An Algorithm for Estimating Parameters from Noisy Data

Joint work by Francois Boulier Anja Korporal, Francois Lemaire, Adrien Poteaux, Wilfrid Perruquetti and Rosane Ushirobira

```
> restart;
  with (DifferentialAlgebra):
  with (Tools):
  Integrate := DifferentialAlgebra0:-Integrate:
  read "Integral.mpl":
```

The nonlinear differential system under study.
Two state variables x1(t), x2(t).
No command.
x1(t) is observed (a data file might be available)
We would like the best fitting values for the three parameters : k12, k21, Ve.

```
> syst := [ diff (x1(t),t) = - k12*x1(t) + k21*x2(t) - Ve*x1(t)/(1
  + x1(t)),
            diff (x2(t),t) =   k12*x1(t) - k21*x2(t),
                      y(t) =       x1(t) ];
```

$$syst := \left[\frac{d}{dt}\, x1(t) = -k12\, x1(t) + k21\, x2(t) - \frac{Ve\, x1(t)}{1 + x1(t)},\ \frac{d}{dt}\, x2(t) = k12\, x1(t) - k21\, x2(t),\ y(t) = x1(t) \right] \tag{1}$$

This system already forms a regular differential chain (or a characteristic set) wrt some natural ranking:

```
> R := DifferentialRing (derivations = [t], blocks = [y,[x1,x2],
  [k12(),k21(),Ve()],[theta1(),theta2(),theta3()]]);
```

$$R := differential\_ring \tag{2}$$

```
> ideal := RosenfeldGroebner (syst, R);
  ideal := ideal [1]:
```

$$ideal := [regular\_differential\_chain] \tag{3}$$

```
> Equations (ideal, solved);
```

$$\left[y(t) = x1(t),\ \frac{d}{dt}\, x1(t) = -\frac{k12\, x1(t)^2 - k21\, x2(t)\, x1(t) + k12\, x1(t) + Ve\, x1(t) - k21\, x2(t)}{1 + x1(t)},\ \frac{d}{dt}\, x2(t) = k12\, x1(t) - k21\, x2(t) \right] \tag{4}$$

We look for the input/output equation: an equation that binds the parameters to derivatives of the output.
We thus perform a change of ranking over the characteristic set (differential elimination).

```
> io_R := DifferentialRing (derivations = [t], blocks = [[x1,x2],y,
```

```
[k12(),k21(),Ve()],[theta1(),theta2(),theta3()]]);
```
$$io\_R := differential\_ring \tag{5}$$

```
> io_ideal := RosenfeldGroebner (ideal, io_R);
```
$$io\_ideal := regular\_differential\_chain \tag{6}$$

```
> Equations (io_ideal, leader = derivative (y(t)));
```
$$\left[\left(\frac{d^2}{dt^2} y(t)\right) y(t)^2 + 2 \left(\frac{d^2}{dt^2} y(t)\right) y(t) + \frac{d^2}{dt^2} y(t) + \left(\frac{d}{dt} y(t)\right) y(t)^2 k12 + \left(\frac{d}{dt} y(t)\right) y(t)^2 k21 + 2 \left(\frac{d}{dt} y(t)\right) y(t)\, k12 + 2 \left(\frac{d}{dt} y(t)\right) y(t)\, k21 + \left(\frac{d}{dt} y(t)\right) k12 + \left(\frac{d}{dt} y(t)\right) k21 + \left(\frac{d}{dt} y(t)\right) Ve + y(t)^2 k21\, Ve + y(t)\, k21\, Ve \right] \tag{7}$$

For legibility, let us introduce new parameters : theta1, theta2, theta3.

```
> relations_among_generators := RosenfeldGroebner (
      [ theta1=k21*Ve,
        theta2=k12+k21,
        theta3=k12+k21+Ve
      ], io_R, singsol=none) [1];
```
$$relations\_among\_generators := regular\_differential\_chain \tag{8}$$

We perform the same change of rankings as above.
The differential elimination is performed over some differential field, defined by generators and relations (those that rename parameters).

```
> F := field (relations = relations_among_generators):
  io_ideal := RosenfeldGroebner (syst, basefield = F, io_R):
  io_eq := Equations (io_ideal[1], notation=diff, leader =
  derivative (y(t))) [1];
```
$$io\_eq := \left(\frac{d^2}{dt^2} y(t)\right) y(t)^2 + 2 \left(\frac{d^2}{dt^2} y(t)\right) y(t) + \frac{d^2}{dt^2} y(t) + \left(\frac{d}{dt} y(t)\right) y(t)^2 \theta 2 + 2 \left(\frac{d}{dt} y(t)\right) y(t)\, \theta 2 + \left(\frac{d}{dt} y(t)\right) \theta 3 + y(t)^2 \theta 1 + y(t)\, \theta 1 \tag{9}$$

