Car-Parking Commutators

The easy way to work these out is to use the general forms calculator that I have placed on my webpage. First you have to load the definitions.

In[15]:=

```
<< forms.m;
```

```
   --   Simple Forms Calculator   -- William L. Burke
$Header: d:/math/bb/forms/forms.m%v 0.4.8 1995/02/20
```

Next we define some shorthands for the basis vectors. We are in a 4 manifold with coordinates (x,y,t,p), t for theta and p for phi. My convention is to write the basis vectors, which you expect to see as d/dx (really with partials) as px, sort of the inverse of dx, and so on.

In[16]:=

```
px = Vector[x];
```

In[17]:=

```
py = Vector[y];
```

In[18]:=

```
pt = Vector[t];
```

In[19]:=

```
pp = Vector[p];
```

Next we give names to the vector fields in the problem:

In[20]:=

```
steer = pp;
```

In[21]:=

```
drive = Cos[t] px + Sin[t] py + Tan[p] pt;
```

Being cautious, let's first work out an obvious commutator to see that the system is working:

In[22]:=

```
Lie[px, x py]
```

Out[22]=

```
Vector[y]
```

Ok, heady with success, we work out the first commutator of these vector fields

In[23]:=

```
Lie[ steer, drive]
```

Out[23]=

$$\mathrm{Sec}[p]^2\ \mathrm{Vector}[t]$$

This is a motion purely in the theta direction. In terms of the car, is a pure rotation. It is clearly independent of the first two vector fields. Let's name it

In[24]:=

```
rotate = Lie[ steer, drive];
```

We will need only one more commutator to close the system, since we can't have more than four linearly independent vectors in four dimensions. First we try

In[27]:=

```
Lie[ rotate, drive ]
```

Out[27]=

$$-(\mathrm{Sec}[p]^2\ \mathrm{Sin}[t]\ \mathrm{Vector}[x]) + \mathrm{Cos}[t]\ \mathrm{Sec}[p]^2\ \mathrm{Vector}[y]$$

This is a vector whose spatial part is perpendicular to drive, and thus it must be linearly independent. This makes up a set of four vectors which span the entire tangent space of our manifold. It is thus possible to park a car in any space whatsoever, at the price of working out double commutators. Jut for fun we also work out

In[28]:=

```
Lie[ rotate, steer ]
```

Out[28]=

$$-2\ \mathrm{Sec}[p]^2\ \mathrm{Tan}[p]\ \mathrm{Vector}[t]$$

We see that this is proportional to steer itself.