Hands-on session on computer methods for mass transport in tissue engineering

Fall 2021

Mojtaba Barzegari <mojtaba.barzegari@kuleuven.be>
Hans Van Oosterwyck <hans.vanoosterwyck@kuleuven.be>

KU LEUVEN

Outline

- Transport equation
- Diffusion, convection, and reaction
- Manually solving diffusion problems
- Computer methods: Introducing Python and Jupyter
- Solving diffusion problems using symbolic computing
- Solving transient problems using numerical computing

Transport Equation

- How a scalar quantity is transported in space
- Commonly used in transport phenomena studies
- From a mathematical point of view:

  a (partial) differential equation to describe the conservation

Transport Equation

- Reaction-Diffusion-Advection equation
- $c = c(x, y, z, t)$, variable of interest
  (e.g. species concentration in mass transfer or temperate in heat transfer)

$$\frac{\partial c}{\partial t} = \underbrace{\nabla . (D \nabla c) +}_{\text{Diffusion}} \underbrace{\nabla . (\mathbf{v} c) +}_{\text{Advection}} \underbrace{f(c)}_{\text{Reaction}}$$

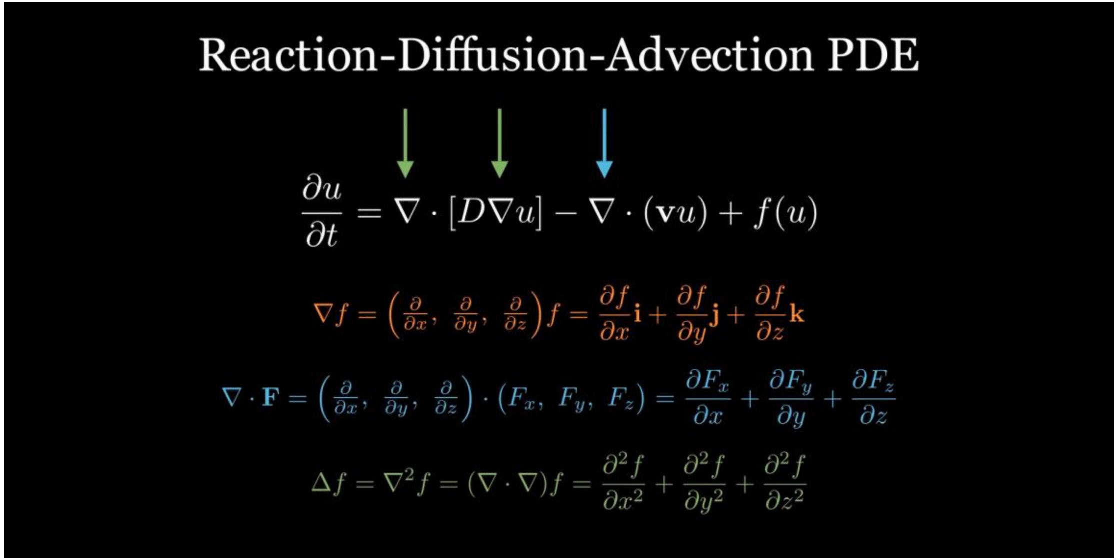
$$\downarrow$$

$$\left(A + 2B \xrightarrow{k} C \;\Rightarrow \frac{\partial [C]}{\partial t} = k[A][B]^2 \right)$$

KU LEUVEN

Reaction-Diffusion-Advection PDE

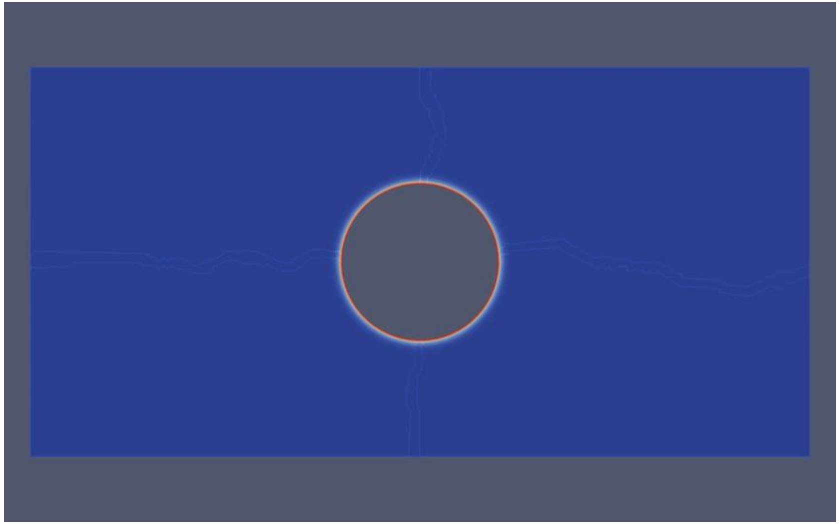
- A short introductory video on the equation: https://www.youtube.com/watch?v=YiIT3p507S0



