

Numerical methods for dynamical systems

Homework nº 4

Goal(s)

- * Implementation of one-step methods for discontinuous dynamocal systems
- ★ Simulation of a few number of systems

For the next exercises, we will consider different dynamical systems to test the method. In particular, we will consider — F4

$$\dot{y} = \begin{cases} -\frac{2}{21} - \frac{120(t-5)}{1+4(t-5)^2} & \text{if } t \le 10\\ -2y & \text{if } t > 10 \end{cases}$$

with y(0) = 1.0 and the simulation end time is 20 seconds.

— Boucing ball

$$\begin{array}{ll} \dot{y_1} = y_2 \\ \dot{y_2} = -9.81 \end{array} \quad \text{if} \quad (y_1 \leqslant 0 \land y_2 < 0) \Rightarrow \begin{pmatrix} y_1 \\ y_2 \end{pmatrix} = \begin{pmatrix} 0.0 \\ -0.8y_2 \end{pmatrix}$$

with $y_1(0) = 10$ and $y_2(0) = 15.0$. Warning : in this system after detecting the event it is necessary to apply a reset function which will change the state vectors. Final simulation time is 20 (or less)

— F1

$$\begin{split} \dot{y}_1 &= y_2 \\ \dot{y}_2 &= \begin{cases} 2ay_2 - (\pi^2 + a^2)y_1 + 1 & \text{if } \lfloor t \rfloor & \text{is even} \\ 2ay_2 - (\pi^2 + a^2)y_1 - 1 & \text{if } \lfloor t \rfloor & \text{is odd} \end{cases} \end{split}$$

with a = 0.1, $y_1(0) = 0$ and $y_2(0) = 0$, the final simulation time is 20 seconds. We recall in Algorithm 1

Data : f_1 the dynamic, f_2 the dynamic, g the zero-crossing function, \mathbf{y}_0 initial condition, t_0 starting time, t_{end} end time, h integration step-size, tol

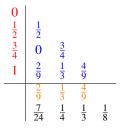
```
t \leftarrow t_0;
\mathbf{y} \leftarrow \mathbf{y}_0;
f \leftarrow f_1;
while t < t_{end} do
      Print(t, y);
      y_1 \leftarrow \text{Euler}(f, t, \mathbf{y}, h);
      y_2 \leftarrow \text{Heun}(f,t,\mathbf{y},h);
      if \textit{ComputeError}(y_1,y_2) is smaller than tol then
             if g(\mathbf{y}) \cdot g(\mathbf{y}_1) < 0 then
                    Compute p(t) from y, f(\mathbf{y}), \mathbf{y}_1 and f(\mathbf{y}_1);
                    [t^{-},t^{+}] = FindZero (g(p(t)));
                    Print (t + t^{-}, p(t^{-}));
                    f \leftarrow f_2;
                    \mathbf{y} \leftarrow p(t^+);
                   t \leftarrow t + t^+;
             end
             \mathbf{y} \leftarrow \mathbf{y}_1;
             t \leftarrow t + h;
             h \leftarrow \text{ComputeNewH} (h, \mathbf{y}_1, \mathbf{y}_2);
       end
      h \leftarrow h/2
end
```

Algorithme 1 : Pseudo code of simulation engine with one-step variable step-size method with discontinuous dynamical systems

Exercise 1 – Implementation

Question 1

We consider a second adaptive Runge-Kutta method which is the Bogacki-Shampine method. Its Butcher tableau is



The integration methods is defined by

$$\mathbf{k}_{1} = f(t_{n}, \mathbf{y}_{n})$$

$$\mathbf{k}_{2} = f(t_{n} + \frac{1}{2}h_{n}, \mathbf{y}_{n} + \frac{1}{2}h_{n}\mathbf{k}_{1})$$

$$\mathbf{k}_{3} = f(t_{n} + \frac{3}{4}h_{n}, \mathbf{y}_{n} + \frac{3}{4}h_{n}\mathbf{k}_{2})$$

$$\mathbf{y}_{n+1} = \mathbf{y}_{n} + h_{n}\left(\frac{2}{9}\mathbf{k}_{1} + \frac{1}{3}\mathbf{k}_{2} + \frac{4}{9}\mathbf{k}_{3}\right)$$

$$\mathbf{k}_{4} = f(t_{n} + h_{n}, \mathbf{y}_{n+1})$$

$$\mathbf{z}_{n+1} = \mathbf{y}_{n} + h_{n}\left(\frac{7}{24}\mathbf{k}_{1} + \frac{1}{4}k_{2} + \frac{1}{3}\mathbf{k}_{3} + \frac{1}{8}\mathbf{k}_{4}\right)$$

Implement this method (with the computeNewH method)

Question 2

Implement of method to compute a polynomial interpolation of the solution y(t) by using Hermite's Cubic Splines

https://en.wikipedia.org/wiki/Cubic_Hermite_spline

Question 3

Implement a method to search for zeros of univariate functions. In particular, we will use Secant method defined by

$$x_{i+1} = x_i + \frac{x_i - x_{i-1}}{f(x_i) - f(x_{i-1})} f(x_i)$$

It requieres two initial values x_0 and x_1 which shall enclose the solution.

Question 4

Solve the problems with your simulation engine.

TO SUBMIT

— A small report should be sent summarize the answers to the questions.

This report should be associated to the source code.

Send the archive containing the report and the source codes in a mail which title is

[numerical methods for dynamical systems] FIRSTNAME LASTNAME

to alexandre.chapoutot@ensta-paris.fr before the next lecture, Friday October 16, 2020.