Numerical Methods for Astrophysics:

PARTIAL DIFFERENTIAL EQUATIONS (PDEs)

Michela Mapelli

Partial Differential Equations (PDEs). Concept

PDEs: differential equations that involve more than 1 variable

PDEs ARE UBIQUITOUS IN PHYSICS/ASTROPHYSICS

Examples:

- **wave equation**

$$-\frac{1}{c^2}\frac{\partial^2\psi}{\partial t^2}+\nabla^2\psi=0$$

- **where Laplacian operator in Cartesian coordinates is**

$$\nabla^2=\frac{\partial^2}{\partial x^2}+\frac{\partial^2}{\partial y^2}+\frac{\partial^2}{\partial z^2}$$

Other examples: diffusion equation, Laplace equation, Schroedinger equation, gravitational-wave equation, stellar interior equations

Partial Differential Equations (PDEs). Concept

1. **BOUNDARY-VALUE PDE PROBLEMs:**

   we know only boundary conditions

   If there is NO TIME EVOLUTION, simple case:

   STATIONARY SOLUTION

   can be solved with FINITE-DIFFERENCE METHODS (FDMs)

2. **INITIAL-VALUE PDE PROBLEMs:**

   we know initial conditions

   but PDE evolves with time

   can be solved with FDMs but requires one more trick

Partial Differential Equations (PDEs). Boundary-value problem with finite-difference methods (FDMs)

Practical example, Laplace's equation: $$\nabla^2\phi = 0$$

In cartesian coordinates $$\frac{\partial^2\phi}{\partial x^2} + \frac{\partial^2\phi}{\partial y^2} = 0$$

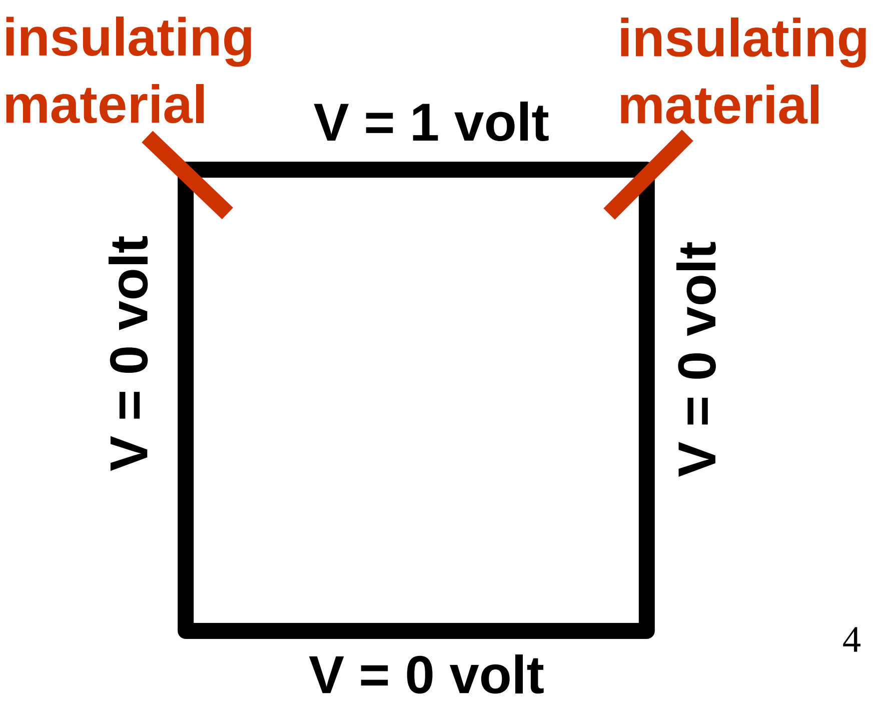
where φ is the electrostatic potential in absence of electric charges

NO TIME DEPENDENCE!

e.g. the case of a 2-dimensional
empty box
with conducting walls

top wall V = 1 volt

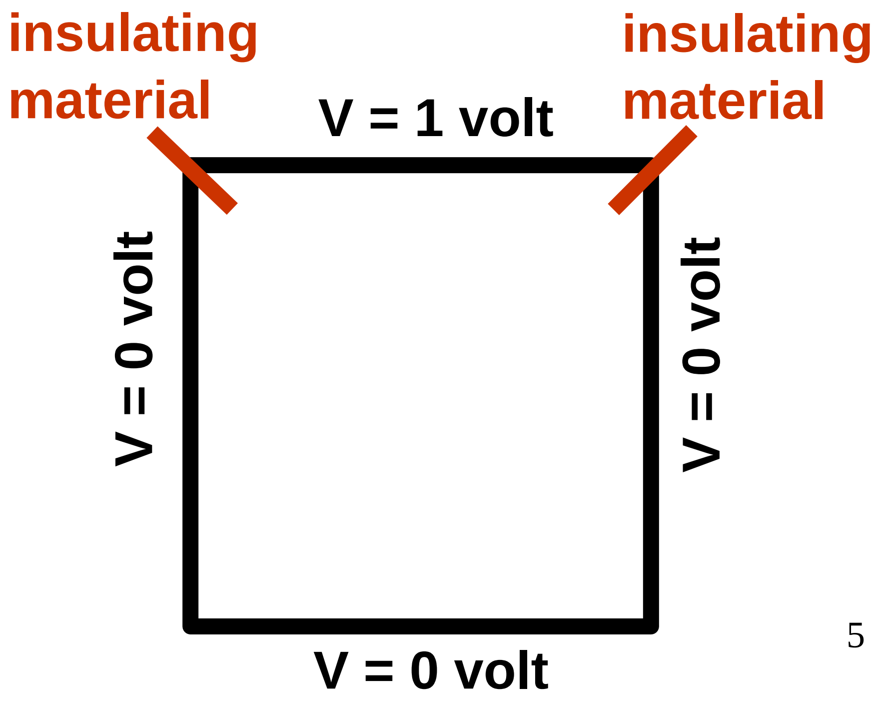
other walls (insulated from top wall)
V = 0 volt



