GPoM : 7 Retro-modelling

Sylvain Mangiarotti & Mireille Huc

2020-02-18

Detection of miscellaneous chaotic systems

The question underlying the expression retro-modelling is the following: is it possible to retrieve the original formulation of a given dynamical system from a set of observed time series generated by this system (and without any extra information)? To answer this question, the Rössler system[1] was extensively used in the previous vignettes, in particular in vignettes `3 Modelling` and `6 Sensitivity`, to show that the global modelling is very efficient to retrieve this system under various conditions of noise, subsampling, short length time series, etc. In the present vignette, other three-dimensional chaotic systems are considered in order to check the generality of the technique. Seventeen other systems are considered in the present vignette which equations can be loaded as follows:

```
  data("allMod_nVar3_dMax2")
  names(allMod_nVar3_dMax2)
```

```
##  [1] "L63"      "R76"      "BS81"     "L84"      "NH86"     "GT92"
##  [7] "SprF"     "SprH"     "SprK"     "SprO"     "SprP"     "SprG"
## [13] "SprM"     "SprQ"     "SprS"     "CS2004"   "Li2007"   "Cord2012"
```

These systems were used to produce time series. These time series can also be directly loaded as follows:

```
  data("TSallMod_nVar3_dMax2")
```

In the following experiments, these time series are used as an input of the `GPoM` package to check if the sets of original equations can be retrieved.

The Nosé-Hoover-1986 system

The Nosé-Hoover system was discovered by Shūichi Nosé[2] and William Hoover[3] in 1986 (it was rediscovered by Sprott in 1994, known as Sprott-A). This system reads

```
visuEq(K = allMod_nVar3_dMax2$NH86,  substit = 1)
```

```
## dx/dt = 0.2 y
##
## dy/dt = 1 y z  -1 x
##
## dz/dt = 3  -1 y^2
```

[1] O. Rössler, An Equation for Continuous Chaos, *Physics Letters*, **57A**(5), 1976, 397-398.

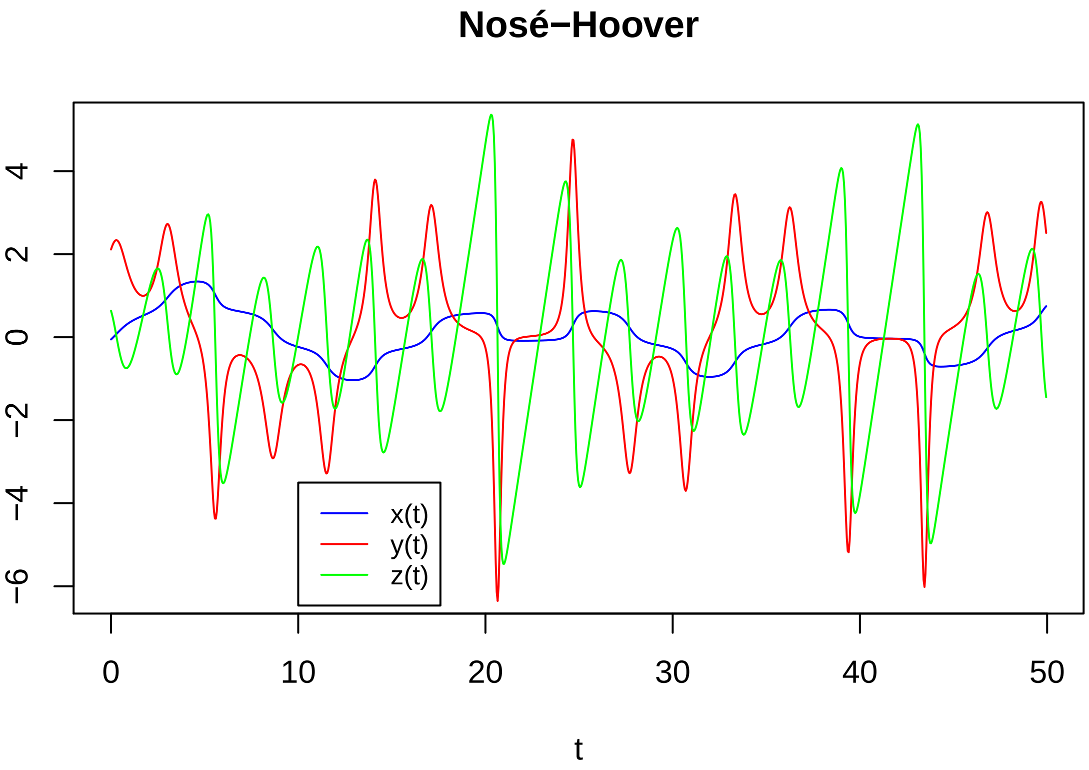
[2] S. Nosé, A molecular dynamics method for simulations in the canonical ensemble, *Molecular Physics*, **52** (2), 255-268, 1984.

[3] W. G. Hoover, Nonlinear conductivity and entropy in the two-body Boltzmann gas, *Journal of Statistical Physics*, **42**(3/4), 587-600, 1986.

Data

This system is particularly interesting because, despite their deterministic couplings, the three variables are characterized by quite different signals:

```
TS <- TSallMod_nVar3_dMax2$NH86$reconstr
plot(TS[,1], TS[,2], type = 'l', ylim = c(-6.2,5.2), xlab = 't', ylab = '',
     main = 'Nosé-Hoover', col = 'blue')
lines(TS[,1], TS[,3], type = 'l',  col = 'red')
lines(TS[,1], TS[,4], type = 'l',  col = 'green')
legend(10,-3.5, c("x(t)", "y(t)", "z(t)"), col=c('blue', 'red', 'green'), lty=1, cex = 0.8)
```



And the three variables are almost completely decorrelated.

```
# Compute the correlation
  cor(TS[,2:4])
```

Indeed, correlation does not exceed ~0.142 (the maximum correlation is observed between x and z), and it is almost zero between x and y (~0.025) and between y and z (~0.016). Obviously, for such a type of dynamic, the correlation may be particulary misleading information for whom would like to detect the existence of a causal coupling.

Global modelling

The `GPoM` package [4], [5] is used to investigate the relations existing between the three time series $x(t)$, $y(t)$ and $z(t)$ shown above. A drastic model selection is performed by the `gPoMo` function when trying to obtain

[4] S. Mangiarotti, 2015. Low dimensional chaotic models for the plague epidemic in Bombay, *Chaos, Solitons & Fractals*, **81**(A), 184-196.

[5] S. Mangiarotti, M. Peyre & M. Huc, 2016. A chaotic model for the epidemic of Ebola virus disease in West Africa (2013–2016). *Chaos*, **26**, 113112.

a global model from a set of observational time series. The numerical integration is then used to obtain an optimal model among the few models preselected by the function:

```
# modelling from time series starting
outNH86 <- gPoMo(data = TS[,2:4], tin = TS[,1], dMax = 2, nS = c(1,1,1),
              show = 1,
              IstepMin = 10, IstepMax = 15000, nPmin = 3, nPmax = 5,
              method = 'rk4')
```

With the only exception of model #6, the preselected models are either diverging (models #1, #2, #4, #5, #7 and #10) or converging to a fixed point (#3, #8 and #9). Equations for model #6 effectively correspond to the original system:

```
visuEq(K = outNH86$models$model6,  substit = 1, approx = 2)
```

```
## dx/dt = 0.196 y
##
## dy/dt = 0.854 y z  -1.079 x
##
## dz/dt = 2.744  -0.916 y^2
```

Therefore, despite the absence of correlation observed between the three variables, the proper formulation is directly and completely retrieved. Note that allowing models of larger size (e.g. with `nPmax = 6`), three other models of nontrivial solution can be obtained (models #12, #14 and #15). The model #6 should be preferred from these for its more parsimonious formulation.

The results obtained for this system are especially interesting since providing an obvious evidence that correlation is not a good tool for the detection of linkages between dynamical variables. Contrarily, the retro-modelling technique enables not only to detect the couplings. Furthermore, it goes far beyound, by making it possible to retrieve the original and complete formulation of the coupling between the three variables.

The Genesio-Tesi system (1992)

The following differential equations was introduced by Roberto Genesio and Alberto Tesi in 1992[6]

$d^3x/dt^3 + a.d^2x/dt^2 + b.dx/dt + x(1+x) = 0,$

(for a = 0.446 and b = 1.1) which can be rewritten in the canonical form:

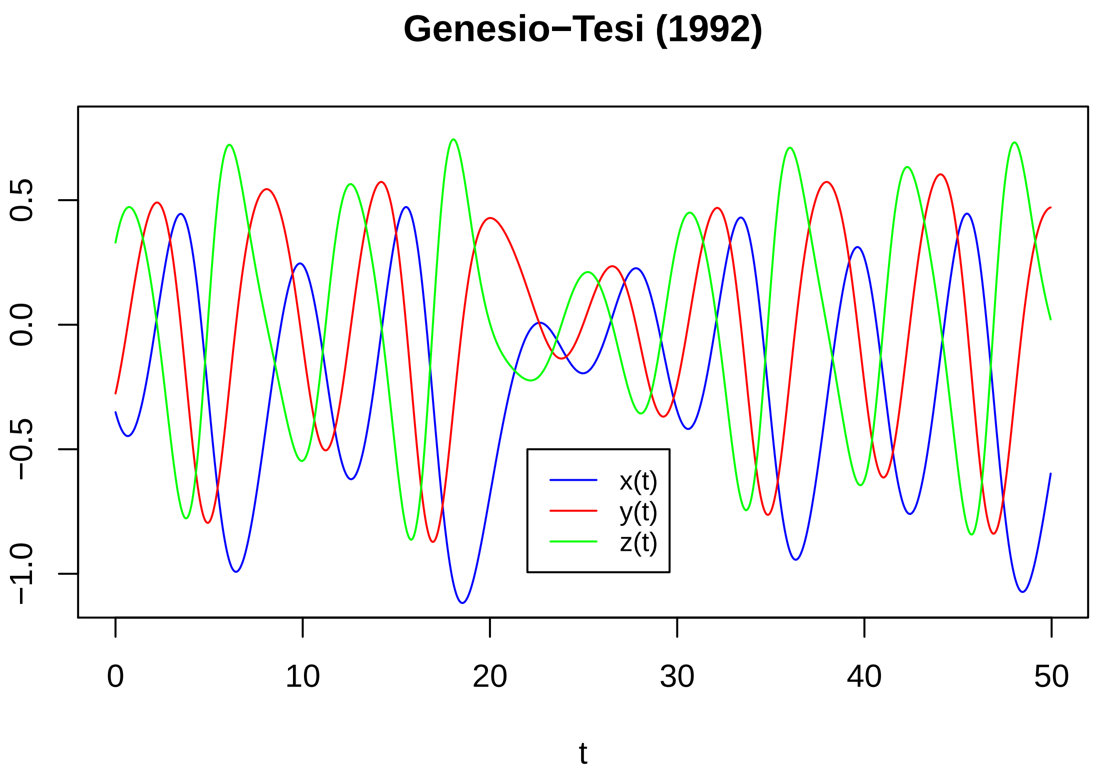
```
visuEq(K = allMod_nVar3_dMax2$GT92,  substit = 1)
```

```
## dx/dt = 1 y
##
## dy/dt = 1 z
##
## dz/dt = -0.446 z  -1.1 y  -1 x  -1 x^2
```

The following set of time series $x(t)$, $y(t)$ and $z(t)$ was produced from this system.

```
  TS <- TSallMod_nVar3_dMax2$GT92$reconstr
  plot(TS[,1], TS[,2], type = 'l', ylim = c(-1.1, 0.8), xlab = 't', ylab = '',
       main= 'Genesio-Tesi (1992)', col = 'blue')
  lines(TS[,1], TS[,3], type = 'l', col = 'red')
  lines(TS[,1], TS[,4], type = 'l', col = 'green')
  legend(22,-0.5, c("x(t)", "y(t)", "z(t)"), col=c('blue', 'red', 'green'), lty=1, cex = 0.8)
```

[6] R. Genesio & A. Tesi, Harmonic balance methods for the analysis of chaotic dynamics in nonlinear systems, *Automatica*, **28**(3), 531-548, 1992.



And the global modelling technique was then applied to this set of time series:

```
# modelling from time series starting
outGT92 <- gPoMo(data = TS[,2:4], tin = TS[,1], dMax = 2, nS = c(1,1,1),
                 show = 1,
                 IstepMin = 10, IstepMax = 15000, nPmin = 6, nPmax = 6,
                 method = 'rk4')
```

Performing the same analysis as for the system Nosé-Hoover (see upper in the text), the formulation of the original system can be directly retrieved (model #1):

```
visuEq(K = outGT92$models$model1,  substit = 1, approx = 2)
```

```
## dx/dt = 0.995 y
##
## dy/dt = 0.994 z
##
## dz/dt = -0.437 z  -1.093 y  -0.98 x  -0.981 x^2
```

Note that obtained models of smaller size have period-1 cycles (these can be obtained by setting `nPmin = 3`). The complexity of the observed signal is thus not reproduced by these models and these can be rejected. Allowing models of larger size (nPmax = 7), interesting solutions will be otbained (in terms of phase portrait), but their formulation will be less concise and the previous formulation obtained with model #1 should thus be preferred.

The Sprott systems

Numerous systems were discovered (some rediscovered) by Julian Clinton Sprott in the 1990s[7]. Several of these are considered in this section, namely systems Sprott-F, H, K, O, P, G, M, Q and S.

[7] J. C. Sprott, Some simple chaotic flows, *Physical Review E*, **50**(2), 647-650, 1994.

