Simulating industrial processes – mathematical models and numerical solutions

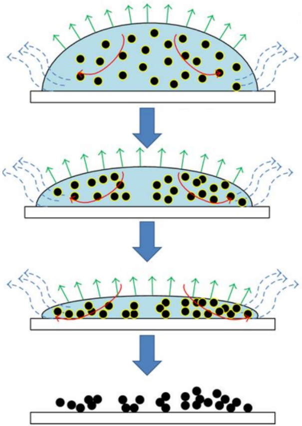
Jonna Roden

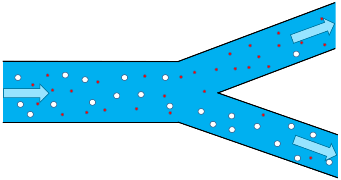
26th November 2021

Structure of the Talk

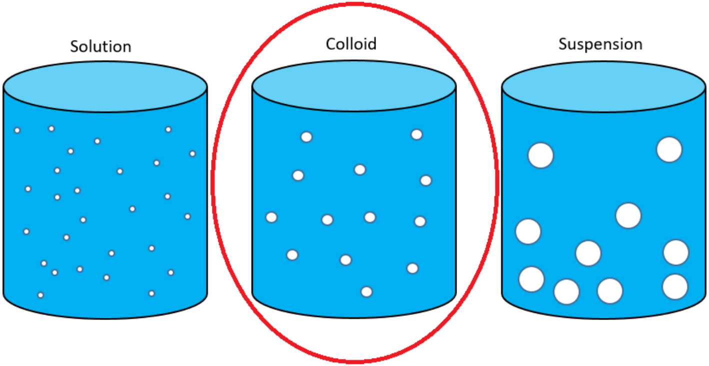
- Motivation!
- Modelling
- Numerical methods
- Some pictures!

What do these pictures have in common?



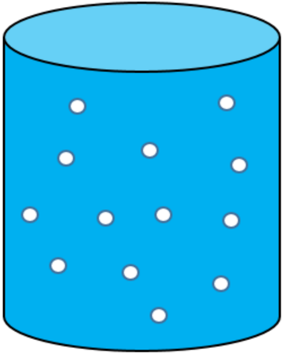


They all look like this ...



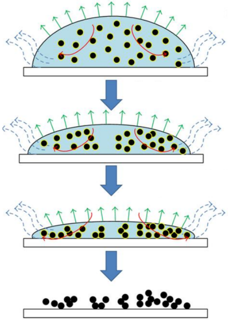
Different modelling approaches

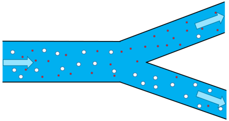
- ODEs for N particles AND n water molecules, $n \gg N$ (impossible computations)
- SDEs for N particles (expensive computations)
- PDEs for the N particle density (impossible computations)
- PDEs for the 1 particle density (good compromise)



What effects can be described with a (non-local) PDE model?

- Forces
- Particle interactions
- Multiple species
- Self-propelled particles
- Anisotropic particles
- Different geometries
- ...





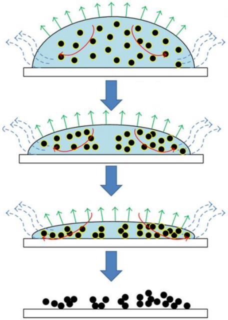
Diffusion and advection

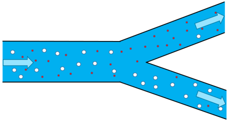
$$\partial_t \rho = \nabla^2 \rho - \nabla \cdot (\rho \vec{w}) \qquad \text{in } \Sigma$$

$$0 = \frac{\partial \rho}{\partial n} - \rho \vec{w} \cdot \vec{n} \qquad \text{on } \partial\Sigma$$

$$\rho(0, \vec{x}) = \rho_0(\vec{x})$$

where ρ : particle density at $(\vec{x}, t)$, $\quad \Sigma = (0, T) \times \Omega$