Partial Differential Equations (PDEs). Boundary-value problem with finite-difference methods (FDMs)

- divide box into a GRID of points separated by a

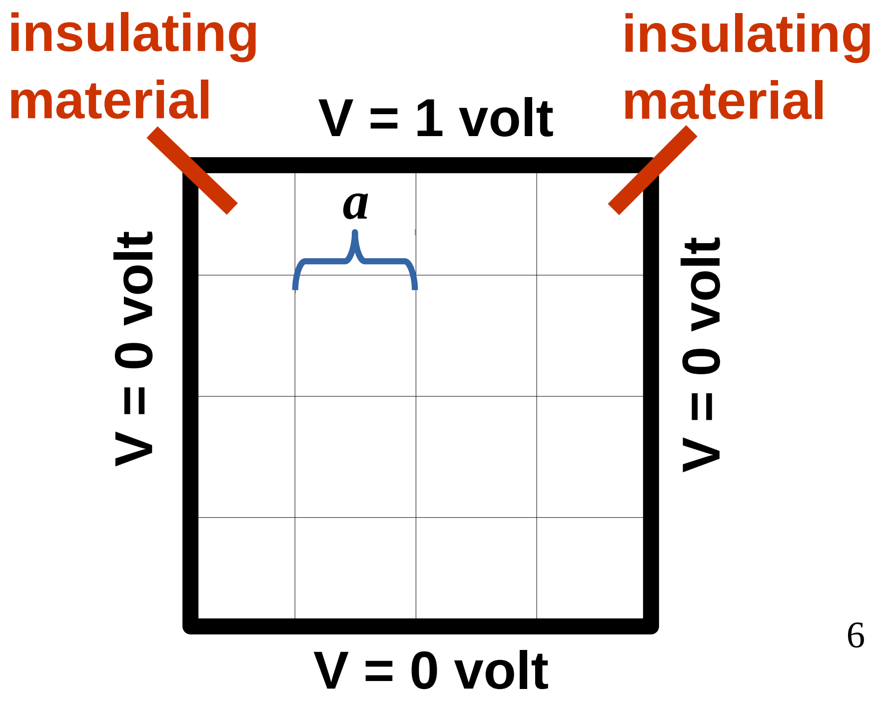
- a can be fixed or variable – let's make it fixed for simplicity



Partial Differential Equations (PDEs). Boundary-value problem with finite-difference methods (FDMs)

- divide box into a GRID of points separated by a

- a can be fixed or variable – let's make it fixed for simplicity



Partial Differential Equations (PDEs). Boundary-value problem with finite-difference methods (FDMs)

- divide box into a GRID of points separated by *a*

- *a* can be fixed or variable – let's make it fixed for simplicity

- use numerical solution of PARTIAL DERIVATIVES (chapter 7) to rewrite Laplace equation

$$\frac{\partial^2 \phi}{\partial x^2} \simeq \frac{1}{a^2}\,[\phi(x+a,y)+\phi(x-a,y)-2\,\phi(x,y)]$$

$$\frac{\partial^2 \phi}{\partial y^2} \simeq \frac{1}{a^2}\,[\phi(x,y+a)+\phi(x,y-a)-2\,\phi(x,y)]$$

Partial Differential Equations (PDEs). Boundary-value problem with finite-difference methods (FDMs)

- **divide box into a GRID of points separated by *a***

- ***a* can be fixed or variable – let's make it fixed for simplicity**

- **use numerical solution of PARTIAL DERIVATIVES (chapter 7) to rewrite Laplace equation**

$$\frac{\partial^2 \phi}{\partial x^2} \simeq \frac{1}{a^2}\,[\phi(x+a,y) + \phi(x-a,y) - 2\,\phi(x,y)]$$

$$\frac{\partial^2 \phi}{\partial y^2} \simeq \frac{1}{a^2}\,[\phi(x,y+a) + \phi(x,y-a) - 2\,\phi(x,y)]$$

- **substitute these back to the main equation**

$$\frac{1}{a^2}\,[\phi(x+a,y) + \phi(x-a,y) + \phi(x,y+a) + \phi(x,y-a) - 4\,\phi(x,y)] = 0$$

Partial Differential Equations (PDEs). Boundary-value problem with finite-difference methods (FDMs)

- remove $1/a^2$ and reshuffle bringing the term $\phi(x,y)$ to the left-hand side

$$\phi(x,y) = \frac{1}{4}\left[\phi(x+a,y) + \phi(x-a,y) + \phi(x,y+a) + \phi(x,y-a)\right]$$

→ we can derive the value of $\phi(x,y)$ at any point in the grid, provided that we solve a set of linear equations

i.e. solving PDEs with a finite difference method is equivalent to solving systems of linear equations

WHAT METHOD TO SOLVE LINEAR EQUATIONS CAN WE USE?