The Spott-F system

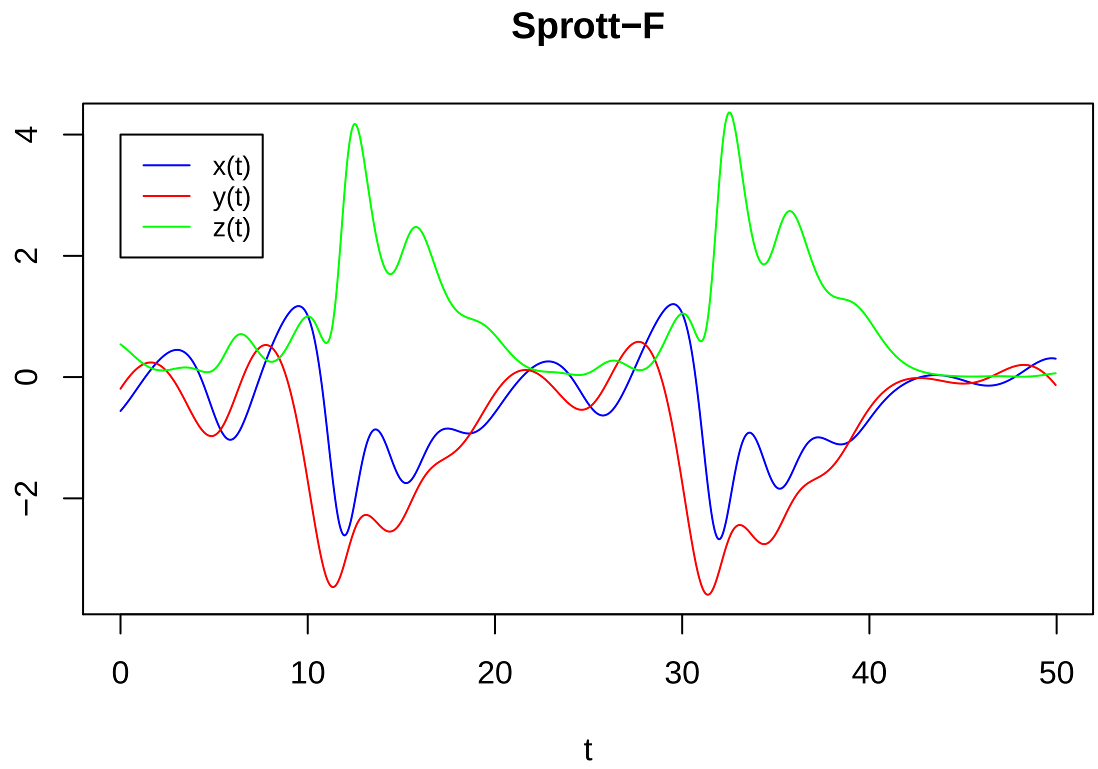
The Sprott-F system reads:

```
visuEq(K = allMod_nVar3_dMax2$SprF,  substit = 1)
```

```
## dx/dt = 1 z  + 1 y
##
## dy/dt = 0.5 y  -1 x
##
## dz/dt = -1 z  + 1 x^2
```

A specific attention was paid uuper to the Nosé-Hoover system for its very low correlation. The Sprott-F system is interesting for the opposite reason: its variables are characterized by a high level of correlation (or anticorrelation) as shown by the following plots $x(t)$, $y(t)$ and $z(t)$:

```
  TS <- TSallMod_nVar3_dMax2$SprF$reconstr
  plot(TS[,1], TS[,2], type = 'l', ylim = c(-3.6,4.2),
       xlab = 't', ylab = '', main = 'Sprott-F', col = 'blue')
  lines(TS[,1], TS[,3], type = 'l', col = 'red')
  lines(TS[,1], TS[,4], type = 'l', col = 'green')
  legend(0,4, c("x(t)", "y(t)", "z(t)"), col=c('blue', 'red', 'green'), lty=1, cex = 0.8)
```



Correlation is high between the three variables: ~0.68 between $x(t)$ and $y(t)$, ~0.74 between $x(t)$ and $-z(t)$, and ~0.77 between $y(t)$ and $-z(t)$:

```
  # Compute the correlation
    cor(TS[,2:4])
```

The global modelling technique was applied to this data set using the `gPoMo` function:

```
# modelling from time series starting
outSprF <- gPoMo(data = TS[,2:4], tin = TS[,1], dMax = 2, nS = c(1,1,1),
              show = 1, IstepMin = 10, IstepMax = 15000, nPmin = 6, nPmax = 6,
              method = 'rk4')
```

Note that allowing a larger range of model size (`nPmin = 3` and `nPmax = 7`), models solutions can be rejected either because they are diverging or oversimplified (models #3-14, #16-24, #26-27 and #30-35), or because they are less parsimonious (models #15, #25, #28 and #29).

The formulation of the original system is thus effectively retrieved:

```
visuEq(K = outSprF$models$model5,  substit = 1, approx = 2)
```

```
## dx/dt = 0.994538 y  -0.00092 x^2
##
## dy/dt = 0.994299 z  -0.005833 x y
##
## dz/dt = -1.108611 y  -0.18608 x^2
```

Similarly, results are found to be straightforward for all the Sprott systems.

The Spott-H system

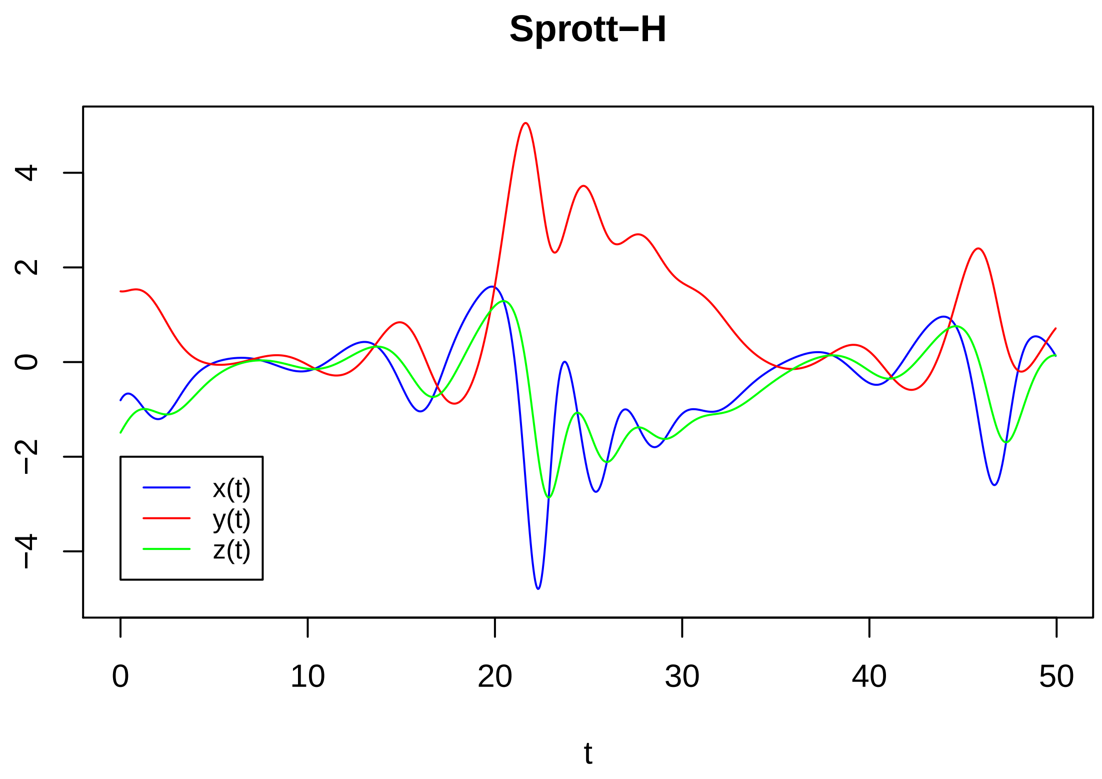
The Sprott-H system reads

```
visuEq(K = allMod_nVar3_dMax2$SprH,  substit = 1)
```

```
## dx/dt = 1 z^2  -1 y
##
## dy/dt = 0.5 y  + 1 x
##
## dz/dt = -1 z  + 1 x
```

Although less marked, the behavior of this system is more or less similar to Sprott-F in terms of correlation (or anticorrelation).

```
  TS <- TSallMod_nVar3_dMax2$SprH$reconstr
  plot(TS[,1], TS[,2], type = 'l', ylim = c(-5,5),
       xlab = 't', ylab = '', main = 'Sprott-H', col = 'blue')
  lines(TS[,1], TS[,3], type = 'l',  col = 'red')
  lines(TS[,1], TS[,4], type = 'l',  col = 'green')
  legend(0, -2, c("x(t)", "y(t)", "z(t)"), col=c('blue', 'red', 'green'), lty=1, cex = 0.8)
```



Correlation is high between $x(t)$ and $z(t)$ (~0.77), and $x(t)$ and $-y(t)$ (~0.63), and lower between $y(t)$ and $z(t)$ (~0.38):

```
  # Compute the correlation
    cor(TS[,2:4])
```

The global modelling technique was applied to this data set:

```
# modelling from time series starting
outSprH <- gPoMo(data = TS[,2:4], tin = TS[,1], dMax = 2, nS = c(1,1,1),
              show = 1,
              IstepMin = 10, IstepMax = 15000, nPmin = 6, nPmax = 6,
              method = 'rk4')
```

The formulation of the original system is directly retrieved:

```
visuEq(K = outSprH$models$model5,  substit = 1, approx = 2)
```

```
## dx/dt = 0.983 z^2  -0.984 y
##
## dy/dt = 0.495 y  + 0.991 x
##
## dz/dt = -0.989 z  + 0.989 x
```

No simpler formulation could enable to reproduce the observed dynamic.

The Spott-K system

The Sprott-K system reads

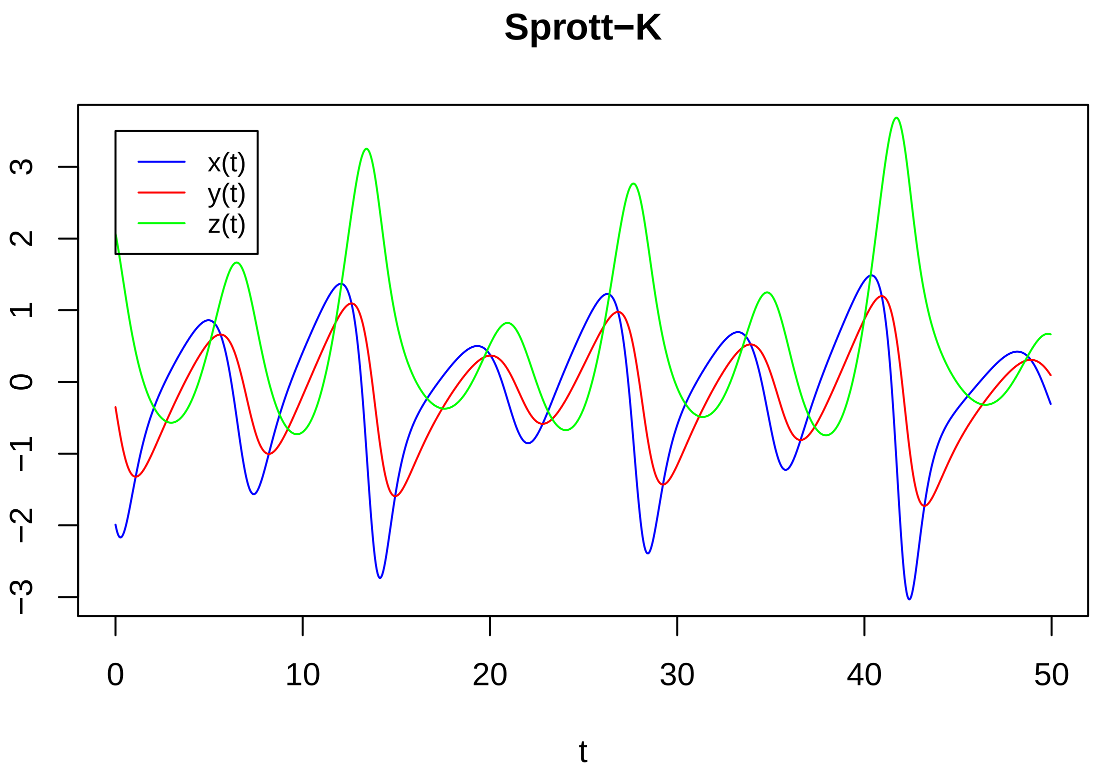
```
visuEq(K = allMod_nVar3_dMax2$SprK,  substit = 1)
```

```
## dx/dt = -1 z  + 1 x y
```

```
##
## dy/dt = -1 y  + 1 x
##
## dz/dt = 0.3 z  + 1 x
```

The following data set was used:

```
TS <- TSallMod_nVar3_dMax2$SprK$reconstr
plot(TS[,1], TS[,2], type = 'l', ylim = c(-3,3.6), xlab = 't', ylab= '',
     main = 'Sprott-K', col = 'blue')
lines(TS[,1], TS[,3], type = 'l', col = 'red')
lines(TS[,1], TS[,4], type = 'l',  col = 'green')
legend(0,3.5, c("x(t)", "y(t)", "z(t)"), col=c('blue', 'red', 'green'), lty=1, cex = 0.8)
```



and the global modelling technique was applied.

```
# modelling from time series starting
outSprK <- gPoMo(data = TS[,2:4], tin = TS[,1], dMax = 2, nS = c(1,1,1),show = 1,
              IstepMin = 10, IstepMax = 15000, nPmin = 6, nPmax = 6, method = 'rk4')
```

The formulation of the original system is directly retrieved:

```
visuEq(K = outSprK$models$model5,  substit = 1, approx = 2)
```

```
## dx/dt = -0.989 z  + 0.981 x y
##
## dy/dt = -0.992 y  + 0.992 x
##
## dz/dt = 0.298 z  + 0.994 x
```

and no simpler formulation enabling to reproduce the observed dynamic was obtained.

