> *restart* : *with*(*DifferentialAlgebra*) :

Clairaut Equation

This is a classical example of a differential equation possessing besides its *general solution* a so-called *singular solution.* The plot shows that here the singular solution (plotted in red) is an envelope of the one-parameter family of solutions making up the general solution. The Maple solver for differential equations is able to find this singular solution, as it uses internally functionality of the DifferentialAlgebra package; more precisely, it uses the Rosenfeld-Gröbner algorithm. Applied to this simple system, it returns two components corresponding to the general and the singular solution, respectively. The general solution is here defined by equations and inequations and the inequation is simply given by the non-vanishing of the separant.

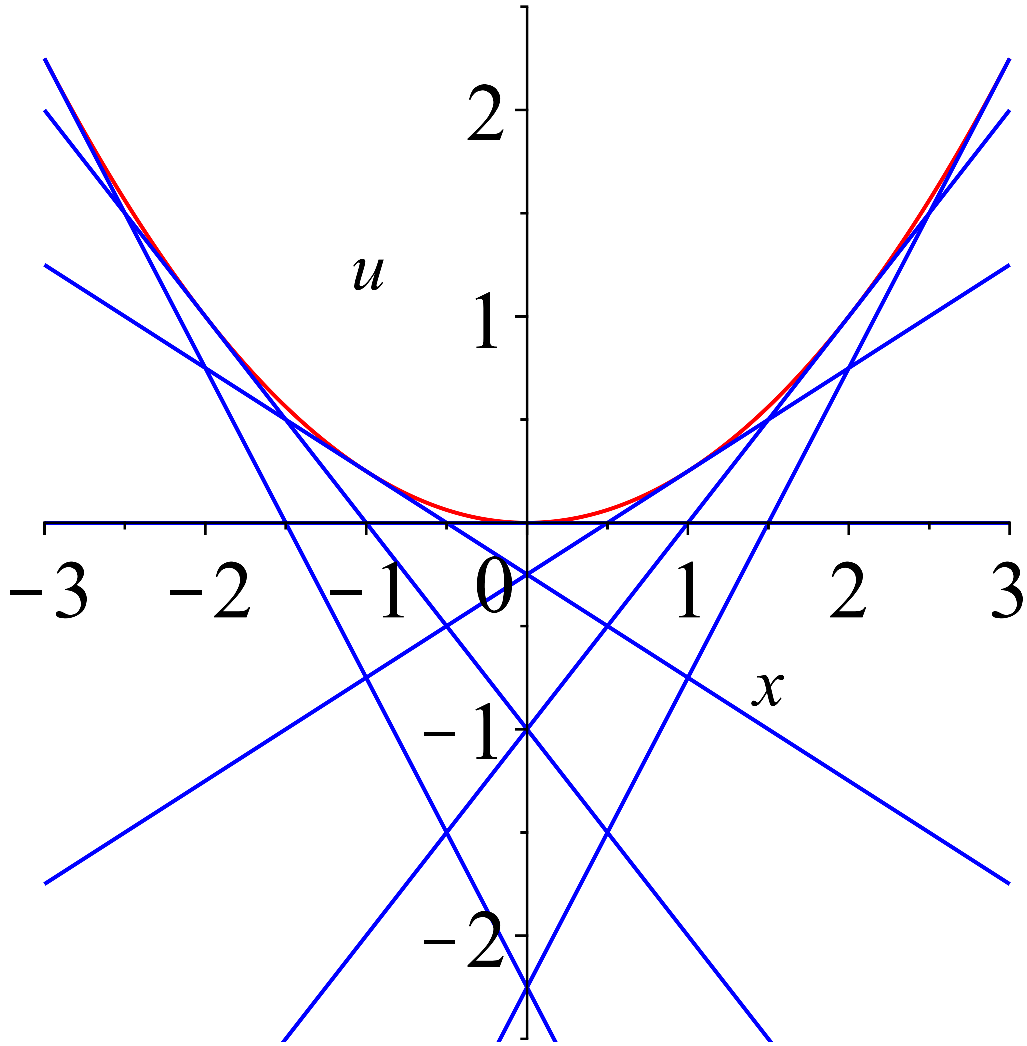
> *clairaut* := *u*(*x*) - *x*·*diff*(*u*(*x*), *x*) + *diff*(*u*(*x*), *x*)^2

$$clairaut := u(x) - x\left(\frac{d}{dx}u(x)\right) + \left(\frac{d}{dx}u(x)\right)^2 \quad \textbf{(1)}$$

> *dsolve*(*clairaut*, *u*(*x*))

$$u(x) = \frac{x^2}{4},\ u(x) = -\_C1^2 + x\,\_C1 \quad \textbf{(2)}$$

> *plot*([*x*^2/4, *seq*(-(*c*/2)^2 + *c* * *x*/2, *c* = -3 ..3)], *x* = -3 ..3, *u* = -2.5 ..2.5, *color* = ["Red", *seq*("Blue", *c* = -3 ..3)])



> $R := DifferentialRing(blocks = [u], derivations = [x])$

$$R := differential\_ring \tag{3}$$

> $G := RosenfeldGroebner([clairaut], R)$

$$G := [regular\_differential\_chain, regular\_differential\_chain] \tag{4}$$

> $Equations(G[1]); Inequations(G[1]);$

$$\left[u(x) - x\left(\frac{d}{dx}u(x)\right) + \left(\frac{d}{dx}u(x)\right)^2\right]$$

$$\left[2\frac{d}{dx}u(x) - x\right] \tag{5}$$

> $Equations(G[2]); Inequations(G[2]);$

$$\left[4u(x) - x^2\right]$$

$$[\ ] \tag{6}$$

There are also PDEs of Clairaut type. Here the built-in solver of Maple is not yet clever enough to detect the presence of singular solutions. And it also does not find all members of the general solution, but returns only a two-parameter family of solutions. But the output of the Rosenfeld-Gröbner algorithm easily provides us with the one singular solution existing here.

> $clairaut2 := -u(x, y) + x\, diff(u(x, y), x) + y\, diff(u(x, y), y) + diff(u(x, y), x)\, diff(u(x, y), y)$

$$clairaut2 := -u(x, y) + x \left(\frac{\partial}{\partial x} u(x, y)\right) + y \left(\frac{\partial}{\partial y} u(x, y)\right) + \left(\frac{\partial}{\partial x} u(x, y)\right) \left(\frac{\partial}{\partial y} u(x, y)\right) \quad \textbf{(7)}$$

> $pdsolve(clairaut2, u(x, y))$

$$u(x, y) = x\, \_c_1 + y\, \_c_2 + \_c_1\, \_c_2 \quad \textbf{(8)}$$

> $R := DifferentialRing(blocks = [u], derivations = [x, y]) :$

> $G := RosenfeldGroebner([clairaut2], R)$

$$G := [regular\_differential\_chain, regular\_differential\_chain] \quad \textbf{(9)}$$