Partial Differential Equations (PDEs). Boundary-value problem with finite-difference methods (FDMs)

- remove 1/a^2 and reshuffle bringing the term $\phi(x,y)$ to the left-hand side

$$\phi(x,y) = \frac{1}{4}\left[\phi(x+a,y) + \phi(x-a,y) + \phi(x,y+a) + \phi(x,y-a)\right]$$

→ we can derive the value of $\phi(x,y)$ at any point in the grid, provided that we solve a set of linear equations

i.e. solving PDEs with a finite difference method is equivalent to solving systems of linear equations

WHAT METHOD TO SOLVE LINEAR EQUATIONS CAN WE USE?

We simplified derivatives with NUMERICAL derivatives: our model will always be a huge simplification

→ just go for a very simple, ITERATIVE APPROACH, like the RELAXATION technique we have seen for non-linear eqs. (but can be used also for linear eqs, of course)
and can be easily generalized to a system of equations

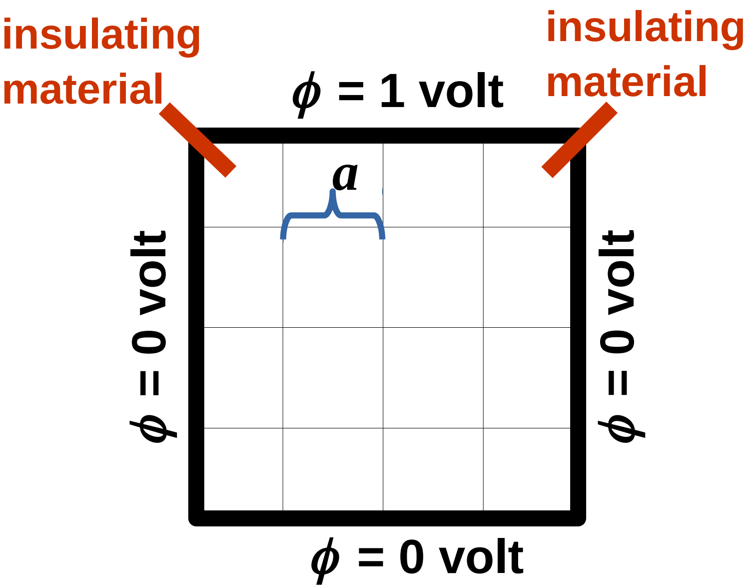
Partial Differential Equations (PDEs). Boundary-value problem with finite-difference methods (FDMs)

- **Steps to solve the equations:**
  1. **assign the boundary values** $\phi[0,:] = 1$

$$\phi[M,:] = \phi[0:M,0] = \phi[0:M,M] = 0$$

  2. **assign to all the other points of the grid a guess value, e.g. zero**

$$\phi[i,j] = 0 \quad \text{if } i \neq (0,M) \text{ and } j \neq (0,M)$$



  3. **solve the equation below by iteration (like relaxation)**

$$\phi(x,y) = \frac{1}{4}\left[\phi(x+a,y) + \phi(x-a,y) + \phi(x,y+a) + \phi(x,y-a)\right]$$

or apply the over-relaxation

$$\phi(x,y) = \frac{(1+\omega)}{4}\left[\phi(x+a,y) + \phi(x-a,y) + \phi(x,y+a) + \phi(x,y-a)\right] - \omega\,\phi(x,y)$$

try with $\omega = 0.9$ and then reduce ω if the result does not converge

$\omega > 0.9$ is often non-stable

Partial Differential Equations (PDEs). Boundary-value problem with finite-difference methods (FDMs)

Where the relaxation formula comes from?

Define $\Delta\phi(x,y) \equiv \phi_{\rm new}(x,y) - \phi(x,y)$

& define $\phi_\omega(x,y) = \phi(x,y) + (1+\omega)\,\Delta\phi(x,y)$

Combining the two of them:

$$\phi_\omega(x,y) = \phi(x,y) + (1+\omega)\,[\phi_{\rm new}(x,y) - \phi(x,y)]$$

Expressing in terms of ϕnew : $\phi_{\rm new}(x,y) = \frac{1}{(1+\omega)}\,\phi_\omega(x,y) + \frac{\omega}{(1+\omega)}\,\phi(x,y)$

Substituting to the left-hand term of

$$\phi_{\rm new}(x,y) = \frac{1}{4}\,[\phi(x+a,y) + \phi(x-a,y) + \phi(x,y+a) + \phi(x,y-a)]$$

