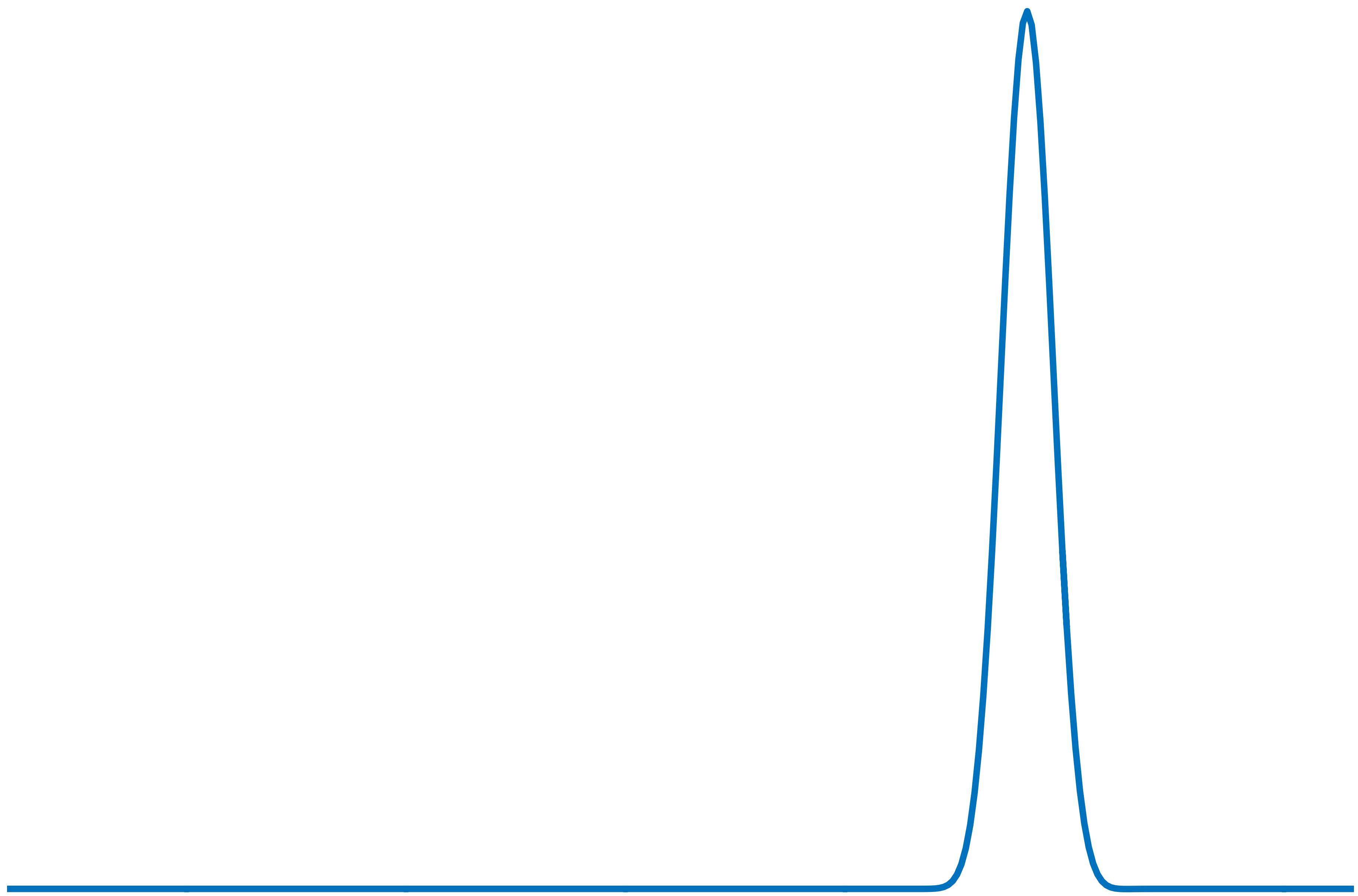
Numerical simulation of 1D convection equation