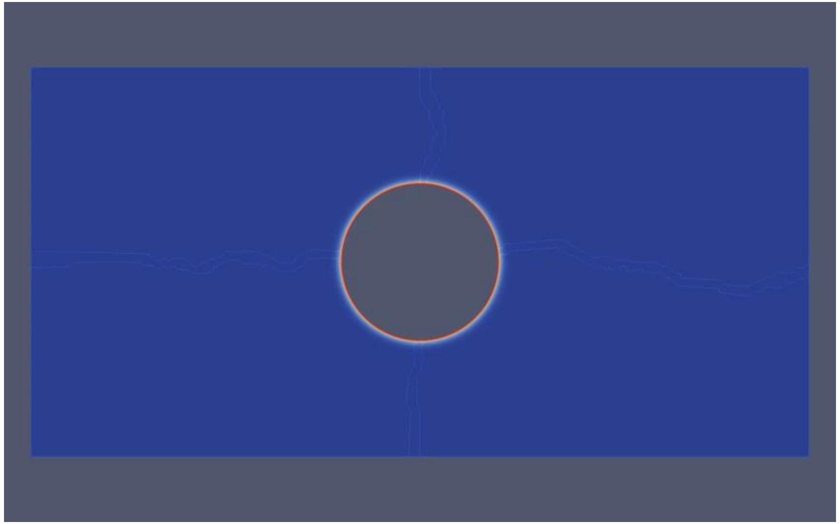
Convection Effect

$$\frac{\partial c}{\partial t} = \nabla.(D\nabla c)$$

$$\frac{\partial c}{\partial t} = \nabla.(D\nabla c) + \nabla.(\mathbf{v}c)$$



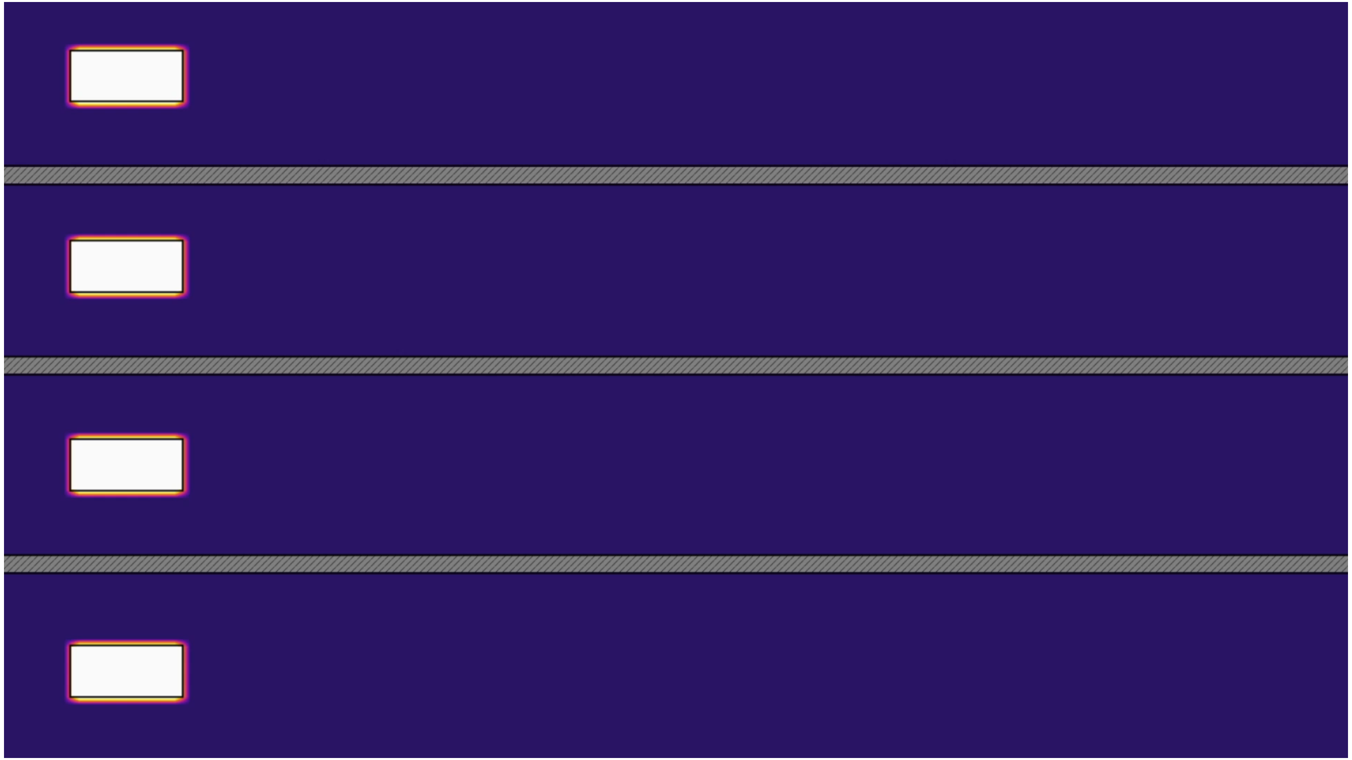
Diffusion



Diffusion-Advection

Convection Effect - Velocity

$$\frac{\partial c}{\partial t} = \nabla.(D\nabla c) + \nabla.(\mathbf{v}c)$$



Increasing

$\mathbf{v}$

Solving Problems Manually

- Integrating the conservative form of equations
- Applying appropriate boundary conditions to obtain integration coefficients
- In case of transient (not steady state) problems: apply initial conditions
- Calculate quantities of desire (e.g. maximum diffusion depth)

Steady State Examples

- Some examples were already on the mass transport slides
- Let's solve the equations for case 3 (slide 18):
  1d diffusion in cellular construct

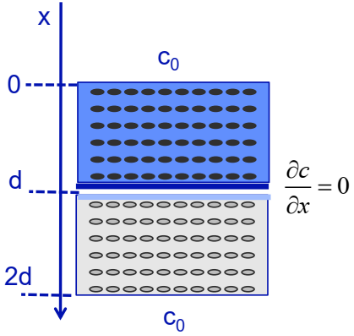
Simple case 3: 1D-diffusion in cellular construct (Cartesian coordinates)

- Passive transport in cellular construct:

$$\frac{\partial c}{\partial t} = D\frac{\partial^2 c}{\partial x^2} - V_{max,cell}\underbrace{\rho_{cell}}_{\text{cell density (\#cells/m}^3\text{)}}$$

  - Steady state solution:
    - At x=0: $c(x=0) = c_0$
    - At x=d: $\Gamma=0$ (impermeable or symmetric)

$$c(x) = \frac{V_{max,cell}\rho_{cell}}{D}\left(\frac{x^2}{2} - dx\right) + c_0$$



  - Maximum diffusion depth (for c(x=d) = 0):

$$d_{max} = \sqrt{2c_0\frac{D}{V_{max,cell}\rho_{cell}}}$$

Mass transport in tissue engineering 18

1D Diffusion in Cellular Construct

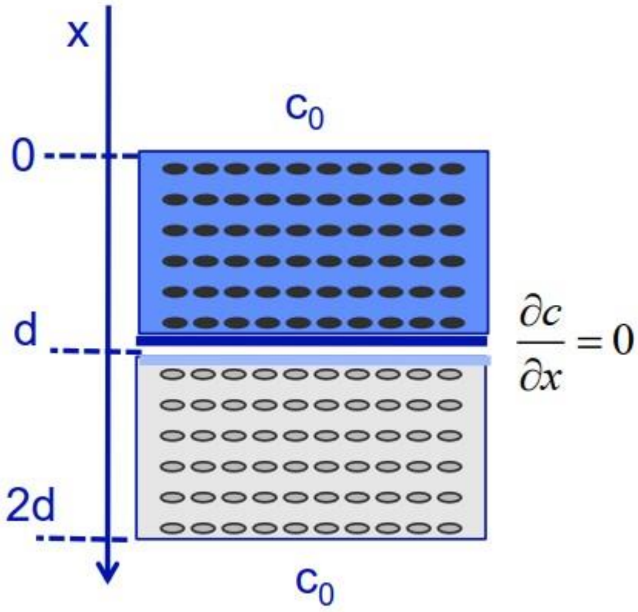
- The problem is to solve the following equation

$$\frac{\partial c}{\partial t} = D\frac{\partial^2 c}{\partial x^2} - V_{max,cell}\rho_{cell}$$

- For a steady state solution, we can neglect the left hand side term

$$D\frac{\partial^2 c}{\partial x^2} - V_{max,cell}\rho_{cell} = 0$$

- Maximum depth of oxygen penetration?



