lie algebra: } Alg1 \quad (5.2)$$

Let's "mix" up these structure equations by changing bases for these two Lie algebras. The Lie AlgebraData command will recompute the structure equations with respect to the new basis.

```
M > LDX := LieAlgebraData([e4, e3 - e4 ,e2 + e1, e1], AlgX);
```

$$LDX := [[e1, e2] = -e1 - e2 - e3 + e4, [e1, e3] = -e3 - e4, [e1, e4] = -2\, e4, [e2, e3] = e3, [e2, e4] = 2\, e4] \quad (5.3)$$

```
Alg1 > LDY := LieAlgebraData([ e1, e2 + e1, e3, e4 + e1 ],
        AlgY);
```

$$LDY := [[e1, e4] = 2\, e1, [e2, e3] = e1, [e2, e4] = e1 + e2, [e3, e4] = -e1 + e2 + e3] \quad (5.4)$$

```
Alg1 > DGsetup(LDX, [p], [chi]);
```

$$\text{Lie algebra: } AlgX \quad (5.5)$$

```
Alg1 > DGsetup(LDY, [q], [sigma]);
```

$$\text{Lie algebra: } AlgY \quad (5.6)$$

Here are the structure equations for *AlgX* and *AlgY*.

```
AlgY > MultiplicationTable(AlgX, "LieTable");
```

$$\left[\begin{array}{cccccc}
 & `|` & p1 & p2 & p3 & p4 \\
 & --- & ---- & ---- & ---- & ---- \\
p1 & `|` & 0 & -p1 - p2 - p3 + p4 & -p3 - p4 & -2\, p4 \\
p2 & `|` & p1 + p2 + p3 - p4 & 0 & p3 & 2\, p4 \\
p3 & `|` & p3 + p4 & -p3 & 0 & 0 \\
p4 & `|` & 2\, p4 & -2\, p4 & 0 & 0
\end{array}\right] \quad (5.7)$$

```
AlgY > MultiplicationTable(AlgY, "LieTable");
```

$$\left[\begin{array}{cccccc} & `|` & q1 & q2 & q3 & q4 \\ & --- & ---- & ---- & ---- & ---- \\ q1 & `|` & 0 & 0 & 0 & 2\,q1 \\ q2 & `|` & 0 & 0 & q1 & q1+q2 \\ q3 & `|` & 0 & -q1 & 0 & -q1+q2+q3 \\ q4 & `|` & -2\,q1 & -q1-q2 & q1-q2-q3 & 0 \end{array}\right] \quad (5.8)$$

Now let's use a direct approach to find an isomorphism by solving :

$$A^i_r A^j_s\, C^t_{ij} = A^t_l\, C^l_{rs}.$$

```
AlgY > phi := FindIsomorphism(AlgX, AlgY, method =
        "direct");
```

$$\phi := [[p1, q4], [p2, q3 - q4], [p3, q2], [p4, q1]] \quad (5.9)$$

Check it -- for example -- $\phi([p_1, p_2]) = [\,\phi(p_1), \phi(p_2)\,]$.

```
AlgY > ApplyHomomorphism(phi, LieBracket(p1, p2));
```

$$q1 - q2 - q3 \quad (5.10)$$

```
AlgY > LieBracket(q4, q3 - q4);
```

$$q1 - q2 - q3 \quad (5.11)$$

Good -- these are equal, as required.

Now let's try a move complicated example.

```
AlgY > LD2 := Retrieve("Mubarakyzanov", 3, [6, 72], Alg2);
```

$$LD2 := [[e1, e6] = 4\,e1, [e2, e5] = e1, [e2, e6] = 3\,e2, [e3, e5] = e2, [e3, e6] = 2\,e3, [e4, e5] = e3, [e4, e6] = e4, [e5, e6] = e4 + e5] \quad (5.12)$$

```
AlgY > DGsetup(LD2);
```

$$\textit{Lie algebra: Alg2} \quad (5.13)$$

```
Alg2 > LDX2 := LieAlgebraData([ e6 -e2 +e1, e4, e3 - e4 ,
        e2 + e1, e3- e2, e5], AlgX2);
```

