Algorithms for the integration of ODE

Centre de Recerca Matemàtica

Lluis Alsedà, Marc Jorba-Cuscó, Josep Sardanyès

November 7, 2023

Previous concepts

Euler's method

Can we do better?
- The Runge-Kutta family
- Linear Multistep methods
- The Taylor method

Cripples, Bastards, and Broken Things

Why we can't predict the weather?

Section 1

Previous concepts

What is an ODE?

An Ordinary Differential Equation (ODE) is an expression

$$\dot{x}(t) = f(t, x(t)),$$

Where:

- $f : \mathbb{R} \times \mathbb{R}^n \mapsto \mathbb{R}^n$ is a known function called the **vector-field**.

What is an ODE?

An Ordinary Differential Equation (ODE) is an expression

$$\dot{x}(t) = f(t, x(t)),$$

Where:

- $f : \mathbb{R} \times \mathbb{R}^n \mapsto \mathbb{R}^n$ is a known function called the **vector-field**.
- $x : \mathbb{R} \to \mathbb{R}^n$ is an unknown function (the dependent variable).

What is an ODE?

An Ordinary Differential Equation (ODE) is an expression

$$\dot{x}(t) = f(t, x(t)),$$

Where:

- $f : \mathbb{R} \times \mathbb{R}^n \mapsto \mathbb{R}^n$ is a known function called the **vector-field**.
- $x : \mathbb{R} \to \mathbb{R}^n$ is an unknown function (the dependent variable).
- t is dependent variable (often called **time**).

What is an ODE?

An Ordinary Differential Equation (ODE) is an expression

$$\dot{x}(t) = f(t, x(t)),$$

Where:

- $f : \mathbb{R} \times \mathbb{R}^n \mapsto \mathbb{R}^n$ is a known function called the **vector-field**.
- $x : \mathbb{R} \to \mathbb{R}^n$ is an unknown function (the dependent variable).
- t is dependent variable (often called **time**).
- An ODE with a prescribed initial condition

$$\begin{cases} \dot{x}(t) = f(t, x(t)), \\ x(t_0) = x_0, \end{cases}$$

  is called a **Cauchy Problem**.

Existence and Uniqueness

Under reasonable conditions (at least, continuous $+$ Lipschitz), the Cauchy problem

$$\begin{cases} \dot{x}(t) = f(t, x(t)), \\ x(t_0) = x_0, \end{cases}$$

has a unique solution. Moreover.

- The solution also verifies the integral equation

$$x(t) = x_0 + \int_{t_0}^{t} f(s, x(s))ds.$$

Existence and Uniqueness

Under reasonable conditions (at least, continuous + Lipschitz), the Cauchy problem

$$\begin{cases} \dot{x}(t) = f(t, x(t)), \\ x(t_0) = x_0, \end{cases}$$

has a unique solution. Moreover.

- The solution also verifies the integral equation

$$x(t) = x_0 + \int_{t_0}^{t} f(s, x(s))ds.$$

- The map $\varphi : \mathbb{R}^{n+2} \mapsto \mathbb{R}^n$ given by

$$\varphi(t_0, t, x_0) = x(t),$$

  is called **flow**.

Existence and Uniqueness

Under reasonable conditions (at least, continuous $+$ Lipschitz), the Cauchy problem

$$\begin{cases} \dot{x}(t) = f(t, x(t)), \\ x(t_0) = x_0, \end{cases}$$

has a unique solution. Moreover.

- The solution also verifies the integral equation

$$x(t) = x_0 + \int_{t_0}^{t} f(s, x(s))ds.$$

- The map $\varphi : \mathbb{R}^{n+2} \mapsto \mathbb{R}^n$ given by

$$\varphi(t_0, t, x_0) = x(t),$$

  is called **flow**.
- The flow inherits the regularity of function f.

First order variational equations

- Given a trajectory $\varphi(t_0, t, x_0)$ of the original system, it holds that,

$$\begin{cases} \frac{d}{dt} D_{x_0}\varphi(t_0, t, x_0) = D_{x_0} f(t, \varphi(t_0, t, x_0)) D_{x_0}\varphi(t_0, t, x_0), \\ D_{x_0}\varphi(t_0, 0, x_0), = I_n. \end{cases}$$

First order variational equations

- Given a trajectory $\varphi(t_0, t, x_0)$ of the original system, it holds that,

$$\begin{cases} \frac{d}{dt} D_{x_0}\varphi(t_0, t, x_0) = D_{x_0} f(t, \varphi(t_0, t, x_0)) D_{x_0}\varphi(t_0, t, x_0), \\ D_{x_0}\varphi(t_0, 0, x_0), = I_n. \end{cases}$$

- Interesting for practical purposes: Newton method, Stability of orbits, Lyapunov spectrum, control theory, ...

First order variational equations

- Given a trajectory $\varphi(t_0, t, x_0)$ of the original system, it holds that,

$$\begin{cases} \frac{d}{dt} D_{x_0}\varphi(t_0, t, x_0) = D_{x_0} f(t, \varphi(t_0, t, x_0)) D_{x_0}\varphi(t_0, t, x_0), \\ D_{x_0}\varphi(t_0, 0, x_0), = I_n. \end{cases}$$

- Interesting for practical purposes: Newton method, Stability of orbits, Lyapunov spectrum, control theory, ...

- Classically, are computed by hand and integrated numerically together with the original differential equation.

First order variational equations

- Given a trajectory $\varphi(t_0, t, x_0)$ of the original system, it holds that,

$$\begin{cases} \frac{d}{dt} D_{x_0}\varphi(t_0, t, x_0) = D_{x_0} f(t, \varphi(t_0, t, x_0)) D_{x_0}\varphi(t_0, t, x_0), \\ D_{x_0}\varphi(t_0, 0, x_0), = I_n. \end{cases}$$

- Interesting for practical purposes: Newton method, Stability of orbits, Lyapunov spectrum, control theory, $\ldots$

- Classically, are computed by hand and integrated numerically together with the original differential equation.

- The whole system is of dimension $n + n^2$.

Section 2

Euler's method

Euler's method

The idea of Euler's method is to produce a linear approximation of the solution.