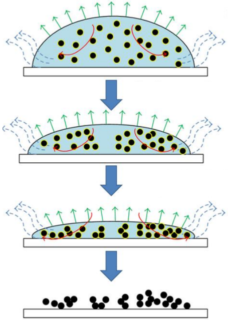
Diffusion, advection and particle interactions

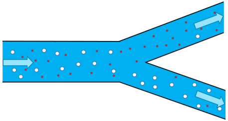
$$\partial_t \rho = \nabla^2 \rho - \nabla \cdot (\rho \vec{w}) + \nabla \cdot \int_\Omega \rho(\vec{x})\rho(\vec{x}\,')\nabla V_2(|\vec{x} - \vec{x}\,'|)d\vec{x}\,' \qquad \text{in } \Sigma$$

$$0 = \frac{\partial \rho}{\partial n} - \rho \vec{w} \cdot \vec{n} + \int_\Omega \rho(\vec{x})\rho(\vec{x}\,')\frac{\partial V_2}{\partial n}(|\vec{x} - \vec{x}\,'|)d\vec{x}\,' \qquad \text{on } \partial\Sigma$$

$$\rho(0, \vec{x}) = \rho_0(\vec{x})$$

where ρ : particle density at $(\vec{x}, t)$, $\quad \Sigma = (0, T) \times \Omega$





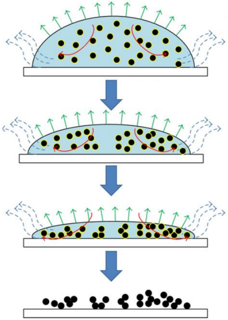
Problem:

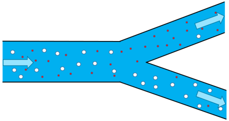
- Particle interaction term is nonlinear and nonlocal (+ nonlocal BCs).

  How do we solve this PDE on complex domains?
  How do we avoid shortcomings of standard methods (FEM/FDM)?

Solution:

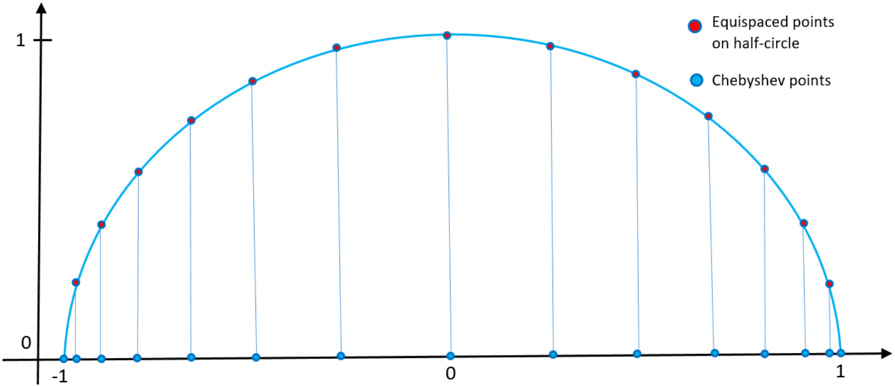
- Pseudospectral methods and spectral element methods.



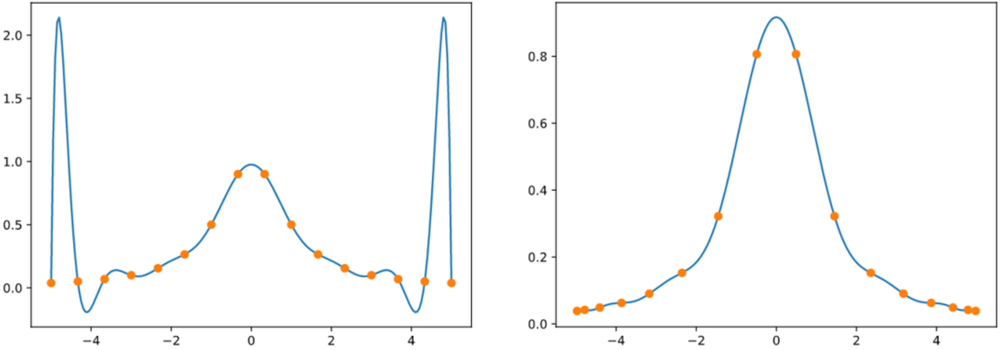


What are pseudospectral methods?

