Numerical Methods for Differential Equations

Ferran Arqué

2018

Contents

Ordinary Differential Equations

Partial Differential Equations

ORDINARY DIFFERENTIAL EQUATIONS

1 Ordinary Differential Equations. Basic concepts

1.1 Introduction and some notation

Given $y' = f(x, y)$, where $\begin{cases} y(x) \in \mathbb{R}^n \\ f : \mathbb{R} \times \mathbb{R}^n \to \mathbb{R}^n \end{cases}$

Definition. We denote by $y(x)$ the exact solution of the ODE system above.

Definition. y_k is the approximation of $y(x_k)$ (after k steps).

Objective. We want to approximate $y(x)$ within a given interval $[x_0, x_n]$.

We know $\begin{cases} x_0 \\ x_1 \\ x_2 \\ \vdots \\ x_n \end{cases}$ We'd like to know $\begin{cases} y(x_0) \\ y(x_1) \\ y(x_2) \\ \vdots \\ y(x_n) \end{cases}$ We find $\begin{cases} y_0 \\ y_1 \\ y_2 \\ \vdots \\ y_n \end{cases}$ (given by a method)

Definition. $\|y(x_n) - y_n\|$ is the **global error**.

Definition. We define the **local truncation error** as the error caused by one iteration, i.e.

$$LTE = \|y(x_k) - y_k\| \qquad \left(\text{assuming the } localizing\ assumption\text{: } y_{k-1} = y(x_{k-1})\right)$$
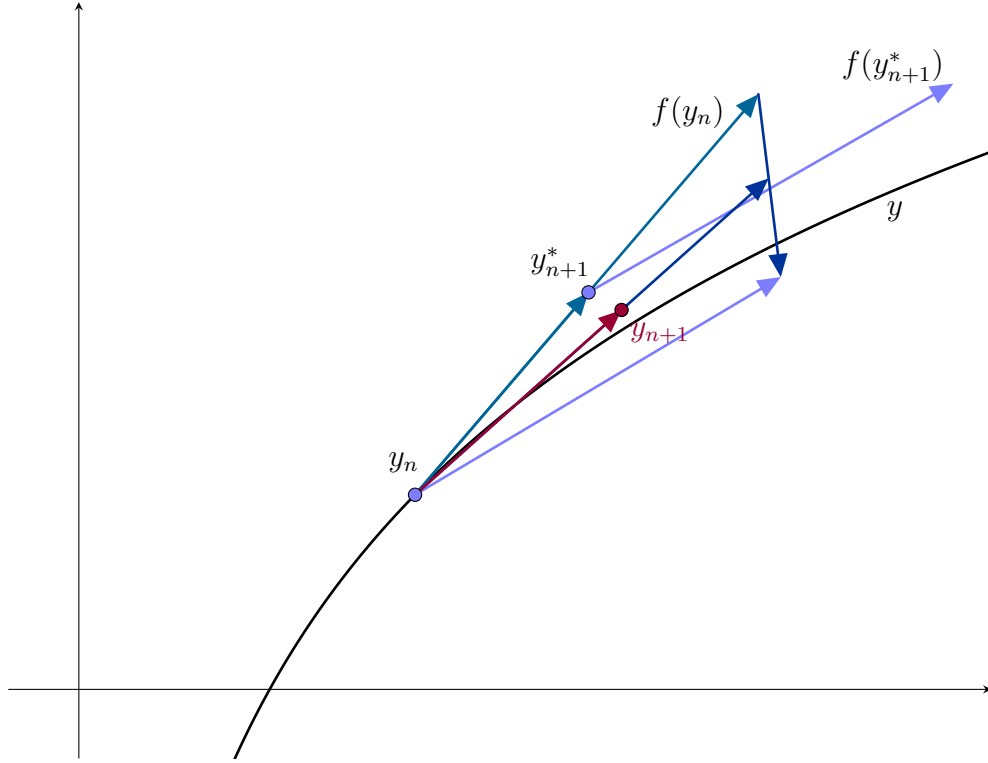
1.2 Euler's method

1.3   Enhanced Euler's method

Figure 1.1: One step of the enhanced Euler's method

Given $y'(x) = f(y(x)), \quad y : \mathbb{R} \to \mathbb{R}, \quad f : \mathbb{R} \to \mathbb{R}$, we'll go through the steps to deduce the enhanced Euler's method with the help of the scheme in Figure 1.1.