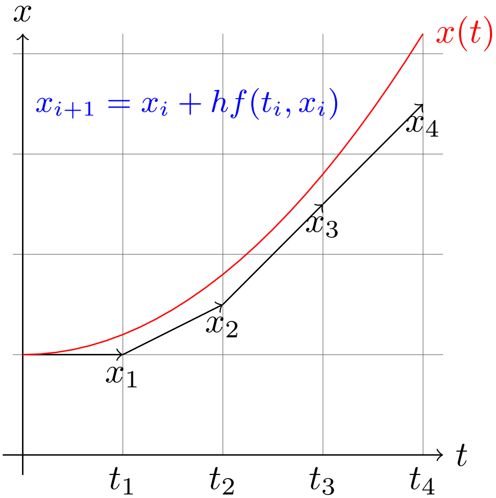
- Given an initial condition (t_0, x_0):

$$\begin{cases} x_1 = x_0 + hf(t_0, x_0), \\ t_1 = t_0 + h. \end{cases}$$

- Here, h is a small quantity called **step**.
- IDEA:

$$\frac{x_1 - x_0}{h} \approx f(t_0, x_0), \quad h = (t_1 - t_2).$$

- The sequence $\{(t_i, x_i)\}_{i \leq N}$ approximates the solution.



Local error

- Let $\{(t_i, x_i)\}_{i \leq N}$ be a sequence of approximations produced by Euler's method.

$$x_{n+1} = x_n + hf(t_n, x_n),$$

Local error

- Let $\{(t_i, x_i)\}_{i \leq N}$ be a sequence of approximations produced by Euler's method.

$$x_{n+1} = x_n + hf(t_n, x_n),$$

- The local error of Euler's method can be estimated by:

$$\sigma(t_n, h) = \|x(t_n) - x_n\| = \mathcal{O}(h^2).$$

(is of order 1).

Local error

- Let $\{(t_i, x_i)\}_{i \leq N}$ be a sequence of approximations produced by Euler's method.

$$x_{n+1} = x_n + hf(t_n, x_n),$$

- The local error of Euler's method can be estimated by:

$$\sigma(t_n, h) = \|x(t_n) - x_n\| = \mathcal{O}(h^2).$$

  (is of order 1).
- Notice that

$$\lim_{h \to 0} \frac{\sigma(t_n, h)}{h} = 0,$$

  (is consistent).

A test equation

- During this lecture we will consider

$$\begin{cases} \dot{x} = x^2 + 2t - t^4, \\ x(0) = 0, \end{cases}$$

  as our test equation.

- This equation can be solved by hand and the solution is t^2.

- This allow us to control the error in a trivial way.

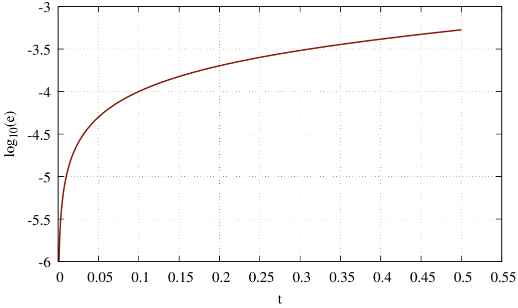


Figure: 500 iterates of the Euler method with step $h = 10^{-3}$

Example: The Kepler Problem

The motion of a test particle about a massive one is governed by Kepler ODE.

$$\begin{cases} \dot{x} = v_x, \\ \dot{y} = v_y, \\ \dot{v_x} = -x/(x^2+y^2)^{3/2}, \\ \dot{v_y} = -y/(x^2+y^2)^{3/2}, \end{cases}$$

- The solutions are known to be conic sections.
- The angular momentum

$$L = xv_y - yv_x,$$

is preserved.

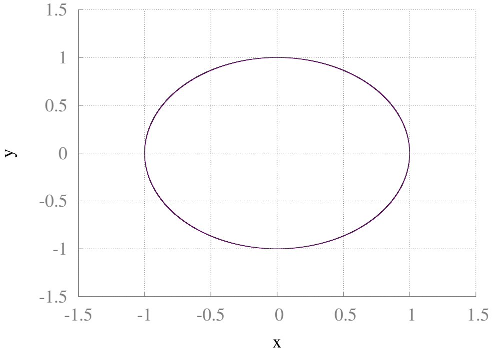


Figure: Circular solution of the Kepler problem.

Example: The Kepler Problem

The motion of a test particle about a massive one is governed by Kepler ODE.

$$\begin{cases} \dot{x} = v_x, \\ \dot{y} = v_y, \\ \dot{v_x} = -x/(x^2+y^2)^{3/2}, \\ \dot{v_y} = -y/(x^2+y^2)^{3/2}, \end{cases}$$

- Local error behaves as expected for t small.
- The errors accumulate and accuracy is lost as t increases.

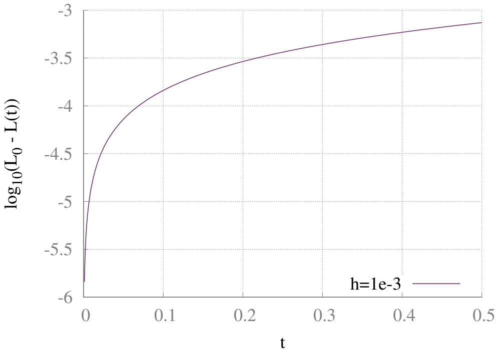


Figure: Error estimated by L. Trajectory with eccentricity 0.1.

A strategy for step size control

- To control the step size it is mandatory to estimate the error.
- We use an extra double iteration of Euler with half the step size.
- The difference between the two predictions behave as

$$e = \frac{1}{2}Kh^2 + \mathcal{O}(h^3)$$

- if $r = e/h > \varepsilon$ we decrease the step

$$h' = 0.9\frac{\varepsilon}{r}h.$$

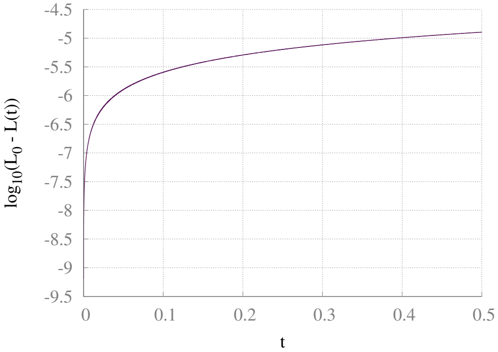


Figure: Error estimated by L. Initial step size 1^3. Final $\approx 10^{-5}$

Generalisations

The concepts of order and consistency can be generalised to any other method to produce the approximation $\{(t_i, x_i)\}_{i \le N}$:

- A method is of **order** p if

$$\sigma(t_n, h) = \|x(t_n) - x_n\| = \mathcal{O}(h^{p+1}).$$

- It is **consistent** if

$$\lim_{h \to 0} \frac{\sigma(t_n, h)}{h} = 0.$$

Section 3

Can we do better?

Improving Euler's method

In Euler's method uses a single value of the vectorfield at a given point of the trajectory to predict the next one.