- We start by defining Chebyshev points $x_1, x_2, \ldots, x_n$:



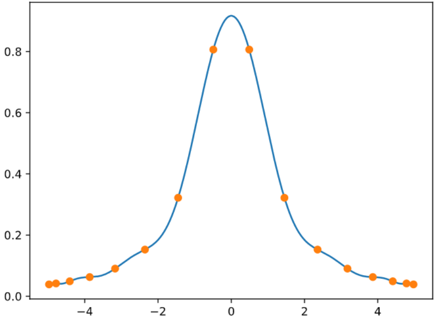
Why Chebyshev points?



Lagrange Interpolation

- Know $f(x_i) := f_i$, want to know $f(x)$ for all x.
- Lagrange polynomial, $p(x)$, such that $p(x_i) = f_i$

$$p(x) = \sum_i f_i \prod_{m \neq i} \frac{x - x_m}{x_i - x_m}.$$



Examples:

Choosing $i = 1, 2$ we have $p(x) = f_1 \frac{x - x_2}{x_1 - x_2} + f_2 \frac{x - x_1}{x_2 - x_1}.$

Choosing $i = 1, 2, 3$ we have

$$p(x) = f_1 \frac{x - x_2}{x_1 - x_2} \frac{x - x_3}{x_1 - x_3} + f_2 \frac{x - x_1}{x_2 - x_1} \frac{x - x_3}{x_2 - x_3} + f_3 \frac{x - x_1}{x_3 - x_1} \frac{x - x_2}{x_3 - x_2}.$$

Differentiation:

- We know $p(x) = \sum_i f_i \prod_{m \neq i} \frac{x - x_m}{x_i - x_m}$.
- Can take derivative

$$p'(x) = \sum_i f_i \frac{d}{dx} \left(\prod_{m \neq i} \frac{x - x_m}{x_i - x_m} \right).$$

- Then

$$\begin{pmatrix} p'(x_1) \\ p'(x_2) \\ \ldots \\ p'(x_n) \end{pmatrix} = D_x \begin{pmatrix} f_1 \\ f_2 \\ \ldots \\ f_n \end{pmatrix}$$

- We can do the same thing for higher derivatives and extend this idea to two dimensions.

Solving a PDE in 1D

We can now solve a PDE on a 1D domain. The PDE

$$\begin{aligned} \partial_t \rho &= \partial_{xx} \rho && \text{in } \Omega \\ 0 &= \partial_x \rho \cdot \vec{n} && \text{on } \partial\Omega \end{aligned}$$

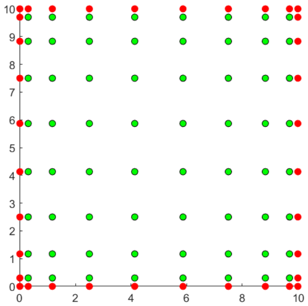
becomes a system of ODEs

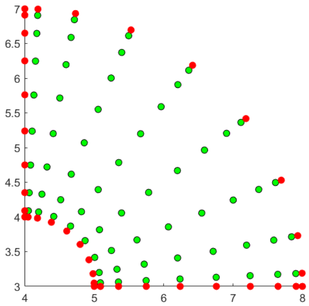
$$\begin{aligned} \partial_t \vec{\rho} &= D_{xx} \vec{\rho} && \text{in } \Omega \\ 0 &= D_x \vec{\rho} \cdot \vec{n} && \text{on } \partial\Omega \end{aligned}$$

where $\vec{\rho} = (\rho(x_1), \rho(x_2), ..., \rho(x_n))$.

Now use a DAE solver (e.g. in Matlab)!