$$LDX2 := [[e1, e2] = -e2, [e1, e3] = -e2 - 2\,e3, [e1, e4] = e2 + e3 - 4\,e4 - e5, [e1, e5] = e2 + e3 - 3\,e5, [e1, e6] = e3 - e4 - e5 - e6, [e2, e6] = e2 + e3, [e3, e6] = -e5, [e4, e6] = -e2 - e3 + e4 + e5, [e5, e6] = 2\,e2 + 2\,e3 - e4 - 2\,e5] \quad (5.14)$$

```
Alg2 > LDY2 := LieAlgebraData([e1 + e5, e3 - e5 ,e2 + e1,
        e4 + e1 + e6, e6 - e5, e5 ], AlgY2);
```

$$LDY2 := [[e1, e2] = e1 - e3 - e6, [e1, e3] = -e1 + e6, [e1, e4] = 3\,e1 - e2 + e4 - e5 - 4\,e6, [e1, e5] = 3\,e1 + e4 - e5 - 3\,e6, [e2, e3] = e1 - e6, [e2, e4] = e1 + 3\,e2 - e4 \quad (5.15)$$

$$+ e5 + 2\, e6, [e2, e5] = 2\, e1 + 2\, e2 - e3 - e4 + e5, [e2, e6] = -e1 + e3 + e6, [e3, e4] = e1 + 3\, e3 - e6, [e3, e5] = 3\, e3, [e3, e6] = e1 - e6, [e4, e5] = 2\, e1 - e2 + 2\, e4 - 2\, e5 - 4\, e6, [e4, e6] = e1 + e2 - e4 + e5, [e5, e6] = e1 - e4 + e5 - e6]$$

```
Alg2 > DGsetup(LDX2, [p], [chi]);
```

Lie algebra: AlgX2 **(5.16)**

```
Alg2 > DGsetup(LDY2, [q], [sigma]);
```

Lie algebra: AlgY2 **(5.17)**

The direct approach leads to a dense set of 90 quadratic equations for 36 unknowns -- this is too hard to solve.

```
Alg2 > #t := time():
Alg2 > #FindIsomorphism(AlgX2, AlgY2, method = "direct");
Alg2 > #(time() - t)/60;
```

Now let's try a different approach where we shall use Lie's Theorem to change bases to put the structure equations for both Lie algebras into a simpler form.

```
AlgY2 > phi := FindIsomorphism(AlgX2, AlgY2);
```

$$\phi := \left[[p1, q4], [p2, -q1 + q4 - q5], [p3, q1 + q2 - q4 + q5 + q6], [p4, q3], [p5, q1 + q2 - q3], \left[p6, \frac{1}{3}\, q1 + q2 + \frac{5}{3}\, q6\right]\right] \qquad \textbf{(5.18)}$$

Use the Query command to check that ϕ is a homomorphism. Check also that ϕ is injective.

```
>  Query (phi, "Homomorphism");
```

true **(5.19)**

```
AlgY2 > HomomorphismSubalgebras(phi, "Kernel");
```

[ ] **(5.20)**

Notes on Lie's Theorem

```
> Acrobat("C:\\Documents and
  Settings\\Ian\\Desktop\\Srni2010\\Worksheets\\NotesOnLieTheore
  mPresentation.pdf");
```

1076 **(6.1)**

An Illustrative Example

We illustrate the algorithm described in the last section (using the *ad* representation).

First we define the algebra we shall use.

```
> restart: with(DifferentialGeometry):
> with(Library):
```

1 Solvable Algebras

Definition: A Lie algebra $\mathfrak{g}$ is solvable if it admits a descending chain of subalgebras

$$\mathfrak{g} = \mathfrak{g}_0 \supset \mathfrak{g}_1 \supset \mathfrak{g}_2 \ldots \{\, 0 \},$$

with $\mathfrak{g}_{k+1}$ an ideal in $\mathfrak{g}_k$ and $\mathfrak{g}_k/\mathfrak{g}_{k+1}$ abelian.

Example: The set of upper triangular matrices.

Theorem: Every Lie algebra $\mathfrak{g}$ admits a Levi decomposition

$$\mathfrak{g} = \mathfrak{s} \times \mathfrak{r},$$

where $\mathfrak{s}$ is semi-simple and $\mathfrak{r}$ is the radical - the largest solvable ideal in $\mathfrak{g}$.

Semi-simple algebras have been completely classified. No such classification of solvable algebras is possible.