The Spott-O system

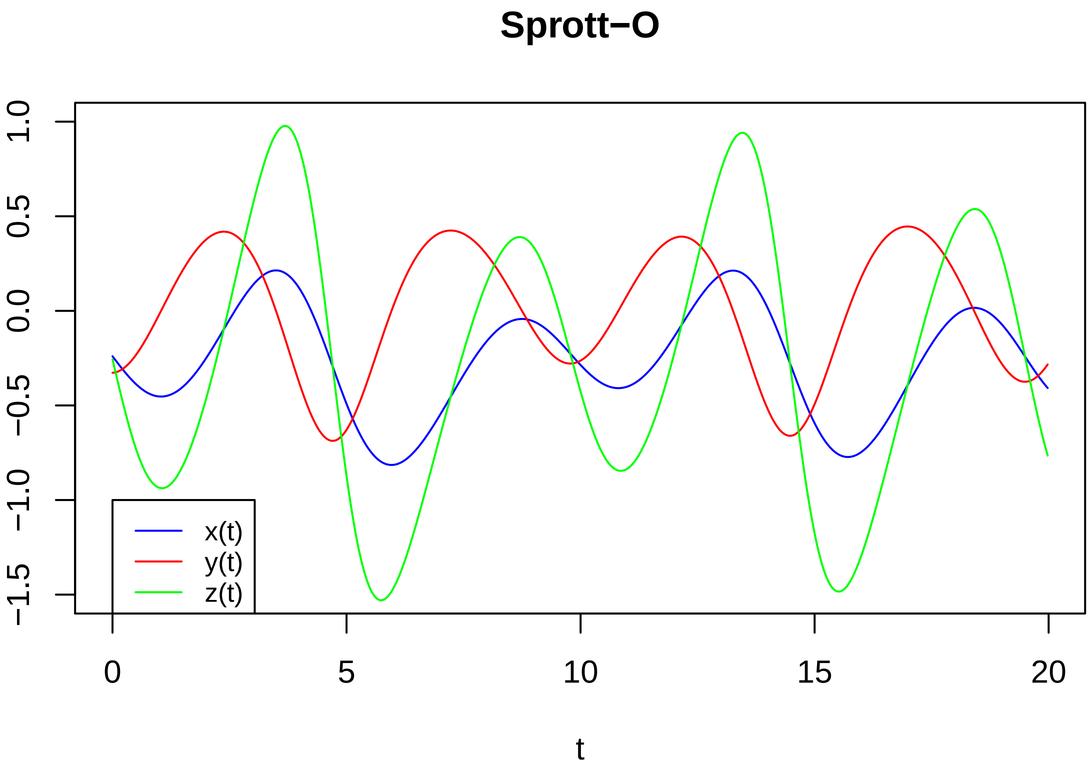
The Sprott-O system reads

```
visuEq(K = allMod_nVar3_dMax2$SprO,  substit = 1)
```

```
## dx/dt = 1 y
##
## dy/dt = -1 z  + 1 x
##
## dz/dt = 2.7 y  + 1 x  + 1 x z
```

The following data set was used:

```
  TS <- TSallMod_nVar3_dMax2$SprO$reconstr
  plot(TS[,1], TS[,2], type = 'l', ylim = c(-1.5,1), main = 'Sprott-O',
       xlab = 't', ylab='', col = 'blue')
  lines(TS[,1], TS[,3], type = 'l', col = 'red')
  lines(TS[,1], TS[,4], type = 'l', col = 'green')
  legend(0,-1, c("x(t)", "y(t)", "z(t)"), col=c('blue', 'red', 'green'), lty=1, cex = 0.8)
```



The global modelling technique was applied.

```
# modelling from time series starting
outSprO <- gPoMo(data = TS[,2:4], tin = TS[,1], dMax = 2, nS = c(1,1,1),
              show = 1,
              IstepMin = 10, IstepMax = 15000, nPmin = 6, nPmax = 6,
              method = 'rk4')
```

The formulation of the original was directly retrieved:

```
visuEq(K = outSprO$models$model2,  substit = 1, approx = 2)
```

```
## dx/dt = 0.999 y
##
## dy/dt = -0.998 z  + 0.998 x
##
## dz/dt = 2.696 y  + 0.995 x  + 0.995 x z
```

and no simpler formulation enabling to reproduce the observed dynamic was obtained.

The Spott-P system

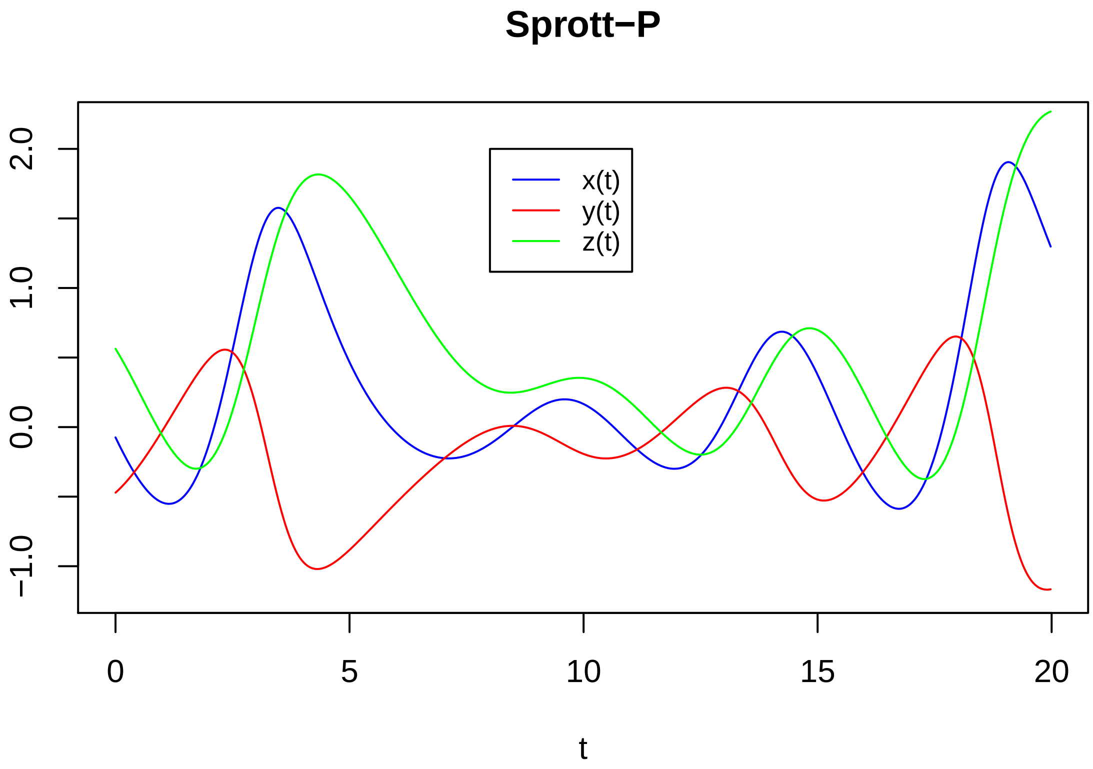
The Sprott-P system reads

```
visuEq(K = allMod_nVar3_dMax2$SprP,  substit = 1)
```

```
## dx/dt = 1 z  + 2.68 y
##
## dy/dt = 1 y^2  -1 x
##
## dz/dt = 1 y  + 1 x
```

The following data set was used:

```
  TS <- TSallMod_nVar3_dMax2$SprP$reconstr
  plot(TS[,1], TS[,2], type = 'l', ylim = c(-1.2,2.2), main='Sprott-P',
       xlab = 't', ylab = '', col = 'blue')
  lines(TS[,1], TS[,3], type = 'l', col = 'red')
  lines(TS[,1], TS[,4], type = 'l', col = 'green')
  legend(8,2, c("x(t)", "y(t)", "z(t)"), col=c('blue', 'red', 'green'), lty=1, cex = 0.8)
```



The global modelling technique was applied.

```
# modelling from time series starting
outSprP<- gPoMo(data = TS[,2:4], tin = TS[,1], dMax = 2, nS = c(1,1,1), show = 1,
                IstepMin = 10, IstepMax = 15000, nPmin = 6, nPmax = 6, method = 'rk4')
```

The formulation of the original system is directly retrieved:

```
visuEq(K = outSprP$models$model5,  substit = 1, approx = 2)
```

```
## dx/dt = 0.998 z  + 2.676 y
##
## dy/dt = 0.996 y^2  -0.998 x
##
## dz/dt = 0.999 y  + 0.999 x
```

No simpler formulation enabling to reproduce the observed dynamic was obtained.

The Spott-G system

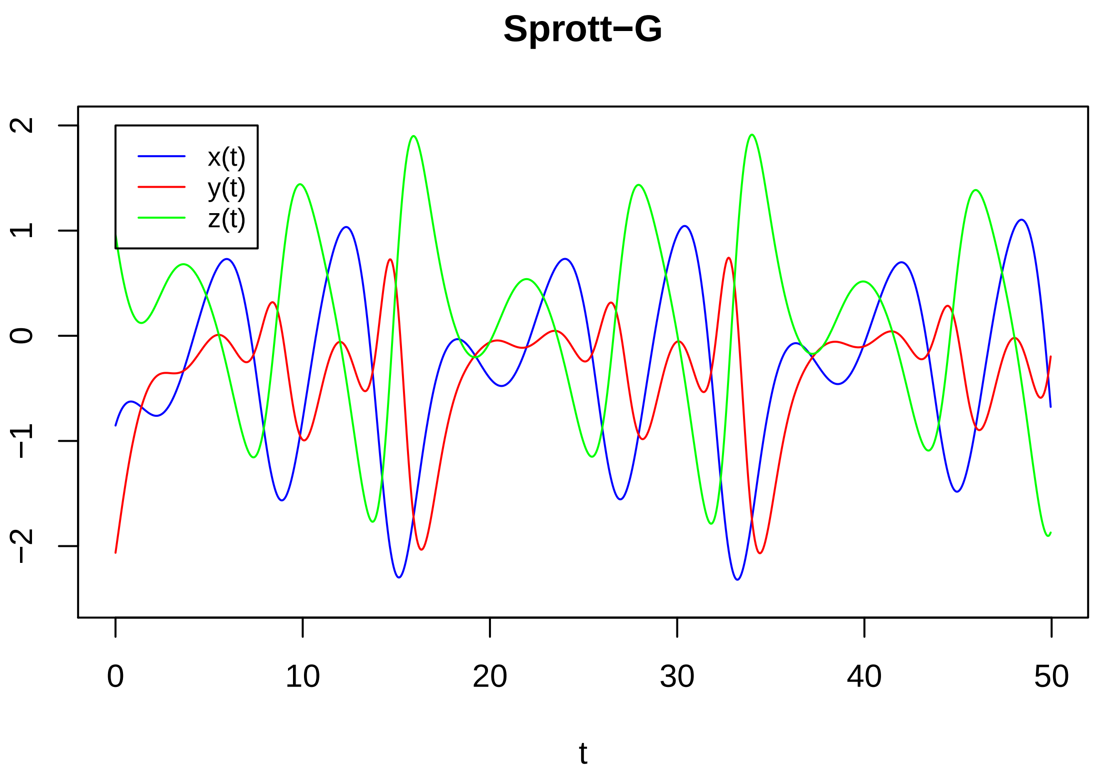
The Sprott-G system reads

```
visuEq(K = allMod_nVar3_dMax2$SprG,  substit = 1)
```

```
## dx/dt = 1 z  + 0.4 x
##
## dy/dt = -1 y  + 1 x z
##
## dz/dt = 1 y  -1 x
```

The following data set was used:

```
  TS <- TSallMod_nVar3_dMax2$SprG$reconstr
  plot(TS[,1], TS[,2], type = 'l', ylim = c(-2.5,2),
       xlab = 't', ylab = '', main='Sprott-G', col = 'blue')
  lines(TS[,1], TS[,3], type = 'l', col = 'red')
  lines(TS[,1], TS[,4], type = 'l', col = 'green')
  legend(0, 2 , c("x(t)", "y(t)", "z(t)"), col=c('blue', 'red', 'green'), lty=1, cex = 0.8)
```



The global modelling technique was then applied.

```
# modelling from time series starting
outSprG <- gPoMo(data = TS[,2:4], tin = TS[,1], dMax = 2, nS = c(1,1,1),
                 show = 1, IstepMin = 10, IstepMax = 15000, nPmin = 3, nPmax = 6, method = 'rk4')
```

The formulation of the original system was directly retrieved:

```
visuEq(K = outSprG$models$model5,  substit = 1, approx = 2)
```

```
## dx/dt = 0.83034 z
##
## dy/dt = 0.56311 x z
##
## dz/dt = 0.00385 z^2  + 0.99406 y  -0.99167 x
```

No simpler formulation enabling to reproduce the observed dynamic was obtained.