Teaching Assistants:

- P. Balty - pierre.balty@uclouvain.be
- G. Poncelet - gilles.poncelet@uclouvain.be

Teacher:

- G. Winckelmans

UCL Université catholique de Louvain

LMECA2660 Homework

Goals

- Validation of the periodic approximation
- Production of a C code to
  - ★ investigate convection phenomena,
  - ★ investigate the numerical properties of different discretisation schemes of the temporal and convective terms.
  - ★ Investigate of the numerical properties of a stretched mesh
  - ★ Investigate the convection of wave packets

Problem statement

- Convection equation (1D-problem):

$$\frac{\partial u}{\partial t} + c\frac{\partial u}{\partial x} = 0$$

with a Gaussian function as initial condition

$$u(x,0) = U\ exp(-\frac{x^2}{\sigma^2})$$

- Analytical solution (initial function moving at a constant velocity c)

$$u(x,t) = U\ exp\left(-\frac{(x-ct)^2}{\sigma^2}\right)$$

Periodic approximation

- The Gaussian function has an unbounded support

  → impossible to solve that problem numerically

- We use a periodic domain of period L instead of an unbounded domain

  → valid approximation as long as $L >> \sigma$

The first part of the homework will allow to assess the validity of this approximation

Reminder on Fourier Transforms and Fourier Series

- Fourier Transform (continuous and non periodic function)

The Fourier transform $\widehat{f}(k)$ of a function $f(x)$ is defined as :

$$\widehat{f}(k) = \mathcal{F}\left(f(x)\right) \quad = \quad \int_{-\infty}^{\infty} f(x) \, \exp\left(-\imath k x\right) dx \;,$$

while the inverse transform is defined as :

$$f(x) = \mathcal{F}^{-1}\left(\widehat{f}(k)\right) \quad = \quad \frac{1}{2\pi} \int_{-\infty}^{\infty} \widehat{f}(k) \, \exp\left(\imath k x\right) dk \;.$$

- For a **periodic** signal: Fourier series (discrete)

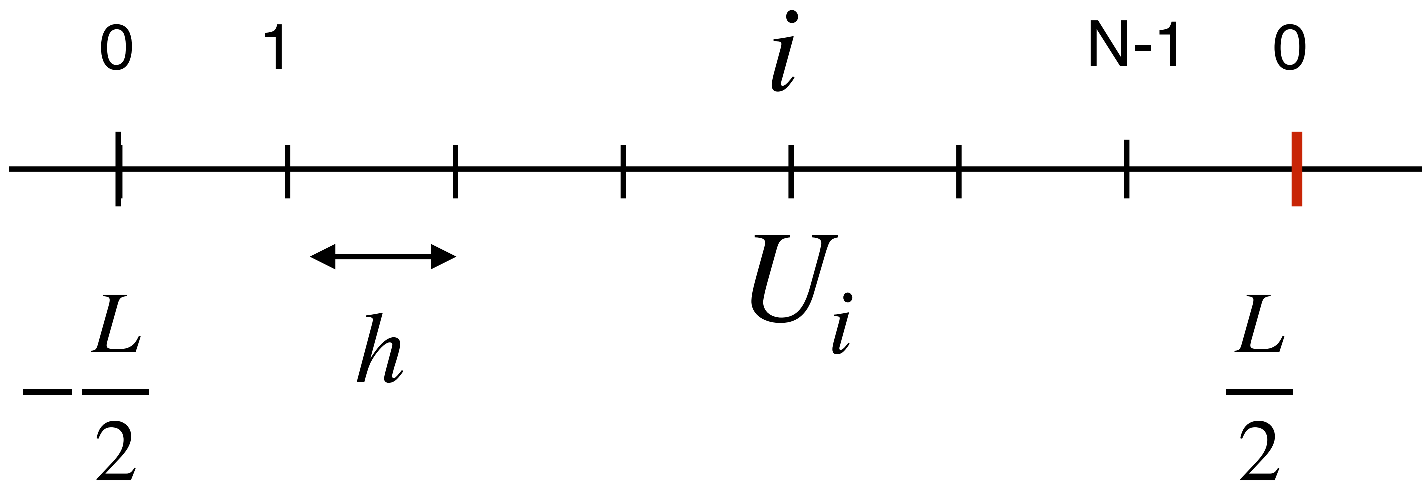
$$f(x) \quad = \quad \sum_{j=-\infty}^{\infty} \widehat{F}(k_j) \exp(\imath k_j x) \;,$$