2 Lie's Theorem

Lie's Theorem: Let $\rho : \mathfrak{g} \to \mathrm{gl}(V)$ be a representation of a solvable algebra. Then there exists $v \in V$ such that for all $x \in \mathfrak{g}$

$$\rho(x)(v) = \lambda_x v \quad \text{or} \quad x \cdot v = \lambda_x v.$$

The vector $v \in V$ is a common eigenvector for ρ.

Corollary: There is a basis for V in which the matrices $\rho(x)$ are upper triangular.

Corollary: Change $\mathfrak{g}_{k+1}$ an ideal in $\boxed{\mathfrak{g}_k}$ $\underline{\text{to}}$ $\mathfrak{g}_{k+1}$ an ideal in $\boxed{\mathfrak{g}}$.

Corollary: The only irreducible representations of a solvable algebra are 1-dimensional.

3 Proof 1

Step 1. Pick a codimension 1 ideal $\mathfrak{h} \in \mathfrak{g}$ and proceed by induction on dimension.

$$x \cdot v_0 = \lambda_x v_0 \quad \text{for all} \quad x \in \mathfrak{h}$$

Choose a $y \in \mathfrak{g}$ with $y \notin \mathfrak{h}$.

Step 2. Calculate

$$W = \{\, v \in V \,|\, x \cdot v = \lambda_x v \quad \text{for all} \quad x \in \mathfrak{h} \,\}.$$

Prove that W is a non-empty, y-invariant subspace.

Step 3. Calculate an eigenvector for the restriction of y to W

Comments. From the viewpoint of symbolic implementation there are two difficulties.

1. One has to solve an eigenvalue-eigenvector problem for each dimension of $\mathfrak{g}$.
2. There are many choices to make at each step of the induction. Different choices give different final answers.
3. Worst case – $\mathfrak{g}$ is abelian.

4 Proof 2

Note that if $x \cdot v_0 = \lambda_x v_0$ and $y \cdot v_0 = \lambda_y v_0$,then

$$\begin{aligned}[x, y] \cdot v &= x \cdot (y \cdot v) - y \cdot (x \cdot v) \\ &= x \cdot (\lambda_y v_0) - y(\lambda_x v_0) = 0.\end{aligned}$$

Step 1. Calculate the derived series, defined by $\mathfrak{g}^{(0)} = \mathfrak{g}$ and

$$\mathfrak{g}^{(i+1)} = [\mathfrak{g}^{(i)}, \mathfrak{g}^{(i)}].$$

The $\mathfrak{g}^{(i)}$ are ideals and $\mathfrak{g}$ is solvable iff $\mathfrak{g}^{(p)} = 0$ for some p.

Step 2. Define subspaces of V by

$$N^{(i)} = \{v \in V \mid x \cdot v = 0 \text{ for all } x \in \mathfrak{g}^{(i)}\} \quad i \geq 1.$$

$$N^{(1)} \subset N^{(2)} \subset \cdots \subset N^{(p)} \subset N^{(p+1)} = V.$$

The subspaces $N^{(i)}$ are ρ invariant and this already puts ρ in block triangular form (unless the algebra is abelian).

Step 3. Repeat the calculation of Step 1 for each quotient representation $N^{(i+1)}/N^{(i)}$.
Example. Suppose $\dim V = 10$. We compute (for example)

$$\begin{aligned}N^{(1)} &= \{v_1, v_2, v_3\}, \\ N^{(2)} &= \{v_1, v_2, v_3, v_4, v_5\}, \\ V &= \{v_1, v_2, v_3, v_4, v_5, v_6, v_7, v_8, v_9, v_{10}\}.\end{aligned}$$

Then the matrices $\rho(x)$ all look like

$$\begin{bmatrix} A_{3\times 3} & * & * \\ 0 & B_{2\times 2} & * \\ 0 & 0 & C_{5\times 5} \end{bmatrix}.$$

The matrices $A_{3\times 3}$, $B_{2\times 2}$, $C_{5\times 5}$ all give new representations $\tilde{\rho}_a$ of $\mathfrak{g}$. Re-calculate the N-flags.

$$\begin{gathered}\{v_1, v_2, v_3\} = \{\{w_1, w_2\}, \{w_3\}\}, \\ \{v_4, v_5\} = \{\{w_4\}, \{w_5\}\}, \\ \{v_6, v_7, v_8, v_9, v_{10}\} = \{\{w_6, w_7, w_8\}, \{w_9, w_{10}\}\},\end{gathered}$$

with

$$A_{3\times 3} = \begin{bmatrix} * & * & * \\ * & * & * \\ 0 & 0 & * \end{bmatrix}, \quad B_{2\times 2} = \begin{bmatrix} * & * \\ 0 & * \end{bmatrix}, \quad C_{5\times 5} = \begin{bmatrix} * & * & * & * & * \\ * & * & * & * & * \\ * & * & * & * & * \\ 0 & 0 & 0 & * & * \\ 0 & 0 & 0 & * & * \end{bmatrix}.$$

Step 3. Divide and conquer. The procedure stops when the matrices $\tilde{\rho}_a(x)$ all commute. Then eigenvalue computations are required.