The Spott-M system

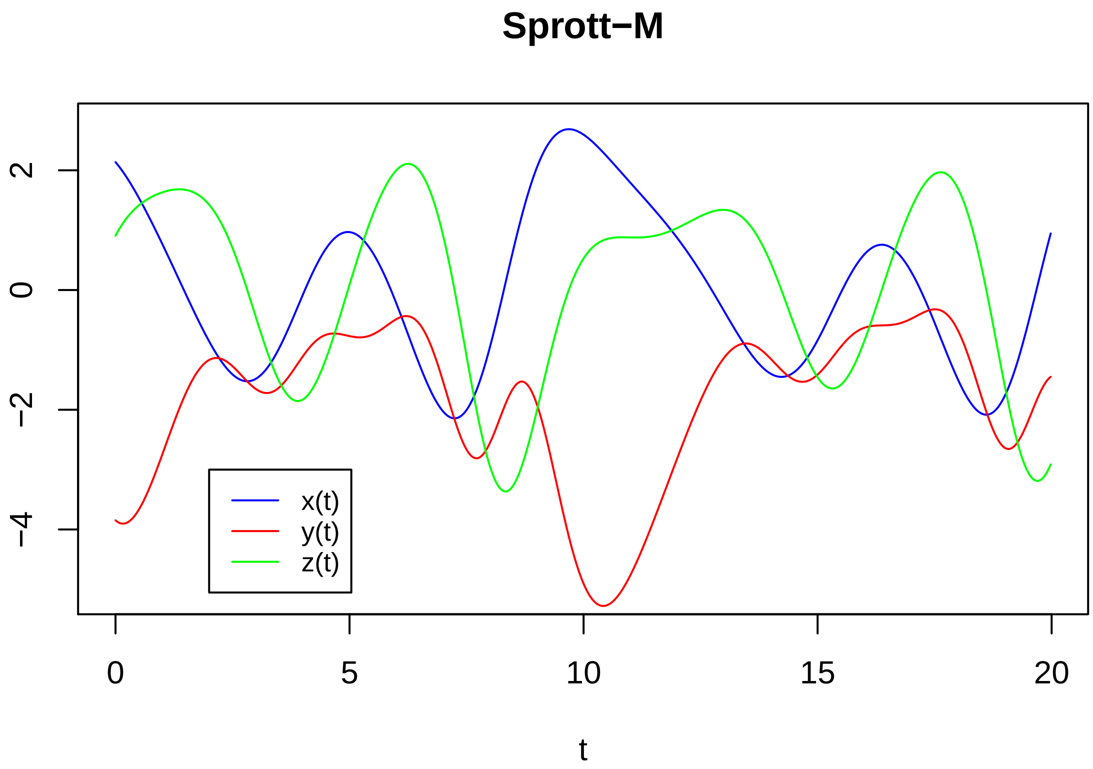
The Sprott-M system reads

```
visuEq(K = allMod_nVar3_dMax2$SprM,  substit = 1)
```

```
## dx/dt = -1 z
##
## dy/dt = -1 y  -1 x^2
##
## dz/dt = 1.7  + 1 y  + 1.7 x
```

The following data set was used:

```
TS <- TSallMod_nVar3_dMax2$SprM$reconstr
plot(TS[,1], TS[,2], type = 'l', ylim = c(-5.1,2.8), main='Sprott-M',
     xlab = 't', ylab = '', col = 'blue')
lines(TS[,1], TS[,3], type = 'l', col = 'red')
lines(TS[,1], TS[,4], type = 'l', col = 'green')
legend(2, -3, c("x(t)", "y(t)", "z(t)"), col=c('blue', 'red', 'green'), lty=1, cex = 0.8)
```



The global modelling technique was then applied.

```
# modelling from time series starting
outSprM <- gPoMo(data = TS[,2:4], tin = TS[,1], dMax = 2, nS = c(1,1,1),
              show = 1,
              IstepMin = 10, IstepMax = 15000, nPmin = 6, nPmax = 6,
              method = 'rk4')
```

The formulation of the original system was directly retrieved:

```
visuEq(K = outSprM$models$model2,  substit = 1, approx = 2)
```

```
## dx/dt = -0.999 z
##
## dy/dt = -0.997 y  -0.997 x^2
##
## dz/dt = 1.695  + 0.997 y  + 1.697 x
```

No simpler formulation enabling to reproduce the observed dynamic was obtained.

The Spott-Q system

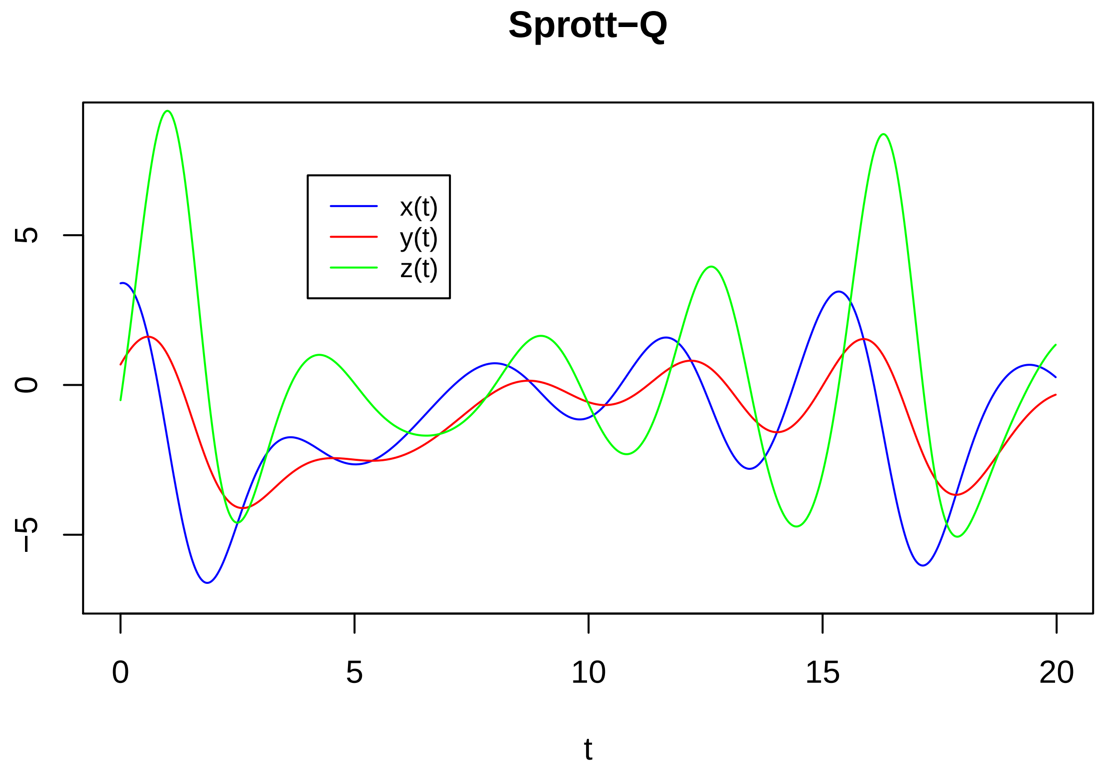
The Sprott-Q system reads

```
visuEq(K = allMod_nVar3_dMax2$SprQ,  substit = 1)
```

```
## dx/dt = -1 z
##
## dy/dt = -1 y  + 1 x
##
## dz/dt = 0.5 z  + 1 y^2  + 3.1 x
```

The following data set was used:

```
  TS <- TSallMod_nVar3_dMax2$SprQ$reconstr
  plot(TS[,1], TS[,2], type = 'l', ylim = c(-7,8.8), main = 'Sprott-Q',
       xlab = 't', ylab = '', col = 'blue')
  lines(TS[,1], TS[,3], type = 'l', col = 'red')
  lines(TS[,1], TS[,4], type = 'l', col = 'green')
  legend(4, 7, c("x(t)", "y(t)", "z(t)"), col=c('blue', 'red', 'green'), lty=1, cex = 0.8)
```



The global modelling technique was applied.

```
# modelling from time series starting
outSprQ <- gPoMo(data = TS[,2:4], tin = TS[,1], dMax = 2, nS = c(1,1,1), show = 1,
                 IstepMin = 10, IstepMax = 15000, nPmin = 6, nPmax = 6, method = 'rk4')
```

The formulation of the original is directly retrieved:

```
visuEq(K = outSprQ$models$model2,  substit = 1, approx = 2)
```

```
## dx/dt = -0.998 z
##
## dy/dt = -0.998 y  + 0.998 x
##
## dz/dt = 0.498 z  + 0.996 y^2  + 3.09 x
```

No simpler formulation enabling to reproduce the observed dynamic was obtained.

The Spott-S system

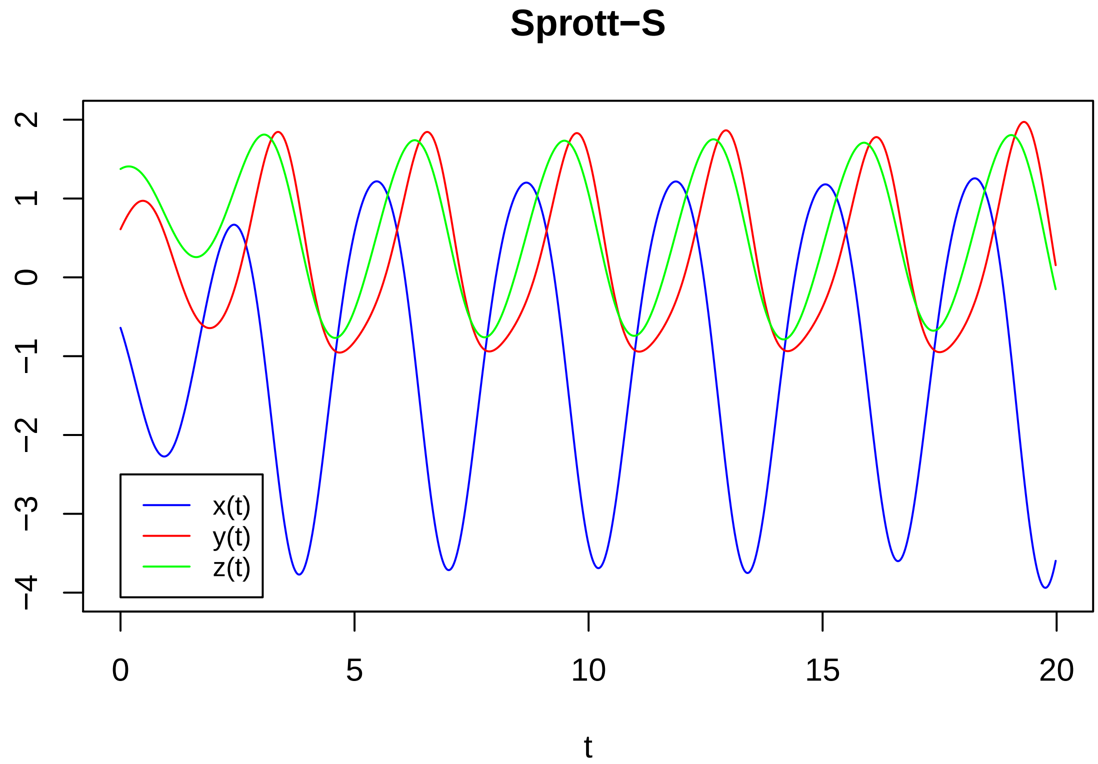
The Sprott-S system reads

```
visuEq(K = allMod_nVar3_dMax2$SprS,  substit = 1)
```

```
## dx/dt = -4 y  -1 x
##
## dy/dt = 1 z^2  + 1 x
##
## dz/dt = 1  + 1 x
```

The following data set was used:

```
  TS <- TSallMod_nVar3_dMax2$SprS$reconstr
  plot(TS[,1], TS[,2], type = 'l', ylim = c(-4,2), main = 'Sprott-S',
       xlab = 't', ylab = '', col = 'blue')
  lines(TS[,1], TS[,3], type = 'l', col = 'red')
  lines(TS[,1], TS[,4], type = 'l', col = 'green')
  legend(0,-2.5, c("x(t)", "y(t)", "z(t)"), col=c('blue', 'red', 'green'), lty=1, cex = 0.8)
```



The global modelling technique was applied.

```
# modelling from time series starting
outSprS<- gPoMo(data = TS[,2:4], tin = TS[,1], dMax = 2, nS = c(1,1,1), show = 1,
                IstepMin = 10, IstepMax = 15000, nPmin = 6, nPmax = 6, method = 'rk4')
```

The formulation of the original system is directly retrieved:

```r
visuEq(K = outSprS$models$model5,  substit = 1, approx = 2)
```

```
## dx/dt = -3.986 y  -0.997 x
##
## dy/dt = 0.995 z^2  + 0.996 x
##
## dz/dt = 0.997  + 0.997 x
```

No simpler formulation enabling to reproduce the observed dynamic was obtained.

The Lorenz-1963 system

The Lorenz system[8] discovered by Edouard N. Lorenz in 1963 is now considered. This system reads

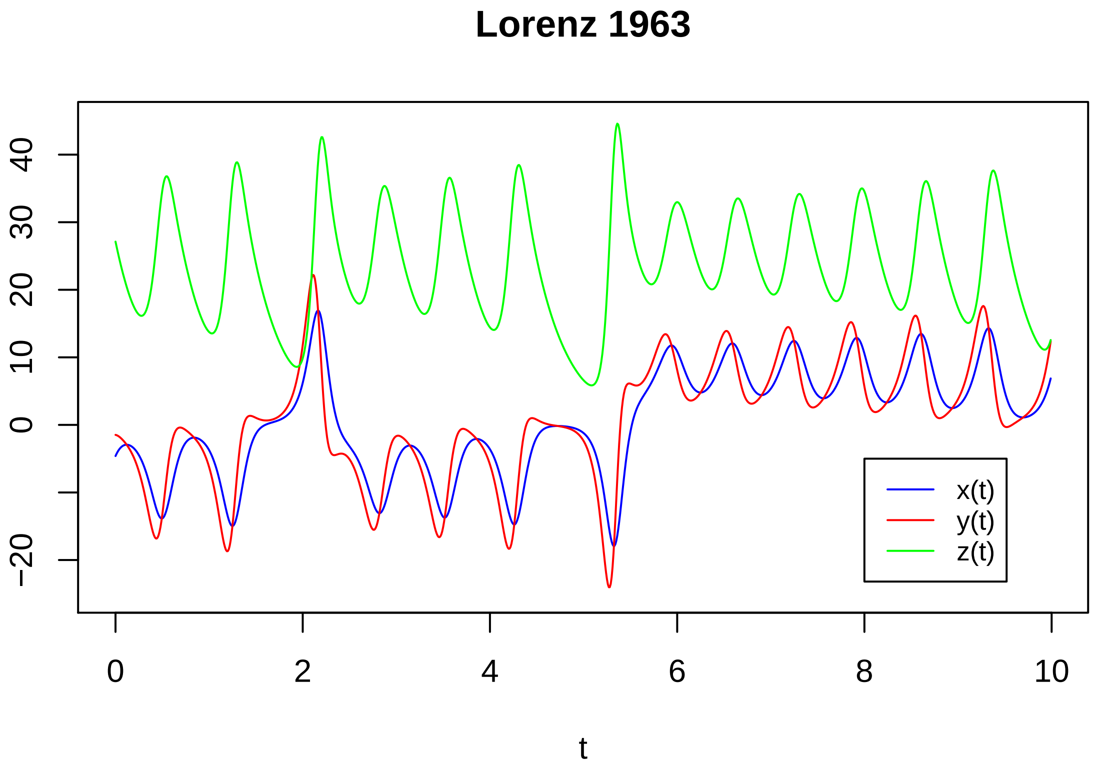
```r
  visuEq(K = allMod_nVar3_dMax2$L63,  substit = 1, approx = 4)
```

```
## dx/dt = 10 y  -10 x
##
## dy/dt = -1 y  + 28 x  -1 x z
##
## dz/dt = -2.6667 z  + 1 x y
```

The following data set was considered for the modelling:

```r
  TS <- TSallMod_nVar3_dMax2$L63$reconstr
  plot(TS[,1], TS[,2], type = 'l', ylim = c(-25,45), xlab = 't', ylab = '',
       main = 'Lorenz 1963', col = 'blue')
  lines(TS[,1], TS[,3], type = 'l', col = 'red')
  lines(TS[,1], TS[,4], type = 'l', col = 'green')
  legend(8, -5, c("x(t)", "y(t)", "z(t)"), col=c('blue', 'red', 'green'), lty=1, cex = 0.8)
```

[8] E. N. Lorenz, Deterministic non-periodic flow, *Journal of the Atmospheric Sciences*, **20**(2), 130–141 (1963).