$$\text{where} \quad \widehat{F}(k_j) \quad = \quad \frac{1}{L} \int_{-L/2}^{L/2} f(x) \, \exp(-\imath k_j x) \, dx$$

Reminder on Fourier Transforms and Fourier Series

- Discretization of the function

Spatial domain



$$x_i = -\frac{L}{2} + ih$$

- Discrete and periodic Fourier Series

Spectral domain

$$f_i = f(x_i) = \sum_{j=-N/2}^{N/2} \hat{F}(k_j) exp(\imath k_j x_i),$$

$$\text{where } \hat{F}_j = \hat{F}(k_j) = \frac{1}{N} \sum_{i=0}^{N-1} f(x_i) exp(-\imath k_j x_i)$$

Discrete Fourier Series in Matlab/Python

- The Fourier coefficients $\hat{F}(k)$ may be obtained using the *fft* function in Matlab or Python.
- Caution: the definition differs and the result has to be divided by N (see the documentation)
- The N coefficients are provided in the following form:

Indices of the vector returned by the fft function

1	2		N/2	N/2+1	N/2+2		N
$\hat{F}(k_0)$	$\hat{F}(k_1)$	$\cdots$	$\hat{F}(k_{N/2-1})$	$\hat{F}(k_{N/2})$	$\hat{F}(k_{-1})$	$\cdots$	$\hat{F}(k_{-(\frac{N}{2}-1)})$

Zero mode (purely real) *Flip-flop mode (purely real)*

Produce a C-code

$$\boxed{\frac{\partial u}{\partial t}} + \boxed{c\frac{\partial u}{\partial x}} = 0$$

Temporal integration: RK4

Convective term:
- *Second order explicit (E2)*
- *E4*
- *Fourth order implicit (I4)*

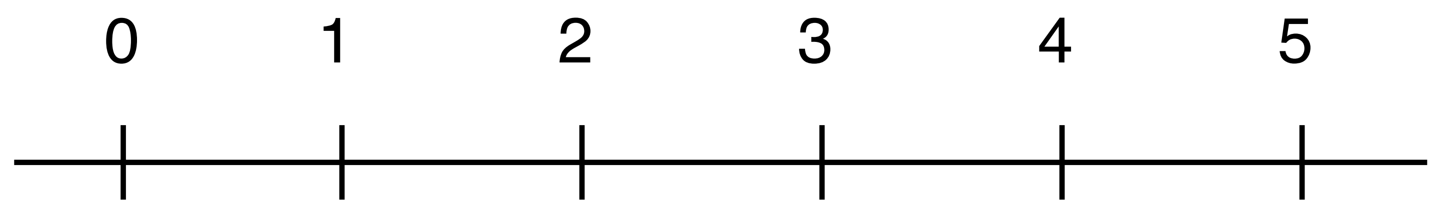
- Lost in C? -> **Google** your issue!

  - http://stackoverflow.com

  - https://openclassrooms.com/courses/apprenez-a-programmer-en-c (french only!)