Chebyshev Points in 2D





Solving a PDE in 2D

We can now solve a PDE on a 2D domain. The PDE

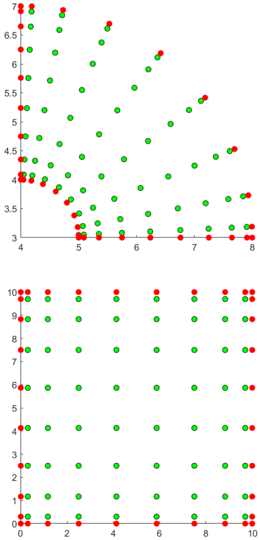
$$\partial_t \rho = \nabla^2 \rho \qquad \text{in } \Omega$$
$$0 = \nabla \rho \cdot \vec{n} \qquad \text{on } \partial\Omega$$

becomes a system of ODEs

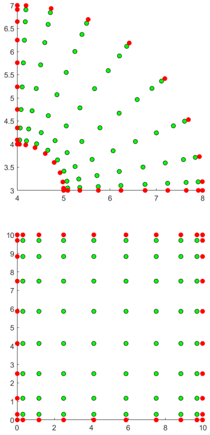
$$\partial_t \vec{\rho} = (D_{xx} + D_{yy})\vec{\rho} \qquad \text{in } \Omega$$
$$0 = (D_x, D_y)\rho \cdot \vec{n} \qquad \text{on } \partial\Omega$$

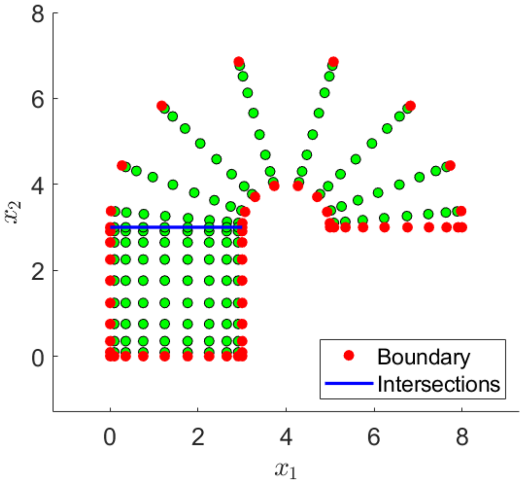
where $\vec{\rho} = (\rho(x_1, y_1), \rho(x_1, y_2), \ldots, \rho(x_n, y_n))$.

Now use a DAE solver (e.g. in Matlab)!



Fancier discretizations in 2D





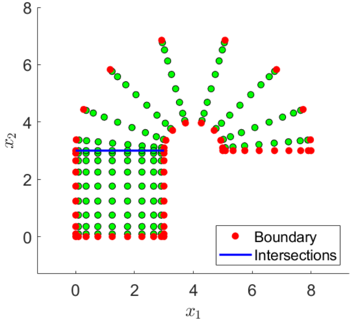
Solving a PDE on a fancy 2D domain

We can now solve a PDE on a *fancy* 2D domain.
The PDE is

$$\begin{aligned} \partial_t \rho &= \nabla^2 \rho && \text{in } \Omega, \\ 0 &= \nabla \rho \cdot \vec{n} && \text{on } \partial\Omega. \end{aligned}$$

Additionally, we need some information on the intersection boundary. We set

$$\begin{aligned} \rho(x_a) &= \rho(x_b) && \text{on } \partial\Omega_{a,b} \\ \nabla \rho(x_a) \cdot \vec{n}_a &= -\nabla \rho(x_b) \cdot \vec{n}_b && \text{on } \partial\Omega_{a,b} \end{aligned}$$



