In this worksheet we demonstrate the computation of complements of the torsion submodule of a finitely presented module over an Ore algebra. There are different procedures for the case of constant and non-constant coefficients. We also show how a complement of the torsion submodule can be used in order to construct a parametrization of the given linear system.

```
> with(Ore_algebra):
> with(OreModules):
```

We demonstrate the computation of complements of the torsion submodule on a linear system of partial differential equations. Therefore, we start by defining the Weyl algebra:

```
> Alg := DefineOreAlgebra(diff=[D1,x1], diff=[D2,x2], diff=[D3,x3],
> polynom=[x1,x2,x3]):
```

The divergence operator is given by:

```
> div := evalm([[D1, D2, D3]]);
```

$$div := \begin{bmatrix} \mathrm{D1} & \mathrm{D2} & \mathrm{D3} \end{bmatrix}$$

We consider the linear system D1 $div\ y = 0$, where $y = (y1, y2, y3)^T$. The *Alg*-module which is associated with this linear system is denoted by M.

```
> D1div := Mult(D1, div, Alg);
```

$$D1div := \begin{bmatrix} \mathrm{D1}^2 & \mathrm{D1\,D2} & \mathrm{D1\,D3} \end{bmatrix}$$

Let us first compute the torsion submodule of M:

```
> Ext := Exti(Involution(D1div, Alg), Alg, 1);
```

$$Ext := \left[\begin{bmatrix} \mathrm{D1} \end{bmatrix}, \begin{bmatrix} \mathrm{D1} & \mathrm{D2} & \mathrm{D3} \end{bmatrix}, \begin{bmatrix} \mathrm{D3} & \mathrm{D2} & 0 \\ 0 & -\mathrm{D1} & \mathrm{D3} \\ -\mathrm{D1} & 0 & -\mathrm{D2} \end{bmatrix} \right]$$

The computation of the first extension module with values in *Alg* of the module presented by the formal adjoint of *D1div* is isomorphic to the torsion submodule of M. It is generated by the row in $Ext[2]$. Hence, $Ext[2]$ yields a presentation of $M\ /\ \mathrm{t}(M)$. The same generating set of torsion elements can be obtained using *TorsionElements* which expresses them in terms of the system variables *y1*, *y2*, *y3*:

```
> ivar := x1,x2,x3:
> TorsionElements(D1div, [y1(ivar),y2(ivar),y3(ivar)], Alg);
```

$$[[\ \tfrac{\partial}{\partial x1}\,\theta_1(x1,\,x2,\,x3) = 0\ ],$$
$$[\ \theta_1(x1,\,x2,\,x3) = (\tfrac{\partial}{\partial x1}\,\mathrm{y1}(x1,\,x2,\,x3)) + (\tfrac{\partial}{\partial x2}\,\mathrm{y2}(x1,\,x2,\,x3)) + (\tfrac{\partial}{\partial x3}\,\mathrm{y3}(x1,\,x2,\,x3))\ ]]$$

Since M is not torsion-free, M is not parametrizable. Of course, $M/t(M)$ is torsion-free. Let us check whether $M/t(M)$ is projective:

```
> RightInverse(Ext[2], Alg);
```

$$[]$$

Since $Ext[2]$ has full row rank and does not admit a right-inverse, the torsion-free *Alg*-module $M/t(M)$ is not projective. If $M/t(M)$ was projective, then the short exact sequence

$$0 \longrightarrow t(M) \longrightarrow M \longrightarrow M/t(M) \longrightarrow 0$$

would *split*, i.e., there would exist an *Alg*-morphism $M/t(M) \to M$ which, composed by the canonical projection $M \to M/t(M)$, would give the identity on $M/t(M)$.