The global modelling technique was applied to this set of series with the hope to retrieve the original equations.

```
# modelling from time series starting
outL63 <- gPoMo(data = TS[,2:4], tin = TS[,1], dMax = 2, nS = c(1,1,1), show = 1,
                IstepMin = 10, IstepMax = 1500, nPmin = 6, nPmax = 7, method = 'rk4')
```

Surprisingly, the simplest model able to reproduce the original phase portrait has six terms, that is, one less term than the original model: the term $-y$ is missing in the second equation. This obtained morel reads

```
visuEq(K = outL63$models$model5,  substit = 1, approx = 2)
```

```
## dx/dt = 9.748 y  -9.748 x
##
## dy/dt = 24.482 x  -0.913 x z
##
## dz/dt = -2.581 z  + 0.968 x y
```

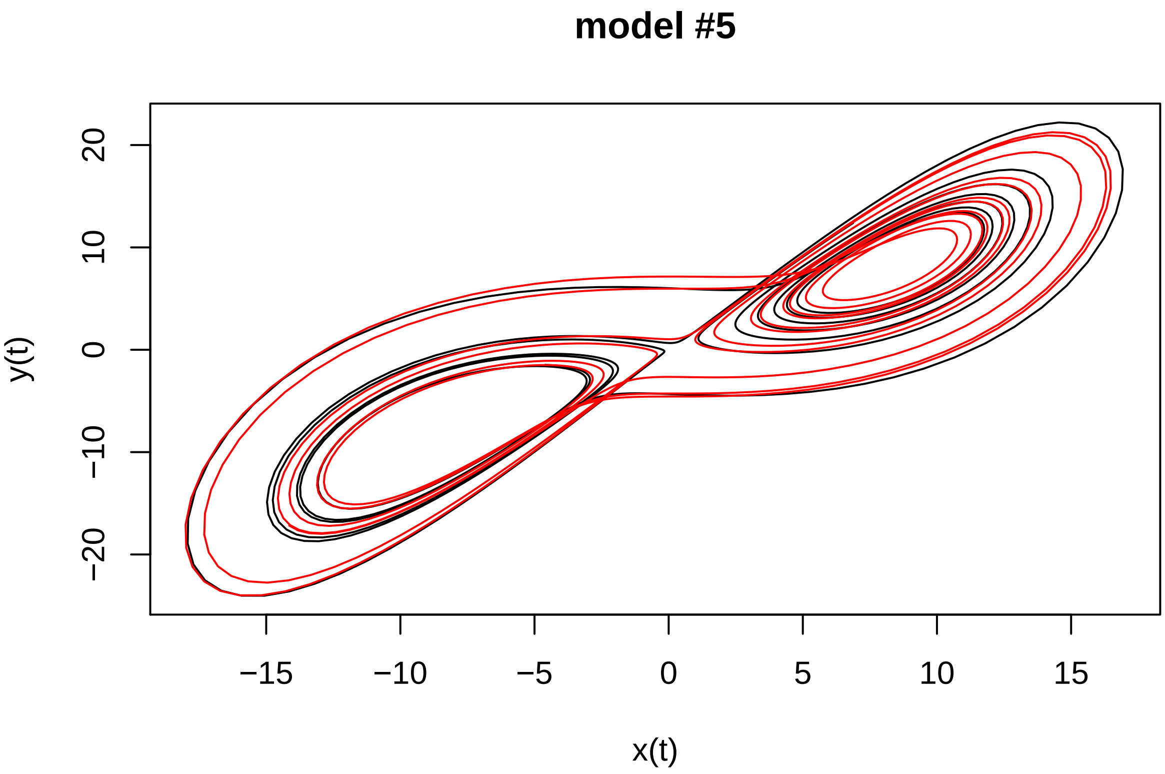
This term is not detected because it is of small influence on the dynamics (actually the dynamic of variable y is more or less similar to the dynamic of x – See upper the observed time series – and its coefficient is 28 times smaller). Three 7-parameter models were also obtained, all having a phase space very similar to the original one. The original model (#18) is included among them :

```
visuEq(K = outL63$models$model18,  substit = 1, approx = 2)
```

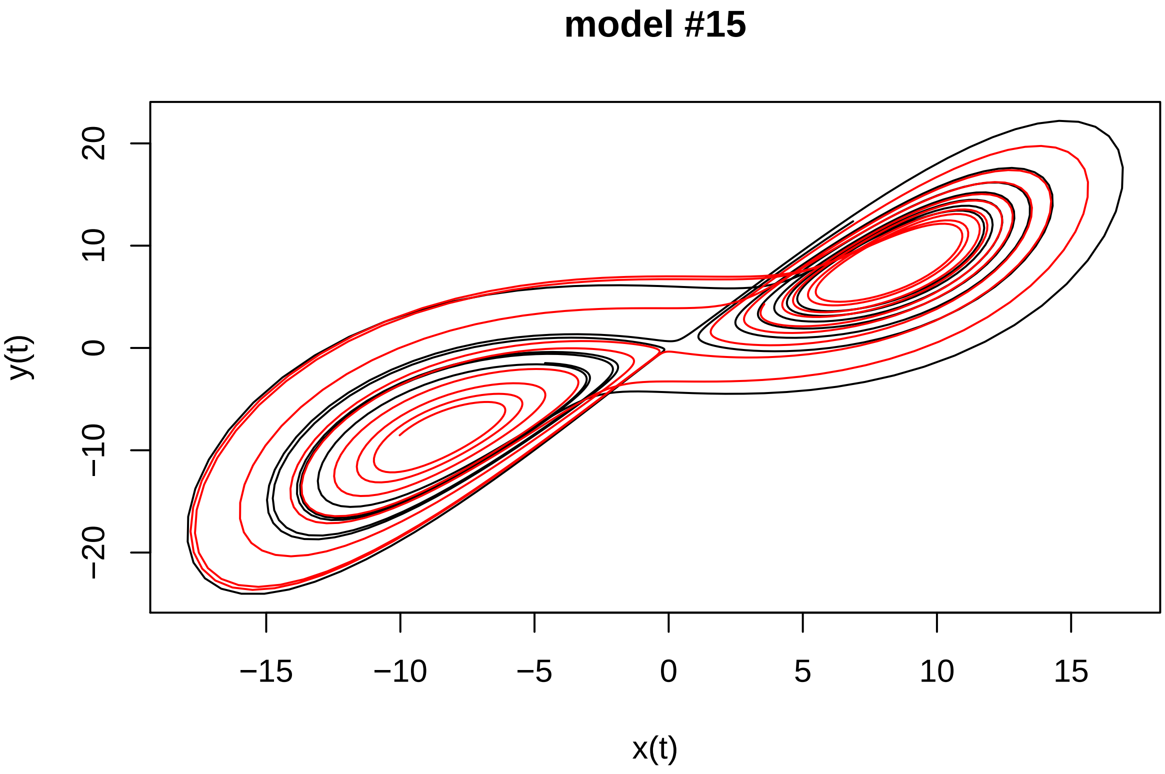
```
## dx/dt = 9.748 y  -9.748 x
##
## dy/dt = -0.605 y  + 25.939 x  -0.941 x z
##
## dz/dt = -2.581 z  + 0.968 x y
```

Since model #5 is more parsomonious, it can be considered that six of the seven terms of the Lorenz system are thus clearly detected. A more in-deep investigation might enable the detection of the term of weaker influence, but this remains unsure. The phase portraits of four potentially valid models are very close to the original phase portraits, it may be quite quite challenging to retrieve the full formulation of the original equations.

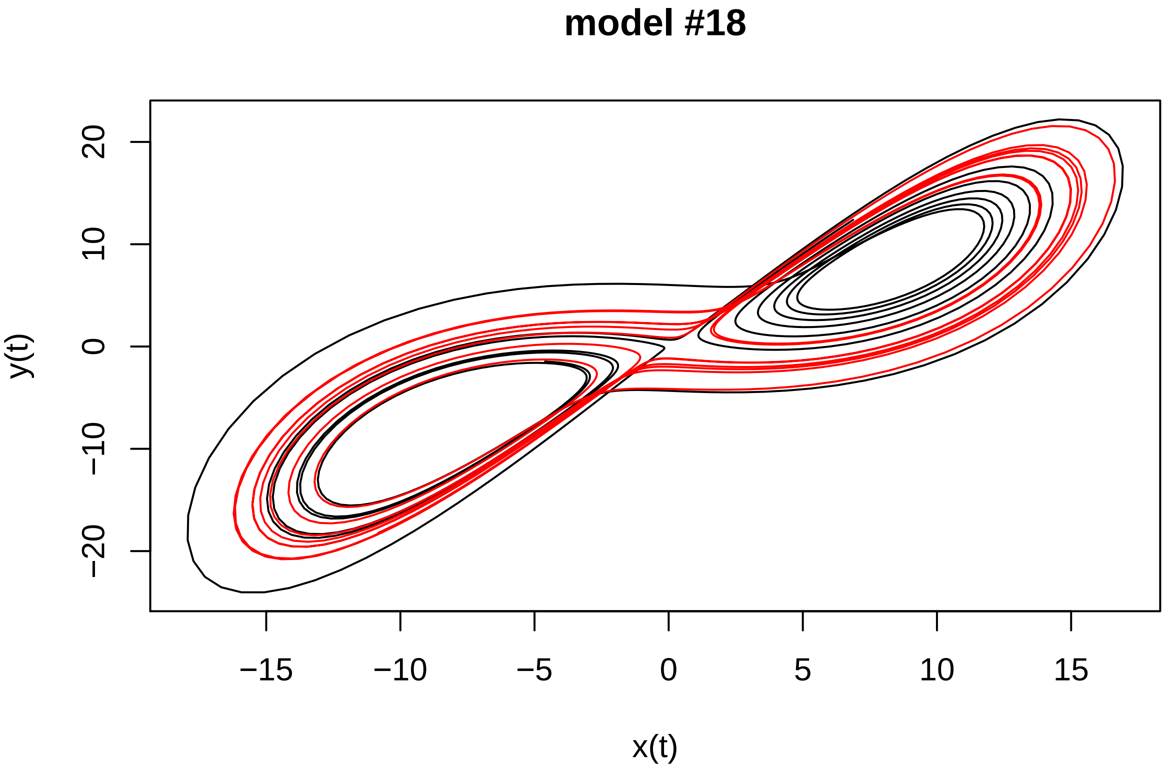
```
plot(TS[,2], TS[,3], type = 'l', xlab = 'x(t)', ylab = 'y(t)', main = 'model #5')
lines(outL63$stockoutreg$model5[,1], outL63$stockoutreg$model5[,2],
      type = 'l', col = 'red')
```



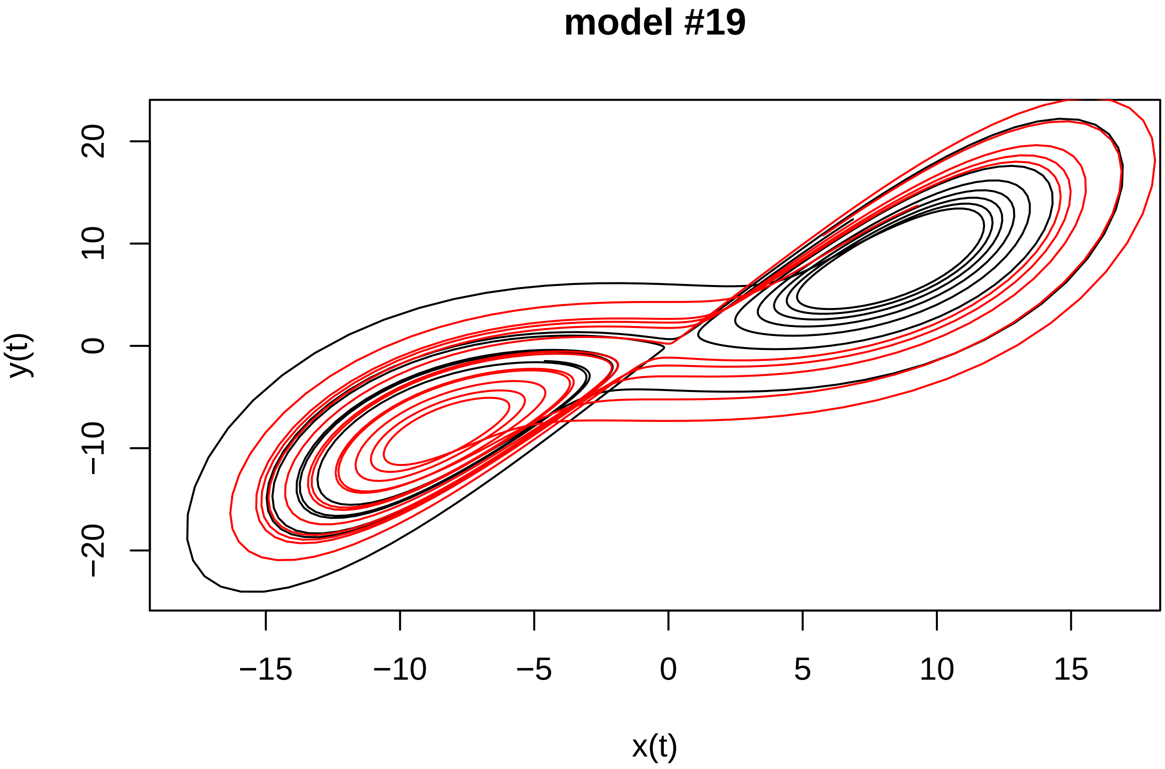
```
plot(TS[,2], TS[,3], type = 'l', xlab = 'x(t)', ylab = 'y(t)', main = 'model #15')
lines(outL63$stockoutreg$model15[,1], outL63$stockoutreg$model15[,2],
      type = 'l', col = 'red')
```



```
plot(TS[,2], TS[,3], type = 'l', xlab = 'x(t)', ylab = 'y(t)', main = 'model #18')
lines(dbl(outL63$stockoutreg$model18[,1]), dbl(outL63$stockoutreg$model18[,2]),
      type = 'l', col = 'red')
```



```
plot(TS[,2], TS[,3], type = 'l', xlab = 'x(t)', ylab = 'y(t)', main = 'model #19')
lines(dbl(outL63$stockoutreg$model19[,1]), dbl(outL63$stockoutreg$model19[,2]),
      type = 'l', col = 'red')
```



A very refined validation technique based on the transition matrices[9] may be a powerful tool to reach such a level of precision for the distinction.