The auxiliary point y_{n+1}^* can be found doing one step of the standard Euler's method, so

$$y_{n+1}^* = y_n + h \cdot f(y_n)$$

To get the point y_{n+1} we compute the average vector of $f(y_{n+1}^*)$ and $f(y_n)$, and with this new vector, we can apply again a step of Euler's method, ending up with our method

$$y_{n+1} = y_n + \frac{h}{2} \cdot \left(f(y_{n+1}^*) + f(y_n) \right)$$

Let's find the LTE:

The Local Truncation Error is given by:

$$LTE = \|method - exact\ solution\|$$

Then

$$LTE = \|y_{n+1} - y(x_{n+1})\| = \left\|y_n + \frac{h}{2}f(y_n) + \frac{h}{2}f(y_{n+1}^*) - \underbrace{y(x_{n+1})}_{y(x_n+h)}\right\| = (*)$$

Applying Taylor on $f(y_{n+1}^*) = f\left(y_n + hf(y_n)\right)$, we have

$$f\left(y_n + hf(y_n)\right) = f(y_n) + hf(y_n)f'(y_n) + \mathcal{O}(h^2)$$

and on $y(x_n + h)$ we have

$$y(x_n + h) = y(x_n) + h \cdot y'(x_n) + \frac{h^2}{2} \cdot y''(x_n) + \mathcal{O}(h^3)$$

(In this case, we expand to second order for later simplifications)

$$(*) = \left\|y_n + \frac{h}{2}f(y_n) + \frac{h}{2}\left(f(y_n) + hf(y_n)f'(y_n) + \mathcal{O}(h^2)\right) - \left(y(x_n) + h \cdot y'(x_n) + \frac{h^2}{2} \cdot y''(x_n) + \mathcal{O}(h^3)\right)\right\| \quad (1)$$

Now, given $y'(x) = f(y(x))$, we have

$$y''(x) = f'(y(x)) \cdot y'(x)$$
$$= f'(y(x)) \cdot f(y(x))$$

and we can rewrite the following expression as:

$$\frac{h^2}{2}f(y(x))f'(y(x)) = \frac{h^2}{2}y''(x)$$

With that and the localising assumption ($y_n = y(x_n)$), we can simplify most of the terms in (1) and we end up with

$$\left\|\cancel{y_n} + \cancel{\frac{h}{2}f(y_n)} + \cancel{\frac{h}{2}f(y_n)} + \cancel{\frac{h^2}{2}f(y_n)f'(y_n)} + \mathcal{O}(h^3) - \left(\cancel{y(x_n)} + \cancel{h \cdot y'(x_n)} + \cancel{\frac{h^2}{2}y''(x_n)} + \mathcal{O}(h^3)\right)\right\| = \mathcal{O}(h^3)$$

So LTE $= \mathcal{O}(h^3)$

Remark. Of course, this method also works for $y : \mathbb{R} \to \mathbb{R}^n$, $f : \mathbb{R}^n \to \mathbb{R}^n$

1.4   Final remarks

- There's an improved Euler's method of order 2 similar to the previous one:

$$y_{n+1} = y_n + hf\left(\frac{y_{n+1}^* + y_n}{2}\right)$$

- If we have a method of order ≥ 2, and we want the value of $y(x^*)$, with x^* off the mesh $(x^* \neq k \cdot h)$, we have some options:

   -Step back and take a step with the right h.

   -Interpolate with the right order.

   -Use a continuous Runge-Kutta method.

2 Runge-Kutta and Linear Multistep Methods

2.1 General Runge-Kutta methods

2.1.1 Embedded R-K

2.2 Linear multistep methods

2.2.1 Generalities

2.2.2 Predictor-Corrector method

2.2.3 Richardson's extrapolation

2.2.4 Convergence of a linear multistep method

Let's see an example of divergence using a linear multistep method:

Example

Given the method

$$y_{n+2} + a_1 y_{n+1} + a_0 y_n = h(b_1 f_{n+1} + b_0 f_n)$$

1) Find a_0, a_1, b_0, b_1 so that the method above has the highest possible order.

2) Try it on

$$\begin{cases} y' = -y \\ y(0) = 1 \end{cases} \qquad (y_0 = 1, y_1 = e^{-h})$$

and prove the method diverges.