In any case, if the above short exact sequence splits, then the image of this morphism in M provides a complement of $\mathrm{t}(M)$ in M. Let us check whether there exists such a morphism, even if M / $\mathrm{t}(M)$ is not projective:

```
> ComplementConstCoeff(div, D1div, Alg);
```

$$[]$$

Since we consider a linear system with constant coefficients, we use *ComplementConstCoeff* to find a complement of the *Alg*-module presented by *div* = *Ext*[2] in the *Alg*-module M presented by *D1div*.

In fact, the system of equations over *Alg* which *ComplementConstCoeff* tried to solve was $R' - R'\, S\, R' = V\, R$, where $R' = Ext[2]$ and R were given and S and V were to be found. For more details, see A. Quadrat, D. Robertz, *Parametrizing all solutions of controllable multidimensional linear systems*, to appear in the Proceedings of the 16th IFAC World Congress, Prague, 2005. We see that there is no solution to the above system, which means that there does not exist a complement of $t(M)$ in M.

Note that the procedure *ComplementConstCoeff* tried to solve $R' - R'\, S\, R' = V\, R$ over the commutative polynomial ring in the indeterminates D1, D2, D3. We now apply *Complement* to the same data which tries to find a solution over the Weyl algebra *Alg*, i.e., we treat the given system as if it actually were a non-constant coefficient one.

```
> C := Complement(div, D1div, Alg);
```

$$C := \left[\begin{bmatrix} -x1\,\mathrm{D1} + 1 & -x1\,\mathrm{D2} & -x1\,\mathrm{D3} \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \begin{bmatrix} -x1 \end{bmatrix}, \begin{bmatrix} x1 \\ 0 \\ 0 \end{bmatrix} \right]$$

Now we have found a solution to $R' - R'\, S\, R' = V\, R$:

```
> S := C[3]:  V := C[2]:
> evalm(div - Mult(div, S, div, Alg) - Mult(V, D1div, Alg));
```

$$\begin{bmatrix} 0 & 0 & 0 \end{bmatrix}$$

In order to find a parametrization of M, even if $\mathrm{t}(M)$ is not the zero module, the parametrization of M / $\mathrm{t}(M)$ (found, e.g., by *Exti*) is glued with the solutions of $R'\ \eta = \tau$, where $R' = Ext[2]$ and the vector τ consists of the generating torsion elements. For this glueing a complement of $\mathrm{t}(M)$ in M is needed. Up to now, the procedure *Parametrization* only checks whether a complement of $\mathrm{t}(M)$ in M can be constructed using a right- or generalized inverse of *Ext*[2]. It does not take advantage yet of the command *Complement*. However, in the present case *Parametrization* returns η + *Ext*[3] ξ and $R'\ \eta = \tau$, where the integrated autonomous element θ_1 is plugged into τ:

```
> Parametrization(D1div, Alg);
```

$$\mathrm{table}([1 = \begin{bmatrix} \eta_1(x1,\, x2,\, x3) + (\frac{\partial}{\partial x3}\,\xi_1(x1,\, x2,\, x3)) + (\frac{\partial}{\partial x2}\,\xi_2(x1,\, x2,\, x3)) \\ \eta_2(x1,\, x2,\, x3) - (\frac{\partial}{\partial x1}\,\xi_2(x1,\, x2,\, x3)) + (\frac{\partial}{\partial x3}\,\xi_3(x1,\, x2,\, x3)) \\ \eta_3(x1,\, x2,\, x3) - (\frac{\partial}{\partial x1}\,\xi_1(x1,\, x2,\, x3)) - (\frac{\partial}{\partial x2}\,\xi_3(x1,\, x2,\, x3)) \end{bmatrix},$$
$$2 = ([\ (\frac{\partial}{\partial x1}\,\eta_1(x1,\, x2,\, x3)) + (\frac{\partial}{\partial x2}\,\eta_2(x1,\, x2,\, x3)) + (\frac{\partial}{\partial x3}\,\eta_3(x1,\, x2,\, x3))\ ] =$$
$$(\theta_1(x1,\, x2,\, x3) = \_\mathrm{F1}(x2,\, x3))\ \&\mathrm{where}\, [\ (\_\mathrm{F1}(x2,\, x3),\ \ \textit{are arbitrary functions.})])$$
$$])$$