Comments.

1. In many cases no eigenvectors need to be computed at all!

2. The flag $N^{(i)}$ is canonical.

```
> with(LieAlgebras):
> with(GroupActions):
> with(LieAlgebraRepresentations):
```

```
Alg2 > DGsetup(Retrieve("Mubarakyzanov", 3, [6, 43], Alg))
       ;
```

Lie algebra: Alg **(7.1)**

```
Alg > LD := LieAlgebraData([e6, e5 + e4, e4 + e6, e3, e2,
      e1], Alg1);
```

$LD := [[e1, e2] = -2\,e1 - e2 + 2\,e3 - e4, [e1, e3] = -e1 + e3, [e1, e4] = -e4 - e5, [e1, e5] = -e5, [e2, e3] = 2\,e1 + e2 - 2\,e3 + e4 - e6, [e2, e6] = -e5, [e3, e4] = -e4 - e5, [e3, e5] = -e5]$ **(7.2)**

```
Alg > DGsetup(LD);
```

Lie algebra: Alg1 **(7.3)**

We will use the profile command to track calls tothe NullSpace and Eigenvalues commands. We shall also use the infolevel command to follow the progress of the algorithm.

The option abelian = "no" for the command SolvableRepresentation stops the algorithm from making any calls to the Eigenvalues command.

```
Alg1 > profile(LinearAlgebra:-NullSpace, LinearAlgebra:-
       Eigenvalues);
Alg1 > infolevel[SolvableRepresentation] := 3:
Alg1 > Basis := SolvableRepresentation(Alg1, abelian =
       "no", output = ["Basis", "Partition"]);
Partition = [1 .. 6]
   P = [[1, 0, 0, 0, 0, 0], [0, 1, 0, 0, 0, 0], [0, 0, 1, 0,
0, 0], [0, 0, 0, 1, 0, 0], [0, 0, 0, 0, 1, 0], [0, 0, 0, 0, 0,
1]]
   Triangularize  block 1 .. 6
Partition = [1 .. 2, 3 .. 4, 5 .. 6]
   P = [[0, 0, 1, 0, 1, 0], [0, 0, 0, 0, 0, 1], [0, 0, -1, 0,
0, 0], [1, 0, 0, 0, 0, 0], [0, 1, 0, 0, 0, 0], [0, 0, 0, 1, 0,
0]]
   Triangularize  block 1 .. 2
   The following matrices commute
      M = [[-1, 0], [-1, -1]]
      Characteristic Polynomial  = 1+_t^2+2*_t
Partition = [1 .. 2, 3 .. 4, 5 .. 6]
   P = [[0, 0, 1, 0, 1, 0], [0, 0, 0, 0, 0, 1], [0, 0, -1, 0,
0, 0], [1, 0, 0, 0, 0, 0], [0, 1, 0, 0, 0, 0], [0, 0, 0, 1, 0,
0]]
```

```
   Triangularize  block 3 .. 4
Partition = [1 .. 2, 3 .. 3, 4 .. 4, 5 .. 6]
   P = [[0, 0, 0, 1, 1, 0], [0, 0, 0, 0, 0, 1], [0, 0, 0, -1,
0, 0], [1, 0, 0, 0, 0, 0], [0, 1, 0, 0, 0, 0], [0, 0, 1, 0, 0,
0]]
   Triangularize  block 5 .. 6
```

$$Basis := [[e4, e5, e6, e1 - e3, e2, e1], [1, 3, 4, 5, 6]] \tag{7.4}$$

Check it.   All the ad matrices should be block upper triangular (with blocks of size [2,1,1,1])

```
AlgX2 > DGsetup(LieAlgebraData(Basis[1], NewAlg2));
```