Some strategies to improve this approach are:

1. Do intermediate evaluations.

Improving Euler's method

In Euler's method uses a single value of the vectorfield at a given point of the trajectory to predict the next one.

Some strategies to improve this approach are:

1. Do intermediate evaluations.

2. Use previously computed values.

Improving Euler's method

In Euler's method uses a single value of the vectorfield at a given point of the trajectory to predict the next one.

Some strategies to improve this approach are:

1. Do intermediate evaluations.

2. Use previously computed values.

3. Use higher order derivatives.

The idea behind the Runge-Kutta methods

- Let us go back to the weak formulation of the Cauchy Problem

$$x(t) = x_0 + \int_{t_0}^{t} f(\tau, x(\tau))d\tau,$$

- The Gaussian quadrature is a method to compute integrals:

$$\int_a^b \psi(\tau)\omega(\tau)d\tau \approx \sum_{i=1}^{s} b_i\psi(c_i),$$

where b_i and c_i depend upon ω (a nonnegative function), a and b.

The idea behind the Runge-Kutta methods

- If we use the weak formulation for a integration step

$$x_{n+1} = x_n + \int_{t_n}^{t_{n+1}} f(\tau, x(\tau))d\tau = x_n + h\int_0^1 f(\tau, x(\tau))d\tau,$$

The idea behind the Runge-Kutta methods

- If we use the weak formulation for a integration step

$$x_{n+1} = x_n + \int_{t_n}^{t_{n+1}} f(\tau, x(\tau))d\tau = x_n + h\int_0^1 f(\tau, x(\tau))d\tau,$$

- We can replace the integral by a quadrature.

$$x_{n+1} = x_n + h\sum_{i=1}^{s} b_i f(t_n + c_i h, x(t_n + c_i h)).$$

The idea behind the Runge-Kutta methods

- If we use the weak formulation for a integration step

$$x_{n+1} = x_n + \int_{t_n}^{t_{n+1}} f(\tau, x(\tau))d\tau = x_n + h\int_0^1 f(\tau, x(\tau))d\tau,$$

- We can replace the integral by a quadrature.

$$x_{n+1} = x_n + h\sum_{i=1}^{s} b_i f(t_n + c_i h, x(t_n + c_i h)).$$

- Here, the quantities $x(t_n + c_i h)$ are not known. In R-K methods are approximated by linear combinations of evaluations of the vectorfield.

General formulation

The family of explicit Runge-Kutta methods of s stages

$$x_{n+1} = x_n + h \sum_{i=1}^{s} b_i k_i,$$

$$\begin{aligned} k_1 &= f(t_n, x_n) \\ k_2 &= f(t_n + c_2 h, x_n + h a_{2,1} k_1) \\ \vdots &= \qquad \vdots \\ k_s &= f\left(t_n + c_s h, x_n + h \sum_{j=1}^{s-1} a_{s,j} k_j\right) \end{aligned}$$

- The methods are consistent if and only if

$$\sum_{i=1}^{s} b_i = 1.$$

- There is more freedom in choosing $a_{i,j}$. A standard choice is

$$\sum_{j=1}^{i-1} a_{i,j} = c_i, \quad i = 2, \ldots, s.$$

- The order of a RK is smaller or equal than the number of stages.

Butcher tableau of a RK method

0					
c_2	$a_{2,1}$				
c_3	$a_{3,1}$	$a_{3,2}$			
$\vdots$	$\vdots$	$\vdots$	$\ddots$		
c_s	$a_{s,1}$	$a_{s,2}$	$\dots$	$a_{s,s-1}$	
	b_1	b_2	$\dots$	b_{s-1}	b_s

Table: General Butcher tableau.

0		
1	1	
	1/2	1/2

Table: Heun's method of 2 stages (order 2).

0			
1/3	1/3		
2/3	0	2/3	
	1/4	0	3/4

Table: Heun's method of 3 stages (order 3).

0				
1/2	1/2			
1/2	0	1/2		
1	0	0	1	
	1/6	2/6	2/6	1/6

Table: Classical R-K method.

Order of R-K methods

- In general, it is not true that a method of s stages has order s.

Order of R-K methods

- In general, it is not true that a method of s stages has order s.
- THM 1: If an explicit s-stage RK method has order p, then $s \leq p$.

Order of R-K methods

- In general, it is not true that a method of s stages has order s.
- THM 1: If an explicit s-stage RK method has order p, then $s \leq p$.
- THM 2: If an explicit s-stage RK method has order $p \geq 5$, then $s > p$.

Order of R-K methods

- In general, it is not true that a method of s stages has order s.
- THM 1: If an explicit s-stage RK method has order p, then $s \leq p$.
- THM 2: If an explicit s-stage RK method has order $p \geq 5$, then $s > p$.
- THM 3 (Butcher): For $p \geq 7$ no explicit R-K method exists of order p with $s = p + 1$ stages.

Order of R-K methods

- In general, it is not true that a method of s stages has order s.
- THM 1: If an explicit s-stage RK method has order p, then $s \leq p$.
- THM 2: If an explicit s-stage RK method has order $p \geq 5$, then $s > p$.
- THM 3 (Butcher): For $p \geq 7$ no explicit R-K method exists of order p with $s = p + 1$ stages.
- THM 4 (Butcher): For $p \geq 8$ no explicit R-K method exists of order p with $s = p + 2$ stages.

Order of R-K methods

- In general, it is not true that a method of s stages has order s.
- THM 1: If an explicit s-stage RK method has order p, then $s \leq p$.
- THM 2: If an explicit s-stage RK method has order $p \geq 5$, then $s > p$.
- THM 3 (Butcher): For $p \geq 7$ no explicit R-K method exists of order p with $s = p + 1$ stages.
- THM 4 (Butcher): For $p \geq 8$ no explicit R-K method exists of order p with $s = p + 2$ stages.

p	1	2	3	4	5	6	7	8
$\min s$	1	2	3	4	6	7	9	11

Table: Minimal number of stages s required to obtain order p.

Performance

- We solve the test equation

$$\begin{cases} \dot{x} = x^2 + 2t - t^4, \\ x(0) = 0, \end{cases}$$

with

1. Euler's method,
2. Heun's method of order 2
3. Classical R-K method (RK4).

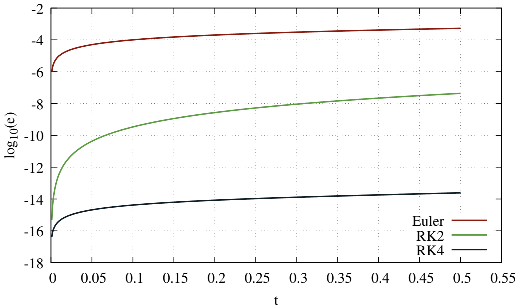


Figure: 500 iterates of Euler's, Heun's and RK4 $h = 10^{-3}$

Error estimation

- Practical computations require local error estimation to control the step size.