FDs and periodic domain

- Spatial discretization: N points numbered from 0 to N-1

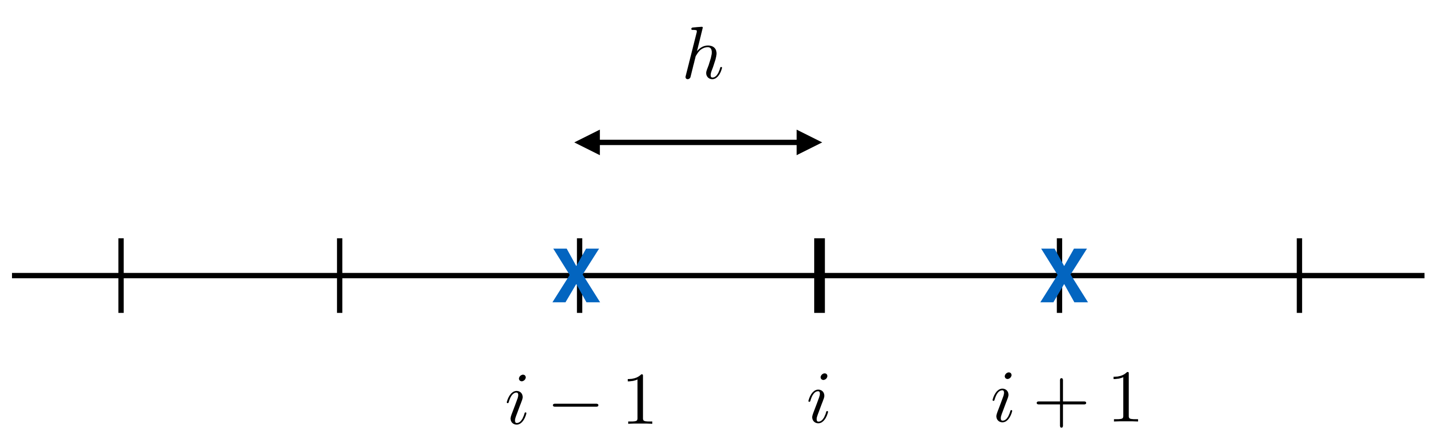


N = 6

- Computation of the first order derivative using a E2 scheme

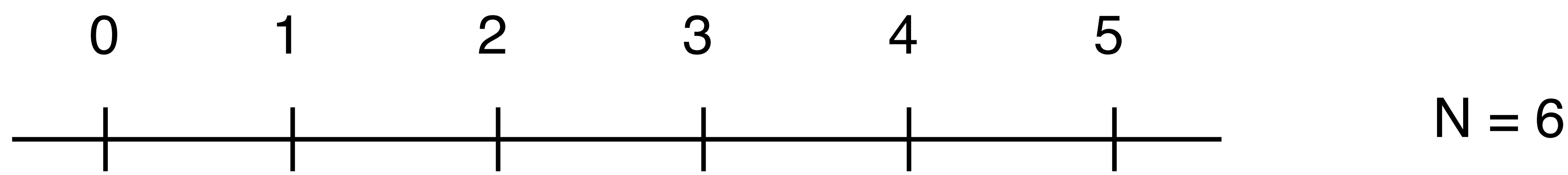
$$\left.\frac{\partial u}{\partial x}\right|_i = \frac{U_{i+1} - U_{i-1}}{2h} + \mathcal{O}(h^2)$$

