A review of classical numerical integration methods.

Elena Celledoni

Atlantic Association for Research in the Mathematical Sciences - SUMMER SCHOOL

Structure-preserving discretization of differential equations

Overview

- Existence and uniqueness of solutions. Continuous dependence on initial data.
- Numerical integration methods *one-step methods* vs *multi-step methods*: examples.
- Runge-Kutta methods.
- Order and convergence of one-step methods.
- Linear Multistep methods: convergence.

The lecture is based on the note published on the webpage of the course under *lecture 2* and in particular section 2, 3, 4, 7.

Numerical integration of ODEs

$$\begin{cases} \dot{y}(t) &= f(t,y), \\ y(t_0) &= y_0, \end{cases} \quad t \in [t_0, t_f],$$

seek for approximations of

$$y_1 \approx y(t_1), y_2 \approx y(t_2), \dots y_n \approx y(t_n)$$

at the values of time $t_1, t_2, \dots, t_n$ in $[t_0, t_f]$.

Numerical integration of ODEs

$$\begin{cases} \dot{y}(t) &= f(t,y), \\ y(t_0) &= y_0, \end{cases} \quad t \in [t_0, t_f],$$

seek for approximations of

$$y_1 \approx y(t_1), y_2 \approx y(t_2), \dots y_n \approx y(t_n)$$

at the values of time $t_1, t_2, \dots, t_n$ in $[t_0, t_f]$.

- **One-step methods:**

$$y_{n+1} \text{ is obtained by using only } y_n \text{ and } f.$$

Numerical integration of ODEs

$$\begin{cases} \dot{y}(t) &= f(t,y), \\ y(t_0) &= y_0, \end{cases} \quad t \in [t_0, t_f],$$

seek for approximations of

$$y_1 \approx y(t_1), y_2 \approx y(t_2), \dots y_n \approx y(t_n)$$

at the values of time $t_1, t_2, \dots, t_n$ in $[t_0, t_f]$.

- **One-step methods:**

$$y_{n+1} \text{ is obtained by using only } y_n \text{ and } f.$$

- **Multi-step methods:**

$$y_{n+1} \text{ is obtained by using } y_{n-k+1}, y_{n-k+2}, \dots y_n \text{ and } f.$$

Examples ...

Runge-Kutta methods

$$
\begin{aligned}
y_{n+1} &= y_n + h \sum_{i=1}^{s} b_i K_i \\
K_i &= f\left(t_n + c_i h, y_n + h \sum_{j=1}^{s} a_{i,j} K_j \right)
\end{aligned}
$$

$$
c := \begin{bmatrix} c_1, \\ c_2, \\ \vdots \\ c_s \end{bmatrix}, \quad b := [b_1, b_2, \ldots, b_s], \quad A := \begin{bmatrix} a_{1,1} & a_{1,2} & \ldots & a_{1,s} \\ a_{2,1} & a_{2,2} & \ldots & a_{2,s} \\ \vdots & & \ldots & \vdots \\ a_{s,1} & a_{s,2} & \ldots & a_{s,s} \end{bmatrix}
$$

Runge-Kutta methods

$$
\begin{aligned}
y_{n+1} &= y_n + h \sum_{i=1}^{s} b_i K_i \\
K_i &= f\left(t_n + c_i h, y_n + h \sum_{j=1}^{s} a_{i,j} K_j \right)
\end{aligned}
$$

$$
c := \begin{bmatrix} c_1, \\ c_2, \\ \vdots \\ c_s \end{bmatrix}, \quad b := [b_1, b_2, \ldots, b_s], \quad A := \begin{bmatrix} a_{1,1} & a_{1,2} & \ldots & a_{1,s} \\ a_{2,1} & a_{2,2} & \ldots & a_{2,s} \\ \vdots & & \ldots & \vdots \\ a_{s,1} & a_{s,2} & \ldots & a_{s,s} \end{bmatrix}
$$