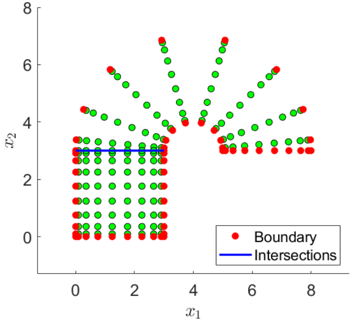
Solving a PDE on a fancy 2D domain

After discretizing, we have

$$\partial_t \vec{\rho} = (D_{xx} + D_{yy})\vec{\rho} \qquad \text{in } \Omega$$
$$0 = (D_x, D_y)\rho \cdot \vec{n} \qquad \text{on } \partial\Omega$$

where $\vec{\rho} = (\rho(x_1, y_1), \rho(x_1, y_2), ..., \rho(x_n, y_n))$, with additional conditions

$$\vec{\rho}_a = \vec{\rho}_b \qquad \text{on } \partial\Omega_{a,b}$$
$$(D_x, D_y)\vec{\rho}_a \cdot \vec{n}_a = -(D_x, D_y)\vec{\rho}_b \cdot \vec{n}_b \qquad \text{on } \partial\Omega_{a,b}$$



Now use a DAE solver (e.g. in Matlab)!

... of course this PDE model...

$$
\begin{aligned}
\partial_t \rho &= \nabla^2 \rho && \text{in } \Omega, \\
0 &= \nabla \rho \cdot \vec{n} && \text{on } \partial\Omega.
\end{aligned}
$$

... is way less complicated...

... than the actual PDE model...

$$\partial_t \rho = \nabla^2 \rho - \nabla \cdot (\rho \vec{w}) + \nabla \cdot \int_\Omega \rho(\vec{x})\rho(\vec{x}\,')\nabla V_2(|\vec{x} - \vec{x}\,'|)d\vec{x}\,' \qquad \text{in } \Sigma$$

$$0 = \frac{\partial \rho}{\partial n} - \rho \vec{w} \cdot \vec{n} + \int_\Omega \rho(\vec{x})\rho(\vec{x}\,')\frac{\partial V_2}{\partial n}(|\vec{x} - \vec{x}\,'|)d\vec{x}\,' \qquad \text{on } \partial\Sigma$$

... but the approach remains the same!

... even if we make it more complicated...

$$\partial_t \rho_1 = \nabla^2 \rho_1 - \nabla \cdot (\rho_1 \vec{w}) + \nabla \cdot \int_\Omega \rho_1(\vec{x}) \left(\rho_1(\vec{x}\,') + \rho_2(\vec{x}\,')\right) \nabla V_2(|\vec{x} - \vec{x}\,'|) d\vec{x}\,'$$

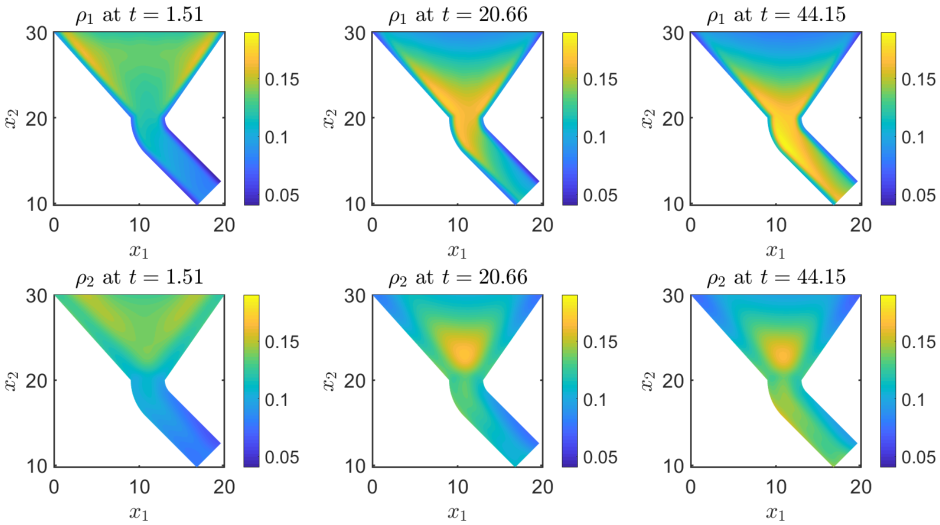
$$\partial_t \rho_2 = \nabla^2 \rho_2 - \nabla \cdot (\rho_2 \vec{w}) + \nabla \cdot \int_\Omega \rho_2(\vec{x}) \left(\rho_2(\vec{x}\,') + \rho_1(\vec{x}\,')\right) \nabla V_2(|\vec{x} - \vec{x}\,'|) d\vec{x}\,'$$