KU LEUVEN

1D Diffusion in Cellular Construct

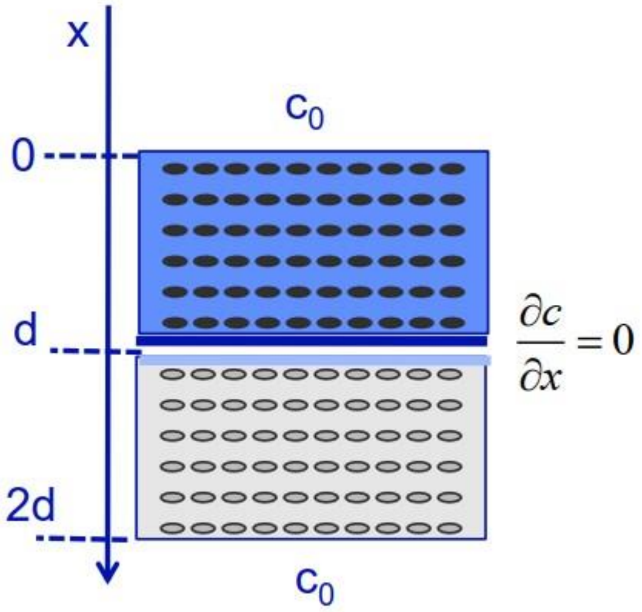
- It is a 1-D problem, so let's say c = c(x)

$$D\frac{d^2c}{dx^2} = V_m\rho \rightarrow \frac{d^2c}{dx^2} = \frac{V_m\rho}{D}$$

- Integrating both sides of the equation yields

$$\int \frac{d^2c}{dx^2}dx = \int \frac{V_m\rho}{D}dx$$

$$\rightarrow \frac{dc}{dx} = \frac{V_m\rho}{D}x + C_1$$

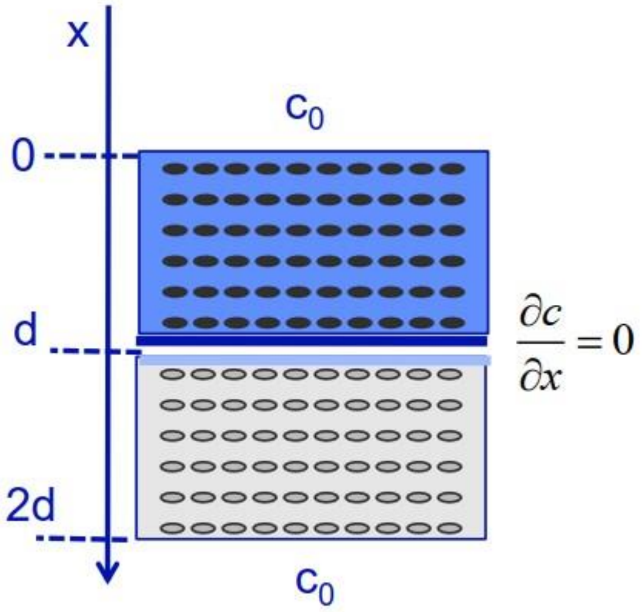


1D Diffusion in Cellular Construct

$$\frac{dc}{dx} = \frac{V_m \rho}{D} x + C_1$$

- We know $\frac{dc}{dx} = 0$ at $\mathrm{x} = \mathrm{d}$

$$0 = \frac{V_m \rho}{D} d + C_1 \rightarrow C_1 = -\frac{V_m \rho d}{D}$$



1D Diffusion in Cellular Construct

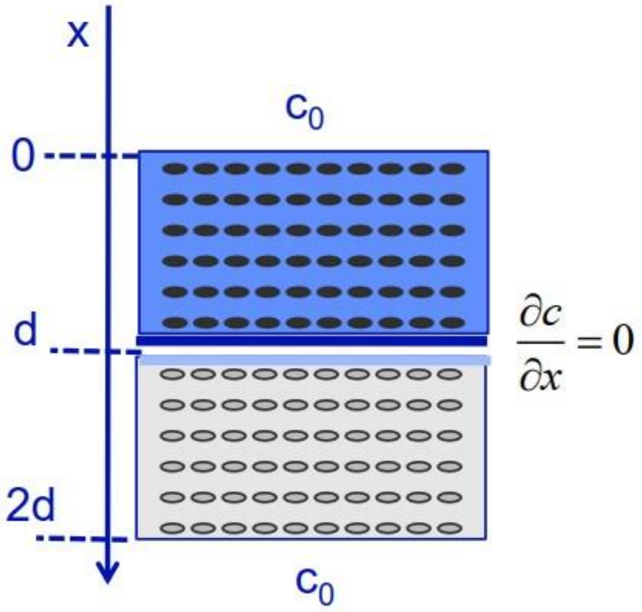
- Still have a derivative term, so we eliminate it

$$\int \frac{dc}{dx} dx = \int \left(\frac{V_m \rho}{D} x + C_1\right) dx$$

$$\rightarrow c(x) = \frac{V_m \rho}{D} \frac{x^2}{2} + C_1 x + C_2$$

- We know $c = \mathrm{c}_0$ at $\mathrm{x} = 0$

$$C_2 = c_0$$



KU LEUVEN

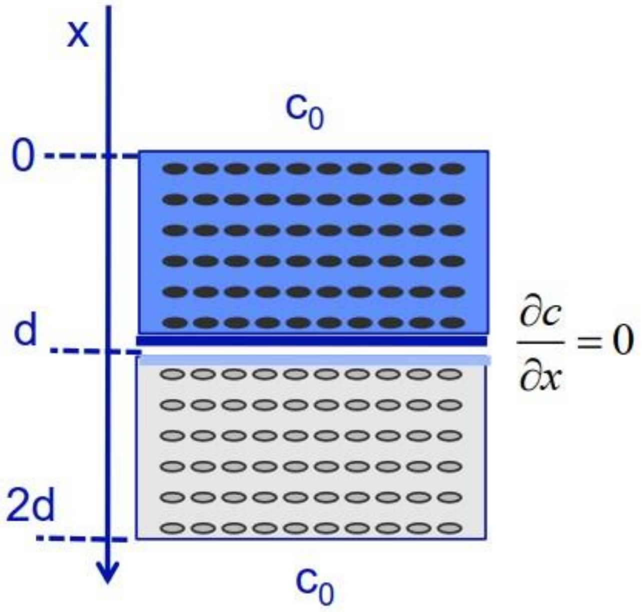
1D Diffusion in Cellular Construct

- So, by knowing C_1 and C_2, we can write the equation of the change of concentration of oxygen throughout the construct:

$$c(x) = \frac{V_m\rho}{D}\frac{x^2}{2} - \frac{V_m\rho d}{D}x + c_0$$

$$\rightarrow c(x) = \frac{V_m\rho}{D}\left(\frac{x^2}{2} - xd\right) + c_0$$

- How to find the maximum diffusion depth?