> $Equations(G[1]); Inequations(G[1]);$

$$\left[-u(x, y) + x \left(\frac{\partial}{\partial x} u(x, y)\right) + y \left(\frac{\partial}{\partial y} u(x, y)\right) + \left(\frac{\partial}{\partial x} u(x, y)\right) \left(\frac{\partial}{\partial y} u(x, y)\right)\right]$$

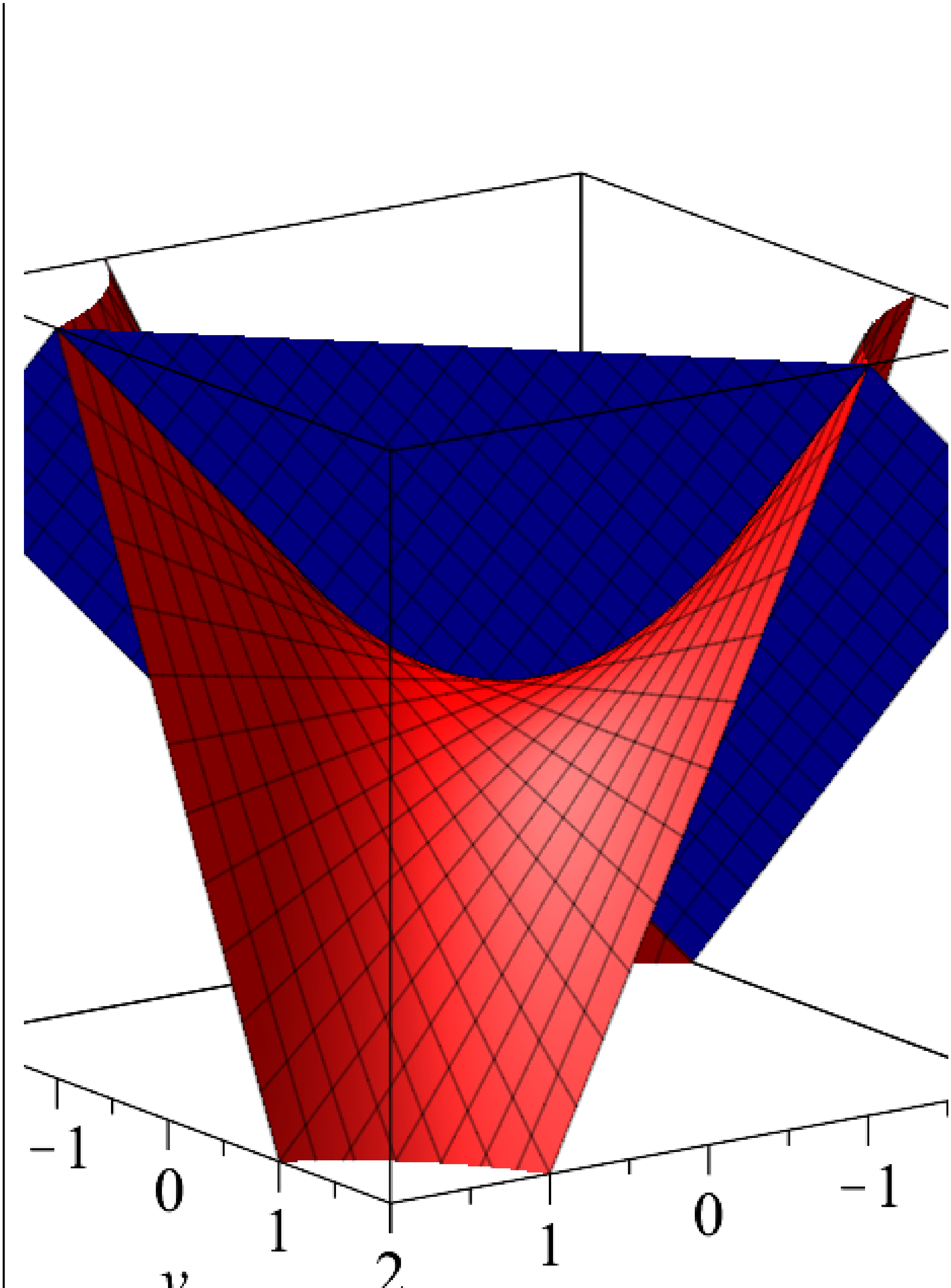
$$\left[x + \frac{\partial}{\partial y} u(x, y)\right] \quad \textbf{(10)}$$

> $Equations(G[2]); Inequations(G[2]);$

$$[u(x, y) + x\, y]$$

$$[\ ] \quad \textbf{(11)}$$

> $plot3d([-x \cdot y, 1 + x + y], x = -2..2, y = -2..2, view = -2..2, color = [\text{"Red"}, \text{"Blue"}])$



−1
0
1
2
y
1
0
−1
x

An Equation due to Chazy

The following equation came up in Chazy's work on the Painleve analysis of third-order equations (note that this is *not* the Chazy equation!). In this kind of analysis, one is concerned with the singularities appearing in solutions, in particular with the existence of so-called *movable singularities* where the location of singularity depends on the initial data. This example is rather special, as the (two-parameter) general solution has no movable pole, but one of the two (one-parameter) singular solutions.

```
> chazy := (diff(u(x), x$2) + u(x)^3·diff(u(x), x))^2 = (u(x)·diff(u(x), x))^2·(4·diff(u(x), x)
      + u(x)^4)
```

$$chazy := \left(\frac{d^2}{dx^2}\,u(x) + u(x)^3\left(\frac{d}{dx}\,u(x)\right)\right)^2 = u(x)^2\left(\frac{d}{dx}\,u(x)\right)^2\left(4\,\frac{d}{dx}\,u(x) + u(x)^4\right) \quad \textbf{(12)}$$

```
> R := DifferentialRing(blocks = [u], derivations = [x]) :
> G := RosenfeldGroebner([chazy], R)
```

$G := [regular\_differential\_chain, regular\_differential\_chain, regular\_differential\_chain]$ **(13)**

```
> Equations(G[1]); Inequations(G[1]);
```

$$\left[2\,u(x)^3\left(\frac{d}{dx}\,u(x)\right)\left(\frac{d^2}{dx^2}\,u(x)\right) - 4\,u(x)^2\left(\frac{d}{dx}\,u(x)\right)^3 + \left(\frac{d^2}{dx^2}\,u(x)\right)^2\right]$$

$$\left[\frac{d^2}{dx^2}\,u(x) + u(x)^3\left(\frac{d}{dx}\,u(x)\right)\right] \quad \textbf{(14)}$$

```
> Equations(G[2]); Inequations(G[2]);
```

$$\left[4\,\frac{d}{dx}\,u(x) + u(x)^4\right]$$

$$[\;] \quad \textbf{(15)}$$

```
> Equations(G[3]); Inequations(G[3]);
```

$$\left[\frac{d}{dx}\,u(x)\right]$$

$$[\;] \quad \textbf{(16)}$$

```
> _EnvExplicit := false : dsolve(chazy, u(x))
```

$$u(x) = RootOf\left(-4 + (4\,\_C1 + 3\,x)\,\_Z^3\right),\; u(x) = \_C1, \quad \textbf{(17)}$$

$$\int^{u(x)} \frac{1}{-\frac{\_a^4}{2} + \frac{\_a^2\,RootOf\left(\_a^4 - 2\,\_Z\,\_a^2 + \_Z^2 - 4\,\_C1\right)}{2} + \_C1}\,d\_a - x - \_C2 = 0$$