$$\textit{Lie algebra: NewAlg2} \tag{7.5}$$

```
AlgX2 > Adjoint();
```

$$\left[\begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}, \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}, \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}, \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 & 0 & -1 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}, \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 2 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}, \begin{bmatrix} -1 & 0 & 0 & 0 & -1 & 0 \\ -1 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & -2 & 0 \\ 0 & 0 & 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}\right] \tag{7.6}$$

```
Alg2 > showprofile();
function              depth    calls     time    time%
bytes   bytes%
--------------------------------------------------------------
-------------
NullSpace                 1        7    0.111   100.00
1328692   100.00
Eigenvalues               0        0    0.000     0.00
 0     0.00
--------------------------------------------------------------
-------------
total:                    1        7    0.111   100.00
1328692   100.00
```

Let's try again - this time using the Eigenvalue command to fully triangularize the representation.

```
NewAlg2 > Basis := SolvableRepresentation(Alg1, abelian
          = "yes", output = ["Basis", "Partition"]);
Partition = [1 .. 6]
   P = [[1, 0, 0, 0, 0, 0], [0, 1, 0, 0, 0, 0], [0, 0, 1, 0,
0, 0], [0, 0, 0, 1, 0, 0], [0, 0, 0, 0, 1, 0], [0, 0, 0, 0, 0,
1]]
   Triangularize  block 1 .. 6
Partition = [1 .. 2, 3 .. 4, 5 .. 6]
   P = [[0, 0, 1, 0, 1, 0], [0, 0, 0, 0, 0, 1], [0, 0, -1, 0,
0, 0], [1, 0, 0, 0, 0, 0], [0, 1, 0, 0, 0, 0], [0, 0, 0, 1, 0,
0]]
   Triangularize  block 1 .. 2
Partition = [1 .. 1, 2 .. 2, 3 .. 4, 5 .. 6]
   P = [[0, 0, 1, 0, 1, 0], [0, 0, 0, 0, 0, 1], [0, 0, -1, 0,
0, 0], [0, 1, 0, 0, 0, 0], [1, 0, 0, 0, 0, 0], [0, 0, 0, 1, 0,
0]]
   Triangularize  block 3 .. 4
Partition = [1 .. 1, 2 .. 2, 3 .. 3, 4 .. 4, 5 .. 6]
   P = [[0, 0, 0, 1, 1, 0], [0, 0, 0, 0, 0, 1], [0, 0, 0, -1,
0, 0], [0, 1, 0, 0, 0, 0], [1, 0, 0, 0, 0, 0], [0, 0, 1, 0, 0,
0]]
   Triangularize  block 5 .. 6
```

$$Basis := [[e5, e4, e6, e1 - e3, e2, e1], [1, 2, 3, 4, 5, 6]] \tag{7.7}$$

```
Alg1 > showprofile();
function              depth    calls     time    time%
bytes   bytes%
--------------------------------------------------------------
-------------
NullSpace                 1       15    0.174    84.88
2729588    86.99
Eigenvalues               1        1    0.031    15.12
408244    13.01
--------------------------------------------------------------
-------------
total:                    2       16    0.205   100.00
3137832   100.00
```

Check it. All the ad matrices should be upper triangular.

```
AlgX2 > DGsetup(LieAlgebraData(Basis[1], NewAlg2));
```

Lie algebra: NewAlg2 **(7.8)**

```
AlgX2 > Adjoint();
```

$$\left[\begin{bmatrix} 0&0&0&0&0&1\\ 0&0&0&0&0&0\\ 0&0&0&0&0&0\\ 0&0&0&0&0&0\\ 0&0&0&0&0&0\\ 0&0&0&0&0&0 \end{bmatrix}, \begin{bmatrix} 0&0&0&0&0&1\\ 0&0&0&0&0&1\\ 0&0&0&0&0&0\\ 0&0&0&0&0&0\\ 0&0&0&0&0&0\\ 0&0&0&0&0&0 \end{bmatrix}, \begin{bmatrix} 0&0&0&0&1&0\\ 0&0&0&0&0&0\\ 0&0&0&0&0&0\\ 0&0&0&0&0&0\\ 0&0&0&0&0&0\\ 0&0&0&0&0&0 \end{bmatrix}, \begin{bmatrix} 0&0&0&0&0&0\\ 0&0&0&0&0&0\\ 0&0&0&0&-1&0\\ 0&0&0&0&0&-1\\ 0&0&0&0&0&0\\ 0&0&0&0&0&0 \end{bmatrix}, \begin{bmatrix} 0&0&-1&0&0&0\\ 0&0&0&0&0&1\\ 0&0&0&1&0&0\\ 0&0&0&0&0&2\\ 0&0&0&0&0&1\\ 0&0&0&0&0&0 \end{bmatrix}, \begin{bmatrix} -1&-1&0&0&0&0\\ 0&-1&0&0&-1&0\\ 0&0&0&0&0&0\\ 0&0&0&1&-2&0\\ 0&0&0&0&-1&0\\ 0&0&0&0&0&0 \end{bmatrix}\right]$$ **(7.9)**

```
Alg1 > unprofile();
Alg1 > infolevel[SolvableRepresentation] := 0:
```