KU LEUVEN

1D Diffusion in Cellular Construct

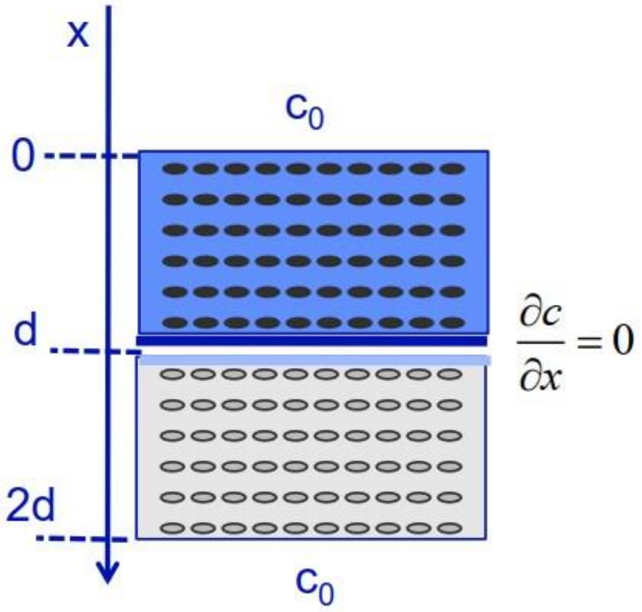
- We know $c(x) = 0$ at $\mathrm{x} = \mathrm{d_{max}}$, so

$$c(x) = \frac{V_m \rho}{D}\frac{x^2}{2} - \frac{V_m \rho d}{D} x + c_0$$

$$x = d_{max}, d = d_{max}, c(x) = 0$$

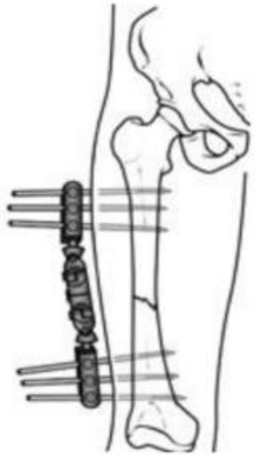
$$\rightarrow 0 = \frac{V_m \rho}{D}\left(\frac{{d_{max}}^2}{2} - {d_{max}}^2\right) + c_0$$

$$\rightarrow d_{max} = \sqrt{2 c_0 \frac{D}{V_m \rho}}$$



A Similar Problem, Solve It as Exercise

- Bone healing in large bone defects
- Derive the expression for the penetration depth $\mathrm{c}(\mathrm{x} = \mathrm{d}) = c_{\mathrm{min}}$
- Bone segmental defect (a symmetry BC?)
- Use the following values:



$$D = 3 \times 10^{-9} \ldots 10^{-10}\ \mathrm{m}^2/\mathrm{s}$$
$$V_{\mathrm{max}} \approx 10^{-18} \ldots 10^{-17}\ \mathrm{mole/cell}.s$$
$$\rho_{\mathrm{cell}} = 10^6 \ldots 10^8\ \mathrm{cells/ml}$$
$$c_0 = 21 \ \ldots 10\%$$
$$c_{\mathrm{min}} = 5 \ \ldots 1\%$$

A Recap on Solubility

- A big difference between the bone healing problem and the cellular construct is the solubility of oxygen.
- The solubility should be taken into account (slide 59)

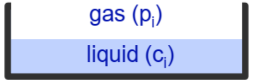
Oxygen solubility

- Partial pressure p_i for molar fraction x_i in gas mixture:

$$\boxed{p_i = x_i p}$$

- Henry's law: steady state concentration c_i of dissolved gas (for ideally dilute solution):

$$\boxed{c_i = \sigma_i p_i}$$



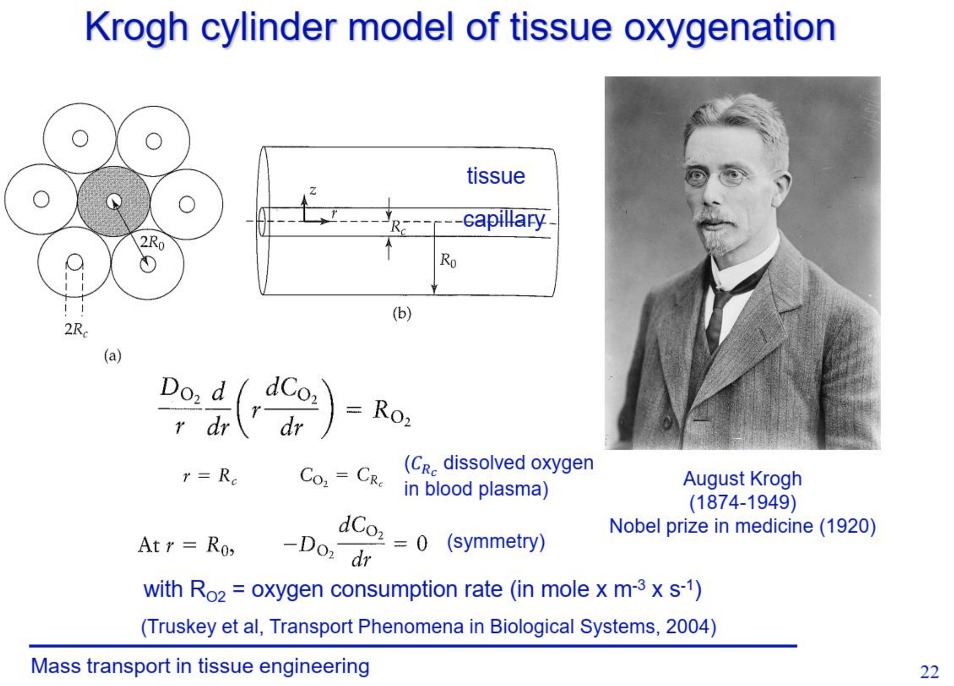
- Solubility σ_i of respiratory gases in blood plasma:

gas	σ (Molar/mmHg)
O_2	1.4×10^{-6}
CO_2	3.3×10^{-5}
CO	1.2×10^{-6}
N_2	7×10^{-7}
He	4.8×10^{-7}