Solving Systems by Elimination

Sometimes the Rosenfeld-Gröbner algorithm is useful, even if an explicit system has to be solved. Like Gröbner bases, it can be used to compute elimination ideals which corresponds here to a partial decoupling of the system. The key here is to set up the right differential ring corresponding to an elimination ranking.

```
> sys := [ diff(u(x), x) = v(x) - u(x)^2, diff(v(x), x) = 4·u(x)·v(x) - 4·u(x)^3, diff(w(x), x)
```

$= w(x)^2 - 2\cdot u(x)^2 + v(x)$]

$$sys := \left[\frac{d}{dx} u(x) = v(x) - u(x)^2, \frac{d}{dx} v(x) = 4\, u(x)\, v(x) - 4\, u(x)^3, \frac{d}{dx} w(x) = w(x)^2 - 2\, u(x)^2 + v(x)\right] \quad \textbf{(18)}$$

> $R := DifferentialRing(blocks = [w, v, u], derivations = [x])$:

> $G := RosenfeldGroebner(sys, R)$

$$G := [regular\_differential\_chain] \quad \textbf{(19)}$$

> $newSys := Equations(G[1]); Inequations(G[1]);$

$$newSys := \left[\frac{d}{dx} w(x) - w(x)^2 - \frac{d}{dx} u(x) + u(x)^2, v(x) - \frac{d}{dx} u(x) - u(x)^2, \frac{d^2}{dx^2} u(x) - 2\, u(x) \left(\frac{d}{dx} u(x)\right)\right]$$

$$[\;] \quad \textbf{(20)}$$

> $solu := dsolve(newSys[3], u(x))$

$$solu := u(x) = \frac{\tan\left(\frac{\_C2 + x}{\_C1}\right)}{\_C1} \quad \textbf{(21)}$$

> $solv := v(x) = solve(eval(subs(solu, newSys[2])), v(x))$

$$solv := v(x) = \frac{2\tan\left(\frac{\_C2 + x}{\_C1}\right)^2 + 1}{\_C1^2} \quad \textbf{(22)}$$

> $solw := dsolve(eval(subs(solu, newSys[1])), w(x))$

$$solw := w(x) = \frac{\tan\left(\frac{\_C2 + x}{\_C1}\right)}{\_C1} \quad \textbf{(23)}$$

Detection of Integrability Conditions

A classical application - corresponding to the determination of a Gröbner basis in commutative algebra - is the detection of "hidden" integrability conditions which in particular allows for deciding whether or not the given system is consistent. We consider here the incompressible Navier-Stokes equations from fluid dynamics. The dependent variables u,v,w describe the 3D velocity vector of the fluid, the dependent variable p the pressure in the fluid. The independent variables are space (x,y,z) and time t. The system returned contains one equation more (number 4 in the output). It can be considered as a Poisson equation for the pressure and it is crucial for the development of numerical schemes to integrate the equations in the form stated. Without this equation, the schemes would not be closed, i.e. one would have more unknowns than equation. For computations like this, it is important to use orderly rankings. Hence again one must carefully set up the differential ring in which the computations take place (in fact, for an elimination ranking the computation blows up!). From a mathematical point of view, this is not

> $ns1 := u[t] + u\cdot u[x] + v\cdot u[y] + w\cdot u[z] + p[x] + \mathrm{mu}\cdot(u[x, x] + u[y, y] + u[z, z]);$
$ns2 := v[t] + u\cdot v[x] + v\cdot v[y] + w\cdot v[z] + p[y] + \mathrm{mu}\cdot(v[x, x] + v[y, y] + v[z, z]);$
$ns3 := w[t] + u\cdot w[x] + v\cdot w[y] + w\cdot w[z] + p[z] + \mathrm{mu}\cdot(w[x, x] + w[y, y] + w[z, z]);$

```
ns4 := u[x] + v[y] + w[z];
```

$$ns1 := u_t + u\, u_x + v\, u_y + w\, u_z + p_x + \mu\,(u_{x,x} + u_{y,y} + u_{z,z})$$

$$ns2 := v_t + u\, v_x + v\, v_y + w\, v_z + p_y + \mu\,(v_{x,x} + v_{y,y} + v_{z,z})$$

$$ns3 := w_t + u\, w_x + v\, w_y + w\, w_z + p_z + \mu\,(w_{x,x} + w_{y,y} + w_{z,z})$$

$$ns4 := u_x + v_y + w_z \tag{24}$$

```
> R := DifferentialRing(blocks = [[u, v, w, p]], derivations = [x, y, z, t], arbitrary = [mu]) :
> G := RosenfeldGroebner([ns1, ns2, ns3, ns4], R)
```

$$G := [regular\_differential\_chain] \tag{25}$$

```
> Inequations(G[1]);
```

$$[\mu] \tag{26}$$

```
> for i from 1 to nops(Equations(G[1])) do
    print(i, Equations(G[1])[i]);
  end
```

$$1,\ \mu\, u_{y,y} + \mu\, u_{z,z} - \mu\, v_{x,y} - \mu\, w_{x,z} - u\, v_y - u\, w_z + v\, u_y + w\, u_z + p_x + u_t$$

$$2,\ \mu\, v_{x,x} + \mu\, v_{y,y} + \mu\, v_{z,z} + u\, v_x + v\, v_y + w\, v_z + p_y + v_t$$

$$3,\ \mu\, w_{x,x} + \mu\, w_{y,y} + \mu\, w_{z,z} + u\, w_x + v\, w_y + w\, w_z + p_z + w_t$$

$$4,\ 2\,\mu\, u_y v_x + 2\,\mu\, u_z w_x + 2\,\mu\, v_y^2 + 2\,\mu\, v_y w_z + 2\,\mu\, v_z w_y + 2\,\mu\, w_z^2 + u\,\mu_x v_y + u\,\mu_x w_z - u\,\mu_y v_x - u\,\mu_z w_x - v\,\mu_x u_y - v\,\mu_y v_y - v\,\mu_z w_y - w\,\mu_x u_z - w\,\mu_y v_z - w\,\mu_z w_z + \mu\, p_{x,x} + \mu\, p_{y,y} + \mu\, p_{z,z} - \mu_x p_x - \mu_x u_t - \mu_y p_y - \mu_y v_t - \mu_z p_z - \mu_z w_t$$

$$5,\ u_x + v_y + w_z \tag{27}$$

Solving Overdetermined Systems of PDEs

The Lie symmetry analysis of PDEs leads to large overdetermined systems of linear PDEs. Here the Rosenfeld-Gröbner algorithm reduces to a kind of differential Buchberger algorithms. The produced additional equations often drastically simplify the system and make it possible to solve it automatically. We consider here as a small example the determination of the symmetries of the Burgers equation. Here the effect is not so pronounced, but one can still see that the system produced by the Rosenfeld-Gröbner algorithm is easier to solve than the original one. Nevertheless, the output shows in our example that the symmetry group is five-dimensional

```
> burgers := diff(u(x, t), t) + u(x, t)·diff(u(x, t), x) = diff(u(x, t), x, x)
```