All one-step methods can be written in the form

$$
y_{n+1} = \Phi_h(t_n, y_n, h, f)
$$

numerical flow

Convergence of Runge-Kutta methods

DEF: A one-step method is **convergent** of order p if and only if

$$\|y(t_N) - y_N\| = \mathcal{O}(h^p).$$

Convergence of Runge-Kutta methods

DEF: A one-step method is **convergent** of order p if and only if

$$\|y(t_N) - y_N\| = \mathcal{O}(h^p).$$

Consider a one step method: $y_{n+1} = \Phi_h(t_n, y_n, h, f)$.
Local truncation error:

$$d_{n+1}(h) := y(t_{n+1}) - z_{n+1}$$

where

$$z_{n+1} := \Phi_h(t_n, y(t_n), h, f).$$

Convergence of Runge-Kutta methods

DEF: A one-step method is **convergent** of order p if and only if

$$\|y(t_N) - y_N\| = \mathcal{O}(h^p).$$

Consider a one step method: $y_{n+1} = \Phi_h(t_n, y_n, h, f)$.
Local truncation error:

$$d_{n+1}(h) := y(t_{n+1}) - z_{n+1}$$

where

$$z_{n+1} := \Phi_h(t_n, y(t_n), h, f).$$

DEF: A one-step method is **consistent** if and only if

$$\lim_{h \to 0} \frac{|d_{n+1}(h)|}{h} = 0.$$

Convergence of Runge-Kutta methods

DEF: A one-step method is **convergent** of order p if and only if

$$\|y(t_N) - y_N\| = \mathcal{O}(h^p).$$

Consider a one step method: $y_{n+1} = \Phi_h(t_n, y_n, h, f)$.
Local truncation error:

$$d_{n+1}(h) := y(t_{n+1}) - z_{n+1}$$

where

$$z_{n+1} := \Phi_h(t_n, y(t_n), h, f).$$

DEF: A one-step method is **consistent** if and only if

$$\lim_{h \to 0} \frac{|d_{n+1}(h)|}{h} = 0.$$

A one-step method is consistent of order p if and only if

$$|d_{n+1}(h)| = \mathcal{O}(h^{p+1}).$$

Convergence of Runge-Kutta methods

DEF: A one-step method is **convergent** of order p if and only if

$$\|y(t_N) - y_N\| = \mathcal{O}(h^p).$$

Consider a one step method: $y_{n+1} = \Phi_h(t_n, y_n, h, f)$.
Local truncation error:

$$d_{n+1}(h) := y(t_{n+1}) - z_{n+1}$$

where

$$z_{n+1} := \Phi_h(t_n, y(t_n), h, f).$$

DEF: A one-step method is **consistent** if and only if

$$\lim_{h \to 0} \frac{|d_{n+1}(h)|}{h} = 0.$$

A one-step method is consistent of order p if and only if

$$|d_{n+1}(h)| = \mathcal{O}(h^{p+1}).$$

Theorem: A RK method is convergent (of order p) if and only if it is consistent (of order p).

Convergence of Euler's method (exercise)

STEP 1: write the error

$$e_{n+1} := y(t_{n+1}) - y_{n+1}$$

by means of e_n and the local truncation error

$$d_{n+1} := y(t_{n+1}) - z_{n+1}, \qquad z_{n+1} = y(t_n) + hf(t_n, y(t_n))$$

Convergence of Euler's method (exercise)

STEP 1: write the error

$$e_{n+1} := y(t_{n+1}) - y_{n+1}$$

by means of e_n and the local truncation error

$$d_{n+1} := y(t_{n+1}) - z_{n+1}, \qquad z_{n+1} = y(t_n) + hf(t_n, y(t_n))$$

STEP 2: use the Lipschitz condition to find a bound for $\|e_{n+1}\|$, $n = 0, \ldots, N-1$.

