Hydraulique à surface libre

TD2: Numerical integration, discretization and schemes for solving differential equations

Professor: C. Ancey
Teaching assistant: T. Trewhela

ENAC / LHE - 2020

Aim: to learn how to discretize and numerically solve differential equations (including ordinary and partial differential equations)

1. Concepts
   - Stability
   - Convergence
   - Computation times and efficiency
2. Notions on numerical integration
3. Numerical discretization
   - Forward
   - Backward
   - Centered
4. Advection diffusion equation solution
   - Explicit/Implicit schemes

Stability

A numerical solution is called *stable* when numerical errors do not increase in the course of the numerical computation. For iterative methods, a stable method is one for which the errors do not diverge to a point that they would be no longer bounded.

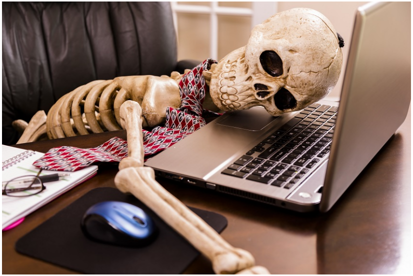
We will address stability in each problem we solve during this session, and it is an important issue to address every time one solve an equation numerically.

A numerical solution converges when the found solution tends to the exact solution as the grid size tends to 0.

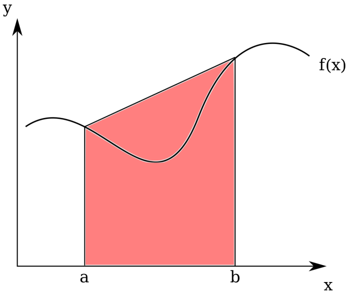
When the exact solution is unknown, the convergence must be proven with various solutions using different grid sizes. If the solutions tend to fixed values and are consistent, then the solution converges.

In all the exercises, convergence must be verified.

Some numerical methods take more time than others. There are several ways to improve computational times, such as vectorization for example.



In Matlab, it is highly advised to vectorize and avoid the usage of *for/while* cycles. Computational time can be evaluated using the *tic/toc* commands.



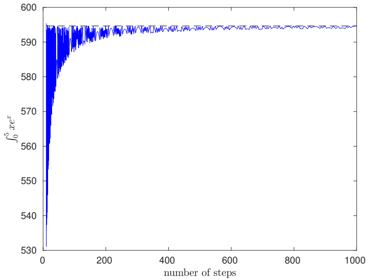
By estimating the Riemann sums numerically, it is possible to approximate a definite integral using the trapeze rule:

$$\int_a^b f(x)dx \approx (b-a)\frac{f(a)+f(b)}{2} \tag{1}$$

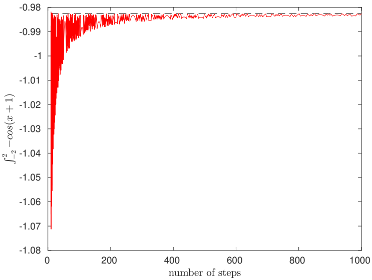
Estimate the integral of the following functions.

1. $xexp(x)$ between 0 and 5
2. $-cos(x+1)$ between -2 and 2

Analyze the convergence of the integration using different step sizes.



600
590
580
570
560
550
540
530
$\int_0^5 xe^x$
0
200
400
600
800
1000
number of steps



-0.98
-0.99
-1
-1.01
-1.02
-1.03
-1.04
-1.05
-1.06
-1.07
-1.08
$\int_{-2}^2 -cos(x+1)$
0
200
400
600
800
1000
number of steps

The Runge-Kutta fourth-order method is a well-known and widely used solver for differential equations.

$$k_1 = f(x_i, y_i) \tag{2}$$

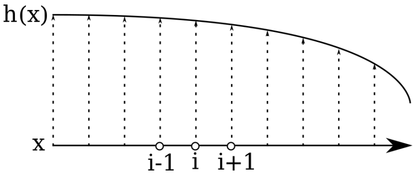
$$k_2 = f(x_i + \frac{1}{2}\Delta x, y_i + \frac{1}{2}k_1\Delta) \tag{3}$$

$$k_3 = f(x_i + \frac{1}{2}\Delta x, y_i + \frac{1}{2}k_2\Delta) \tag{4}$$

$$k_4 = f(x_i + \Delta x, y_i + k_3\Delta) \tag{5}$$

$$y_{i+1} = y_i + h/6(k_1 + 2k_2 + 2k_3 + k_4) \tag{6}$$

Use the Runge-Kutta method to estimate the water-height profile for a rectangular channel of width $B = 0.5$ m and slope $i = 0.05\%$ for a water flow rate $Q = 1.2\ \mathrm{m^3/s}$. Consider a Manning-Strickler coefficient of $K = 55\ \mathrm{m^{1/3}s^{1}}$.



1. Determine the hydraulic regime. To that end, use the Manning-Strickler equation

$$\frac{QK}{\sqrt{i}} = A\ R_h^{2/3} \tag{7}$$

2. Alternatively you can approximate the flow depth using the approximation of infinitely wide channel

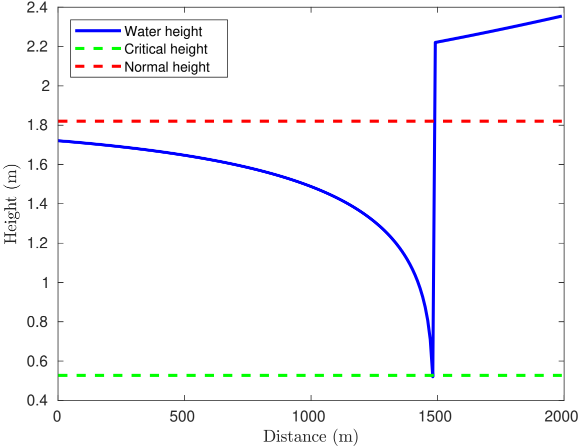
$$h_n = \left(\frac{q}{K\sqrt{i}}\right)^{3/5} \tag{8}$$

3. Determine the critical height $h_c = (q^2/g)^{1/3}$, the hydraulic regime, the boundary conditions and then implement the RK4 method for the equation

$$\frac{dh}{dx} = i\frac{(h_n/h)^{10/3} - 1}{(h_c/h)^3 - 1} \tag{9}$$

4. Evaluate the convergence by changing the size of Δ

Save the results for further comparison.