$$burgers := \frac{\partial}{\partial t}\, u(x,t) + u(x,t)\left(\frac{\partial}{\partial x}\, u(x,t)\right) = \frac{\partial^2}{\partial x^2}\, u(x,t) \tag{28}$$

```
> detsys := PDEtools:-DeterminingPDE(burgers, u(x, t), [zeta, phi, eta](x, t, u))
```

$$detsys := \left\{ \frac{\partial}{\partial t}\,\eta(x,t,u) = -\frac{\left(\frac{\partial^2}{\partial t^2}\,\phi(x,t,u)\right) u}{2},\ \frac{\partial}{\partial u}\,\eta(x,t,u) = -\frac{\frac{\partial}{\partial t}\,\phi(x,t,u)}{2},\ \frac{\partial}{\partial x}\,\eta(x, \right. \tag{29}$$

$$t, u) = \frac{\frac{\partial^2}{\partial t^2} \phi(x, t, u)}{2}, \frac{\partial}{\partial u} \phi(x, t, u) = 0, \frac{\partial}{\partial x} \phi(x, t, u) = 0, \frac{\partial^3}{\partial t^3} \phi(x, t, u) = 0, \frac{\partial}{\partial t} \zeta(x, t,$$

$$u) = \frac{u \left(\frac{\partial}{\partial t} \phi(x, t, u)\right)}{2} + \eta(x, t, u), \frac{\partial}{\partial u} \zeta(x, t, u) = 0, \frac{\partial}{\partial x} \zeta(x, t, u) = \frac{\frac{\partial}{\partial t} \phi(x, t, u)}{2}\Bigg\}$$

> *R* := *DifferentialRing*(*blocks* = [*lex*[eta, phi, zeta]], *derivations* = [*u*, *t*, *x*]) :

> *G* := *RosenfeldGroebner*(*detsys*, *R*)

$$G := [regular\_differential\_chain] \quad \textbf{(30)}$$

> *Equations*(*G*[1]); *Inequations*(*G*[1]);

$$\left[\frac{\partial}{\partial u} \eta(x, t, u) + \frac{\partial}{\partial x} \zeta(x, t, u), \frac{\partial}{\partial u} \phi(x, t, u), \frac{\partial}{\partial u} \zeta(x, t, u), \frac{\partial}{\partial t} \eta(x, t, u) + \left(\frac{\partial}{\partial x} \eta(x, t, u)\right) u, \frac{\partial}{\partial t} \phi(x, t, u) - 2 \frac{\partial}{\partial x} \zeta(x, t, u), \frac{\partial}{\partial t} \zeta(x, t, u) - \left(\frac{\partial}{\partial x} \zeta(x, t, u)\right) u - \eta(x, t, u), \frac{\partial^2}{\partial x^2} \eta(x, t, u), \frac{\partial^2}{\partial x^2} \zeta(x, t, u), \frac{\partial}{\partial x} \phi(x, t, u)\right]$$

$$[\ ] \quad \textbf{(31)}$$

> *pdsolve*(*Equations*(*G*[1]), [zeta(*x*, *t*, *u*), phi(*x*, *t*, *u*), eta(*x*, *t*, *u*)])

$$\{\eta(x, t, u) = (t u - x)\ \_C1 + u\ \_C2 + \_C3, \phi(x, t, u) = -\_C1\ t^2 - 2\ \_C2\ t + \_C4, \zeta(x, t, u) = (-\_C1\ x + \_C3)\ t - \_C2\ x + \_C5\} \quad \textbf{(32)}$$

The Pendulum as a "Differential Algebraic Equation"

The modeling of mechanical systems (either in physics or mechanical engineering) often leads to so-called *systems with constraints,* i.e. systems which mix algebraic and differential equations. Unfortunately, it has become costumary to call such systems *differential algebraic equations (DAE)*, although they have a priori nothing to do with differential algebra and simply correspond to overdetermined systems of ODE. If, however, all equations in such a DAE are polynomial, then differential algebra represents a useful tool for analysing such systems.

We consider here the equations of motion of a pendulum which is described by three parameters: length l, mass m and gravitational acceleration g. Besides the two coordinates of the pendulum bob, we also need as a further unknown the Lagrange multiplier lambda representing the force inside the pendulum string. We first compute with respect to an orderly ranking. We obtain two regular differential systems. The second one describes the equilibria of the pendulum where no motion takes place. The first one describes the dynamics outside of the equilibria (interestingly in form of an implicit equation for y of degree 2).

> *newton1* := *m*·*diff*(*x*(*t*), *t*\$2) = lambda(*t*)·*x*(*t*); *newton2* := *m*·*diff*(*y*(*t*), *t*\$2) = lambda(*t*)·*y*(*t*) − *g*;
con := *x*(*t*)^2 + *y*(*t*)^2 = *l*^2;

$$newton1 := m \left(\frac{d^2}{dt^2} x(t)\right) = \lambda(t)\, x(t)$$

$$newton2 := m\left(\frac{d^2}{dt^2}\,y(t)\right) = \lambda(t)\,y(t) - g$$

$$con := x(t)^2 + y(t)^2 = l^2 \quad \textbf{(33)}$$

> $R := DifferentialRing(blocks = [[\text{lambda}, x, y]], derivations = [t], arbitrary = [m, l, g])$:

> $G := RosenfeldGroebner([newton1, newton2, con], R)$

$$G := [regular\_differential\_chain, regular\_differential\_chain] \quad \textbf{(34)}$$

> $Equations(G[1], solved); Inequations(G[1]);$

$$\left[\frac{d}{dt}\,\lambda(t) = \frac{3\left(\frac{d}{dt}\,y(t)\right)g}{l^2}, \left(\frac{d}{dt}\,y(t)\right)^2 = -\frac{-\lambda(t)\,y(t)^2\,l^2 + \lambda(t)\,l^4 + y(t)^3\,g - y(t)\,l^2\,g}{m\,l^2},\right.$$
$$\left. x(t)^2 = -y(t)^2 + l^2\right]$$

$$\left[x(t), l, m\,l^2, \left(\frac{d}{dt}\,y(t)\right)m\,l^2\right] \quad \textbf{(35)}$$

> $Equations(G[2], solved); Inequations(G[2]);$

$$\left[\lambda(t) = \frac{y(t)\,g}{l^2}, x(t) = 0, y(t)^2 = l^2\right]$$

$$[y(t), l] \quad \textbf{(36)}$$

Now we perform the same computation with respect to a different ranking which tries to eliminate lambda. We obtain again two regular differential systems with the second one (describing the equilibria) unchanged. In the first system, we obtain now an algebraic equation determining the multiplier lambda in terms of x,y and their derivatives.