Error estimation

- Practical computations require local error estimation to control the step size.
- The idea is to use two methods:

$$x_n = x(t_n) + \mathcal{O}(h^{p+1}), \quad \bar{x}_n = x(t_n) + \mathcal{O}(h^{q+1}),$$

here, $q > p$.

Error estimation

- Practical computations require local error estimation to control the step size.
- The idea is to use two methods:

$$x_n = x(t_n) + \mathcal{O}(h^{p+1}), \quad \bar{x}_n = x(t_n) + \mathcal{O}(h^{q+1}),$$

  here, $q > p$.
- The error estimation is

$$\bar{x}_n - x_n = x(t_n) - y_n + \mathcal{O}(h^{p+2}).$$

Error estimation

- Practical computations require local error estimation to control the step size.
- The idea is to use two methods:

$$x_n = x(t_n) + \mathcal{O}(h^{p+1}), \quad \bar{x_n} = x(t_n) + \mathcal{O}(h^{q+1}),$$

  here, $q > p$.
- The error estimation is

$$\bar{x_n} - x_n = x(t_n) - y_n + \mathcal{O}(h^{p+2}).$$

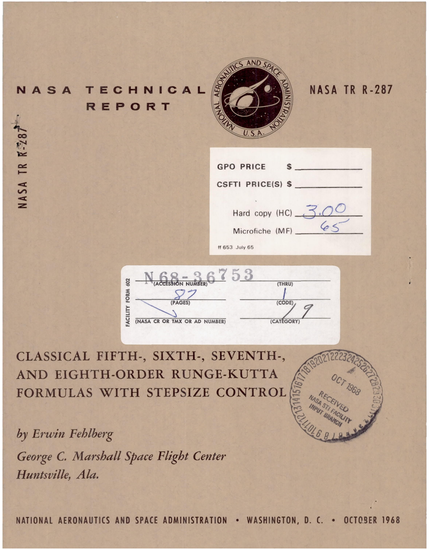
- $\bar{x_n}$ is regarded as the "true solution".

Fehlberg's approach

- Fehlberg considered the following tableau:

$$\begin{array}{c|c} c & A \\ \hline & b' \\ & \hat{b}' \\ \hline & d' \end{array}$$

- Which contains a R-K method of order p and a method of order $p+1$.
- $d' = \hat{b}' - \hat{b}'$ is used for error estimation.
- If we are using the method to compute a quadrature b' and $\hat{b}'$ identical.



Step size control

Let us assume that x_n and $\hat{x}_n$ are approximations provided by methods of order p and $p+1$ respectively.

- Let $\varepsilon < 0$ be a prescribed tolerance.

Step size control

Let us assume that x_n and $\hat{x}_n$ are approximations provided by methods of order p and $p+1$ respectively.

- Let $\varepsilon < 0$ be a prescribed tolerance.
- The estimation of the error is given by

$$\delta = \|x_n - \hat{x}_n\| = \|K_p h^p + \mathcal{O}(h^{p+1})\|.$$

Step size control

Let us assume that x_n and $\hat{x}_n$ are approximations provided by methods of order p and $p+1$ respectively.

- Let $\varepsilon < 0$ be a prescribed tolerance.
- The estimation of the error is given by

$$\delta = \|x_n - \hat{x}_n\| = \|K_p h^p + \mathcal{O}(h^{p+1})\|.$$

- If $\delta < \varepsilon$ we can proceed with the next step (both approximations can be used).

Step size control

Let us assume that x_n and $\hat{x}_n$ are approximations provided by methods of order p and $p+1$ respectively.

- Let $\varepsilon < 0$ be a prescribed tolerance.
- The estimation of the error is given by

$$\delta = \|x_n - \hat{x}_n\| = \|K_p h^p + \mathcal{O}(h^{p+1})\|.$$

- If $\delta < \varepsilon$ we can proceed with the next step (both approximations can be used).
- If not, the step size must be reduced and the approximations recomputed. The new step is

$$0.9\left(\frac{\varepsilon}{\delta}\right)^p h.$$

Linear Multistep methods

- The idea is to use previous steps to improve the accuracy of the method as the integration advances.

Linear Multistep methods

- The idea is to use previous steps to improve the accuracy of the method as the integration advances.
- Originally proposed by Bashforth and Adams (1883).

Linear Multistep methods

- The idea is to use previous steps to improve the accuracy of the method as the integration advances.
- Originally proposed by Bashforth and Adams (1883).
- An stepper of Adams type is given by:

$$x_{n+s} = x_{n+s-1} + h\sum_{j=0}^{s-1} \beta_j f(t_{n+j}, x_{n+j}).$$

Linear Multistep methods

- The idea is to use previous steps to improve the accuracy of the method as the integration advances.
- Originally proposed by Bashforth and Adams (1883).
- An stepper of Adams type is given by:

$$x_{n+s} = x_{n+s-1} + h \sum_{j=0}^{s-1} \beta_j f(t_{n+j}, x_{n+j}).$$

- The constants β_j are chosen to five the highest possible order.

How we get there?

- Assume we have already computed an approximation $x_0, x_1, \ldots x_{n+s-1}$ of order s i.e.

$$x_m = x(t_m) + \mathcal{O}(h^{s+1}).$$

How we get there?

- Assume we have already computed an approximation $x_0, x_1, \dots x_{n+s-1}$ of order s i.e.

$$x_m = x(t_m) + \mathcal{O}(h^{s+1}).$$

- And consider

$$x(t_{n+s}) = x(t_{n+s-1}) + \int_{t_{n+s-1}}^{t_{n+s}} f(\tau, x(\tau)) d\tau.$$

How we get there?

- Assume we have already computed an approximation $x_0, x_1, \dots x_{n+s-1}$ of order s i.e.

$$x_m = x(t_m) + \mathcal{O}(h^{s+1}).$$

- And consider

$$x(t_{n+s}) = x(t_{n+s-1}) + \int_{t_{n+s-1}}^{t_{n+s}} f(\tau, x(\tau))d\tau.$$

- We can approximate $f(t, x(t))$ by

$$P(t) = \sum_{j=0}^{s-1} p_j(t) f(t_{n+j}, x_{n+j}),$$

here, p_j are the Lagrange interpolation polynomials.

Examples

- Adams method for $s = 1$ is Euler's method.