  - ★ Point in the middle of the domain



FDs and periodic domain

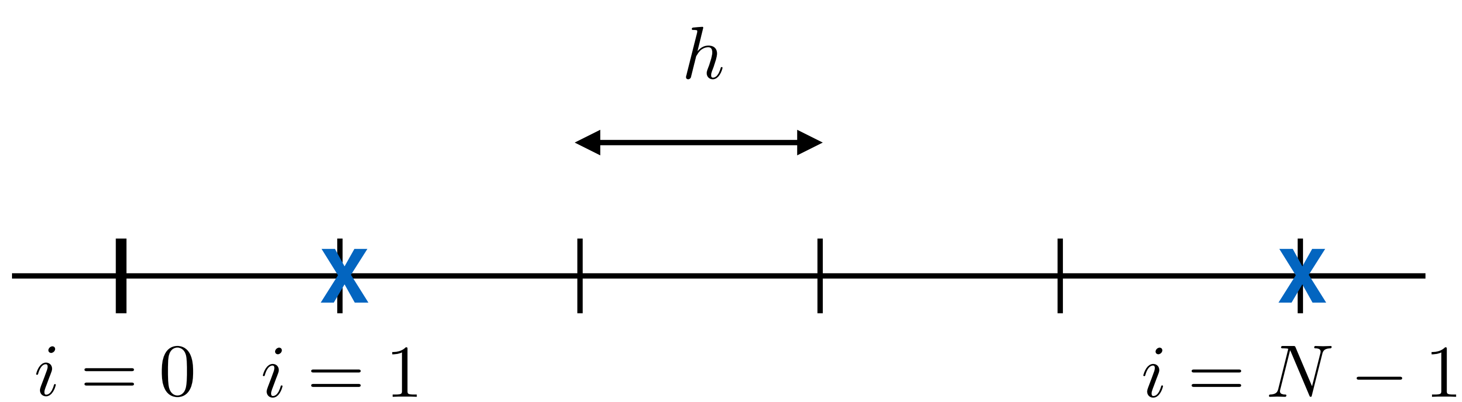
- Spatial discretization: N points numbered from 0 to N-1



- Computation of the first order derivative using a E2 scheme

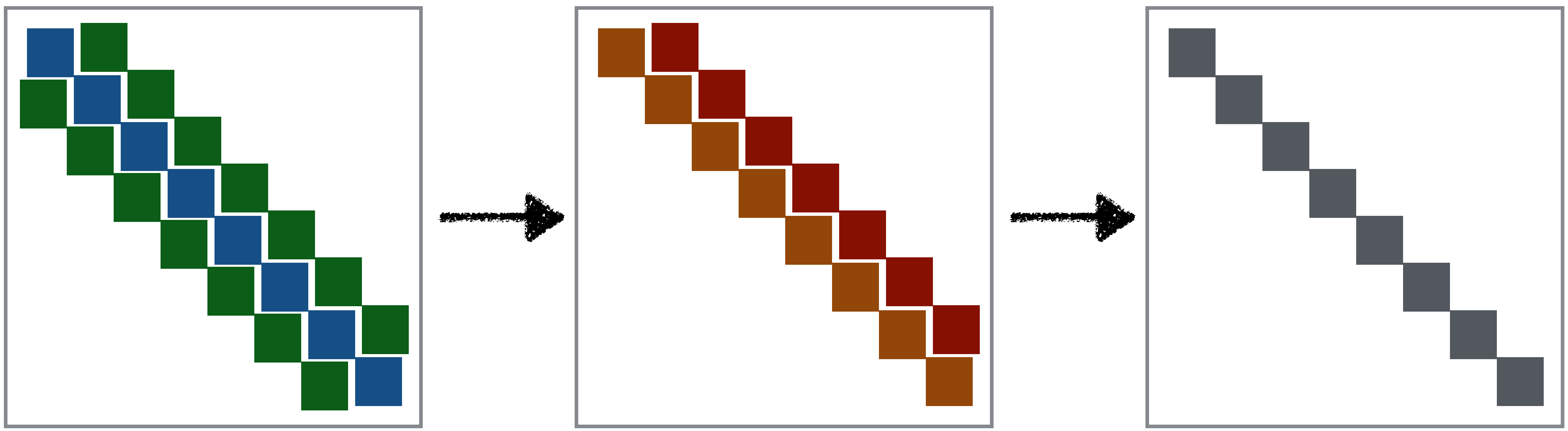
$$\left.\frac{\partial u}{\partial x}\right|_i = \frac{U_{i+1} - U_{i-1}}{2h} + \mathcal{O}(h^2)$$