The Burke and Shaw system (1981)

The system discovered by Bill Burke and Robert Show[10] in 1981 is a Lorenz-like system. This system reads

```
visuEq(K = allMod_nVar3_dMax2$BS81,  substit = 1, approx = 4)
```

```
## dx/dt = -10 y  -10 x
##
## dy/dt = -1 y  -10 x z
##
## dz/dt = 4.272  + 10 x y
```

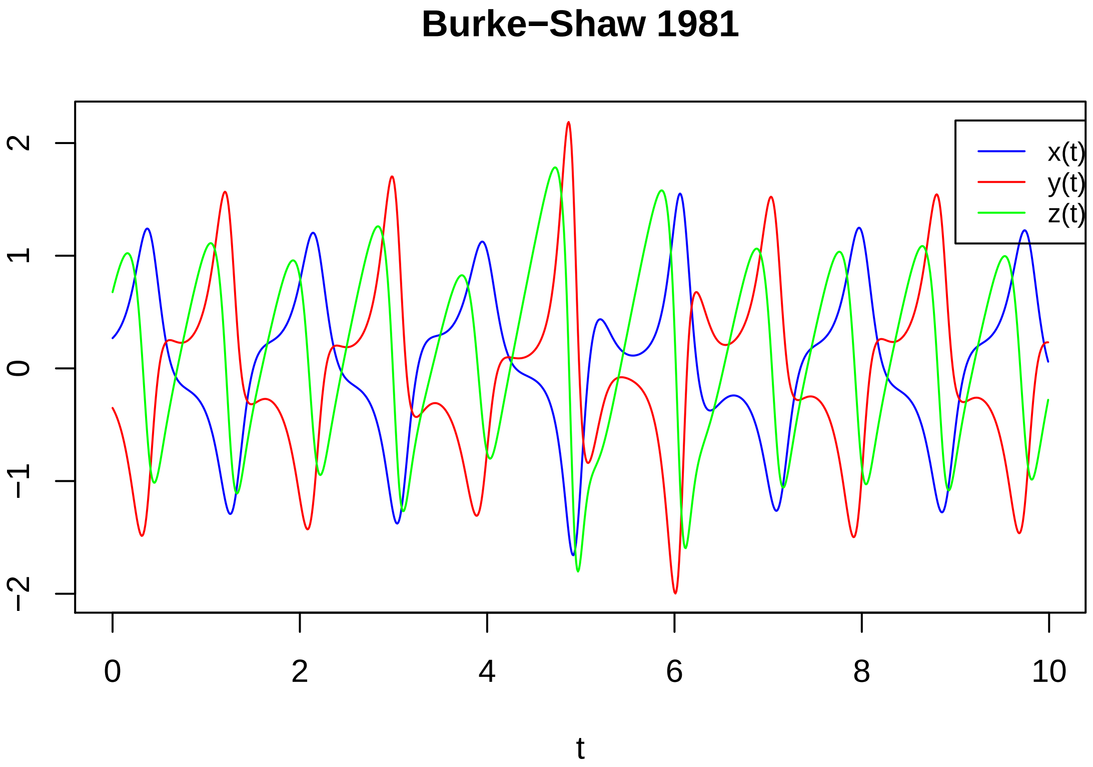
The following set of variable was used to test the possibility to retrieve the original set of equations:

```
TS <- TSallMod_nVar3_dMax2$BS81$reconstr
plot(TS[,1], TS[,2], type = 'l', ylim = c(-2,2.2), main = 'Burke-Shaw 1981',
     xlab = 't', ylab = '', col = 'blue')
lines(TS[,1], TS[,3], type = 'l', col = 'red')
```

[9] Mangiarotti S., Coudret R., Drapeau L. & Jarlan L., Global models' formulation and associated transition matrices for the Rössler system, the electrodissolution of copper in phosphoric acid and the cycles of rainfed wheat observed from satellite in North Morocco. *Supplemental Material* to article "Polynomial Model Search and Global Modelling – two new algorithms for global modelling of chaos" by Mangiarotti S., Coudret R., Drapeau L. & Jarlan L., 2012, *Physical Review E*, **86**(4), 046205.

[10] R. Shaw, Strange attractor, chaotic behavior and information flow, *Zeitschrift für Naturforsch A*, **36**, 80-112, 1981

```
lines(TS[,1], TS[,4], type = 'l', col = 'green')
legend(9, 2.2, c("x(t)", "y(t)", "z(t)"), col=c('blue', 'red', 'green'), lty=1, cex = 0.8)
```



The global modelling technique was applied:

```
# modelling from time series starting
outBS81 <- gPoMo(data = TS[,2:4], tin = TS[,1], dMax = 2, nS = c(1,1,1), show = 1,
                 IstepMin = 10, IstepMax = 1500, nPmin = 5, nPmax = 6, method = 'rk4')
```

Here also, the simplest model has one less term than the original model:

```
visuEq(K = outBS81$models$model3,  substit = 1, approx = 2)
```

```
## dx/dt = -9.77 y  -9.77 x
##
## dy/dt = -9.46 x z
##
## dz/dt = 4.14  + 9.7 x y
```

Three 6-parameter models were also obtained which all have a phase space very similar to the original one. The original model

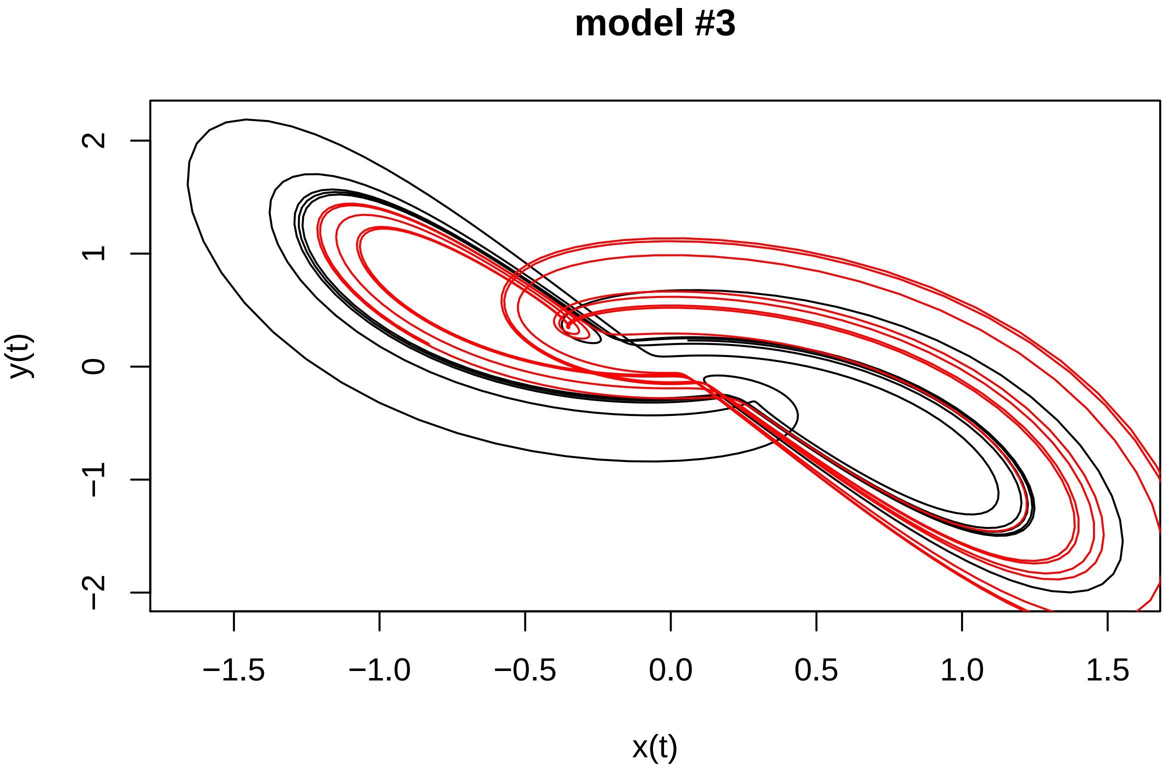
```
visuEq(K = outBS81$models$model11,  substit = 1, approx = 2)
```

```
## dx/dt = -9.766 y  -9.766 x
##
## dy/dt = -0.965 y  -9.627 x z
##
## dz/dt = 4.138  + 9.696 x y
```

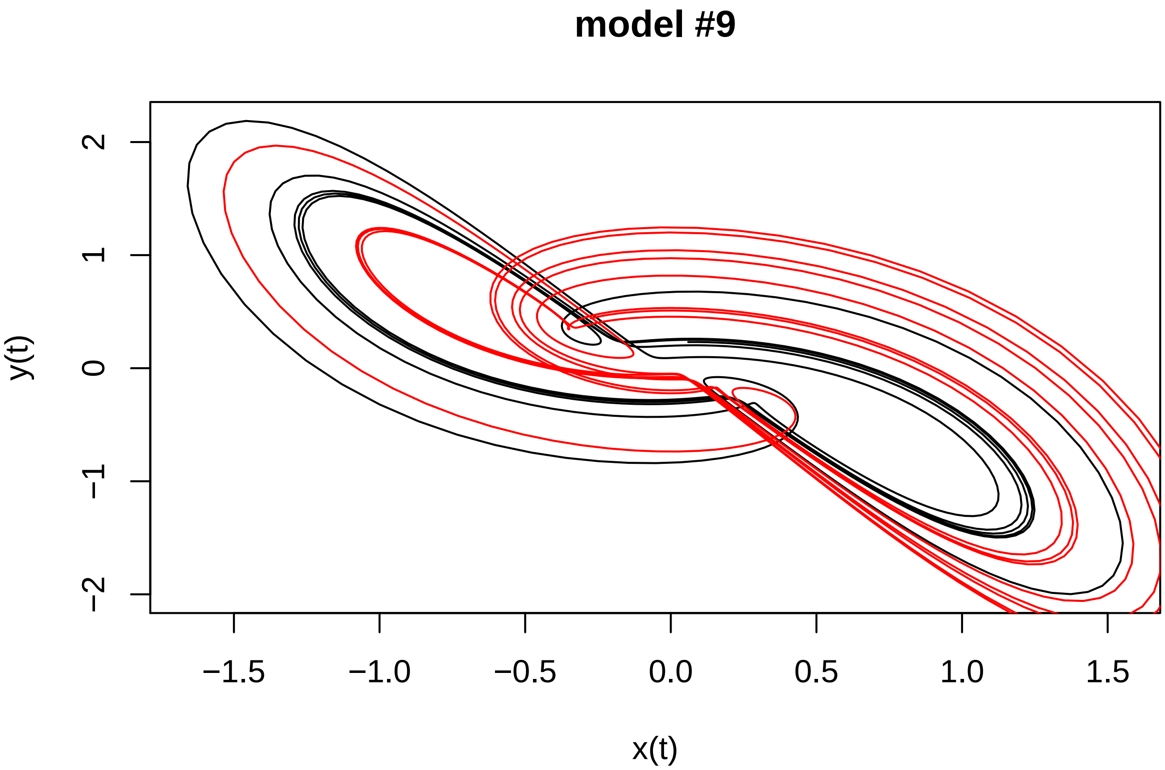
is included among them.

A more in-deep investigation will be required to check if the term of weaker influence can be detected. At this stage, it can already be observed that the phase portrait of the model with the proper formulation (model #11 in red) seems closer to the original phase space (in black).