Application 1 (Lie's Third Theorem)

The Maurer-Cartan forms for a solvable Lie algebra are easily integrated when the adjoint representation is upper triangular.

```
NewAlg > ChangeFrame(NewAlg2);
```

NewAlg2 **(8.1)**

```
NewAlg > MultiplicationTable("ExteriorDerivative");
```

$$d(\theta 1) = -\theta 1 \wedge \theta 6 - \theta 2 \wedge \theta 6 - \theta 3 \wedge \theta 5$$

$$d(\theta 2) = -\theta 2 \wedge \theta 6 - \theta 5 \wedge \theta 6$$

$$d(\theta 3) = \theta 4 \wedge \theta 5$$

$$d(\theta 4) = \theta 4 \wedge \theta 6 - 2\,\theta 5 \wedge \theta 6$$

$$d(\theta 5) = -\theta 5 \wedge \theta 6$$

$$d(\theta 6) = 0\,\theta 1 \wedge \theta 2$$ **(8.2)**

```
NewAlg2 > DGsetup([x1, x2, x3, x4, x5, x6], M);
```

frame name: M **(8.3)**

```
M > LiesThirdTheorem(NewAlg2, M, output = "forms");
```

$$\left[dx1 - dx2 + x5\,dx3 + x6\,x5^2\,dx4 + \left(x5\,x4 + 2\,x5^2 + 1 + x6\,x5\,x4 + 2\,x6\,x5^2\right)dx5 - x1\,dx6, dx2 - dx5 - x2\,dx6, dx3 + (x6 - 1)\,x5\,dx4 + (x4 + 2\,x5)\,x6\,dx5, dx4 + 2\,dx5 + x4\,dx6, dx5 - x5\,dx6, dx6\right] \quad \textbf{(8.4)}$$

Application 2 (Isomorphisms of Solvable Lie Algebras)

Let $\phi : \mathfrak{g} \to \overline{\mathfrak{g}}$ be an isomorphism between two solvable Lie algebras. Then ϕ preserves corresponding filtrations -- $\phi : N^{(i)} \to \overline{N}^{(i)}$.

This dramatically simplifies the algebraic equations for finding ϕ.

It also implies that there is a quasi--normal form for solvable algebras.

Application 3 (Canonical coordinates for simple G spaces)

Let G be a Lie group acting regularly on a manifold M. A slice Σ is a (local) cross-section to the orbits on which the isotropy subalgebras are equal. Actions with slices are called simple G spaces. In this case we have (locally)

$$M \cong G\backslash H \times N$$

This, together with results given in Applications 1 and 2, leads to the idea of canonical coordinates for solvable simple G spaces.

Here is an example. We define 3 infinitesimal transformation groups $\Gamma_1, \Gamma_2, \Gamma_3$.
The corresponding abstract Lie algebras are all isomorphic and we have adjusted the bases for these infinitesimal groups to have identical structure equations.

```
M1 > DGsetup([x, y, z, w], M);
```

frame name: M **(10.1)**

```
M1 > Gamma1 := evalDG([z*D_x, y*D_y/w+D_w, -y*z*D_y/w-z*D_w,
     x*(z+1)*D_x/z+D_z]);
```

$$\Gamma1 := \left[z\,D\_x, \frac{y\,D\_y}{w} + D\_w, -\frac{y\,z\,D\_y}{w} - z\,D\_w, \frac{x\,(z+1)\,D\_x}{z} + D\_z\right] \quad \textbf{(10.2)}$$

```
M1 > Gamma2 := evalDG([x*D_x, 2*w*z*D_y+w^2*z*D_w/y, -2*w^3*
     z^3*D_y/y-w^4*z^3*D_w/y^2, x*ln(x)*D_x+y^2*D_y/(w^2*z^2)
     +y*D_z/(w^2*z)]);
```