Examples

- Adams method for $s = 1$ is Euler's method.
- For $s = 2$
$$x_{n+2} = x_{n+1} + h\left[\frac{3}{2}f(t_{n+1}, x_{n+1}) - \frac{1}{2}f(t_n, x_n)\right].$$

Examples

- Adams method for $s = 1$ is Euler's method.
- For $s = 2$

$$x_{n+2} = x_{n+1} + h\left[\frac{3}{2}f(t_{n+1}, x_{n+1}) - \frac{1}{2}f(t_n, x_n)\right].$$

- For $s = 3$

$$x_{n+3} = x_{n+2} + h\left[\frac{23}{12}f(t_{n+2}, x_{n+2}) - \frac{4}{3}f(t_{n+1}, x_{n+1}) + \frac{5}{12}f(t_n, x_n)\right].$$

Examples

- Adams method for $s = 1$ is Euler's method.
- For $s = 2$

$$x_{n+2} = x_{n+1} + h\left[\frac{3}{2}f(t_{n+1}, x_{n+1}) - \frac{1}{2}f(t_n, x_n)\right].$$

- For $s = 3$

$$x_{n+3} = x_{n+2} + h\left[\frac{23}{12}f(t_{n+2}, x_{n+2}) - \frac{4}{3}f(t_{n+1}, x_{n+1}) + \frac{5}{12}f(t_n, x_n)\right].$$

- In general, an Adams method of s steps has order s.

Error estimation: Minle device

- Error estimation for LMM can be approached with similar ideas to RKM.

Error estimation: Minle device

- Error estimation for LMM can be approached with similar ideas to RKM.
- Milne device: Use two steppers of the same order.

Error estimation: Minle device

- Error estimation for LMM can be approached with similar ideas to RKM.
- Milne device: Use two steppers of the same order.
- One of them is implicit (this will be discussed later).

Error estimation: Minle device

- Error estimation for LMM can be approached with similar ideas to RKM.
- Milne device: Use two steppers of the same order.
- One of them is implicit (this will be discussed later).
- The other one is explicit and it is used only for error estimation.

Error estimation: Minle device

- Error estimation for LMM can be approached with similar ideas to RKM.
- Milne device: Use two steppers of the same order.
- One of them is implicit (this will be discussed later).
- The other one is explicit and it is used only for error estimation.
- When adjusting the step size a remeshing of the approximated points is required.

The Taylor method

Given a Cauchy problem:

$$\begin{cases} \dot{x} = f(t,x), \\ x(0) = x_0. \end{cases}$$

- If we differentiate the ODE w.r.t. t, we get:

$$\ddot{x} = \partial_t f(t,x) + D_x f(t,x)\dot{x} = \partial_t f(t,x) + D_x f(t,x) f(t,x).$$

The Taylor method

Given a Cauchy problem:

$$\begin{cases} \dot{x} = f(t, x), \\ x(0) = x_0. \end{cases}$$

- If we differentiate the ODE w.r.t. t, we get:

$$\ddot{x} = \partial_t f(t, x) + D_x f(t, x)\dot{x} = \partial_t f(t, x) + D_x f(t, x) f(t, x).$$

- In general, we can get all the derivatives of the solution as a recurrence depending on the derivatives of lower order.

The Taylor method

Given a Cauchy problem:

$$\begin{cases} \dot{x} = f(t,x), \\ x(0) = x_0. \end{cases}$$

- If we differentiate the ODE w.r.t. t, we get:

$$\ddot{x} = \partial_t f(t,x) + D_x f(t,x)\dot{x} = \partial_t f(t,x) + D_x f(t,x) f(t,x).$$

- In general, we can get all the derivatives of the solution as a recurrence depending on the derivatives of lower order.
- Indeed, if we name the **normalized derivatives**

$$X_i = \frac{1}{i!} x^{(i)}(t_0), \qquad F_i = \frac{1}{i!} \left(f(t,x(t))\right)^{(i)} |_{x=x_0},$$

then:

$$X_i = \frac{1}{i+1} F_i.$$

The Taylor method

- Going up to order N we can construct a Taylor Polynomial of the solution:

$$x(t_0 + h) \approx \sum_{i=0}^{N} X_i h^i.$$

The Taylor method

- Going up to order N we can construct a Taylor Polynomial of the solution:

$$x(t_0 + h) \approx \sum_{i=0}^{N} X_i h^i.$$

- The error is $\mathcal{O}(h^{N+1})$. Given N, we can pick h small enough so the approximation has error below some prescribed tolerance.

The Taylor method

- Going up to order N we can construct a Taylor Polynomial of the solution:

$$x(t_0 + h) \approx \sum_{i=0}^{N} X_i h^i.$$

- The error is $\mathcal{O}(h^{N+1})$. Given N, we can pick h small enough so the approximation has error below some prescribed tolerance.
- Then, we can produce the next point of the solution as

$$x_1 = \sum_{i=0}^{N} X_i h^i.$$

The Taylor method

- Going up to order N we can construct a Taylor Polynomial of the solution:

$$x(t_0 + h) \approx \sum_{i=0}^{N} X_i h^i.$$

- The error is $\mathcal{O}(h^{N+1})$. Given N, we can pick h small enough so the approximation has error below some prescribed tolerance.
- Then, we can produce the next point of the solution as

$$x_1 = \sum_{i=0}^{N} X_i h^i.$$

- For the next step, we re-compute the Taylor expansion of the solution about x_1.

The Taylor method

- The main practical issue of this process is to compute the terms of the recurrence:

$$F_i = \frac{1}{i!} \left(f(t, x(t)) \right)^{(i)} |_{x=x_0},$$

(this can be achieved by means of automatic differentiation).

The Taylor method

- The main practical issue of this process is to compute the terms of the recurrence:

$$F_i = \frac{1}{i!} \left(f(t, x(t)) \right)^{(i)} |_{x=x_0},$$

(this can be achieved by means of automatic differentiation).

- Given a threshold, there is an optimal choice of order and step-size.

The Taylor method

- The main practical issue of this process is to compute the terms of the recurrence:

$$F_i = \frac{1}{i!} \left(f(t, x(t))\right)^{(i)} |_{x=x_0},$$

(this can be achieved by means of automatic differentiation).

- Given a threshold, there is an optimal choice of order and step-size.
  1. The optimal step-size is $\approx e^{-2}\rho(t)$ where $\rho(t)$ is the radius of convergence of the series.

The Taylor method

- The main practical issue of this process is to compute the terms of the recurrence:

$$F_i = \frac{1}{i!}\left(f(t, x(t))\right)^{(i)}\Big|_{x=x_0},$$

(this can be achieved by means of automatic differentiation).