> $RM := DifferentialRing(R, blocks = [\text{lambda}, [x, y]])$:

> $G := RosenfeldGroebner([newton1, newton2, con], RM)$

$$G := [regular\_differential\_chain, regular\_differential\_chain] \quad \textbf{(37)}$$

> $Equations(G[1], solved); Inequations(G[1]);$

$$\left[\lambda(t) = -\frac{-\left(\frac{d}{dt}\,y(t)\right)^2 m\,l^2 - y(t)^3\,g + y(t)\,l^2\,g}{y(t)^2\,l^2 - l^4}, \frac{d^2}{dt^2}\,y(t) =\right.$$
$$\left. -\frac{-\left(\frac{d}{dt}\,y(t)\right)^2 y(t)\,m\,l^2 - y(t)^4\,g + 2\,y(t)^2\,l^2\,g - l^4\,g}{y(t)^2\,m\,l^2 - m\,l^4}, x(t)^2 = -y(t)^2 + l^2\right]$$

$$[x(t), y(t)^2\,l^2 - l^4, y(t)^2\,m\,l^2 - m\,l^4] \quad \textbf{(38)}$$

> $Equations(G[2], solved); Inequations(G[2]);$

$$\left[\lambda(t) = \frac{y(t)\,g}{l^2}, x(t) = 0, y(t)^2 = l^2\right]$$

$$[y(t), l] \quad \textbf{(39)}$$

In the theory of DAEs the notion of the *index* of the system is important (actually, many different notions of an index have been developed). One way to interpret the index consists in studying how many derivatives we need of our equations to obtain the output of the Rosenfeld-Gröbner algorithm. The simplest way to get this value is to add as a "bookkeeping" device some undetermined right hand side ϵ to each equation so that we can simply check in the result which derivatives appear of this right hand side. In our case we obtain then actually many different regular systems. The first one is the most relevant one, as it describes the actual dynamics, and one can see that it contains a third derivative of the "perturbation" ϵ_3. Hence we are dealing with a DAE of index 3.

```
> newton1 := m·diff(x(t), t$2) - lambda(t)·x(t) = epsilon[1](t);
  newton2 := m·diff(y(t), t$2) - lambda(t)·y(t) - g = epsilon[2](t);
  con := x(t)^2 + y(t)^2 - l^2 = epsilon[3](t);
```

$$newton1 := m\left(\frac{d^2}{dt^2}x(t)\right) - \lambda(t)\,x(t) = \epsilon_1(t)$$

$$newton2 := m\left(\frac{d^2}{dt^2}y(t)\right) - \lambda(t)\,y(t) - g = \epsilon_2(t)$$

$$con := x(t)^2 + y(t)^2 - l^2 = \epsilon_3(t) \tag{40}$$

```
> R := DifferentialRing(blocks = [[lambda, x, y], [epsilon[1], epsilon[2], epsilon[3]]],
      derivations = [t], arbitrary = [m, l, g]) :
> G := RosenfeldGroebner([newton1, newton2, con], R)
```

$$G := [regular\_differential\_chain, regular\_differential\_chain, regular\_differential\_chain, regular\_differential\_chain, regular\_differential\_chain, regular\_differential\_chain, regular\_differential\_chain] \tag{41}$$

```
> Equations(G[1], solved); Inequations(G[1]);
```

$$\left[\frac{d}{dt}\lambda(t) = -\left(6\left(\frac{d}{dt}y(t)\right)x(t)\,y(t)\,\epsilon_1(t) + 6\left(\frac{d}{dt}y(t)\right)y(t)^2\epsilon_2(t) + 6\left(\frac{d}{dt}\right.\right.\right.$$

$$\left.y(t)\right)y(t)^2 g - 6\left(\frac{d}{dt}y(t)\right)\epsilon_2(t)\,\epsilon_3(t) - 6\left(\frac{d}{dt}y(t)\right)\epsilon_2(t)\,l^2 - 6\left(\frac{d}{dt}y(t)\right)\epsilon_3(t)\,g$$

$$-6\left(\frac{d}{dt}y(t)\right)l^2 g + 4\lambda(t)\,y(t)^2\left(\frac{d}{dt}\epsilon_3(t)\right) - 4\lambda(t)\left(\frac{d}{dt}\epsilon_3(t)\right)\epsilon_3(t) - 4\lambda(t)\left(\frac{d}{dt}\right.$$

$$\left.\epsilon_3(t)\right)l^2 + 2x(t)\,y(t)^2\left(\frac{d}{dt}\epsilon_1(t)\right) - 2x(t)\left(\frac{d}{dt}\epsilon_1(t)\right)\epsilon_3(t) - 2x(t)\left(\frac{d}{dt}\epsilon_1(t)\right)l^2$$

$$-3x(t)\left(\frac{d}{dt}\epsilon_3(t)\right)\epsilon_1(t) + 2y(t)^3\left(\frac{d}{dt}\epsilon_2(t)\right) - y(t)^2\left(\frac{d^3}{dt^3}\epsilon_3(t)\right)m - 2y(t)\left(\frac{d}{dt}\right.$$

$$\epsilon_2(t)\Big)\,\epsilon_3(t) - 2\,y(t)\left(\frac{d}{dt}\,\epsilon_2(t)\right)l^2 + \left(\frac{d^3}{dt^3}\,\epsilon_3(t)\right)\epsilon_3(t)\,m + \left(\frac{d^3}{dt^3}\,\epsilon_3(t)\right)m\,l^2\Big) \Big/$$

$$\left(2\,y(t)^2\,\epsilon_3(t) + 2\,y(t)^2\,l^2 - 2\,\epsilon_3(t)^2 - 4\,\epsilon_3(t)\,l^2 - 2\,l^4\right),\ \left(\frac{d}{dt}\,y(t)\right)^2 =$$

$$-\frac{1}{4\,\epsilon_3(t)\,m + 4\,m\,l^2}\Big(-4\left(\frac{d}{dt}\,y(t)\right)y(t)\left(\frac{d}{dt}\,\epsilon_3(t)\right)m - 4\,\lambda(t)\,y(t)^2\,\epsilon_3(t)$$

$$-4\,\lambda(t)\,y(t)^2\,l^2 + 4\,\lambda(t)\,\epsilon_3(t)^2 + 8\,\lambda(t)\,\epsilon_3(t)\,l^2 + 4\,\lambda(t)\,l^4 - 4\,x(t)\,y(t)^2\,\epsilon_1(t)$$

$$+4\,x(t)\,\epsilon_1(t)\,\epsilon_3(t) + 4\,x(t)\,\epsilon_1(t)\,l^2 - 4\,y(t)^3\,\epsilon_2(t) - 4\,y(t)^3\,g + 2\,y(t)^2\left(\frac{d^2}{dt^2}\,\epsilon_3(t)\right)m$$

$$+4\,y(t)\,\epsilon_2(t)\,\epsilon_3(t) + 4\,y(t)\,\epsilon_2(t)\,l^2 + 4\,y(t)\,\epsilon_3(t)\,g + 4\,y(t)\,l^2\,g - 2\,\Big(\frac{d^2}{dt^2}$$

$$\epsilon_3(t)\Big)\,\epsilon_3(t)\,m - 2\left(\frac{d^2}{dt^2}\,\epsilon_3(t)\right)m\,l^2 + \left(\frac{d}{dt}\,\epsilon_3(t)\right)^2 m\Big),\ x(t)^2 = -y(t)^2 + \epsilon_3(t) + l^2\Big]$$

$$\Big[x(t),\ \epsilon_3(t)\,m + m\,l^2,\ 2\left(\frac{d}{dt}\,y(t)\right)\epsilon_3(t)\,m + 2\left(\frac{d}{dt}\,y(t)\right)m\,l^2 - y(t)\left(\frac{d}{dt}\,\epsilon_3(t)\right)m,$$
$$y(t)^2\,\epsilon_3(t) + y(t)^2\,l^2 - \epsilon_3(t)^2 - 2\,\epsilon_3(t)\,l^2 - l^4\Big] \qquad \textbf{(42)}$$