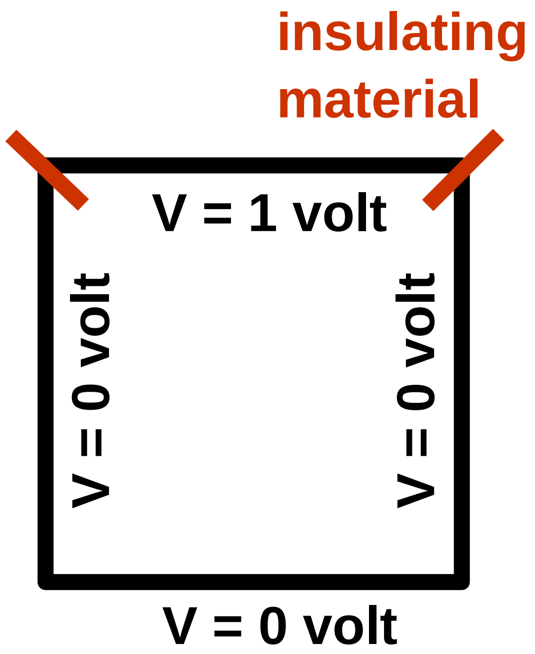
and rearranging, we get:

$$\phi_\omega(x,y) = \frac{(1+\omega)}{4}\,[\phi(x+a,y) + \phi(x-a,y) + \phi(x,y+a) + \phi(x,y-a)] - \omega\,\phi(x,y)$$

Partial Differential Equations (PDEs). Exercise

EXERCISE:

Consider an empty 2D box with conducting walls. The boundary conditions are that one wall is kept at voltage V = 1 volt (for example, the first row of the matrix is kept at voltage V = 1 volt), while the other three walls are insulated from it and are at 0 volts. We want to calculate the potential $\phi(x, y)$ at each point in the 2D box. Plot the result by using the function matplotlib.pyplot.imshow().



Partial Differential Equations (PDEs). Exercise

Suggestions: Require a tolerance delta $= 10^{-3}$ *(smaller tolerances require a significant computing time). For the definition of tolerance you can use*

```
delta=numpy.linalg.norm(phi-phiold)
```

where phi and phiold are the new and the old iteration of the matrix (this treats the two matrices as two vectors and calculates the norm of their difference).
Create a grid of $(M + 1) \times (M + 1)$ *cells, with* $M = 100$. *Rows* 0 *and* M *and columns* 0 *and* M *are the walls of the box (i.e. the boundaries) and must be assigned the given voltage. The result should look like Figure 49 and should require* ≈ 3000 *iterations to reach the required tolerance (for the above definition of tolerance).*

Partial Differential Equations (PDEs). Exercise

To produce the plot use:

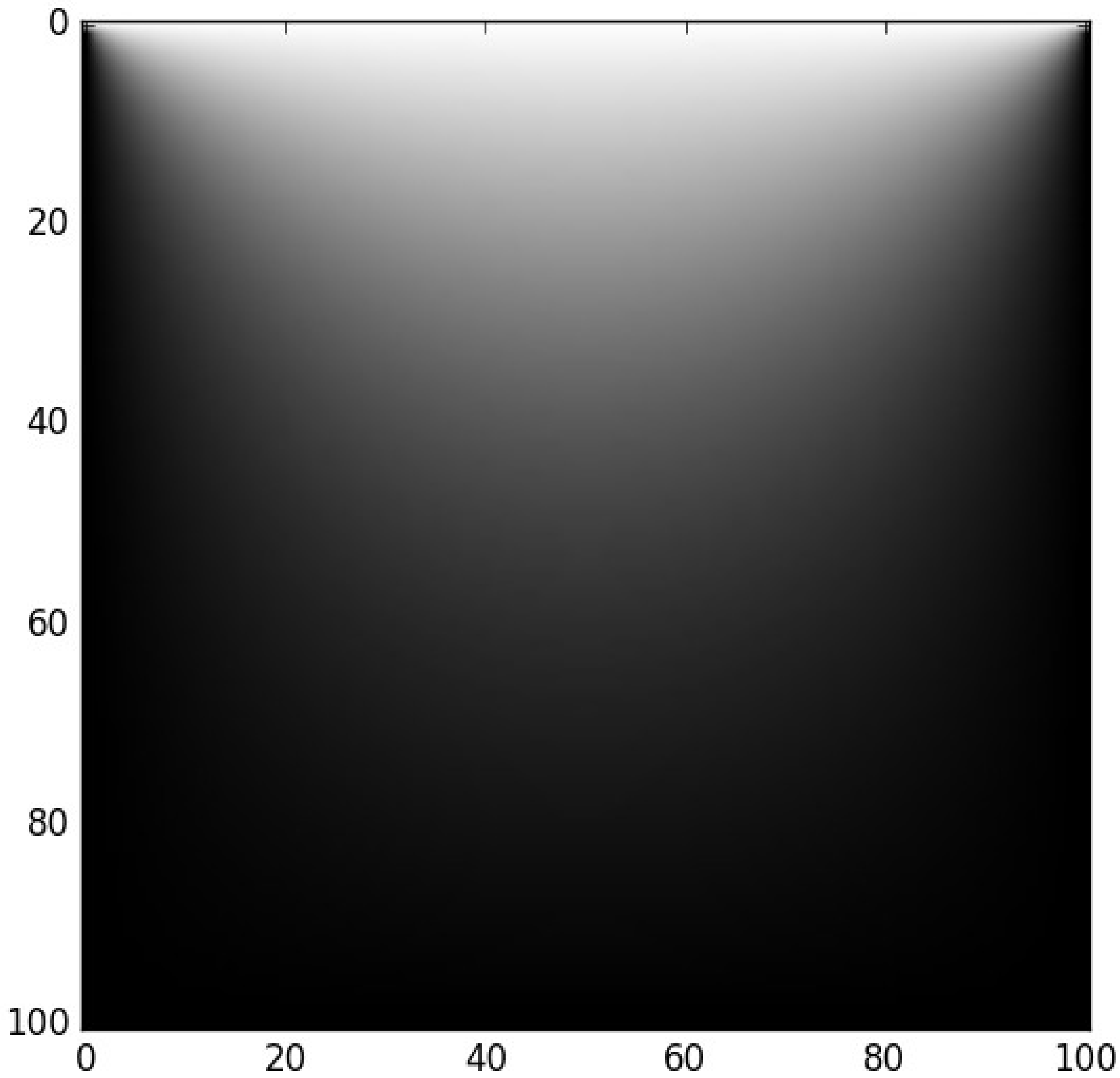
```
import matplotlib.pyplot as plt
plt.imshow(phi) #where phi is the final matrix
plt.gray()
plt.show()
```