Using the result of *Complement*, we are now in position to complete the glueing of the parametrization of M:

```
> P := evalm(ApplyMatrix(S, [_F1(x2,x3)], Alg) + ApplyMatrix(Ext[3],
> [xi[1](ivar),xi[2](ivar),xi[3](ivar)], Alg));
```

$$P := \begin{bmatrix} x1\ \_F1(x2, x3) + (\frac{\partial}{\partial x3}\xi_1(x1, x2, x3)) + (\frac{\partial}{\partial x2}\xi_2(x1, x2, x3)) \\ -(\frac{\partial}{\partial x1}\xi_2(x1, x2, x3)) + (\frac{\partial}{\partial x3}\xi_3(x1, x2, x3)) \\ -(\frac{\partial}{\partial x1}\xi_1(x1, x2, x3)) - (\frac{\partial}{\partial x2}\xi_3(x1, x2, x3)) \end{bmatrix}$$

We check that P is a parametrization of the given linear system D1 $div\ y = 0$:

```
> ApplyMatrix(D1div, P, Alg);
```

$$\begin{bmatrix} 0 \end{bmatrix}$$

We can prove that P parametrizes all the smooth solutions of D1 $div\, y = 0$. For more details, see A. Quadrat, D. Robertz, *Parametrizing all solutions of controllable multidimensional linear systems*, to appear in the Proceedings of the 16th IFAC World Congress, Prague, 2005.

Now, we treat a linear system of PDEs having polynomial coefficients:

```
> Alg := DefineOreAlgebra(diff=[D1,x1], diff=[D2,x2], polynom=[x1,x2]):
```

We enter the system matrix:

```
> R := evalm([[x1*D1, 1, D2],[1, x1*D1, D2]]);
```

$$R := \begin{bmatrix} x1\,\mathrm{D1} & 1 & \mathrm{D2} \\ 1 & x1\,\mathrm{D1} & \mathrm{D2} \end{bmatrix}$$

The left *Alg*-module associated with $R\,y = 0$ is again denoted by M. We compute the first extension module with values in *Alg* of the left *Alg*-module presented by the formal adjoint of R:

```
> Ext2 := Exti(Involution(R, Alg), Alg, 1);
```

$$Ext2 := \left[\begin{bmatrix} -1 + x1\,\mathrm{D1} & 0 \\ 0 & -1 + x1\,\mathrm{D1} \end{bmatrix}, \begin{bmatrix} 1 & -1 & 0 \\ 0 & x1\,\mathrm{D1} + 1 & \mathrm{D2} \end{bmatrix}, \begin{bmatrix} \mathrm{D2} \\ \mathrm{D2} \\ -1 - x1\,\mathrm{D1} \end{bmatrix}\right]$$

The torsion submodule $\mathrm{t}(M)$ of M is generated by the rows of $Ext2[2]$. This information can also be obtained using *TorsionElements*:

```
> ivar := x1,x2:
> TorsionElements(R, [y1(ivar),y2(ivar),y3(ivar)], Alg);
```

$$\left[\begin{bmatrix} -\theta_1(x1, x2) + x1\,(\frac{\partial}{\partial x1}\theta_1(x1, x2)) = 0 \\ -\theta_2(x1, x2) + x1\,(\frac{\partial}{\partial x1}\theta_2(x1, x2)) = 0 \end{bmatrix},\right.$$

$$\left.\begin{bmatrix} \theta_1(x1, x2) = \mathrm{y1}(x1, x2) - \mathrm{y2}(x1, x2) \\ \theta_2(x1, x2) = \mathrm{y2}(x1, x2) + x1\,(\frac{\partial}{\partial x1}\mathrm{y2}(x1, x2)) + (\frac{\partial}{\partial x2}\mathrm{y3}(x1, x2)) \end{bmatrix}\right]$$