Here is the best we can by normalizing the leading coefficient and collecting coefficients.

```
> init_ioeq := Initial (io_eq, R):
  nio_eq := io_eq / init_ioeq:
  map (normal, collect (nio_eq, [theta1,theta2,theta3]));
```
$$\frac{y(t)\, \theta 1}{y(t) + 1} + \frac{\left(\frac{d}{dt} y(t)\right) y(t)\, (y(t) + 2)\, \theta 2}{y(t)^2 + 2\, y(t) + 1} + \frac{\left(\frac{d}{dt} y(t)\right) \theta 3}{y(t)^2 + 2\, y(t) + 1} + \frac{d^2}{dt^2} y(t) \tag{10}$$

Using the new algorithm by Francois Boulier, Francois Lemaire, Georg Regensburger and Markus Rosenkranz [ISSAC 2013] the d/dt operator can be factored.

```
> DifferentialEquation (nio_eq, y(t), io_R);
```

$$\frac{y(t)\,\theta 1}{y(t)+1}+\theta 2\left(\frac{\mathrm{d}}{\mathrm{d}t}\left(\frac{y(t)^2}{y(t)+1}\right)\right)-\theta 3\left(\frac{\mathrm{d}}{\mathrm{d}t}\left(\frac{1}{y(t)+1}\right)\right)+\frac{\mathrm{d}^2}{\mathrm{d}t^2}\,y(t) \qquad \textbf{(11)}$$

Integrating twice from a to t, one gets an integral equation that does not involve any derivative of y(t)

```
> nio_integral_eq := IteratedIntegral (nio_eq, y(t), a, R);
```

$$nio\_integral\_eq := \theta 1\left(\int_a^t\int_a^t \frac{y(t)}{y(t)+1}\,\mathrm{d}t\,\mathrm{d}t\right)+\theta 2\left(\int_a^t \frac{y(t)^2}{y(t)+1}\,\mathrm{d}t-\frac{y(a)^2\,(t-a)}{y(a)+1}\right) \qquad \textbf{(12)}$$

$$-\theta 3\left(\int_a^t \frac{1}{y(t)+1}\,\mathrm{d}t-\frac{t-a}{y(a)+1}\right)+y(t)-\left(\frac{\mathrm{d}}{\mathrm{d}t}\,y(t)\bigg|_{t=a}\right)(t-a)-y(a)$$

```
> simplify (diff (nio_integral_eq, t, t) - nio_eq);
```

$$0 \qquad \textbf{(13)}$$

The following linear system was obtained by a FORTRAN program.
The integral terms were evaluated over a data file, produced by numerical simulation using (theta1,theta2,theta3) = (15,6,3).
The fourth estimated parameter is y'(a).
There is no noise.

```
> with (LinearAlgebra):
```

theta1	theta2	theta3	y'(a)

```
> A :=
   <<0.4083644761      |-1.015167119      |-.7785369835E-01 |
  -1.062500000      >,
    <0.4199821783      |-1.044013043      |-.8054768885E-01 |
  -1.078125000      >,
    <0.4317449934      |-1.073217409      |-.8330065203E-01 |
  -1.093750000      >,
    <0.4436519943      |-1.102777846      |-.8611329960E-01 |
  -1.109375000      >,
    <0.4557022428      |-1.132691954      |-.8898634729E-01 |
  -1.125000000      >,
    <0.4678947897      |-1.162957303      |-.9192051454E-01 |
  -1.140625000      >,
    <0.4802286742      |-1.193571439      |-.9491652412E-01 |
  -1.156250000      >,
    <0.4927029244      |-1.224531873      |-.9797510178E-01 |
  -1.171875000      >>;
>  b := < 1.834618824, 1.857634424, 1.880499144, 1.903211175,
  1.925768696, 1.948169879, 1.970412890, 1.992495885 >;
```

$$A := \begin{bmatrix} 0.4083644761 & -1.015167119 & -0.07785369835 & -1.062500000 \\ 0.4199821783 & -1.044013043 & -0.08054768885 & -1.078125000 \\ 0.4317449934 & -1.073217409 & -0.08330065203 & -1.093750000 \\ 0.4436519943 & -1.102777846 & -0.08611329960 & -1.109375000 \\ 0.4557022428 & -1.132691954 & -0.08898634729 & -1.125000000 \\ 0.4678947897 & -1.162957303 & -0.09192051454 & -1.140625000 \\ 0.4802286742 & -1.193571439 & -0.09491652412 & -1.156250000 \\ 0.4927029244 & -1.224531873 & -0.09797510178 & -1.171875000 \end{bmatrix}$$

$$b := \begin{bmatrix} 1.834618824 \\ 1.857634424 \\ 1.880499144 \\ 1.903211175 \\ 1.925768696 \\ 1.948169879 \\ 1.970412890 \\ 1.992495885 \end{bmatrix} \quad \mathbf{(14)}$$

Not so bad.

```
> <theta1, theta2, theta3, Eval (diff(y(t),t),t=a)> = LeastSquares
  (A, b);
```

$$\begin{bmatrix} \theta 1 \\ \theta 2 \\ \theta 3 \\ \left. \frac{\mathrm{d}}{\mathrm{d}t} y(t) \right|_{t=a} \end{bmatrix} = \begin{bmatrix} 14.7545202797175 \\ 5.90181419824832 \\ 2.99272815933586 \\ -1.91408899045239 \end{bmatrix} \quad \mathbf{(15)}$$