$$0 = \frac{\partial \rho_1}{\partial n} - \rho_1 \vec{w} \cdot \vec{n} + \int_\Omega \rho_1(\vec{x}) \left(\rho_1(\vec{x}\,') + \rho_2(\vec{x}\,')\right) \frac{\partial V_2}{\partial n}(|\vec{x} - \vec{x}\,'|) d\vec{x}\,' \quad \text{on } \partial\Sigma$$

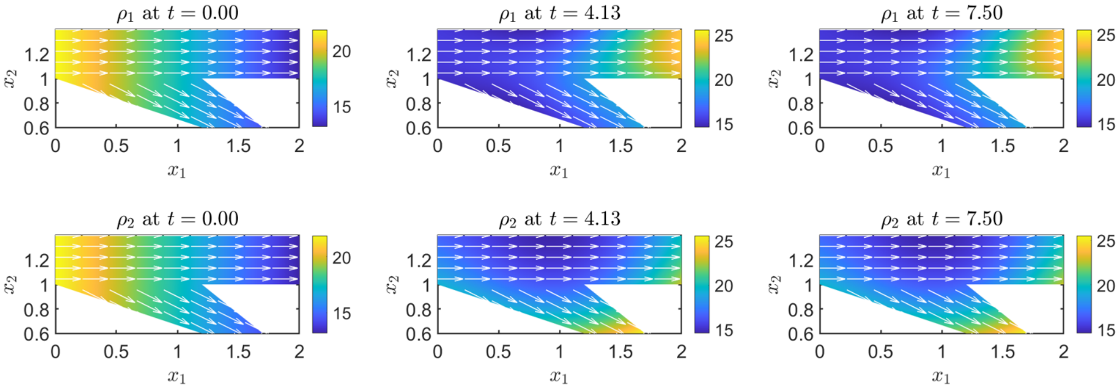
$$0 = \frac{\partial \rho_2}{\partial n} - \rho_2 \vec{w} \cdot \vec{n} + \int_\Omega \rho_2(\vec{x}) \left(\rho_2(\vec{x}\,') + \rho_1(\vec{x}\,')\right) \frac{\partial V_2}{\partial n}(|\vec{x} - \vec{x}\,'|) d\vec{x}\,' \quad \text{on } \partial\Sigma$$

... the approach remains the same!

Some cool results on fancy 2D domains

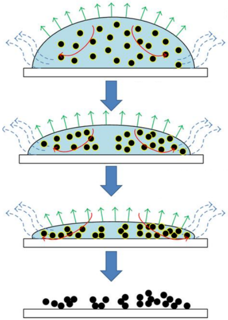


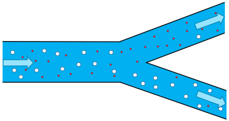
Some cool results on fancy 2D domains



Summary

- Open source implementation.
- Can do cool optimization problems using this.
- Can use it for actual industry problems.





References

M. Aduamoah, B. D. Goddard, J. W. Pearson and J. C. Roden
PDE-Constrained Optimization Models and Pseudospectral Methods for Multiscale Particle Dynamics
Preprint 2021.

A. Nold, B.D. Goddard, P. Yatsyshin, N. Savva and S. Kalliadasis.
Pseudospectral methods for density functional theory in bounded and unbounded domains.
Journal of Computational Physics, 334, 639-664, 2017.
`https://datashare.is.ed.ac.uk/handle/10283/2647` (2DChebClass)

References: Figures

Interpolation Plots https://www.johndcook.com/blog/2017/11/06/chebyshev-interpolation/