Now, solve the same exercise with relaxation, as proposed in equation 233.

$$\phi(x, y) = \frac{(1+\omega)}{4}\left[\phi(x+a, y) + \phi(x-a, y) + \phi(x, y+a) + \phi(x, y-a)\right] - \omega\,\phi(x, y)$$

Suggestions: Choose $\omega = 0.9$ (higher ω might be unstable). With $\omega = 0.9$ you should reach a tolerance of delta $= 10^{-3}$ in $\approx 200 - 300$ iterations. The script is much faster with the **relaxation**.

Partial Differential Equations (PDEs). Exercise



Partial Differential Equations (PDEs). Initial-value PDEs with finite difference methods

INITIAL-VALUE PDEs: PDEs for which we know the initial conditions and we must calculate the time evolution of one or more variables.

Example: One-dimensional DIFFUSION EQUATION

$$\frac{\partial \phi}{\partial t} = D \frac{\partial^2 \phi}{\partial x^2}$$

where ϕ is $\phi(x,t)$ = physical quantity that is diffusing in time and space (e.g. temperature, density of particles, etc)

***D* = diffusion coefficient**

Suppose we know the initial conditions

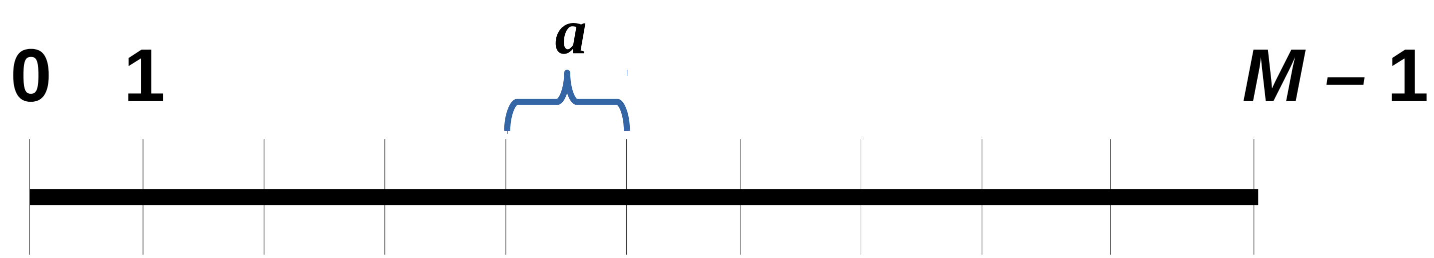
We will try to solve this with the forward-time centered-space (FTCS) method for initial-value PDEs

Partial Differential Equations (PDEs). FTCS method

- First consider the spatial dependence of ϕ(x,t) and write numerical derivative:

$$\frac{\partial^2 \phi}{\partial x^2} = \frac{\phi(x+a,t) + \phi(x-a,t) - 2\,\phi(x,t)}{a^2}$$

Where we chose a spatial step *a*
and we have gridded our spatial domain into a grid of *M* points



Partial Differential Equations (PDEs). FTCS method

- First consider the spatial dependence of ϕ(x,t) and write numerical derivative:

$$\frac{\partial^2 \phi}{\partial x^2} = \frac{\phi(x+a,t) + \phi(x-a,t) - 2\,\phi(x,t)}{a^2}$$

Where we chose a spatial step *a*
and we have gridded our spatial domain into a grid of *M* points

- Substitute the above numerical derivative into the diffusion equation:

$$\frac{\partial \phi}{\partial t} = \frac{D}{a^2}\,\left[\phi(x+a,t) + \phi(x-a,t) - 2\,\phi(x,t)\right]$$

We have a set of *M* ordinary differential equations, one per each point of the spatial grid

Partial Differential Equations (PDEs). FTCS method

We must solve a set of ordinary differential equations with one of the methods we learned

WHICH ONE?

Partial Differential Equations (PDEs). FTCS method

We must solve a set of ordinary differential equations with one of the methods we learned

WHICH ONE?