This approach to define an index has the drawback that it depends on the chosen ranking. If we try as above to obtain as fast as possible an equation of lambda, then we obtain a different value for the index, namely only 2. The difference is easy to explain. This time we obtain an algebraic equation for lambda, above it was a differential equation and thus required one differentiation more.

```
> RM := DifferentialRing(R, blocks = [lambda, [x, y], [epsilon[1], epsilon[2], epsilon[3]]]) :
> G := RosenfeldGroebner([newton1, newton2, con], R)
```

G := [regular_differential_chain, regular_differential_chain, regular_differential_chain, regular_differential_chain, regular_differential_chain, regular_differential_chain, regular_differential_chain] **(43)**

```
> Equations(G[1], solved); Inequations(G[1]);
```

$$\Big[\frac{d}{dt}\,\lambda(t) = -\Big(6\left(\frac{d}{dt}\,y(t)\right)x(t)\,y(t)\,\epsilon_1(t) + 6\left(\frac{d}{dt}\,y(t)\right)y(t)^2\,\epsilon_2(t) + 6\,\Big(\frac{d}{dt}$$

$$y(t)\Big)\,y(t)^2\,g - 6\left(\frac{d}{dt}\,y(t)\right)\epsilon_2(t)\,\epsilon_3(t) - 6\left(\frac{d}{dt}\,y(t)\right)\epsilon_2(t)\,l^2 - 6\left(\frac{d}{dt}\,y(t)\right)\epsilon_3(t)\,g$$

$$-6\left(\frac{d}{dt}y(t)\right)l^2 g+4\lambda(t)\,y(t)^2\left(\frac{d}{dt}\epsilon_3(t)\right)-4\lambda(t)\left(\frac{d}{dt}\epsilon_3(t)\right)\epsilon_3(t)-4\lambda(t)\left(\frac{d}{dt}\epsilon_3(t)\right)l^2+2x(t)\,y(t)^2\left(\frac{d}{dt}\epsilon_1(t)\right)-2x(t)\left(\frac{d}{dt}\epsilon_1(t)\right)\epsilon_3(t)-2x(t)\left(\frac{d}{dt}\epsilon_1(t)\right)l^2-3x(t)\left(\frac{d}{dt}\epsilon_3(t)\right)\epsilon_1(t)+2y(t)^3\left(\frac{d}{dt}\epsilon_2(t)\right)-y(t)^2\left(\frac{d^3}{dt^3}\epsilon_3(t)\right)m-2y(t)\left(\frac{d}{dt}\epsilon_2(t)\right)\epsilon_3(t)-2y(t)\left(\frac{d}{dt}\epsilon_2(t)\right)l^2+\left(\frac{d^3}{dt^3}\epsilon_3(t)\right)\epsilon_3(t)\,m+\left(\frac{d^3}{dt^3}\epsilon_3(t)\right)m\,l^2\Bigg)\Bigg/$$

$$\left(2y(t)^2\epsilon_3(t)+2y(t)^2 l^2-2\epsilon_3(t)^2-4\epsilon_3(t)\,l^2-2l^4\right),\ \left(\frac{d}{dt}y(t)\right)^2=$$

$$-\frac{1}{4\epsilon_3(t)\,m+4m\,l^2}\Bigg(-4\left(\frac{d}{dt}y(t)\right)y(t)\left(\frac{d}{dt}\epsilon_3(t)\right)m-4\lambda(t)\,y(t)^2\epsilon_3(t)-4\lambda(t)\,y(t)^2 l^2+4\lambda(t)\,\epsilon_3(t)^2+8\lambda(t)\,\epsilon_3(t)\,l^2+4\lambda(t)\,l^4-4x(t)\,y(t)^2\epsilon_1(t)+4x(t)\,\epsilon_1(t)\,\epsilon_3(t)+4x(t)\,\epsilon_1(t)\,l^2-4y(t)^3\epsilon_2(t)-4y(t)^3 g+2y(t)^2\left(\frac{d^2}{dt^2}\epsilon_3(t)\right)m+4y(t)\,\epsilon_2(t)\,\epsilon_3(t)+4y(t)\,\epsilon_2(t)\,l^2+4y(t)\,\epsilon_3(t)\,g+4y(t)\,l^2 g-2\left(\frac{d^2}{dt^2}\epsilon_3(t)\right)\epsilon_3(t)\,m-2\left(\frac{d^2}{dt^2}\epsilon_3(t)\right)m\,l^2+\left(\frac{d}{dt}\epsilon_3(t)\right)^2 m\Bigg),\ x(t)^2=-y(t)^2+\epsilon_3(t)+l^2\Bigg]$$

$$\left[x(t),\ \epsilon_3(t)\,m+m\,l^2,\ 2\left(\frac{d}{dt}y(t)\right)\epsilon_3(t)\,m+2\left(\frac{d}{dt}y(t)\right)m\,l^2-y(t)\left(\frac{d}{dt}\epsilon_3(t)\right)m,\ y(t)^2\epsilon_3(t)+y(t)^2 l^2-\epsilon_3(t)^2-2\epsilon_3(t)\,l^2-l^4\right] \quad \textbf{(44)}$$

Some Applications in Control Theory

In physics, one usual only observes the behaviour of a given system. Engineers want to influence and to measure the behaviour of a system. Thus they distinguish three types of dependent variables: the vector **x** represents some internal variables of the system called *state*, the vector **u** some controls with which the system can be influenced called *input* and the vector **y** of quantities which can be observed or measured called *output*. The independent variable is the time t. Typically, one has then a differential system of the form **x**'=**f**(**x**,**u**,**p**), **y**=**g**(**x**,**u**,**p**) where the vector **p** represents some parameters. A first important problem is now to obtain a so-called *input-output* representation of the system, i.e. a differential system that describes the dependency of the output **y** on the input **u** without using the state **x**. Obviously, this corresponds again to an elimination problem from a differential algebraic point of view. A second question is the problem of *observability*: is it possible to reconstruct the state **x**, if we know the input **u** and the output **y**. Finally, we take a look at the problem of *parameter identifiability*: does the knowledge of the input **u** and the output **y** suffice to determine the values of the parameters **p**?