- Given a threshold, there is an optimal choice of order and step-size.
  1. The optimal step-size is $\approx e^{-2}\rho(t)$ where $\rho(t)$ is the radius of convergence of the series.
  2. The optimal order is linear in the number of digits D. For a single step, the global computational cost is $\mathcal{O}(D^4)$.

The Taylor method

SUMMARY:

- The Taylor method is based in producing a Taylor polynomial of the solution at each step.

The Taylor method

SUMMARY:

- The Taylor method is based in producing a Taylor polynomial of the solution at each step.

- Both, the order and the step-size can be updated optimally according to a prescribed accuracy.

The Taylor method

SUMMARY:

- The Taylor method is based in producing a Taylor polynomial of the solution at each step.
- Both, the order and the step-size can be updated optimally according to a prescribed accuracy.
- The Taylor method is extremely competitive when high accuracy is required.

Taylor vs RKF78

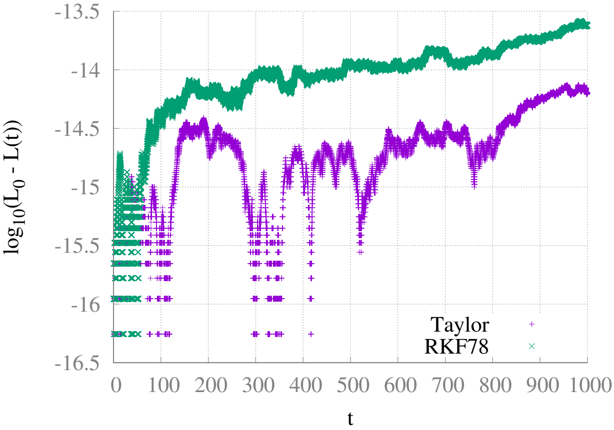


Figure: Integration of 1000 units of time using Taylor and RKF78 of an orbit with $e = 0.5$.

Section 4

Cripples, Bastards, and Broken Things

Stiffness

- Let us consider a linear ODE

$$\dot{x} = Mx, \qquad x(0) = I.$$

Stiffness

- Let us consider a linear ODE

$$\dot{x} = Mx, \qquad x(0) = I.$$

- Suppose that M can be diagonalized by a change $D = S^{-1}MS$,

Stiffness

- Let us consider a linear ODE

$$\dot{x} = Mx, \qquad x(0) = I.$$

- Suppose that M can be diagonalized by a change $D = S^{-1}MS$,
- The equation $\dot{y} = Dy$ is a system of uncoupled equations.

Stiffness

- Let us consider a linear ODE

$$\dot{x} = Mx, \quad x(0) = I.$$

- Suppose that M can be diagonalized by a change $D = S^{-1}MS$,
- The equation $\dot{y} = Dy$ is a system of uncoupled equations.
- Suppose now that there exist an eigenvalue $-\lambda$ with $\lambda >> 1$.

Stiffness

- Let us consider a linear ODE

$$\dot{x} = Mx, \qquad x(0) = I.$$

- Suppose that M can be diagonalized by a change $D = S^{-1}MS$,
- The equation $\dot{y} = Dy$ is a system of uncoupled equations.
- Suppose now that there exist an eigenvalue $-\lambda$ with $\lambda >> 1$.
- Its associated equations is

$$\begin{cases} \dot{x} = -\lambda x, \\ x(0) = 1, \end{cases}$$

and has solution $x(t) = \exp(-\lambda t)$

Domain of stability of Euler's method

- Let us apply Euler's method:

$$\dot{x} = f(t, x), \qquad x_{n+1} = x_n + hf(t_n, x_n)$$

Domain of stability of Euler's method

- Let us apply Euler's method:

$$\dot{x} = f(t, x), \qquad x_{n+1} = x_n + hf(t_n, x_n)$$

- For the previous equation ($\dot{x} = -\lambda x$) the Euler method becomes

$$x_{n+1} = x_n - h\lambda x_n = (1 - h\lambda)x_n \quad \Longrightarrow \quad x_n = (1 - h\lambda)^n.$$

Domain of stability of Euler's method

- Let us apply Euler's method:

$$\dot{x} = f(t, x), \qquad x_{n+1} = x_n + hf(t_n, x_n)$$

- For the previous equation ($\dot{x} = -\lambda x$) the Euler method becomes

$$x_{n+1} = x_n - h\lambda x_n = (1 - h\lambda)x_n \quad \Longrightarrow \quad x_n = (1 - h\lambda)^n.$$

- This implies that, to have $x_n \to 0$, h has to be small enough:

$$h < \frac{2}{\lambda}.$$

Domain of stability of Euler's method

- Let us apply Euler's method:

$$\dot{x} = f(t, x), \qquad x_{n+1} = x_n + hf(t_n, x_n)$$

- For the previous equation ($\dot{x} = -\lambda x$) the Euler method becomes

$$x_{n+1} = x_n - h\lambda x_n = (1 - h\lambda)x_n \quad \Longrightarrow \quad x_n = (1 - h\lambda)^n.$$

- This implies that, to have $x_n \to 0$, h has to be small enough:

$$h < \frac{2}{\lambda}.$$

- Domain of stability: $\mathcal{D} = \{z \in \mathbb{C} \ : \ |1 + z| < 1\}$.

The implicit Euler method

- Let us see what happens if we use an implicit Euler method:

$$\dot{x} = f(t, x), \qquad x_{n+1} = x_n + hf(t_{n+1}, x_{n+1})$$

The implicit Euler method

- Let us see what happens if we use an implicit Euler method:

$$\dot{x} = f(t, x), \qquad x_{n+1} = x_n + hf(t_{n+1}, x_{n+1})$$

- For the equation $\dot{x} = -\lambda x$ we obtain

$$x_{n+1} = x_n - h\lambda x_{n+1} \quad \Longrightarrow \quad (1 + h\lambda)x_{n+1} = x_n,$$

The implicit Euler method

- Let us see what happens if we use an implicit Euler method:

$$\dot{x} = f(t, x), \qquad x_{n+1} = x_n + hf(t_{n+1}, x_{n+1})$$

- For the equation $\dot{x} = -\lambda x$ we obtain

$$x_{n+1} = x_n - h\lambda x_{n+1} \quad \Longrightarrow \quad (1 + h\lambda)x_{n+1} = x_n,$$

- and then

$$x_n = \frac{1}{(1 + h\lambda)^n},$$

  which goes to zero for any $h > 0$ and $\lambda > 0$.

A-stability

- The implicit Euler method has domain of stability $\Re(z) < 0$.

A-stability

- The implicit Euler method has domain of stability $\Re(z) < 0$.

- The domain of stability can be computed for other methods.