  - ★ Point on the edge of the domain (periodic condition)



Implicit scheme — Thomas algorithm

Forward pass **Backward pass**



+ Little trick to use Thomas Algorithm for a periodic problem

C language: dos and don’ts

- Periodic domain: modulo is your best friend!

```
0  % 10 = 0
1  % 10 = 1
11 % 10 = 1
```

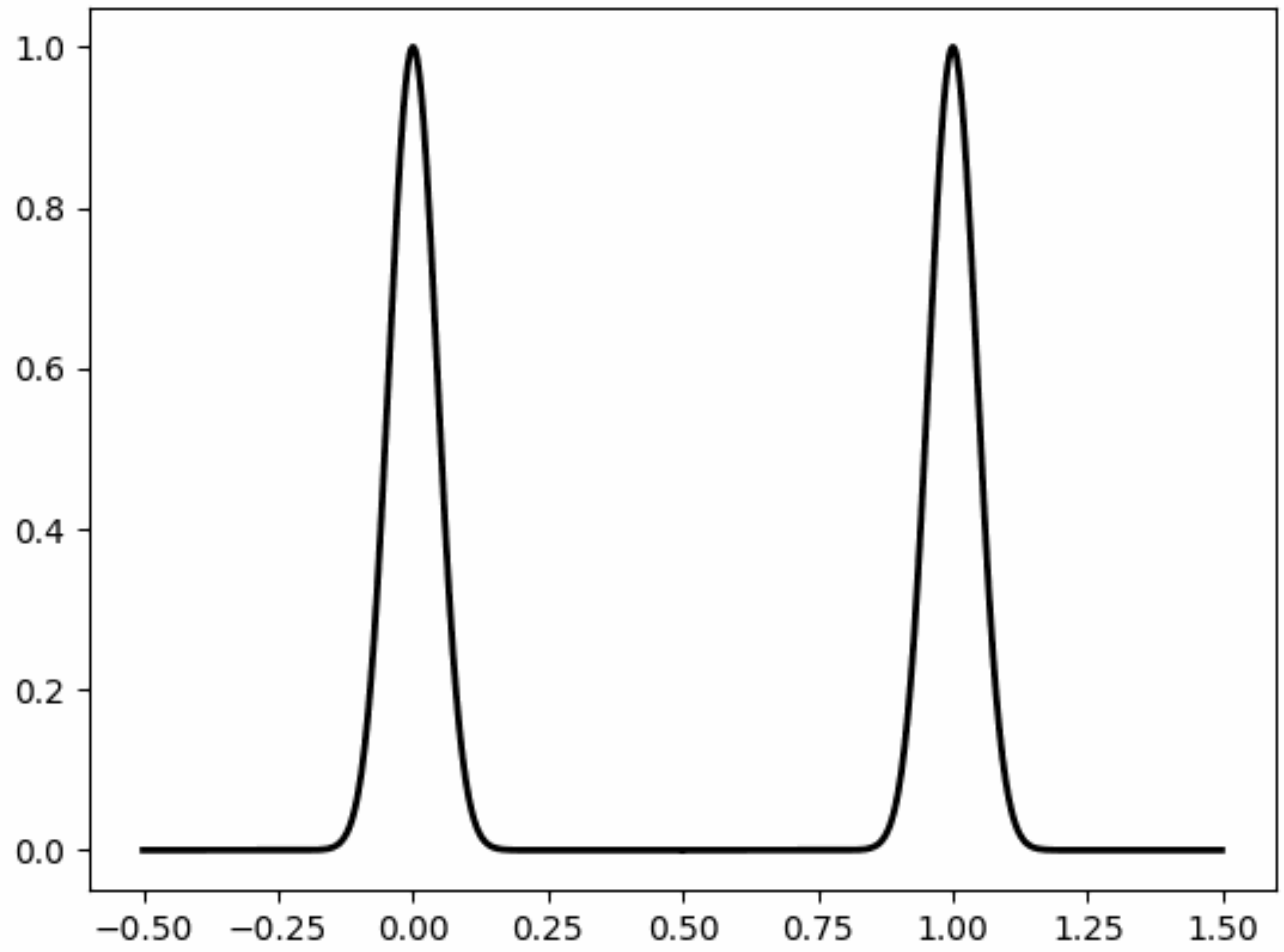
- never do a *if* condition inside a *for* loop

```
for (i=0; i<n; ++i){
   if (i==0) ….
   else ….
}
```