1) We want $y(x_n + 2h) - y_{n+2}$

We assume $y_{n+1} = y(x_n + h), y_n = y(x_n)$ (localizing assumption).

$$y(x_n + 2h) - y_{n+2} = y(x_n + 2h) - \left[-a_1 y_{n+1} - a_0 y_n + h\big(b_1 f(y_{n+1}) + b_0 f(y_n)\big) \right] \underset{\text{loc.as.}}{=}$$

$$= y(x_n + 2h) - \left[-a_1 y(x_n + h) - a_0 y(x_n) + hb_1 \underbrace{f\left(y(x_n + h)\right)}_{y'(x_n+h)} + hb_0 \underbrace{f\left(y(x_n)\right)}_{y'(x_n)} \right]$$

As usual, we expand in powers of h. We'll expand to order 3

$$y(x_n) + 2hy'(x_n) + \frac{4h^2}{2}y''(x_n) + \frac{8h^3}{6}y'''(x_n) + o(h^4) -$$

$$- \left[-a_1 \left(y(x_n) + hy'(x_n) + \frac{h^2}{2}y''(x_n) + \frac{h^3}{6}y'''(x_n) + o(h^4) \right) - \right.$$

$$- a_0 y(x_n)$$

$$+ hb_1 \left(y'(x_n) + hy''(x_n) + \frac{h^2}{2}y'''(x_n) + o(h^3) \right) +$$

$$\left. + hb_0 y'(x_n) \right]$$

Let's group by powers of h and assume the right conditions to obtain the highest possible order:

$$h^0 \longrightarrow y(x_n) + a_1 y(x_n) + a_0 y(x_n) = 0$$

$$h^1 \longrightarrow 2hy'(x_n) + a_1 hy'(x_n) - hb_1 y'(x_n) - hb_0 y'(x_n) = 0$$

$$h^2 \longrightarrow 2h^2 y''(x_n) + a_1 \frac{1}{2} h^2 y''(x_n) - b_1 h^2 y''(x_n) = 0$$

$$h^3 \longrightarrow \frac{8h^3}{6} y'''(x_n) + a_1 \frac{h^3}{6} y'''(x_n) - b_1 h \left(\frac{h^2}{2} y'''(x_n) \right) = 0$$

With that, we get the system of equations

$$\begin{cases} 1 + a_1 + a_0 = 0 \\ 2 + a_1 - b_1 - b_0 = 0 \\ 2 + \dfrac{a_1}{2} - b_1 = 0 \\ \dfrac{8}{6} + \dfrac{a_1}{6} - \dfrac{b_1}{2} = 0 \end{cases}$$

And we end up with

$$a_0 = -5, \quad a_1 = 4, \quad b_0 = 2, \quad b_1 = 4$$

2) Our method is
$$y_{n+2} + 4y_{n+1} - 5y_n = h(4f_{n+1} + 2f_n)$$
and with
$$\begin{cases} y' = -y \\ y(0) = 1, \; y(h) = e^{-h} \end{cases} \qquad (y(x) = e^{-x})$$
we have

$$y_{n+2} + 4y_{n+1} - 5y_n = h(-4y_{n+1} - 2y_n)$$

We'll find a solution of the form

$$y_n = c_1 (\quad)^n + c_2 (\quad)^n$$

and we'll see that it diverges.

$$\lambda^2 + 4\lambda - 5 + 4h\lambda + 2h = 0$$
$$\lambda^2 + (4(1 + h))\lambda + (2h - 5) = 0$$

$$\lambda = \frac{-4(1+h) \pm \sqrt{4^2(1+h)^2 - 4(2h-5)}}{2}$$

Let's expand the discriminant

$$\sqrt{4^2(1+2h+h^2) - 8h + 20} = \sqrt{36 + 24h + 16h^2} = 6\sqrt{1 + \frac{4}{6}h + \frac{4^2}{6^2}h^2} \underset{\text{Taylor}}{=}$$

$$= 6\left(1 + \frac{1}{2}\left(\frac{4}{6}h + \frac{4^2}{6^2}h^2\right) + o(h^2)\right) =$$

$$= 6\left(1 + \frac{1}{3}h + o(h^2)\right)$$

So

$$\lambda = \frac{-4 - 4h \pm (6 + 2h + o(h^2))}{2} = \begin{array}{c} \nearrow \quad 1 - h + o(h^2) \\ \\ \searrow \quad -5 - 3h + o(h^2) \end{array}$$

$$\implies y_n = c_1\left(1 - h + o(h^2)\right)^n + c_2\left(-5 - 3h + o(h^2)\right)^n$$