A-stability

- The implicit Euler method has domain of stability $\Re(z) < 0$.
- The domain of stability can be computed for other methods.
- A method is A-stable if $\Re(z) < 0$ is contained in its domain of stability.

A-stability

- The implicit Euler method has domain of stability $\Re(z) < 0$.
- The domain of stability can be computed for other methods.
- A method is A-stable if $\Re(z) < 0$ is contained in its domain of stability.
- No explicit R-K method can be A-stable.

A-stability

- The implicit Euler method has domain of stability $\Re(z) < 0$.
- The domain of stability can be computed for other methods.
- A method is A-stable if $\Re(z) < 0$ is contained in its domain of stability.
- No explicit R-K method can be A-stable.
- For any order p there exists an Implicit R-K method which is A-stable.

A-stability

- The implicit Euler method has domain of stability $\Re(z) < 0$.
- The domain of stability can be computed for other methods.
- A method is A-stable if $\Re(z) < 0$ is contained in its domain of stability.
- No explicit R-K method can be A-stable.
- For any order p there exists an Implicit R-K method which is A-stable.
- Dahlquist second barrier: The highest order of an A-stable multistep method is 2.

A remark on implicit methods

- Implicit methods are meant to deal with stiff equations.

A remark on implicit methods

- Implicit methods are meant to deal with stiff equations.
- Those methods require to solve an algebraic equation to compute each step.

A remark on implicit methods

- Implicit methods are meant to deal with stiff equations.
- Those methods require to solve an algebraic equation to compute each step.
- As a consequence, implicit methods are computationally more expensive than the explicit ones.

A remark on implicit methods

- Implicit methods are meant to deal with stiff equations.
- Those methods require to solve an algebraic equation to compute each step.
- As a consequence, implicit methods are computationally more expensive than the explicit ones.
- It is not a good idea to use implicit methods “just in case”.

A remark on implicit methods

- Implicit methods are meant to deal with stiff equations.
- Those methods require to solve an algebraic equation to compute each step.
- As a consequence, implicit methods are computationally more expensive than the explicit ones.
- It is not a good idea to use implicit methods "just in case".
- It is important to know if an equation is stiff.

A feature from Taylor

- The Taylor method also fails to deal with stiffness.
- However, this pathological behaviour can be detected by means of the Taylor series of the solution.
- In the Figure, we plot the first 16 Taylor coefficients of $\exp(-10^4 t)$:

$$\sum_{i=0}^{\infty} \frac{(\lambda t)^k}{k!}, \qquad \lambda = -10^4.$$

- The coefficients increase before the factorial becomes dominant.

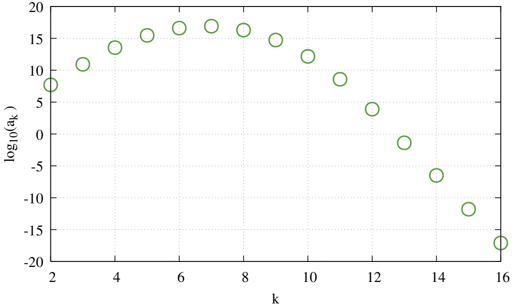


Figure: Taylor coefficients of the function $\exp(-\lambda t)$.

Fail in Fehlberg strategy

- Let us consider the ODE

$$\begin{cases} \dot{x} = \alpha x + \cos(t) - \alpha \sin(t), \\ x(0) = 0, \end{cases}$$

which has $x(t) = \sin t$ as the exact solution. Let us choose $\alpha = 10^{-4}$.

Fail in Fehlberg strategy

- Let us consider the ODE

$$\begin{cases} \dot{x} = \alpha x + \cos(t) - \alpha \sin(t), \\ x(0) = 0, \end{cases}$$

  which has $x(t) = \sin t$ as the exact solution. Let us choose $\alpha = 10^{-4}$.
- A test of the error of integration is to compute the solution at $t = 2\pi$, and it should be zero.

Fail in Fehlberg strategy

- Let us consider the ODE

$$\begin{cases} \dot{x} = \alpha x + \cos(t) - \alpha \sin(t), \\ x(0) = 0, \end{cases}$$

  which has $x(t) = \sin t$ as the exact solution. Let us choose $\alpha = 10^{-4}$.
- A test of the error of integration is to compute the solution at $t = 2\pi$, and it should be zero.
- If we use a RKF78, asking for an accuracy of 10^{-12}, we obtain than $x(2\pi)$ is, approximately, -2.383702×10^{-7}.

Fail in Fehlberg strategy

- Let us consider the ODE

$$\begin{cases} \dot{x} = \alpha x + \cos(t) - \alpha \sin(t), \\ x(0) = 0, \end{cases}$$

  which has $x(t) = \sin t$ as the exact solution. Let us choose $\alpha = 10^{-4}$.
- A test of the error of integration is to compute the solution at $t = 2\pi$, and it should be zero.
- If we use a RKF78, asking for an accuracy of 10^{-12}, we obtain than $x(2\pi)$ is, approximately, -2.383702×10^{-7}.
- An alternative to the Fehlberg step size control is a step size control developed (later) by J. Verner.

Fail in Fehlberg strategy

- Let us consider the ODE

$$\begin{cases} \dot{x} = \alpha x + \cos(t) - \alpha \sin(t), \\ x(0) = 0, \end{cases}$$

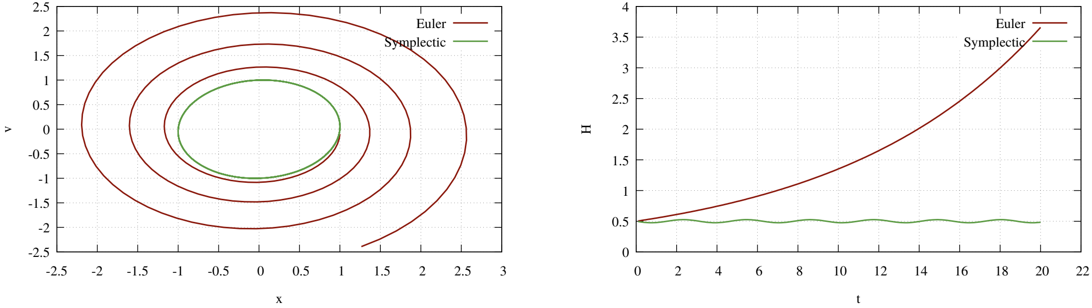
  which has $x(t) = \sin t$ as the exact solution. Let us choose $\alpha = 10^{-4}$.
- A test of the error of integration is to compute the solution at $t = 2\pi$, and it should be zero.
- If we use a RKF78, asking for an accuracy of 10^{-12}, we obtain than $x(2\pi)$ is, approximately, -2.383702×10^{-7}.
- An alternative to the Fehlberg step size control is a step size control developed (later) by J. Verner.
- Using a Runge-Kutta-Verner for the previous example we obtain that $x(2\pi)$ is, approximately, $-3.747003 \times 10^{-16}$.