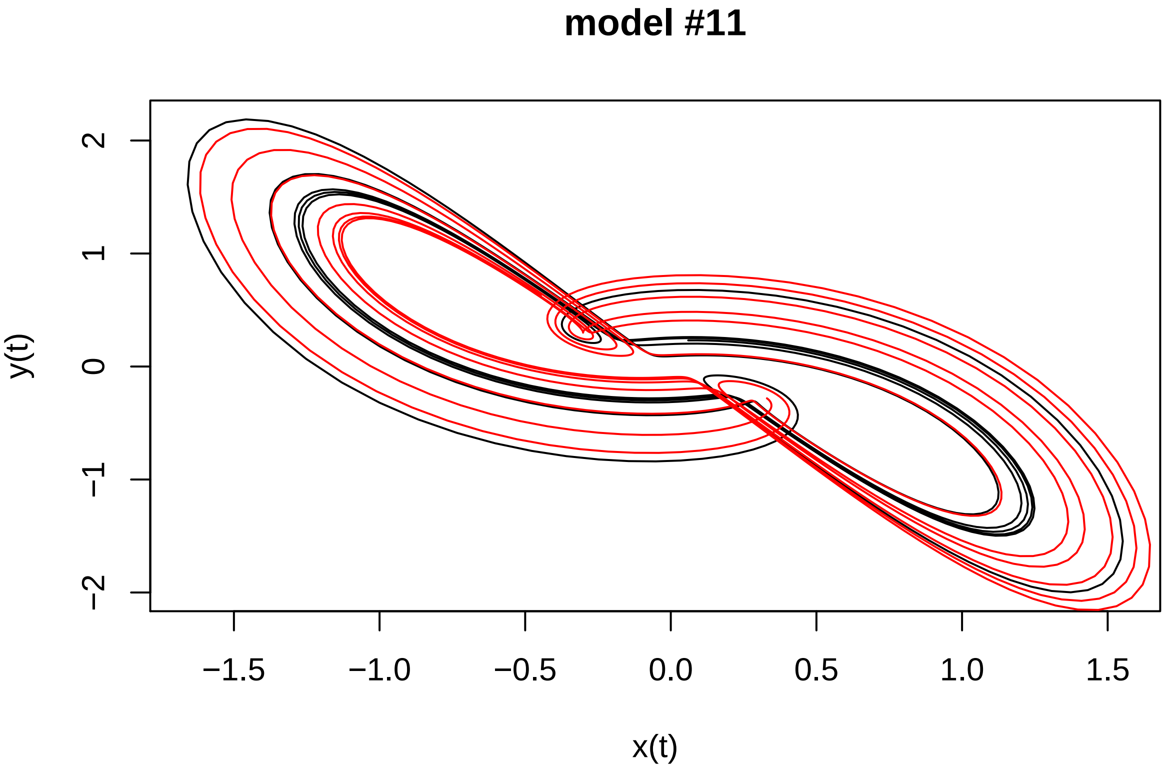
```
plot(TS[,2], TS[,3], type = 'l', xlab = 'x(t)', ylab = 'y(t)', main = 'model #3')
lines(outBS81$stockoutreg$model3[,1], outBS81$stockoutreg$model3[,2],
      type = 'l', col = 'red')
```



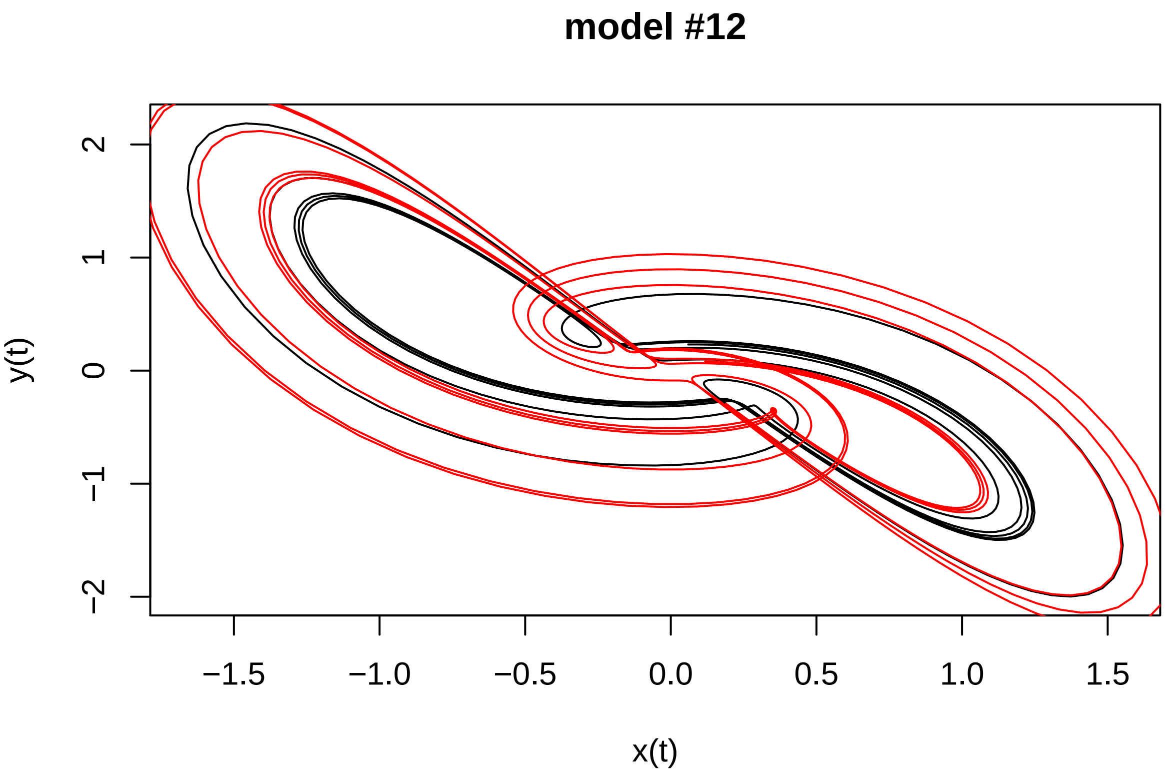
```
plot(TS[,2], TS[,3], type = 'l', xlab = 'x(t)', ylab = 'y(t)', main = 'model #9')
lines(outBS81$stockoutreg$model9[,1], outBS81$stockoutreg$model9[,2],
      type = 'l', col = 'red')
```



```
plot(TS[,2], TS[,3], type = 'l', xlab = 'x(t)', ylab = 'y(t)', main = 'model #11')
lines(outBS81$stockoutreg$model11[,1], outBS81$stockoutreg$model11[,2],
      type = 'l', col = 'red')
```



```
plot(TS[,2], TS[,3], type = 'l', xlab = 'x(t)', ylab = 'y(t)', main = 'model #12')
lines(outBS81$stockoutreg$model12[,1], outBS81$stockoutreg$model12[,2],
      type = 'l', col = 'red')
```



The Lorenz-1984 system

The Lorenz-1984 system[11] was discovered by E.N. Lorenz in 1984. This system is much more complex than the systems previously considered.

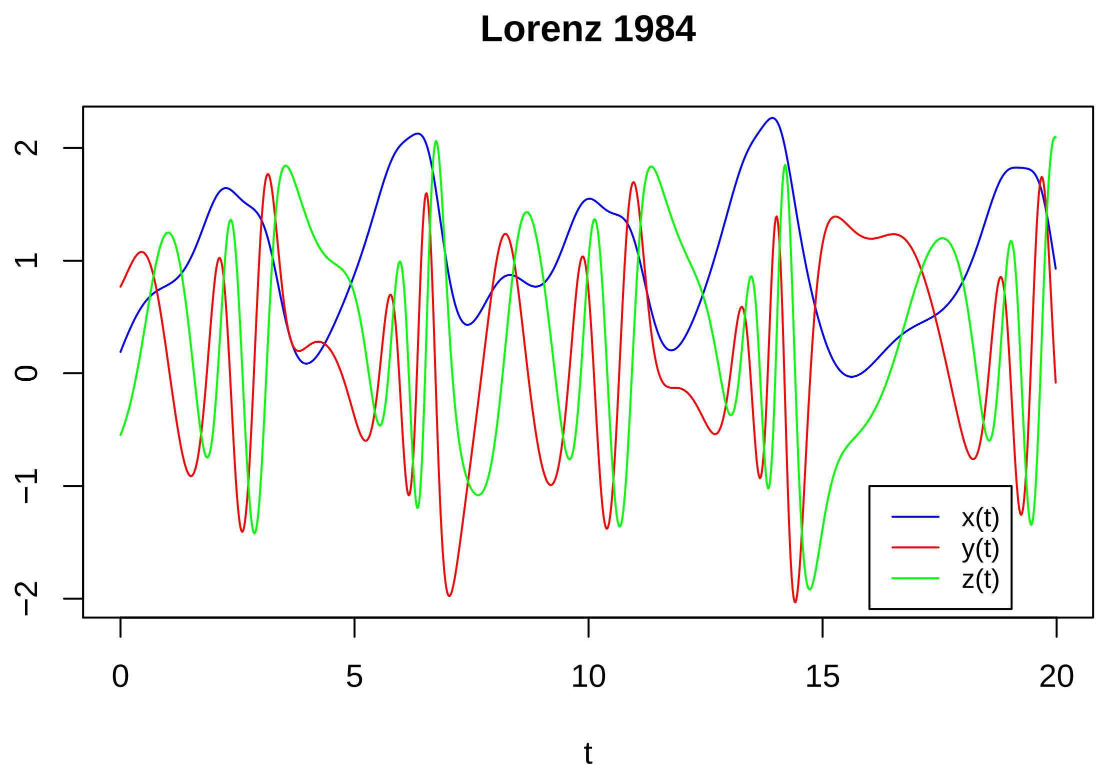
```
visuEq(K = allMod_nVar3_dMax2$L84,  substit = 1)
```

```
## dx/dt = 2  -1 z^2  -1 y^2  -0.25 x
##
## dy/dt = 1  -1 y  -4 x z  + 1 x y
##
## dz/dt = -1 z  + 1 x z  + 4 x y
```

It is also a quadratic system, but it includes eleven terms. The following data set was used for the analysis:

```
TS <- TSallMod_nVar3_dMax2$L84$reconstr
plot(TS[,1], TS[,2], type = 'l', ylim = c(-2,2.2), main = 'Lorenz 1984',
     xlab = 't', ylab = '', col = 'blue')
lines(TS[,1], TS[,3], type = 'l', col = 'red')
lines(TS[,1], TS[,4], type = 'l', col = 'green')
legend(16, -1, c("x(t)", "y(t)", "z(t)"), col=c('blue', 'red', 'green'), lty=1, cex = 0.8)
```

[11] E. N. Lorenz, Irregularity: a fundamental property of the atmosphere, *Tellus A*, **36**, 98-110, 1984.



The global modelling technique was applied to this set of series.

```
# modelling from time series starting
outL84 <- gPoMo(data = TS[,2:4], tin = TS[,1], dMax = 2, nS = c(1,1,1), show = 1,
               IstepMin = 10, IstepMax = 15000, nPmin = 5, nPmax = 11, method = 'rk4')
```

The simplest obtained model has only five terms

```
visuEq(K = outL84$models$model3,  substit = 1, approx = 2)
```

```
## dx/dt = 0.676  -0.684 z^2
##
## dy/dt = -3.672 x z
##
## dz/dt = 0.219 x z  + 3.849 x y
```

However, its attractor is periodic and symmetric. Therefore, it is obviously not a precise approximation of the observed time series. The first model showing a high level of complexity is the model #9. This model has 6 terms

```
visuEq(K = outL84$models$model9,  substit = 1, approx = 2)
```

```
## dx/dt = 0.676  -0.684 z^2
##
## dy/dt = -3.672 x z
##
## dz/dt = -0.951 z  + 0.949 x z  + 3.865 x y
```

Although several terms are missing (two in the first equation, three in the second one), it can be noticed that the detected terms are effectively present in the original system.

Many other models were obtained (models with 7 terms: #19 and #22; with 8 terms: #34, #40 and #49; with 9 terms: #55, #64, #76 and #77; with 10 terms: #83, #92-94, #97-99, #111 and #113; or with 11 terms: #119, #128, #129, #133, #134, #136, #139, #140-142, #147-149 and #155-158). Most of these models have strongly negative values along the x abscissa and can thus be rejected. Only a few of them may be considered as potentially acceptable (models #9, #55, #77, #113, #119, #141 and #158). A visual inspection of the phase portraits obviously shows that the best agreement is obtained with the model #141.

```
visuEq(K = outL84$models$model141,  substit = 1, approx = 2)
```

```
## dx/dt = 1.993  -0.995 z^2  -0.996 y^2  -0.251 x
##
## dy/dt = 0.963  -0.946 y  -3.86 x z  + 0.945 x y
##
## dz/dt = -0.951 z  + 0.949 x z  + 3.865 x y
```

which formulation effectively corresponds to the original Lorenz-1984 system.

The Chlouverakis-Sprott system (2004)

The Chlouverakis-Sprott system was adapted from the Lorenz-1984 system. It was introduced by Konstantinos Chlouverakis and Julian Sprott in 2004.[12]

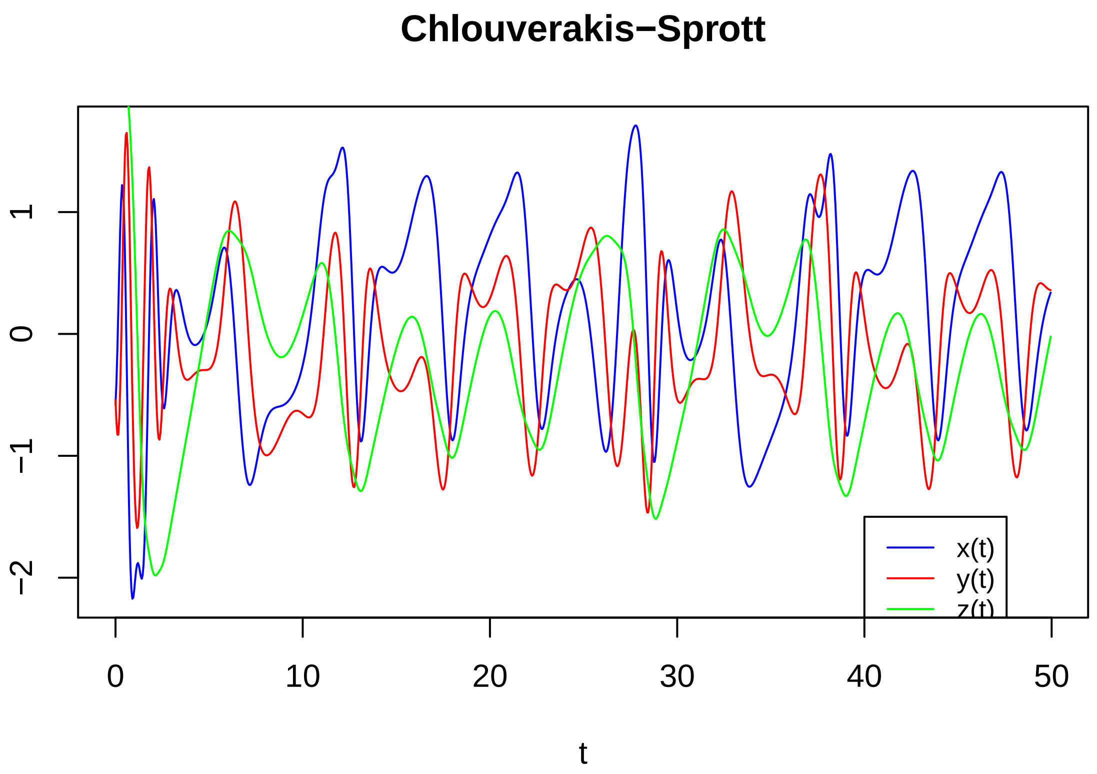
```
visuEq(K = allMod_nVar3_dMax2$CS2004,  substit = 1)
```

```
## dx/dt = 0.4  -3 y z  + 1 x z
##
## dy/dt = 0.1 y z  + 3 x z
##
## dz/dt = 1  -1 y^2  -1 x^2
```

The following data set was used for the analysis:

```
  TS <- TSallMod_nVar3_dMax2$CS2004$reconstr
  plot(TS[,1], TS[,2], type = 'l', xlab = 't', ylab ='',
       main='Chlouverakis-Sprott', col = 'blue')
  lines(TS[,1], TS[,3], type = 'l', xlab = 't', ylab = 'y(t)', col = 'red')
  lines(TS[,1], TS[,4], type = 'l', xlab = 't', ylab = 'z(t)', col = 'green')
  legend(40, -1.5, c("x(t)", "y(t)", "z(t)"), col=c('blue', 'red', 'green'), lty=1, cex = 0.8)
```

[12] K. E. Chlouverakis & J. C. Sprott, A comparison of correlation and Lyapunov dimensions, *Physica D*, **200**, 156-164, 2004.