- use **calloc** instead of **malloc**

```
double* x = (double*) calloc(n,sizeof(double))
```

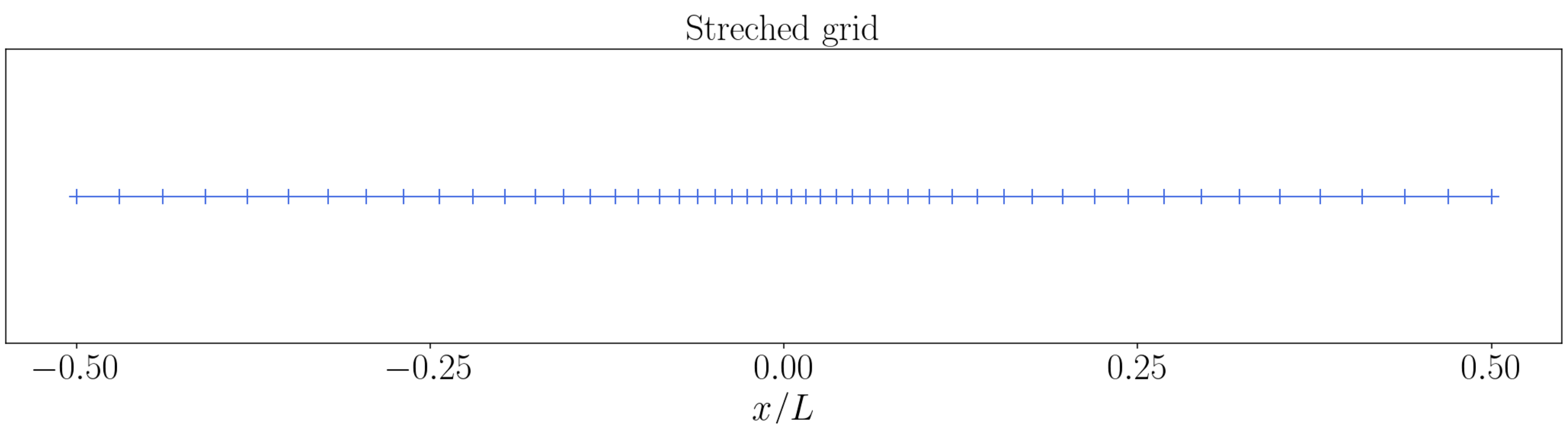
Pure convection case: example



Non uniform grid spacing

- Mapping between the « numerical space » ξ with uniform spacing to the « physical space » with non uniform resolution

$$x = g(\xi) = \xi - a\frac{L}{2\pi} sin\left(2\pi\frac{\xi}{L}\right) \qquad \frac{dx}{d\xi} = g'(\xi) = 1 - acos\left(2\pi\frac{\xi}{L}\right)$$