What are your conclusions on the results?

Based on Taylor's series, a function can be expressed as

$$f(x+h) \approx f(x) + hf'(x) + \frac{1}{2}h^2 f''(x) + \ldots \tag{10}$$

where we can approximate a derivative of order 1 in three different ways

1. Forward step

$$f'(x) \approx \frac{f(x+h) - f(x)}{h} \tag{11}$$

2. Backward step

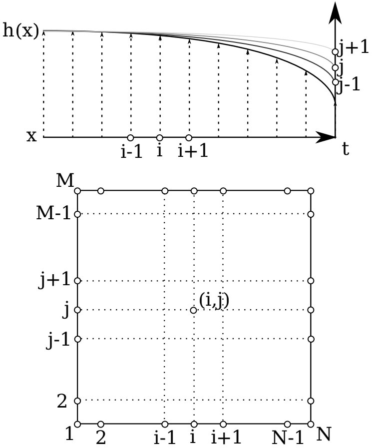
$$f'(x) \approx \frac{f(x) - f(x-h)}{h} \tag{12}$$

3. 2nd order centered step

$$f'(x) \approx \frac{f(x+h) - 2f(x) + f(x-h)}{h^2} \tag{13}$$

Solve the water-height profile using a forward and a centered scheme. What are the differences?

If we consider now a two-dimensional function (e.g. varying in time and space), we can discretize in both dimensions. Usually we use different subscripts to designate each variation or iteration for each dimension. This creates a grid where the numerical solution must be implemented in every point or element of the grid.



Numerical grid and discretization

A widely-studied equation on fluid dynamics is the advection-diffusion equation

$$\frac{\partial C}{\partial t} + v\frac{\partial C}{\partial x} = D\frac{\partial^2 C}{\partial x^2} \tag{14}$$

Discretize the equation using a forward approximation, once for both time, t, and twice for length, x. By using i as the subscript for domain x and j as the subscript for domain t, the equation can be written explicitly for $C_{i+1,j}$. Consider Δt as the spacing for t and Δx as the spacing for x. Do you see a problem or restriction for the equation in terms of stability? For an answer, think on errors related to the grid and how they may propagate.

Discretized advection-diffusion equation

$$C_{i,j+1} = C_{i,j} - v\frac{\Delta t}{\Delta x}(C_{i+1,j} - C_{i,j}) + D\frac{\Delta t}{\Delta x^2}(C_{i+1,j} - 2Ci, j + C_{i-1,j}) \tag{15}$$

Two parameters appear controlling the equation stability

$$Co = v\frac{\Delta t}{\Delta x} \quad \text{and} \quad v_N = D\frac{\Delta t}{\Delta x^2} \tag{16}$$

These are called, Courant and von Neumann stability conditions.

Now we can solve the system by using the following boundary conditions

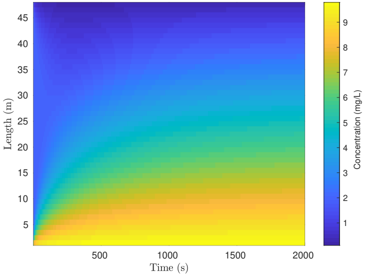
$$C(x, t = 0) = 2 \text{ mg/L} \quad (17)$$

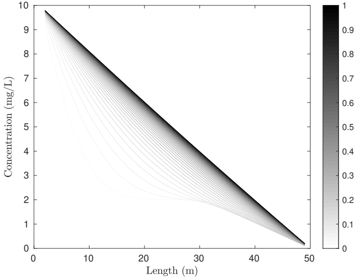
$$C(x = 0 \text{ m}, t) = 10 \text{ mg/L} \quad (18)$$

$$C(x = 50 \text{ m}, t) = 0 \text{ mg/L} \quad (19)$$

Consider also $D = 1 \text{ m}^2/\text{s}$ and $v = 0.01$ m/s. After solving compare your results to another velocity $v = 0.5$ m/s. How you adapt your code?

Explicit scheme: Results





An alternative is to write the advection-diffusion equation using backward finite differences.
Rewrite the equation using backward finite differences for both time and space. What is the difference from the previous case?
Solve your equations using the same conditions as for the explicit scheme. Hint: Write your problem in matrix form.
Compare your results to the previous scheme in terms of stability, calculation time. What do you think about these alternatives?

The implicit scheme can be expressed as

$$\frac{C_{i,j} - C_{i,j-1}}{\Delta t} = -v\frac{\Delta t}{\Delta x}(C_{i+1,j} - C_{i,j}) + D\frac{\Delta t}{\Delta x^2}(C_{i+1,j} - 2Ci,j + C_{i-1,j}) \quad (20)$$

which can be rewritten in matrix form

$$\mathbf{AC_j} = \mathbf{C_{j-1}} \quad (21)$$

where $A_{i,i} = 1 + 2v_N + Co$ and $A_{i,i1} = A_{i,i+1} = v_N + Co$. Don't forget the boundary conditions!