(Molar = M = mole/l)

Another Example

- Krogh cylinder model of tissue oxygenation (slide 22)
- Radial diffusion in cylindrical coordinate



Krogh Cylinder Model

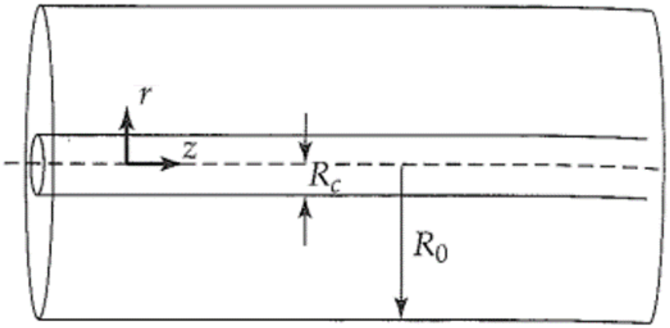
- The problem is solving the following equation

$$\frac{D_{O_2}}{r}\frac{d}{dr}\left(r\frac{dC_{O_2}}{dr}\right) = R_{O_2}$$

- Simplifying the notations:

$$c = C_{O_2}, D = D_{O_2}, R = R_{O_2}$$

$$\frac{D}{r}\frac{d}{dr}\left(r\frac{dc}{dr}\right) = R$$



KU LEUVEN

Krogh Cylinder Model

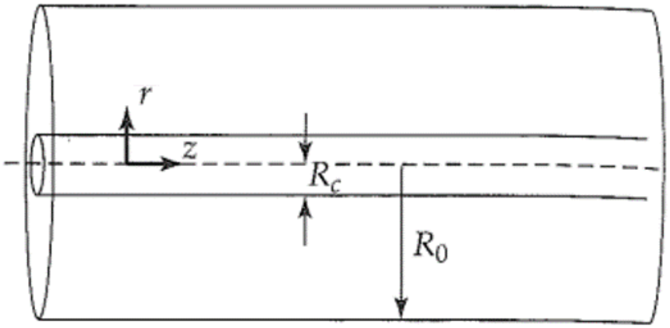
- The procedure is exactly the same as before

$$\int \frac{d}{dr}(r\frac{dc}{dr})dr = \int \frac{Rr}{D}dr$$

$$\rightarrow r\frac{dc}{dr} = \frac{Rr^2}{2D} + C_1$$

$$r = R_0, \frac{dc}{dr} = 0$$

$$0 = \frac{RR_0^2}{2D} + C_1 \rightarrow C_1 = \frac{-RR_0^2}{2D}$$



KU LEUVEN

Krogh Cylinder Model

- Let's continue to integrate

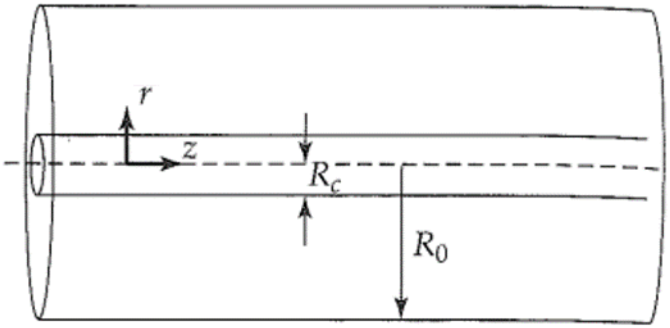
$$\int \frac{dc}{dr} dr = \int \frac{Rr}{2D} dr + \int \frac{C_1}{r} dr$$

$$\rightarrow c(r) = \frac{Rr^2}{4D} + C_1 \ln(r) + C_2$$

$$r = R_c, c = C_{R_c}$$

$$C_{R_c} = \frac{RR_c^2}{4D} - \frac{RR_0^2}{2D} \ln R_c + C_2$$

$$\rightarrow C_2 = \frac{R}{4D}\left(2R_0^2 \ln R_c - R_c^2\right) + C_{R_c}$$

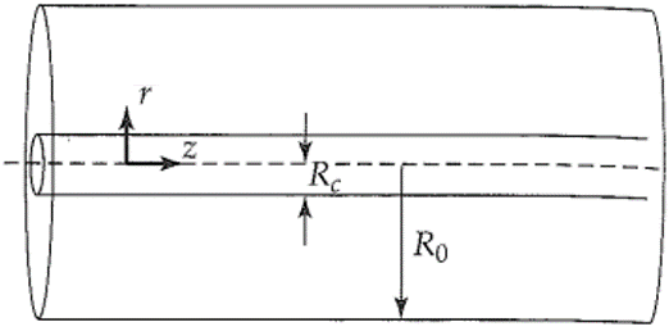


KU LEUVEN

Krogh Cylinder Model

- And now, assembling everything together yields

$$C = \frac{Rr^2}{4D} - \frac{RR_0^2}{2D}\ln(r) + \frac{R}{4D}(2R_0^2 \ln R_c - R_c^2) + C_{R_c}$$



KU LEUVEN

Handling the Complexity with Computers

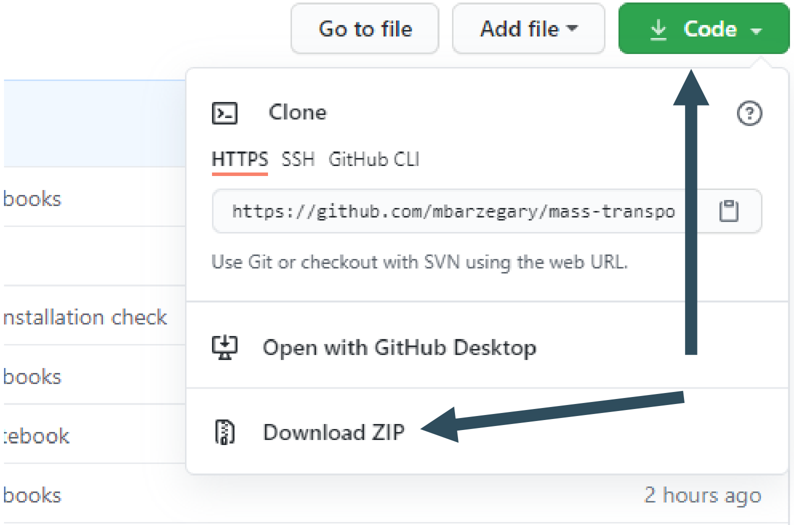
- Solving problems with symbolic computing
- MATLAB, Maple, and Python are the common options
- Python is becoming dominant computing language
- Jupyter is a great tool to facilitate working with Python