Since M is not torsion-free, M is not parametrizable. In order to find a parametrization by "integration of torsion elements", we first check whether $M/t(M)$ is projective:

```
> RightInverse(Ext2[2], Alg);
```

$$[]$$

Since *Ext2*[2] has full row rank and does not admit a right-inverse, $M \;/\; \mathrm{t}(M)$ is not projective. Nevertheless, let us check whether there exists a complement of $\mathrm{t}(M)$ in M:

```
> C := Complement(Ext2[2], R, Alg);
```

$$C := \left[\left[\begin{array}{ccc} \frac{1}{2} & \frac{1}{2} & 0 \\ \frac{1}{2} & \frac{1}{2} & 0 \\ 0 & 0 & 1 \end{array}\right], \left[\begin{array}{cc} 0 & 0 \\ \frac{1}{2} & \frac{1}{2} \end{array}\right], \left[\begin{array}{cc} \frac{1}{2} & 0 \\ \frac{-1}{2} & 0 \\ 0 & 0 \end{array}\right] \right]$$

Again, *Complement* found a solution to the system of equations $R' - R'\,S\,R' = V\,R$, where $R' = $ *Ext2*[2]:

```
> S := C[3]:  V := C[2]:
> evalm(Ext2[2] - Mult(Ext2[2], S, Ext2[2], Alg) - Mult(V, R, Alg));
```

$$\left[\begin{array}{ccc} 0 & 0 & 0 \\ 0 & 0 & 0 \end{array}\right]$$

As in the previous case, *Parametrization* yields $\eta + $ *Ext2*[3] ξ and the system $R'\,\eta = \tau$, where τ consists of the integrated autonomous elements:

```
> Parametrization(R, Alg);
```

$$\mathrm{table}([1 = \left[\begin{array}{c} \eta_1(x1,\, x2) + (\frac{\partial}{\partial x2}\, \xi_1(x1,\, x2)) \\ \eta_2(x1,\, x2) + (\frac{\partial}{\partial x2}\, \xi_1(x1,\, x2)) \\ \eta_3(x1,\, x2) - \xi_1(x1,\, x2) - x1\, (\frac{\partial}{\partial x1}\, \xi_1(x1,\, x2)) \end{array}\right],$$
$$2 = \left(\left[\begin{array}{c} \eta_1(x1,\, x2) - \eta_2(x1,\, x2) \\ \eta_2(x1,\, x2) + x1\, (\frac{\partial}{\partial x1}\, \eta_2(x1,\, x2)) + (\frac{\partial}{\partial x2}\, \eta_3(x1,\, x2)) \end{array}\right] = \left[\begin{array}{c} -\_F1(x2)\, x1 \\ \_F1(x2)\, x1 \end{array}\right] \right)$$
$$])$$

Using the result of *Complement*, we can glue the parametrization *Ext2*[3] of $M \;/\; \mathrm{t}(M)$ with the integration of the torsion elements:

```
> P := evalm(ApplyMatrix(S, evalm([[-_F1(x2)*x1],[_F1(x2)*x1]]), Alg) +
> ApplyMatrix(Ext2[3], [xi[1](ivar)], Alg));
```

$$P := \left[\begin{array}{c} -\frac{1}{2}\, \_F1(x2)\, x1 + (\frac{\partial}{\partial x2}\, \xi_1(x1,\, x2)) \\ \frac{1}{2}\, \_F1(x2)\, x1 + (\frac{\partial}{\partial x2}\, \xi_1(x1,\, x2)) \\ -\xi_1(x1,\, x2) - x1\, (\frac{\partial}{\partial x1}\, \xi_1(x1,\, x2)) \end{array}\right]$$

We can check that P is a parametrization of $R\,y = 0$:

```
> ApplyMatrix(R, P, Alg);
```

$$\left[\begin{array}{c} 0 \\ 0 \end{array}\right]$$