A smart option is simply the EULER'S METHOD

because

1. we have a very large system of equations to solve
2. we have already done a huge approximation with the numerical derivatives in space

Euler's method equation: $$\phi(x, t+h) \simeq \phi(x,t) + h\,\frac{\mathrm{d}\phi(x,t)}{\mathrm{d}t}$$

$$\longrightarrow \quad \phi(x, t+h) \simeq \phi(x,t) + h\,\frac{\partial\phi(x,t)}{\partial t}$$

Partial Differential Equations (PDEs). FTCS method

Diffusion equation written with finite differences

$$\frac{\partial \phi}{\partial t} = \frac{D}{a^2}\,\left[\phi(x+a,t) + \phi(x-a,t) - 2\,\phi(x,t)\right]$$

Euler's equation: $$\phi(x,t+h) \simeq \phi(x,t) + h\,\frac{\partial \phi(x,t)}{\partial t}$$

Rewrite Euler's equation in terms of $\partial\phi(x,t)/\partial t$:

$$\frac{\partial \phi(x,t)}{\partial t} \simeq \frac{\phi(x,t+h) - \phi(x,t)}{h}$$

Substitute the eq. into diffusion equation

$$\frac{\phi(x,t+h) - \phi(x,t)}{h} \simeq \frac{D}{a^2}\,\left[\phi(x+a,t) + \phi(x-a,t) - 2\,\phi(x,t)\right]$$

Multiply by *h* and nd rewrite in terms of $\phi(x,t+h)$

$$\phi(x,t+h) \simeq \phi(x,t) + \frac{h\,D}{a^2}\,\left[\phi(x+a,t) + \phi(x-a,t) - 2\,\phi(x,t)\right]$$

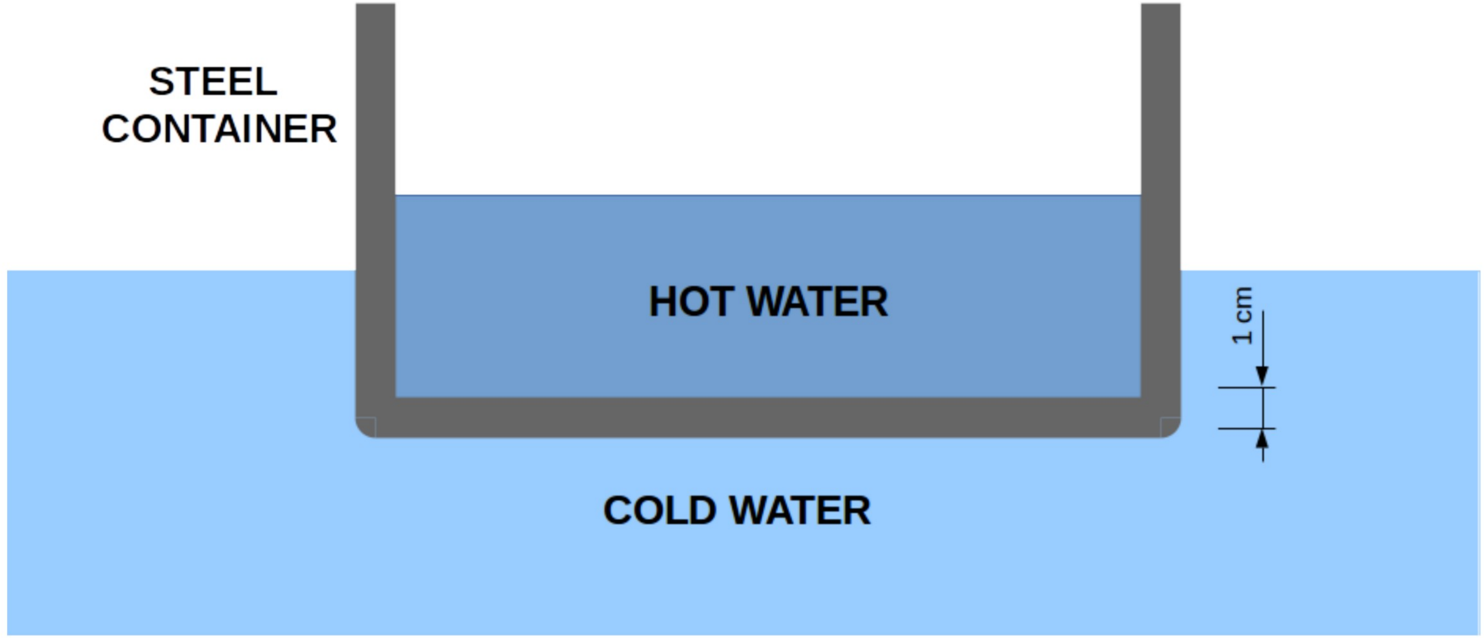
basic equation of the forward-time centered-space (FTCS) method for initial-value PDEs, applied to diffusion equation.

Partial Differential Equations (PDEs). Exercise

EXERCISE:

Write a python script to solve the diffusion equation with the FTCS method for the following system.