We will now try to answer these three questions for a very simple control system with two state variables and one in- and output, respectively. Furthermore, the system depends on one parameter called λ. Again it is crucial to use the right ranking for this computation. Our output consists of three regular differential systems. The first one corresponds to the "generic" case. The fourth equation in it provides us with an input-output representation relating u and y without any reference to the state **x**. The third equation represents an algebraic equation for λ in terms of y and u. Hence our parameter is algebraically identifiable (for given y and u only finitely many values are possible). The second equation allows us to determine x_1 from u and y so that this state variable is algebraically observable. By contrast, the first equation is still a differential equation for x_2 and hence this state variable is not observable. The second differential system tells us that in the special case u=0, the parameter λ is no longer identifiable. The output is still obtainable via a differential equation, but not of the state variables is observable.

```
> state1 := diff(x[1](t), t) = x[1](t) - lambda(t)·u(t);
  state2 := diff(x[2](t), t) = x[2](t)·(1 - x[1](t));
  out := y(t) = lambda(t)·x[1](t);
  para := diff(lambda(t), t);
```

$$state1 := \frac{d}{dt} x_1(t) = x_1(t) - \lambda(t)\, u(t)$$

$$state2 := \frac{d}{dt} x_2(t) = x_2(t)\, \left(1 - x_1(t)\right)$$

$$out := y(t) = \lambda(t)\, x_1(t)$$

$$para := \frac{d}{dt} \lambda(t) \qquad (45)$$

```
> R := DifferentialRing(blocks = [[x[1], x[2]], lambda, y, u], derivations = [t]) :
> G := RosenfeldGroebner([state1, state2, out, para], R)
```

$$G := [regular\_differential\_chain, regular\_differential\_chain, regular\_differential\_chain] \qquad (46)$$

```
> for i from 1 to nops(Equations(G[1])) do
    print(i, Equations(G[1])[i]);
  end
```

$$1, \left(\frac{d}{dt} x_2(t)\right) \left(\frac{d}{dt} y(t)\right) - \left(\frac{d}{dt} x_2(t)\right) y(t) - x_2(t)\, \lambda(t)\, y(t)\, u(t) - x_2(t) \left(\frac{d}{dt} y(t)\right) + x_2(t)\, y(t)$$

$$2, x_1(t) \left(\frac{d}{dt} y(t)\right) - x_1(t)\, y(t) + \lambda(t)\, y(t)\, u(t)$$

$$3, \lambda(t)^2\, u(t) + \frac{d}{dt} y(t) - y(t)$$

$$4, \left(\frac{d^2}{dt^2} y(t)\right) u(t) - \left(\frac{d}{dt} y(t)\right) \left(\frac{d}{dt} u(t)\right) - \left(\frac{d}{dt} y(t)\right) u(t) + y(t) \left(\frac{d}{dt} u(t)\right) \qquad (47)$$

> $Inequations(G[1]);$

> $Equations(G[2]); Inequations(G[2]);$

$$\left[\left(\frac{d}{dt}x_2(t)\right)\lambda(t) - x_2(t)\,\lambda(t) + x_2(t)\,y(t),\ \lambda(t)\,x_1(t) - y(t),\ \frac{d}{dt}\lambda(t),\ \frac{d}{dt}y(t) - y(t),\ u(t)\right]$$

$$[\lambda(t)] \tag{48}$$

> $Equations(G[3]); Inequations(G[3]);$

$$\left[\frac{d}{dt}x_1(t) - x_1(t),\ \frac{d}{dt}x_2(t) + x_1(t)\,x_2(t) - x_2(t),\ \lambda(t),\ y(t)\right]$$

$$[\ ] \tag{49}$$

Analysis of a Biochemical Model

Biochemical reactions can be modelled by systems of ODEs which are obtained more or less automatically from the chemical reaction equations. While we will be playing here with a rather small model, these models can be very huge and model reduction is a big issue. A typical feature of biochemical reactions is the presence of multiple time scales, i.e. some reactions are much faster than others. This allows for certain types of reductions.

We study here a simple, but very classical model: a substrate S transforms into a product P under the influence of an enzyme E. At an intermediate step a complex C is formed. The reaction E+S -> C is reversible with reaction rate constants k_1, k_{-1} for the two directions. The reaction C -> E+P is irreversible with a reaction rate constant k_2. The principle of mass-action kinetics allows to derive automatically an ODE system from the stochiometric coefficients in the reactions. They are collected in the stochiometric matrix N and V is a vector whose entries describe the reaction rates of the three involved reactions.

> $with(LinearAlgebra):$

> $X := \langle E(t), S(t), C(t), P(t)\rangle :$

> $V := \langle k[1] * E(t) * S(t), k[-1] * C(t), k[2] * C(t)\rangle :$

> $N := \langle\langle -1, -1, 1, 0\rangle \mid \langle 1, 1, -1, 0\rangle \mid \langle 1, 0, -1, 1\rangle\rangle :$

> $madm := map(diff, X, t) = N . V;$

$$madm := \begin{bmatrix} \frac{d}{dt}E(t) \\ \frac{d}{dt}S(t) \\ \frac{d}{dt}C(t) \\ \frac{d}{dt}P(t) \end{bmatrix} = \begin{bmatrix} -k_1\,E(t)\,S(t) + k_{-1}\,C(t) + k_2\,C(t) \\ -k_1\,E(t)\,S(t) + k_{-1}\,C(t) \\ k_1\,E(t)\,S(t) - k_{-1}\,C(t) - k_2\,C(t) \\ k_2\,C(t) \end{bmatrix} \tag{50}$$

This is a final model which can be analysed using the theory of ODEs. From a differential algebraic point of view, there is nothing to do. Such an explicit system generates a prime differential ideal and all initials and separants are 1. According to biochemists, the irreversible reaction C -> E+P is often much faster than the reversible one E+S -> C, i.e. the value of k_2 is much greater than those of k_1 and k_{-1}. In the so-called *quasi-steady state approximation*, one thus tries to set $\frac{d}{dt}C(t)=0$ in the hope to obtain a simpler (in particular, a smaller) model that can be analysed more easily. However, this is problematic from a mathematical point of view, as we will now show using differential algebra. To avoid unnecessary case distinctions, we treat the reaction rate constants as elements of the base field of our ring of differential polynomials. We obtain two regular differential systems which are indeed simpler - unfortunately, they are too simple, as they do not describe any dynamics any more.

```
> R := DifferentialRing(blocks = [[C, E, P, S], [k[1](), k[-1](), k[2]()]], derivations = [t]) :
> madm_approx := [ seq(lhs(madm)[i] = rhs(madm)[i], i = 1 .. Dimension(X)), diff(C(t), t)
     = 0 ];
```

$$madm\_approx := \left[\frac{d}{dt}E(t)=-k_1E(t)S(t)+k_{-1}C(t)+k_2C(t), \frac{d}{dt}S(t)=-k_1E(t)S(t)+k_{-1}C(t), \frac{d}{dt}C(t)=k_1E(t)S(t)-k_{-1}C(t)-k_2C(t), \frac{d}{dt}P(t)=k_2C(t), \frac{d}{dt}C(t)=0\right] \quad (51)$$

```
> Field := field(generators = [k[1], k[-1], k[2]]) :
> G := RosenfeldGroebner(madm_approx, R, basefield = Field);
```

$$G := [regular\_differential\_chain, regular\_differential\_chain] \quad (52)$$

```
> Equations(G);
```

$$\left[\left[\frac{d}{dt}E(t), \frac{d}{dt}P(t), C(t), S(t)\right], \left[\frac{d}{dt}P(t), \frac{d}{dt}S(t), C(t), E(t)\right]\right] \quad (53)$$