Let's find c_1 and c_2 imposing the initial conditions

$$\begin{cases} 1 = c_1 + c_2 \implies c_1 = 1 - c_2 \\ e^{-h} = c_1\left(1 - h + o(h^2)\right) + c_2\left(-5 - 3h + o(h^2)\right) \end{cases}$$

$$\implies e^{-h} = (1 - h + o(h^2)) + c_2\left(\underbrace{-5 - 3h + o(h^2) - 1 + h + o(h^2)}_{-6-2h+o(h^2)}\right)$$

$$\implies c_2 = \frac{e^{-h} - 1 + h + o(h^2)}{-6 - 2h + o(h^2)} \underset{\text{Taylor } e^{-h}}{=} \frac{1 - h + o(h^2) - 1 + h + o(h^2)}{-6 - 2h + o(h^2)} \simeq \frac{-1}{6 + 2h}$$

$$\implies c_1 \simeq 1 + \frac{1}{6 + 2h} = \frac{7 + 2h}{6 + 2h}$$

So

$$y_n = \frac{7 + 2h}{6 + 2h}\left(1 - h + o(h^2)\right)^n + \frac{-1}{6 + 2h}\left(-5 - 3h + o(h^2)\right)^n$$

and the term $(-5)^n$ will cause the solution to diverge.

3 Stiff Problems

PARTIAL DIFFERENTIAL EQUATIONS

4 Partial Differential Equations. Generalities on their solution
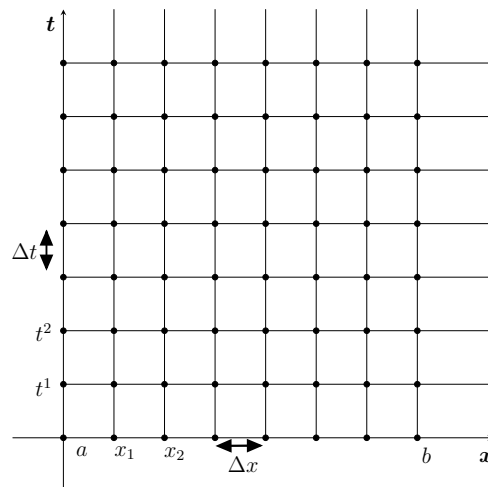
4.1 Finite differences

Example (1D Heat equation)

Our problem is:

$$\begin{cases} u_t - u_{xx} = f \\ u(x,0) = u_0(x) \quad \longleftarrow \text{ Initial condition (IC)} \\ \left.\begin{array}{l} u(a,t) = u_a \\ u(b,t) = u_b \end{array}\right\} \quad \longleftarrow \text{ Boundary conditions (BC)} \end{cases}$$
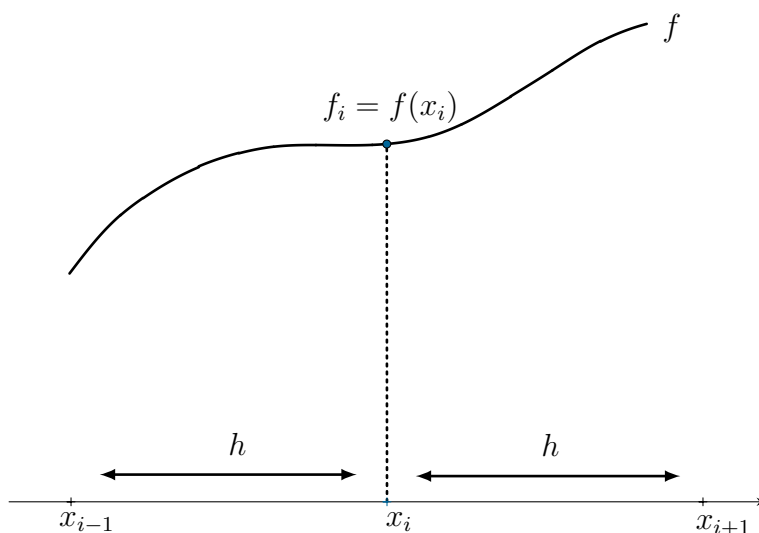
With $t \geq 0$, $x \in [a,b]$

We discretize x and t:

Idea: We'll impose $U_t(x_i, t^n) - U_{xx}(x_i, t^n) = f(x_i, t^n)$

4.1.1 Numerical derivatives

We'll use Taylor's expansion on f to find an approximation for the derivatives:

(1) $f_{i+1} = f_i + h f_i' + \frac{h^2}{2} f_i'' + \frac{h^3}{3!} f_i''' + \ldots$

(2) $f_{i-1} = f_i - h f_i' + \frac{h^2}{2} f_i'' - \frac{h^3}{3!} f_i''' + \ldots$

Approximation of the first derivative:

- Forward differences (1)
$$f_i' = \frac{f_{i+1} - f_i}{h} + \mathcal{O}(h)$$

- Backwards differences (2)
$$f_i' = \frac{f_i - f_{i-1}}{h} + \mathcal{O}(h)$$

- Centered differences (1)-(2)
$$f_i' = \frac{f_{i+1} - f_{i-1}}{2h} + \mathcal{O}(h^2)$$

Remark. With centered differences, it converges faster, but sometimes the other two methods are preferable for ease in computations.