The flat base of a container made of 1 cm thick stainless steel is initially at a uniform temperature of $T_0 = 293$ K. The container is placed in a bath of cold water at $T_{bath} = 273$ K and is filled with hot water at $T_{hot} = 323$ K. Calculate the temperature profile of the steel as a function of distance x from the hot side to the cold side (from 0 to 1 cm) and as a function of time. The system is shown in Figure 50.



Partial Differential Equations (PDEs). Exercise

EXERCISE:

Write a python script to solve the diffusion equation with the FTCS method for the following system.

The flat base of a container made of 1 cm thick stainless steel is initially at a uniform temperature of $T_0 = 293$ K. The container is placed in a bath of cold water at $T_{bath} = 273$ K and is filled with hot water at $T_{hot} = 323$ K. Calculate the temperature profile of the steel as a function of distance x from the hot side to the cold side (from 0 to 1 cm) and as a function of time. The system is shown in Figure 50.
Thermal conduction is described by the diffusion equation as

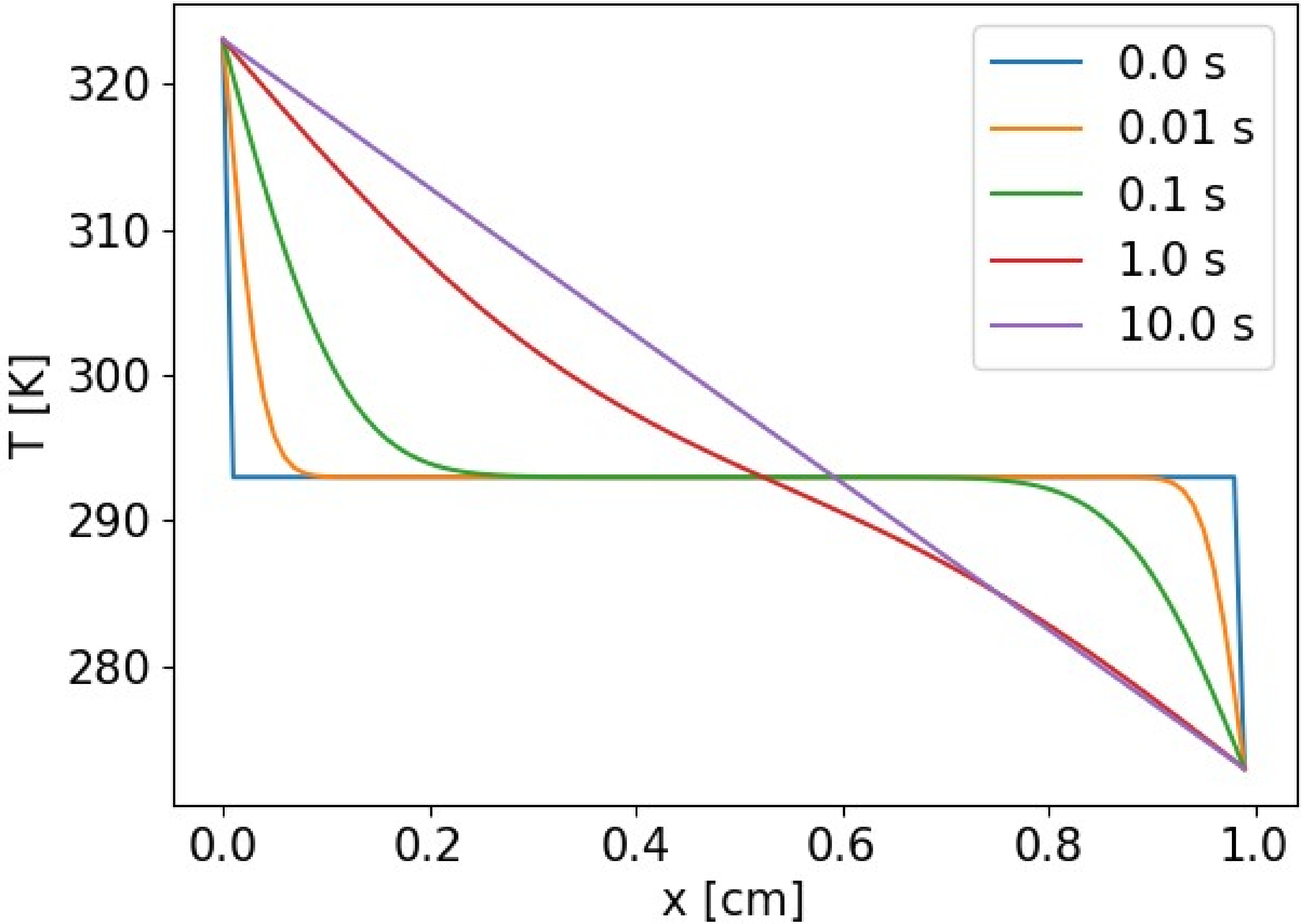
$$\frac{\partial T}{\partial t} = D \frac{\partial^2 T}{\partial x^2}, \tag{239}$$

where $D = 4.25 \times 10^{-2}\ \mathrm{cm}^2\ \mathrm{s}^{-1}$ for stainless steel. Use equation 238 to solve the problem.
Plot the temperature profile of the steel as a function of x at four different times $t = 0.01$, 0.1, 1 and 10 s.