The global modelling technique was applied to this set of time series.

```
# modelling from time series starting
outCS2004 <- gPoMo(data = TS[,2:4], tin = TS[,1], dMax = 2, nS = c(1,1,1), show = 1,
              IstepMin = 10, IstepMax = 15000, nPmin = 3, nPmax = 8, method = 'rk4')
```

Three types of models were obtained (type-1: #7, #13, #15, #23, #25, #38, #40, #43; type-2: #12, #22, #27, #37; and type-3: #26, #30, #41, #44-46, #49) among which only type-3 models seem compatible with the original phase portrait (the other types can only reproduce a small part of the original phase space). Some spurious effects being observed in models #30 and #49, only five models may be potential candidates: models #26, #41 and #44-46. The smallest model (#26) obtained with the third type has the following formulation

```
visuEq(K = outCS2004$models$model26,  substit = 1, approx = 2)
```

```
## dx/dt = 0.364  -2.731 y z  + 0.954 x z
##
## dy/dt = 2.722 x z
##
## dz/dt = 0.983  -0.983 y^2  -0.987 x^2
```

It is almost identical to the original system since only one term is missing in the second equation.

The complete formulation is obtained in model #41. However, it is difficult to tell which model is the best when comparing models #26, #41, #44 and #45. These results suggest that the original system may be reductible to a 7-term sytem.

The Li system (2007)

A toroidal and symmetrical system was introduced by Dequan Li in 2007[13]:

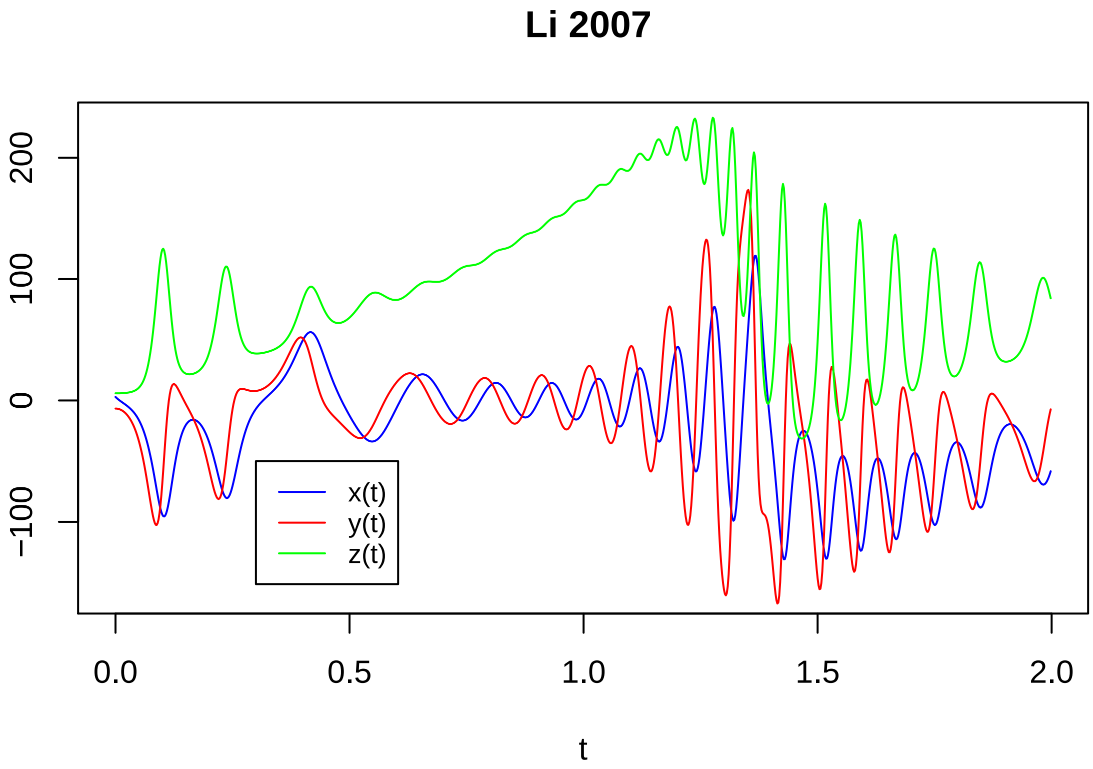
[13] Dequan Li, A three-scroll chaotic attractor, *Physics Letters A*, **372**, 387-393, 2007.

```
visuEq(K = allMod_nVar3_dMax2$Li2007,  substit = 1, approx = 2)
```

```
## dx/dt = 40 y  -40 x  + 0.16 x z
##
## dy/dt = 20 y  + 55 x  -1 x z
##
## dz/dt = 1.833 z  + 1 x y  -0.65 x^2
```

The following data set was obtained from this system:

```
  TS <- TSallMod_nVar3_dMax2$Li2007$reconstr
  plot(TS[,1], TS[,2], type = 'l', ylim = c(-160, 230), xlab = 't',
       ylab = '', main= ' Li 2007', col = 'blue')
  lines(TS[,1], TS[,3], type = 'l', xlab = 't', ylab = 'y(t)', col = 'red')
  lines(TS[,1], TS[,4], type = 'l', xlab = 't', ylab = 'z(t)', col = 'green')
  legend(0.3, -50, c("x(t)", "y(t)", "z(t)"), col=c('blue', 'red', 'green'), lty=1, cex = 0.8)
```



The global modelling technique was then applied to this set of time series.

```
# modelling from time series starting
outLi2007<- gPoMo(data = TS[,2:4], tin = TS[,1], dMax = 2, nS = c(1,1,1), show = 1,
                  IstepMin = 10, IstepMax = 15000, nPmin = 9, nPmax = 9, method = 'rk4')
```

Only the model #13 is able to reproduce the original phase portrait.

```
visuEq(K = outLi2007$models$model13,  substit = 1, approx = 2)
```

```
## dx/dt = 37.726 y  -22.108 x
##
## dy/dt = -0.478 x z
##
## dz/dt = 1.716 z  + 0.897 x y  -0.586 x^2
```

The same result is obtained when allowing models of smaller sizes (e.g. `nPmin = 3`). The original formulation is thus appropriately retrieved.

The Cord system (Aguirre & Letellier 2012)

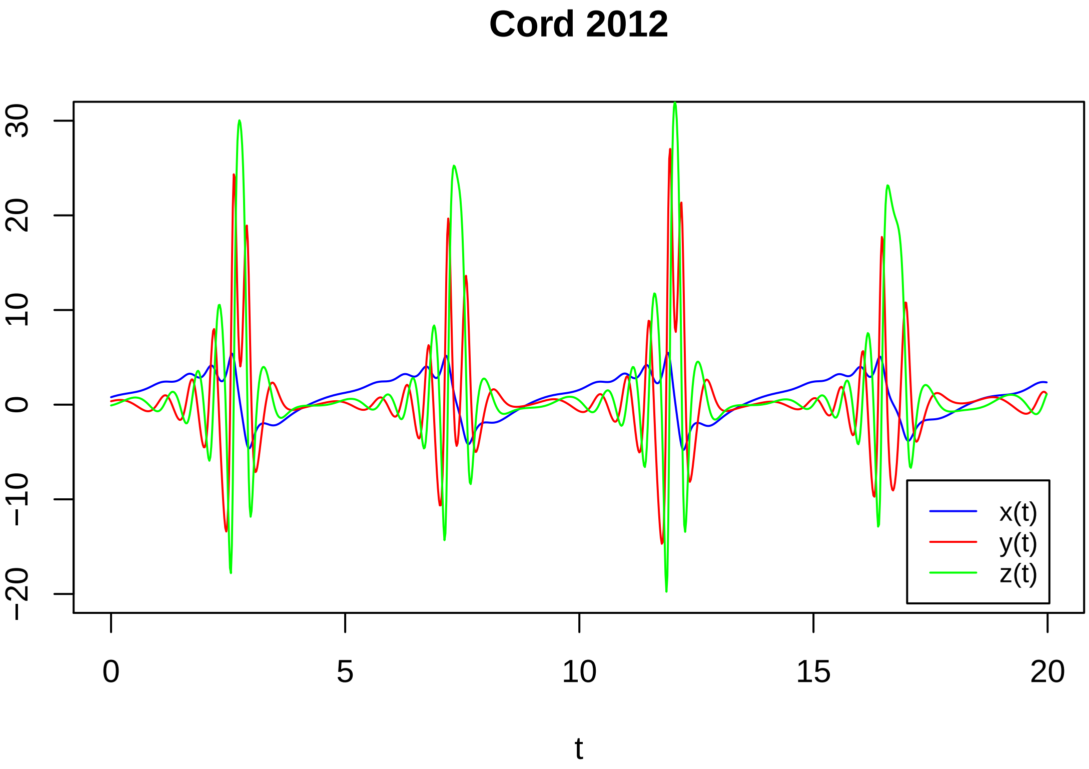
The cord attractor[14] was introduced by Luis Aguirre and Christophe Letellier in 2012 by a modification of the Lorenz-1984 system. The system reads:

```
visuEq(K = allMod_nVar3_dMax2$Cord2012,  substit = 1)
```

```
## dx/dt = 2.064  -1 z  -1 y  -0.258 x
##
## dy/dt = 1  -1 y  -4.033 x z  + 1 x y
##
## dz/dt = -1 z  + 1 x z  + 4.033 x y
```

The following data set was used for the analysis:

```
  TS <- TSallMod_nVar3_dMax2$Cord2012$reconstr
  plot(TS[,1], TS[,2], type = 'l', ylim = c(-20,30),
       xlab = 't', ylab = '', main = 'Cord 2012', col = 'blue')
  lines(TS[,1], TS[,3], type = 'l', xlab = 't', ylab = 'y(t)', col = 'red')
  lines(TS[,1], TS[,4], type = 'l', xlab = 't', ylab = 'z(t)', col = 'green')
  legend(17, -8, c("x(t)", "y(t)", "z(t)"), col=c('blue', 'red', 'green'), lty=1, cex = 0.8)
```



The global modelling technique was applied to this set of time series.

[14] C. Letellier & L. A. Aguirre, Required criteria for recognizing new types of chaos : Application to the “cord” attractor, *Physical Review E*, **85**, 036204, 2012.

```r
# modelling from time series starting
outCord2012 <- gPoMo(data = TS[,2:4], tin = TS[,1], dMax = 2, nS = c(1,1,1), show = 1,
                     IstepMin = 10, IstepMax = 15000, nPmin = 7, nPmax = 11, method = 'rk4')
```

Several obtained models were able to reproduce, more or less, the original phase space are obtained. The smallest models (#9 and #10) have seven terms and are not chaotic. Their formulation is similar to the original system. All the terms detected in model #9 are effectively included in the original system (four terms are not detected).

```r
visuEq(K = outCord2012$models$model9,  substit = 1, approx = 2)
```

```
## dx/dt = 1.66  -0.916 z  -0.897 y
##
## dy/dt = -3.058 x z  + 0.741 x y
##
## dz/dt = 0.699 x z  + 3.319 x y
```

Only one spurious term (lower in magnitude compared to the others) is detected in model #10 (term z^2 in the first equations) and five terms are not detected:

```r
visuEq(K = outCord2012$models$model10,  substit = 1, approx = 2)
```

Among all the obtained models, the phase portrait of model #62 is the more similar to the original one.

```r
visuOutGP(out17,selecmod = 62, prioMinMax = 'model')
```

In this formulation, eight of the eleven terms are correctly detected. The same spurious term is wrongly detected (term `z^2` in the first equations).

```r
visuEq(K = outCord2012$models$model62,  substit = 1, approx = 2)
```

The cord system cannot be retrieved completely in the present case, possibly because its formulation is too complex (too many terms) and not minimal. The most important terms can be detected.

Conclusion

In this vignette, it is shown that, in many cases, the original systems can be fully retrieved. It is in particular the case when the formulation of the original system is concise enough. But it is not always the case. When the systems formulation is more complex (more terms), an exhaustive detection becomes more difficult and only one part of the algebraic terms can be detected. In such conditions the global modelling technique can be used to obtain a reduced formulation of the dynamical systems.