We thus make a new attempt to exploit the different time scales. We essentially forget everything about the irreversible reaction except that it exists and implies a conservation law under our quasi-steady state assumption. Its effect is captured by assuming the the rate "constants" of the reversible reaction are no longer constant, but a yet unknown function $F_1(t)$. Thenwe choose our ranking in such a way that we can eliminate this unknown function.

```
> sys := [ diff(E(t), t) = - F[1](t) + k[2]*C(t), diff(S(t), t) = - F[1](t),
           diff(C(t), t) = - k[2]*C(t) + F[1](t), diff(P(t), t) = k[2]*C(t),
           0 = k[1]*E(t)*S(t) - k[-1]*C(t) ];
```

$$sys := \left[\frac{d}{dt}E(t)=-F_1(t)+k_2C(t), \frac{d}{dt}S(t)=-F_1(t), \frac{d}{dt}C(t)=-k_2C(t)+F_1(t), \frac{d}{dt}P(t)=k_2C(t), 0=k_1E(t)S(t)-k_{-1}C(t)\right] \quad (54)$$

```
> R := DifferentialRing(blocks = [F[1], [C, E, P, S], [k[1](), k[-1](), k[2]()]], derivations
     = [t]) :
> G := RosenfeldGroebner(sys, R, basefield = Field);
```

$$G := [regular\_differential\_chain, regular\_differential\_chain, regular\_differential\_chain] \quad (55)$$

```
> Equations(G[1], solved); Inequations(G[1]);
```

$$\left[F_1(t) = -\frac{-E(t)S(t)^2k_1^2k_2 - E(t)S(t)k_1k_{-1}k_2}{E(t)k_1k_{-1}+S(t)k_1k_{-1}+k_{-1}^2}, \frac{d}{dt}E(t)\right.$$

$$= \frac{E(t)^2\, S(t)\, k_1^2\, k_2}{E(t)\, k_1\, k_{-1} + S(t)\, k_1\, k_{-1} + k_{-1}^2}, \frac{d}{dt} P(t) = \frac{E(t)\, S(t)\, k_1\, k_2}{k_{-1}}, \frac{d}{dt} S(t) =$$

$$- \frac{E(t)\, S(t)^2\, k_1^2\, k_2 + E(t)\, S(t)\, k_1\, k_{-1}\, k_2}{E(t)\, k_1\, k_{-1} + S(t)\, k_1\, k_{-1} + k_{-1}^2}, C(t) = \frac{E(t)\, S(t)\, k_1}{k_{-1}} \Bigg]$$

$$\left[k_{-1}, E(t)\, k_1\, k_{-1} + S(t)\, k_1\, k_{-1} + k_{-1}^2\right] \quad \textbf{(56)}$$

We are here only interested in the first returned differential ideal, as it describes the dynamics of the systems. The expression for $\frac{d}{dt} S(t)$ is almost the famous *Michaelis-Menten formula*. For really deriving it, we must do some games with the constants. We introduce as additional constants the initial values of our unknown functions and two new constants K and Vmax. These constants satisfy some relations: the first two are obvious from a chemical point of view, the latter two define the new constants. Furthermore, we take into account some conservation laws that can be easily derived from the stochiometric matrix of the given reaction. Again, we put the constants into the base field to reduce the number of cases to be considered.

```
> R := DifferentialRing(blocks = [F[1], [E, C, P, S], [k[1](), k[-1](), k[2](), C0(), E0(),
      P0(), S0(), K(), Vmax()]],
                        derivations = [t]) :
> with(Tools) :
> relations_among_params := PretendRegularDifferentialChain([P0 = 0, C0 = 0, K = k[-1]
      /k[1], Vmax = k[2]*E0], R);
```

$$relations\_among\_params := regular\_differential\_chain \quad \textbf{(57)}$$

```
> Field := field(generators = [k[1], k[-1], k[2], C0, E0, P0, S0, K, Vmax],
                 relations = relations_among_params) :
> conservation_laws := [E(t) + C(t) = E0 + C0, S(t) + C(t) + P(t) = S0 + C0 + P0];
```

$$conservation\_laws := [E(t) + C(t) = E0 + C0, S(t) + C(t) + P(t) = S0 + C0 + P0] \quad \textbf{(58)}$$

```
> G := RosenfeldGroebner([ op(sys), op(conservation_laws) ], R, basefield = Field);
```

$$G := [regular\_differential\_chain] \quad \textbf{(59)}$$

```
> Equations(G[1], solved); Inequations(G[1]);
```

$$\Bigg[F_1(t) = - \frac{-S(t)^2\, Vmax - S(t)\, K\, Vmax}{S(t)^2 + 2\, S(t)\, K + E0\, K + K^2}, \frac{d}{dt} S(t) = - \frac{S(t)^2\, Vmax + S(t)\, K\, Vmax}{S(t)^2 + 2\, S(t)\, K + E0\, K + K^2}, E(t)$$

$$= \frac{E0\, K}{S(t) + K}, C(t) = \frac{S(t)\, E0}{S(t) + K}, P(t) = - \frac{S(t)^2 + S(t)\, E0 - S(t)\, S0 + S(t)\, K - S0\, K}{S(t) + K}, k_1$$

$$= \frac{k_{-1}}{K}, k_2 = \frac{Vmax}{E0}, C0 = 0, P0 = 0\Bigg]$$

$$\left[E0, K, S(t) + K, S(t)^2 + 2\, S(t)\, K + E0\, K + K^2\right] \quad \textbf{(60)}$$

```
> mmf := op(Equations(G[1], solved, leader = diff(S(t), t)))
```

$$mmf := \frac{d}{dt} S(t) = - \frac{S(t)^2\, Vmax + S(t)\, K\, Vmax}{S(t)^2 + 2\, S(t)\, K + E0\, K + K^2} \quad \textbf{(61)}$$

This expression is almost the Michaelis-Menton formula. The exact formula is obtained by using one more approximation, namely the assumption that S0 is much larger than E0 (one does not need much enzyme for the reaction). This assumption implies that the term E0*K can be neglected which allows for further simplifications.

> $mmf := normal(subs(E0 \cdot K = 0, mmf))$

$$\frac{d}{dt} S(t) = -\frac{S(t)\ Vmax}{S(t) + K} \quad \textbf{(62)}$$

>