<u>Approximation of the second derivative</u> (1)+(2)

$$f_i'' = \frac{f_{i+1} - 2f_i + f_{i-1}}{h^2} + \mathcal{O}(h^2)$$

There are higher order approximations, but these are the standard, and are more than enough.

Now, we'll present some notation to differentiate between the problem we want to solve and the numerical problem.

Definition.

The **continuous problem** is $\begin{cases} \mathcal{L}(u) = f & \text{in } \Omega,\, t > 0 \\ \mathcal{B}(u) = q & \text{in } \partial\Omega,\, t > 0 \\ u(x, 0) = u_0(x) \end{cases}$

Where $\mathcal{L}$ is any differential operator.

It's what we want to solve.

The **discrete problem** is $\begin{cases} L(u) + \tau = f \\ B(u) + \bar{\tau} = q \\ u(x, 0) = u_0(x) \end{cases}$

The **numerical problem** is $\begin{cases} L(U) = f \\ B(U) = q \\ U(x, 0) = u_0(x) \end{cases}$

Let's go back to our 1D heat equation:

The continuous problem is

$$\begin{cases} u_t - u_{xx} = f & t > 0,\, x \in (a, b) \\ u(a, t) = u_a \\ u(b, t) = u_b \\ u(x, 0) = u_0(x) \end{cases}$$

<u>Notation:</u> $u(x_i, t^n) = u_i^n$
$x_i = a + i\Delta x \quad i = 0, \ldots N \quad x_0 = a,\, x_N = b \qquad$ (sometimes $\Delta x = h$).
$t^n = n\Delta t \, n \geq 0$

We have the following explicit method:

4.1.2   Forward Time Centered Space method (FTCS)

- Approximate u_t using forward differences (FT)

- Approximate u_{xx} using centered differences (CD)

Our discretized problem is

$$\frac{u_i^{n+1} - u_i^n}{\Delta t} + \mathcal{O}(\Delta t) - \frac{u_{i+1}^n - u_i^n + u_{i-1}^n}{\Delta x^2} + \mathcal{O}(\Delta x^2) = f_i^n$$

We define $\tau = \mathcal{O}(\Delta t, \Delta x^2)$

Neglecting τ, the numerical problem we want to solve is

$$\begin{cases} \dfrac{U_i^{n+1} - U_i^n}{\Delta t} - \dfrac{U_{i+1}^n - U_i^n + U_{i-1}^n}{\Delta x^2} = f_i^n \\ U_0^{n+1} = u_a(t^{n+1}) \\ U_N^{n+1} = u_b(t^{n+1}) \\ U_i^0 = u_0(x_i) \end{cases}$$

Now we define r as

$$r = \frac{\Delta t}{\Delta x^2}$$

So

$$U_i^{n+1} = rU_{i-1}^n + (1 - 2r)U_i^n + rU_{i+1}^n + \Delta t f_i^n \qquad i = 1, \dots, N-1$$

For $i = 1$,

$$U_1^{n+1} = (1 - 2r)U_1^n + rU_2^n + \Delta t f_1^n + ru_a^n$$

And for $i = N - 1$,

$$U_{N-1}^{n+1} = rU_{N-2}^n + (1 - 2r)U_{N-1}^n + \Delta t f_{N-1}^n + ru_b^n$$

Thus,

$$\overline{U^{n+1}} = \underset{\sim}{B}\,\overline{U^n} + \overline{F} + \overline{G}$$

Where

$$\overline{U^p} = \left(U_1^p, U_2^p, \dots, U_{N-1}^p\right)^T \qquad \overline{F} = \Delta t \left(f_1^n, f_2^n, \dots, f_{N-1}^n\right)^T \qquad \overline{G} = (ru_a^n, 0, \dots, 0, ru_b^n)^T$$

$$\underset{\sim}{B} = \begin{pmatrix} 1-2r & r & & & & \\ r & 1-2r & r & & & \\ & \ddots & \ddots & \ddots & & \\ & & r & 1-2r & r & \\ & & & r & 1-2r \end{pmatrix}$$

16

5 Numerical Solution of PDEs with the Finite Difference Method

6 Introduction to Boundary Value Problems

7 Quality Control of Solutions