Introduction to Python and Jupyter

- Learning Python is super easy
- Go through the provided materials
- Getting started
  - Installing Jupyter locally
  - Running the examples on free online services

Running the Examples Locally

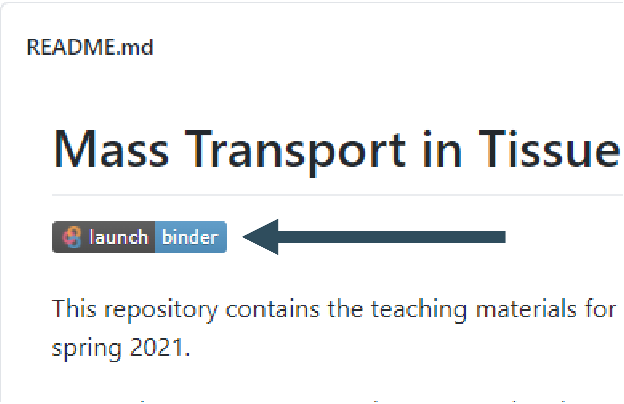
- Go to https://github.com/mbarzegary/mass-transport-tissue-engineering-fall2021
- Download the code by clicking “Code” and then “Download ZIP”
- Open the downloaded code and follow the installation guide (PDF file)
- View the notebooks 0 to 5



KU LEUVEN

Running the Examples Online

- Go to https://github.com/mbarzegary/mass-transport-tissue-engineering-fall2021
- Click on the “launch binder” badge
- Wait until the server starts, and then, click on the notebooks 0 to 5 to view them

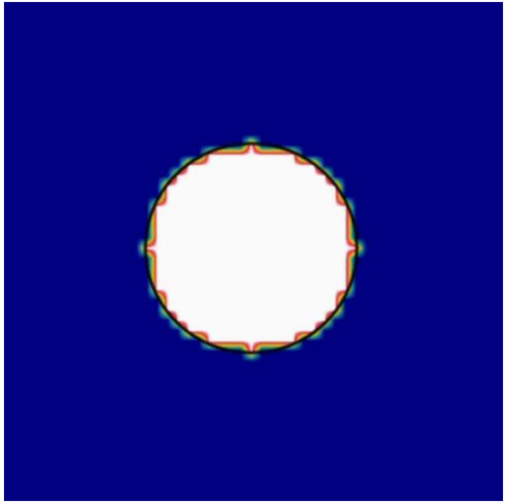


KU LEUVEN

Solving Transient Problems

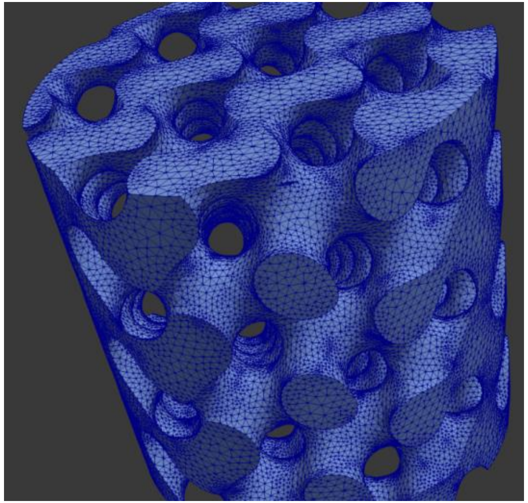
- How to solve more complex models?
- Real application models usually cannot be solved analytically (symbolically)
- A simple example of a transient case:

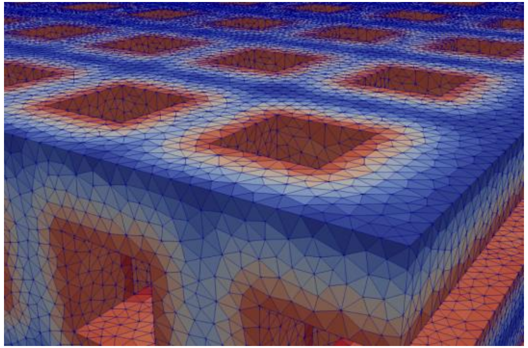
$$\frac{\partial c}{\partial t} = \nabla . (D \nabla c) \rightarrow \frac{\partial \mathrm{c}}{\partial \mathrm{t}} = \mathrm{D}\left(\frac{\partial^2 \mathrm{c}}{\partial \mathrm{x}^2} + \frac{\partial^2 \mathrm{c}}{\partial \mathrm{y}^2}\right)$$



Numerical Computing

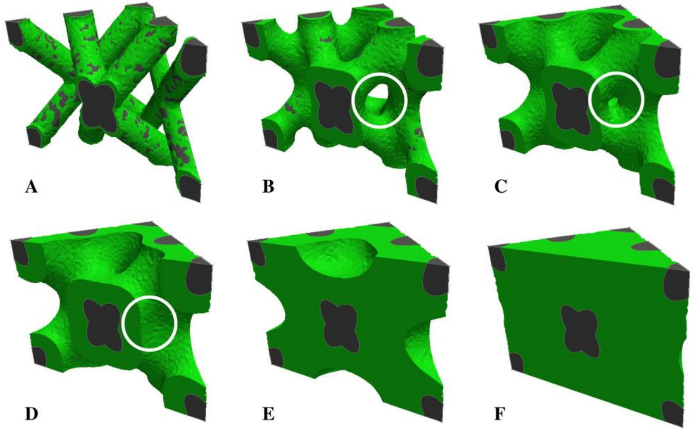
- This is where numerical computing comes into play
- Solving the equations numerically on a mesh
- Widely used in a broad range of engineering fields

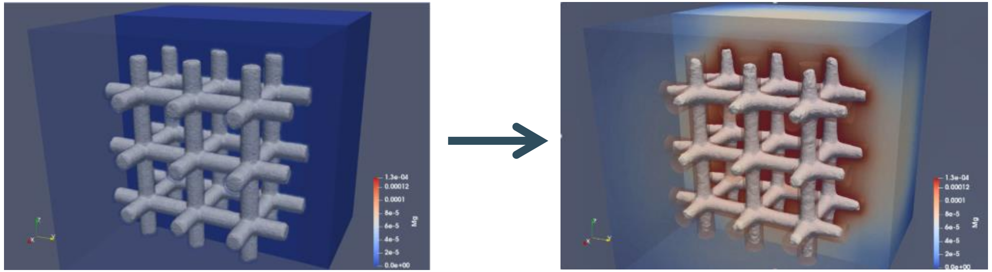




Examples in Tissue Engineering

- Tissue remodeling
- Oxygenation
- Scaffold degradation
- Cell proliferation / tissue growth
- Cell viability
- …