$$\Gamma2 := \left[x\,D\_x, 2\,w\,z\,D\_y + \frac{w^2\,z\,D\_w}{y}, -\frac{2\,w^3\,z^3\,D\_y}{y} - \frac{w^4\,z^3\,D\_w}{y^2}, x\ln(x)\,D\_x + \frac{y^2\,D\_y}{z^2\,w^2}\right. \quad \textbf{(10.3)}$$

$$+\frac{y\,D\_z}{z\,w^2}\Big]$$

```
M1 > Gamma3 := evalDG([D_x+D_z, D_x/w, -sin(y)*D_x/w, (z-sin
      (y))*D_x+D_y/cos(y)+(z-sin(y)+1)*D_z]);
```

$$\Gamma 3 := \left[D\_x+D\_z,\ \frac{D\_x}{w},\ -\frac{\sin(y)\,D\_x}{w},\ (z-\sin(y))\,D\_x+\frac{D\_y}{\cos(y)}+(z-\sin(y)+1)\,D\_z\right] \tag{10.4}$$

The new command AdaptedActionCoordinates gives a transformation bringing these actions to a canonical form:

```
M > DGsetup([x1, x2, x3, x4], N):
M > phi1 := GroupActions:-AdaptedActionCoordinates(Gamma1, [x
    = 0, y = 0, z = 1, w = 1], N);
```

$$\phi 1 := \left[x1=\frac{x}{z},\ x2=-w,\ x3=-1+z,\ x4=\frac{w}{y}\right] \tag{10.5}$$

```
M > phi2 := GroupActions:-AdaptedActionCoordinates(Gamma2, [x
    = 1, y = 1, z = 1, w = 1], N);
```

$$\phi 2 := \left[x1=\ln(x),\ x2=-\frac{y}{z\,w},\ x3=\frac{-y+z^2\,w^2}{y},\ x4=\frac{w\sqrt{z}}{\sqrt{y}}\right] \tag{10.6}$$

```
Temp2 > phi3 := GroupActions:-AdaptedActionCoordinates
        (Gamma3, [x = 1, y = 0, z = 0, w = 1], N);
```

$$\phi 3 := [x1=z-\sin(y),\ x2=w\sin(y)+(x-z)\,w,\ x3=\sin(y),\ x4=w] \tag{10.7}$$

Here are the canonical forms:

```
M > Pushforward(phi1, InverseTransformation(phi1), Gamma1);
```

$$[D\_x1,\ -D\_x2,\ (x3+1)\,D\_x2,\ x1\,D\_x1+D\_x3] \tag{10.8}$$

```
M > Pushforward(phi2, InverseTransformation(phi2), Gamma2);
```

$$\left[D\_x1,\ -D\_x2,\ (x3+1)\,D\_x2,\ \ln\left(e^{x1}\right)\,D\_x1+D\_x3\right] \tag{10.9}$$

```
M > Pushforward(phi3, InverseTransformation(phi3), Gamma3);
```

$$[D\_x1,\ D\_x2,\ -x3\,D\_x2,\ x1\,D\_x1+D\_x3] \tag{10.10}$$

Perfect -- virtually identical.

Application 4 (ODE Integration)

Finally, we return to the problem of scalar ODE integration. We saw that Maple would correctly integrate a certain 4th order ODE but that problems arose if we made a simple change of variables.

Remarkably, we can use the ideas developed in this lecture to effectively undo this change of variables.

Here is the differential equation.

```
> de2 := -3*(diff(u(t), t, t))*(diff(u(t), t, t, t, t))+18*
  (diff(u(t), t, t, t))*(diff(u(t), t, t))*(diff(u(t), t))-9*
  (diff(u(t), t, t))^3+18*(diff(u(t), t, t))^2*(diff(u(t), t))
  ^2+9*(diff(u(t), t, t))*(diff(u(t), t))^4-3*(diff(u(t), t))
  ^2*(diff(u(t), t, t, t, t))-2*(diff(u(t), t, t, t))*(diff(u
  (t), t))^3+2*(diff(u(t), t))^6+5*(diff(u(t), t, t, t))^2;
```