Convergence of Euler's method (exercise)

STEP 1: write the error

$$e_{n+1} := y(t_{n+1}) - y_{n+1}$$

by means of e_n and the local truncation error

$$d_{n+1} := y(t_{n+1}) - z_{n+1}, \qquad z_{n+1} = y(t_n) + hf(t_n, y(t_n))$$

STEP 2: use the Lipschitz condition to find a bound for $\|e_{n+1}\|$, $n = 0, \dots, N-1$.

STEP 3: To end the proof use

Lemma

If

$$s_{n+1} \leq (1 + hL)s_n + D, \qquad n = 0, 1, \dots, N-1$$

then

$$s_m \leq (1 + hL)^N s_0 + D\frac{(1 + hL)^N - 1}{hL}, \quad \forall m \leq N,$$

and

$$s_m \leq e^{L(t_f - t_0)} s_0 + D\frac{e^{L(t_f - t_0)} - 1}{hL}, \quad \forall m \leq N.$$

STEP 4: Prove the lemma (if you have more time).

Order conditions for Runge-Kutta methods

We consider autonomous systems of ODEs

$$\begin{cases} \dot{y}(t) &= f(y), \\ y(t_0) &= y_0, \end{cases} \quad t \in [t_0, t_f],$$

a non-autonomous system can always be written as an autonomous one with one more equation adding one extra equation:

$$\dot{t} = 1$$

To decide the order of a Runge-Kutta method we compare the Taylor expansion of the numerical and the exact solution.
The method has order p if the two expansion match up to the p-th power of h.
Order conditions: algebraic conditions on the coefficients of the Runge-Kutta method which need to be satisfied for the method to have a certain order.

Order conditions: B-series

Expanding the exact solution in a Taylor expansion:

$$\begin{aligned} y(h) &= y(0) + h\dot{y}(0) + \frac{h^2}{2!}\ddot{y}(0) + \frac{h^3}{3!}\dddot{y}(0) + \dots \\ &= y(0) + hf(y_0) + \frac{h^2}{2!}f'(y_0)f(y_0) + \frac{h^3}{3!}(f''(f(y_0), f(y_0)) + f'(y_0)f'(y_0)f(y_0)) \end{aligned}$$

The same terms appear in the Taylor expansion of the numerical solution $y_1 \approx y(h)$ but with different coefficients. We call them elementary differentials. Each elementary differential can be associated to a rooted tree τ, we denote the elementary differentials by

$$F(\tau)(y)$$

for $\tau \in T$ a rooted tree. Both the exact and the numerical solution of the system of ODEs admit an expansion in powers of h, indexed on the set of rooted threes T and in terms of elementary differentials.
We rewrite these expansions in the following format

$$B(a, y) = a(\emptyset)y + \sum_{\tau \in T} h^{|\tau|} a(\tau) F(\tau)(y)$$

This is called a B-series (Butcher series).

Linear multistep methods

A k-step method has the form

$$\sum_{i=0}^{k} \alpha_i y_{n+i} = h \sum_{i=0}^{k} \beta_i f_{n+i}, \quad \alpha_k \neq 0, \quad |\alpha_0| + |\beta_0| > 0$$

DEF A LMS is 0-stable if and only if the polynomial

$$\rho(\xi) = \sum_{i=0}^{k} \alpha_i \xi^i$$

is such that

- all roots ξ_j of $\rho(\xi)$ are such that $|\xi_j| \leq 1$;
- roots such that $|\xi_j| = 1$ are simple.

Convergence of multistep methods

DEF A LMS is convergent on $[t_0, t_f]$ iff

$$\lim_{h \to 0} (y(t_f) - y_N) = 0$$

assuming the starting values are such that

$$\lim_{h \to 0} (y(t_0 + ih) - y_i) = 0, \quad i = 0, \ldots, k-1$$

DEF A LMS is consistent if $\sum_{i=0}^{k} \alpha_i = 0$.

Theorem

A LMS is convergent if and only if it is 0-stable and consistent.