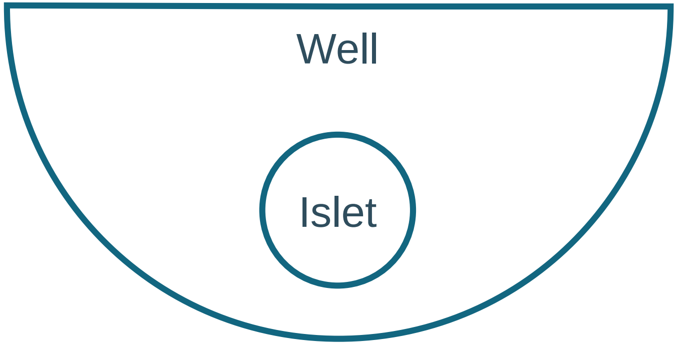




KU LEUVEN

Example: Cell Viability for Pancreatic Islets

- Islet transplantation to treat type 1 diabetes
- Will the implanted islets survive?
- The mathematical model consists of a (nonlinear) reaction-diffusion-advection equation



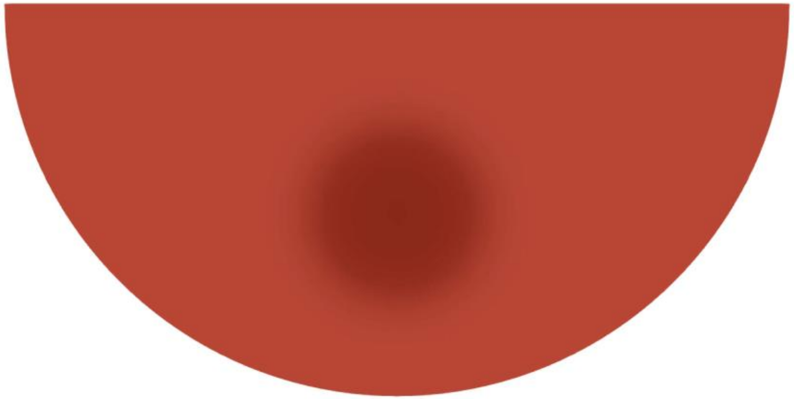
$$\frac{\partial c}{\partial t} + \nabla \cdot (-D\nabla c) = R - \mathbf{u} \cdot \nabla c$$

$$R_{O_2} = R_{\max,O_2} \cdot \varphi \frac{c_{gluc}}{c_{gluc} + C_{MM,gluc}} \cdot \frac{c_{O_2}}{c_{O_2} + C_{MM,O_2}} \cdot \delta(c_{O_2} > C_{cr})$$

KU LEUVEN

Example: Cell Viability for Pancreatic Islets

- One islet

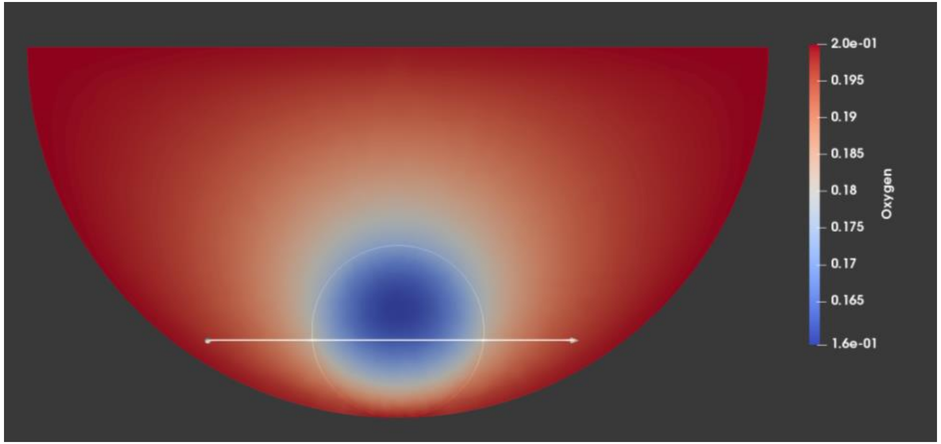


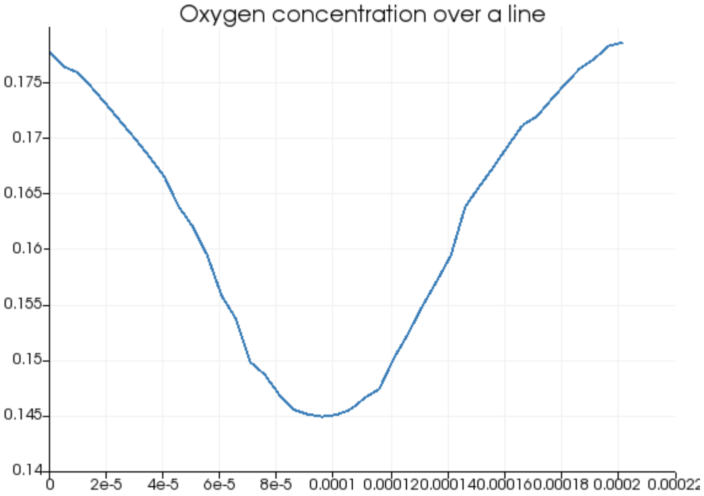
- Two islets

KU LEUVEN

Example: Cell Viability for Pancreatic Islets

- Checking viability (hypoxia)