$$de2 := -3\left(\frac{d^2}{dt^2}u(t)\right)\left(\frac{d^4}{dt^4}u(t)\right)+18\left(\frac{d^3}{dt^3}u(t)\right)\left(\frac{d^2}{dt^2}u(t)\right)\left(\frac{d}{dt}u(t)\right)-9\left(\frac{d^2}{dt^2}u(t)\right)^3+18\left(\frac{d^2}{dt^2}u(t)\right)^2\left(\frac{d}{dt}u(t)\right)^2+9\left(\frac{d^2}{dt^2}u(t)\right)\left(\frac{d}{dt}u(t)\right)^4-3\left(\frac{d}{dt}u(t)\right)^2\left(\frac{d^4}{dt^4}u(t)\right)-2\left(\frac{d^3}{dt^3}u(t)\right)\left(\frac{d}{dt}u(t)\right)^3+2\left(\frac{d}{dt}u(t)\right)^6+5\left(\frac{d^3}{dt^3}u(t)\right)^2 \quad \textbf{(11.1)}$$

We calculate its symmetry algebra using a command from the PDEtools package.

```
P > Symmetries := PDEtools:-Infinitesimals(de2);
```

$$Symmetries := [\_\xi_1(t,u)=1, \_\eta_1(t,u)=0], [\_\xi_1(t,u)=t, \_\eta_1(t,u)=0], [\_\xi_1(t,u)=0, \_\eta_1(t,u)=1], [\_\xi_1(t,u)=e^u, \_\eta_1(t,u)=0], [\_\xi_1(t,u)=0, \_\eta_1(t,u)=e^{-u}], [\_\xi_1(t,u)=0, \_\eta_1(t,u)=t\,e^{-u}] \quad \textbf{(11.2)}$$

Convert to standard DG format.

```
> DGsetup([t, u], P);
```

frame name: P **(11.3)**

```
P > Gamma := convert([Symmetries], DGvector);
```

$$\Gamma := [D\_t, t\,D\_t, D\_u, e^u\,D\_t, e^{-u}\,D\_u, t\,e^{-u}\,D\_u] \quad \textbf{(11.4)}$$

Calculate the structure equations

```
> LD := LieAlgebraData(Gamma, alg);
```

$$LD := [[e1, e2]=e1, [e1, e6]=e5, [e2, e4]=-e4, [e2, e6]=e6, [e3, e4]=e4, [e3, e5]=-e5, [e3, e6]=-e6, [e4, e5]=-e1, [e4, e6]=-e2+e3] \quad \textbf{(11.5)}$$

```
P > DGsetup(LD);
```

Lie algebra: alg **(11.6)**

Calculate the Levi decomposition.

```
> Levi := LeviDecomposition();
```

$$Levi := [[e1, e2+e3, e5], [e2-e3, e4, e6]] \quad \textbf{(11.7)}$$

```
alg > LieAlgebraData(Levi[1], alg2);
```

$$[[e1, e2] = e1, [e2, e3] = -e3] \tag{11.8}$$

```
alg > DGsetup(%);
```

$$\textit{Lie algebra: alg2} \tag{11.9}$$

```
alg > SolvableRepresentation(alg2);
```

$$[[e1, e3, e2]] \tag{11.10}$$

Calculate the canonical coordinates for the radical.

```
alg > DGsetup([s, v], Q);
```

$$\textit{frame name: Q} \tag{11.11}$$

```
Sym1 > Rad := evalDG([Gamma[1], Gamma[5], Gamma[2] + Gamma
       [3]]);
```

$$Rad := [D\_t, e^{-u} D\_u, t\, D\_t + D\_u] \tag{11.12}$$

```
> phi := GroupActions:-AdaptedActionCoordinates(Rad, [t =0, u =
  0], Q);
```

$$\phi := \left[s = t, v = -\frac{1}{e^{-u}}\right] \tag{11.13}$$

Perfect -- this is exactly the transformation used to arrive at de2.

```
P >  simplify(PDETools:-dchange({t = s, u(t) = log(v(s))},
    de2, [s, v(s)]));
```

$$-\frac{-5\left(\frac{d^3}{ds^3}\, v(s)\right)^2 + 3\left(\frac{d^2}{ds^2}\, v(s)\right)\left(\frac{d^4}{ds^4}\, v(s)\right)}{v(s)^2} \tag{11.14}$$