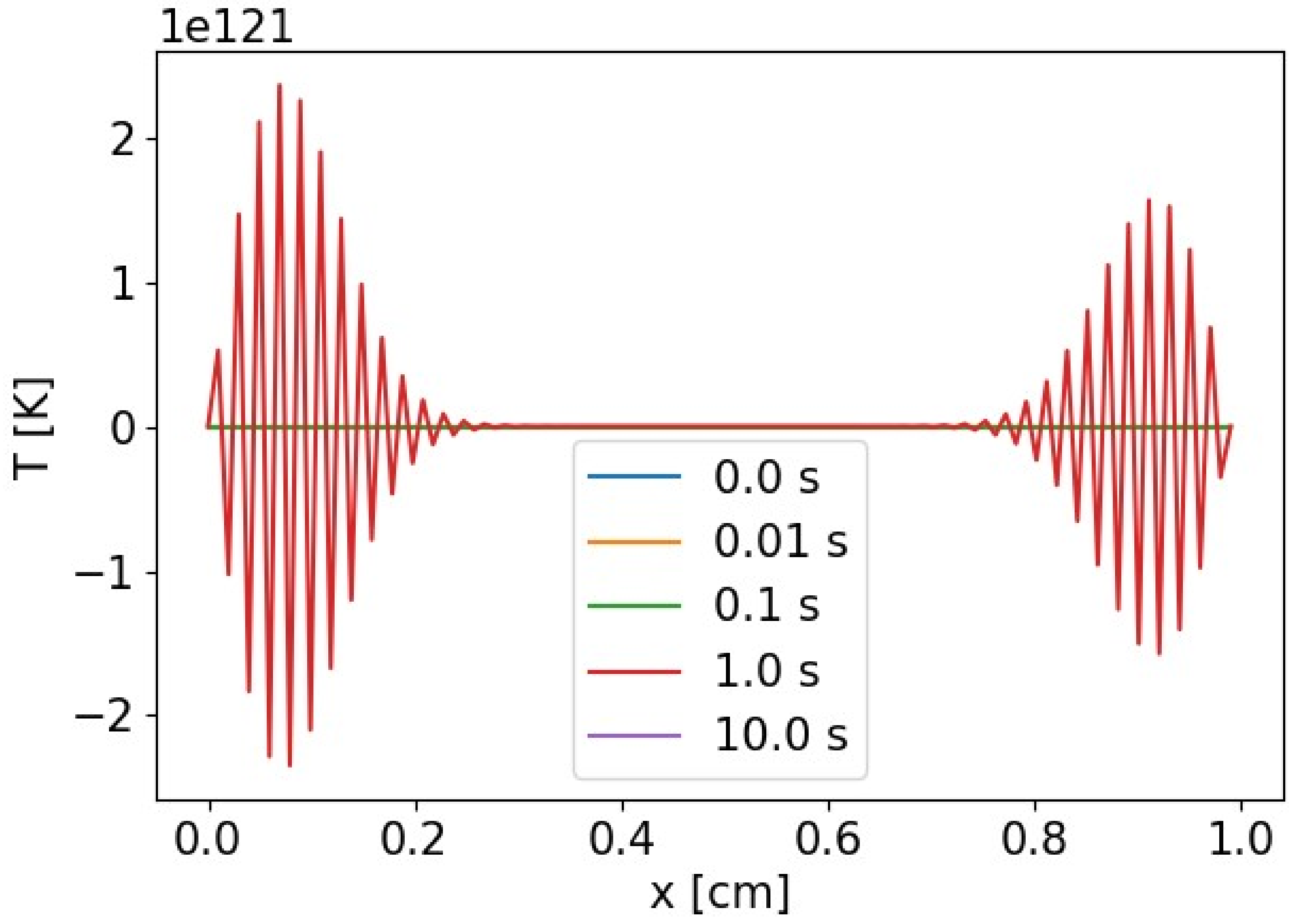
The result should look like Figure 51.

Suggestion: use a spatial grid with M+1 point, where M=100. Choose h=0.001. What happens if you change h and why?

Partial Differential Equations (PDEs). Exercise



Partial Differential Equations (PDEs). Exercise



Partial Differential Equations (PDEs). Exercise

ADDITIONAL SUGGESTIONS for the EXERCISE:

- here you have both spatial step (a) and temporal step (h)
- select spatial step a as a = L/N
  where L=length of steel, N=# of grid points (suggested N=100)
  In this way you control the resolution by simply changing N
- define a constant const=D*h/(a*a) and calculate it just once
  before the loops: you will reduce computing time and numerical errors
- you need two loops: an INNER one for the spatial calculation
  an OUTER one for the temporal calculation
- solve with a loop over the spatial coordinates at each time-step

$$\phi(x, t+h) \simeq \; = \phi(x,t) + \frac{h\,D}{a^2}\,[\phi(x+a,t) + \phi(x-a,t) - 2\,\phi(x,t)]$$

- SAVE RAM AND DO NOT unnecessarily COMPLICATE THE PROBLEM:
  phi can be a matrix (over time and spatial position)
  but this requires more RAM + you are not able to generalize in 2D

→ define phi as an array with one index for the spatial motion

You can print the time steps of interest without using a matrix!