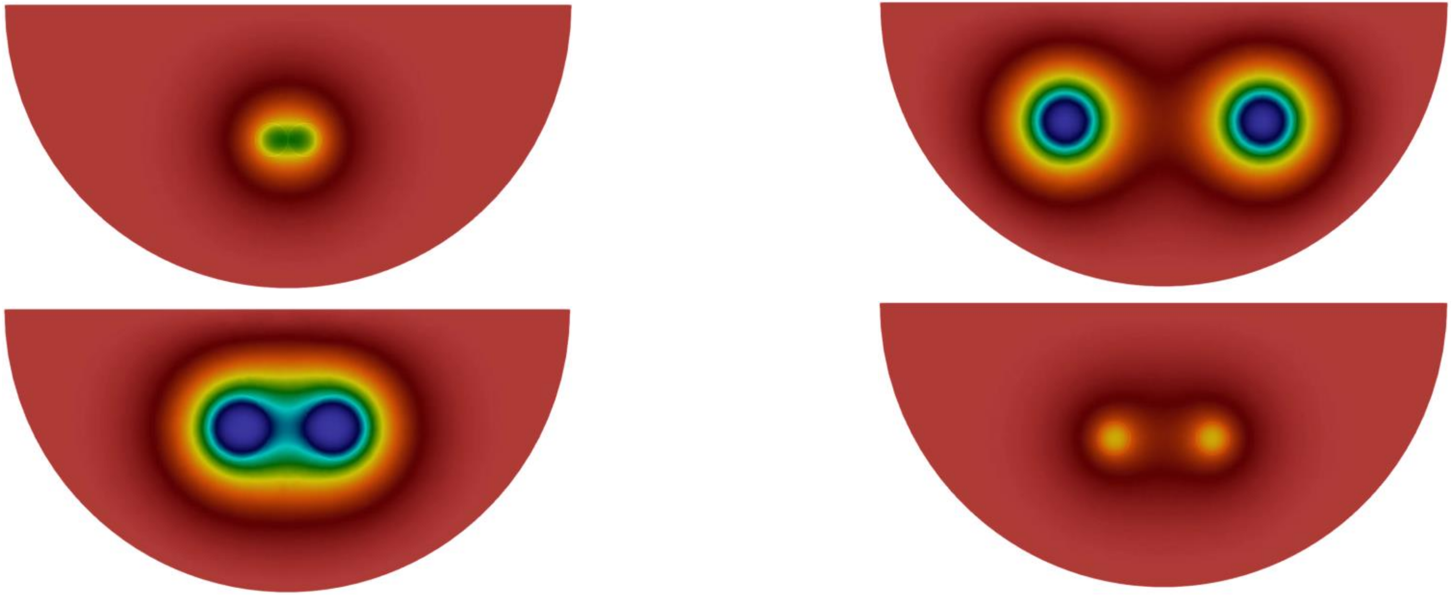




KU LEUVEN

Example: Cell Viability for Pancreatic Islets

- Changing the size and distance of islets



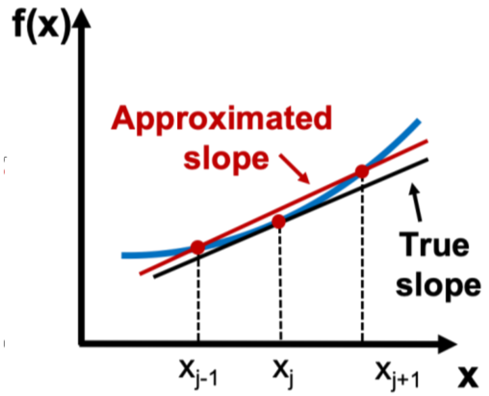
So, How It Works?

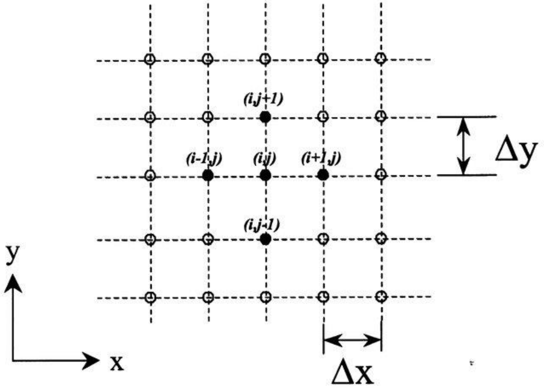
- The core idea is to approximate the derivation terms on a discrete space and time

$$\frac{\partial U}{\partial t} = D\left(\frac{\partial^2 U}{\partial x^2} + \frac{\partial^2 U}{\partial y^2}\right)$$

$$\frac{u_{i,j}^{(n+1)} - u_{i,j}^{(n)}}{\Delta t} = D\left[\frac{u_{i+1,j}^{(n)} - 2u_{i,j}^{(n)} + u_{i-1,j}^{(n)}}{\Delta x^2} + \frac{u_{i,j+1}^{(n)} - 2u_{i,j}^{(n)} + u_{i,j-1}^{(n)}}{\Delta y^2}\right]$$

$$u_{i,j}^{(n+1)} = u_{i,j}^{(n)} + D\Delta t\left[\frac{u_{i+1,j}^{(n)} - 2u_{i,j}^{(n)} + u_{i-1,j}^{(n)}}{\Delta x^2} + \frac{u_{i,j+1}^{(n)} - 2u_{i,j}^{(n)} + u_{i,j-1}^{(n)}}{\Delta y^2}\right]$$

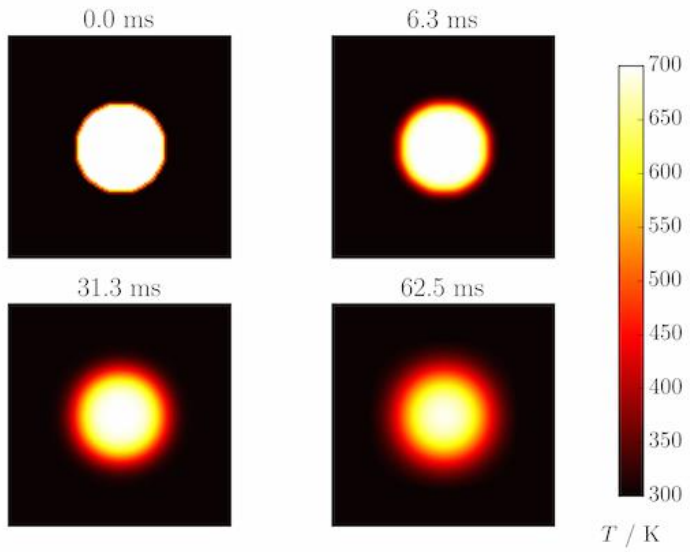




KU LEUVEN

A Simple Implementation

- The discretized form can be simply implemented in Python
- Check out the full implementation in the Jupyter notebooks



References

- GA Truskey, F Yuan, DF Katz, Transport Phenomena in Biological Systems, Pearson Education, 2004
- Peter Buchwald, FEM-based oxygen consumption and cell viability models for avascular pancreatic islets, Theoretical Biology and Medical Modelling, 6, 2009
- Christian Hill, Learning Scientific Programming with Python, Cambridge University Press, 2020