Energy drift

Let us consider hamiltonian model:

$$H = \frac{1}{2}(p^2 + \omega x^2),$$

And integrate if with Euler's method and Symplectic Euler's method ($\omega = 1$ and $h = 0.1$)



Artefacts

- The Chirikov Standard Map (SM) is a well known Area Preserving Map (APM).

$$\begin{cases} \theta_{n+1} = \theta_n + p_{n+1}, \\ p_{n+1} = p_n + h\sin(\theta_n) \end{cases}$$

- It can be obtained from applying a symplectic Euler method to a pendulum.
- The SM is a simple model for non-integrable APMs. Meaning that it exhibits chaotic behaviour.

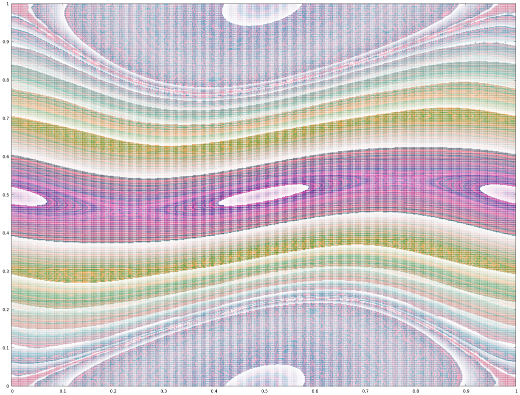


Figure: Phase portrait of Standard Map ($h = 0.5$)

Section 5

Why we can’t predict the weather?

The Lorenz system

The Lorenz system is a simplified model for atmospheric convection:

$$\begin{cases} \dot{x} = \sigma(y - x), \\ \dot{y} = x(\rho - z) - y, \\ \dot{z} = xy - \beta z \end{cases}$$

- For suitable values of the parameters, it exhibits chaotic behaviour.
- The motion is driven by an attractor of Hausdorff dimension ≈ 2.06.
- The flow is dissipative and there are two repealing limit cycles.

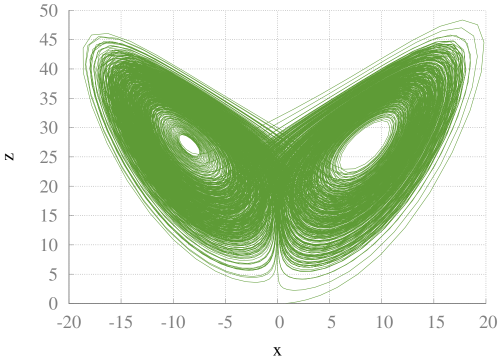


Figure: $x - z$ projection of the Attractor. $\sigma = 10$, $\rho = 28$, $\beta = 8/3$. Integration time: 500. Initial condition $(1, 0, 0)$.

The Lorenz system

The Lorenz system is a simplified model for atmospheric convection:

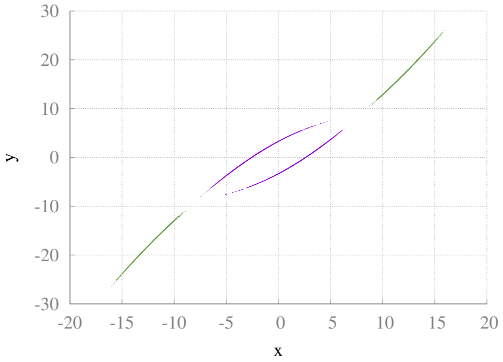


Figure: Poincaré maps $\{z = 25\}$. Purple points correspond to crossings with $\dot{z} < 0$. Green points with $\dot{z} > 0$.

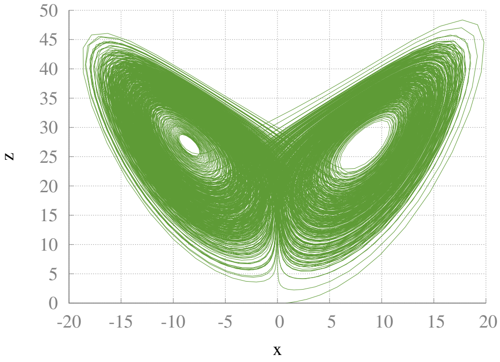


Figure: $x - z$ projection of the Attractor. $\sigma = 10$, $\rho = 28$, $\beta = 8/3$. Integration time: 500. Initial condition $(1, 0, 0)$.

Growth of the error due to dynamics

Let us start an integration at $(1,0,0)$ and $(1,0,0)+v$, check the outputs and track the norm of the directional derivative w.r.t. $v=(10^{-8},0,0)$.

T	e	$\|\nabla_v \varphi_T\|$
10	2.931815e-08	2.9318251e-08
20	7.950019e-08	7.9494469e-08
30	1.534007e-04	1.5333987e-04
40	9.850263e-01	9.7055406e-01
50	1.820953e+01	1.5527040e+04

- For small times the propagation of error is controlled.
- For $T=30$, the initial error has been amplified by 10^4.
- For $T=50$, is amplified by 10^{12}.

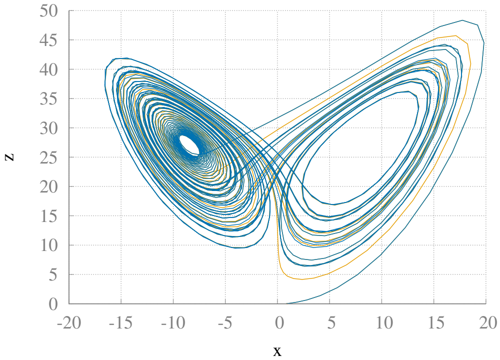


Figure: Two trajectories with initial distance 10^{-8}. Integration time: 50.

References

- Hairer, Ernst; Nørsett, Syvert Paul; Wanner, Gerhard (1993), “Solving ordinary differential equations I: Nonstiff problems”, Berlin, New York: Springer-Verlag, ISBN 978-3-540-56670-0.
- Iserles, Arieh (1996), “A First Course in the Numerical Analysis of Differential Equations”, Cambridge University Press, ISBN 978-0-521-55655-2.
- Butcher, John C. (2008), “Numerical Methods for Ordinary Differential Equations”, New York: John Wiley and Sons, ISBN 978-0-470-72335-7.
- Jorba, Àngel and Zou, Maorong, “A software package for the numerical integration of ODEs by means of high-order Taylor methods”. Exp. Math., 14(1):99–117, 2005.