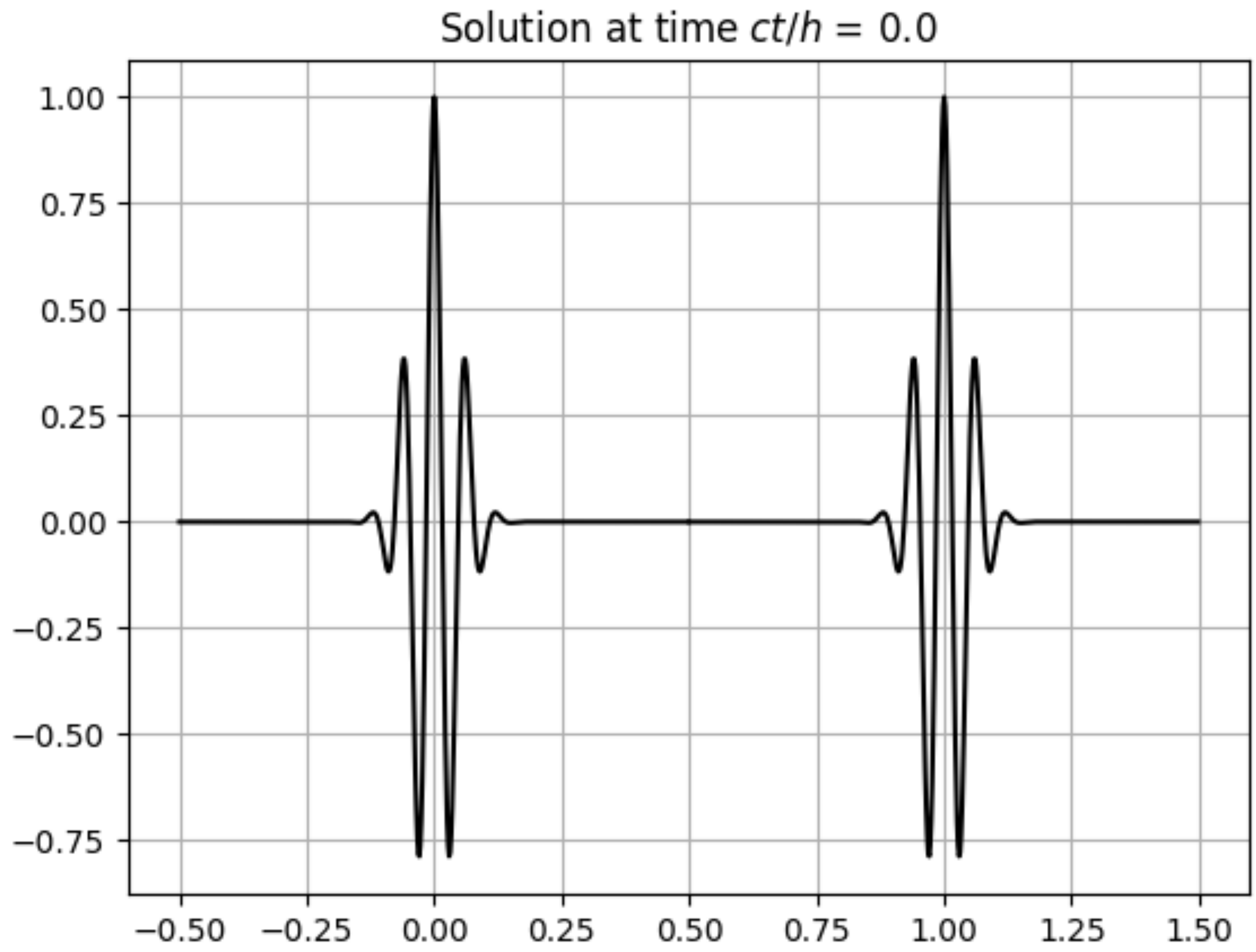


- Obtain the transport equation in a conservative form

$$\frac{\partial v}{\partial t} + \frac{\partial (bv)}{\partial \xi} = 0$$

- Perform the numerical simulation with $a = 1/2$ and $N = 128$

Advection of a wave-packet



Practical information

- Hand over: **18th March at 6 pm**
- Moodle (report + code)
- No fancy covers needed
- French or English
- Questions?

  Wednesday 4.15 to 6.15 pm (starting next week)