ME-474 Numerical Flow Simulation

Exercise: unsteady convection-diffusion

Fall 2022

The aim of this exercise is to implement a FVM code in Matlab to solve the 1D unsteady convection-diffusion equation,

$$\frac{\partial(\rho\phi)}{\partial t} + \frac{\partial(\rho u\phi)}{\partial x} = \frac{\partial}{\partial x}\left(\Gamma\frac{d\phi}{dx}\right),$$

as well as the special cases of pure diffusion ($u = 0$) and pure convection ($\Gamma = 0$). You can reuse the code from week 4 (steady convection-diffusion), and simply wrap the relevant part in a time integration loop.

For the temporal scheme, use the theta method. Define early on in the code a variable `theta`, so that the scheme can be easily modified, simply by choosing the value of $\theta \in [0, 1]$. Using for instance upwind differencing for the convective term (with $u > 0$) and central differencing for the diffusive term, the coefficients of the algebraic equation

$$a_P\phi_P^{n+1} = a_W\phi_W^{n+1} + a_E\phi_E^{n+1} + b(\boldsymbol{\phi}^n)$$

to be solved at each time step are the following:

$$a_P = \frac{\rho_P\Delta x}{\Delta t} + \theta\,(\rho u)_e + \frac{\theta\,\Gamma_e}{\delta x_{PE}} + \frac{\theta\,\Gamma_w}{\delta x_{WP}},$$

$$a_W = \theta\,(\rho u)_w + \frac{\theta\,\Gamma_w}{\delta x_{WP}},$$

$$a_E = \frac{\theta\,\Gamma_e}{\delta x_{PE}},$$

$$b = \left[\frac{\rho\Delta x}{\Delta t} - (1-\theta)\left((\rho u)_e + \frac{\Gamma_w}{\delta x_{WP}} + \frac{\Gamma_e}{\delta x_{PE}}\right)\right]\phi_P^n + (1-\theta)\left((\rho u)_w + \frac{\Gamma_w}{\delta x_{WP}}\right)\phi_W^n + \left(\frac{(1-\theta)\Gamma_e}{\delta x_{PE}}\right)\phi_E^n.$$

Use the following parameters:

- a domain $x \in [0, L]$ with L of your choice between 1 and 5 m;
- a uniform mesh with a number of nodes n of your choice,
- a constant density $\rho = 1$ kg/m^3;
- a constant velocity $u = 0$ for pure diffusion, and a value $u > 0$ otherwise;
- a constant diffusion coefficient $\Gamma = 0$ for pure convection and $\Gamma > 0$ otherwise;
- Dirichlet boundary conditions: $\phi(x = 0, t) = \phi(x = L, t) = \phi_w = 0$ at all times;

Run simulations over an interval $t \in [0, T]$ with T of the order of 5-10 s. Try different schemes: explicit Euler ($\theta = 0$), Crank-Nicolson ($\theta = 0.5$), implicit Euler ($\theta = 1$).

1 Pure diffusion

Here $u = 0$ and $\Gamma = 0.1$ kg/(m.s).

Start the simulation with a uniform initial condition $\phi(x, t = 0) = \phi_0$.

Plot spatial snapshots $\phi(x, t_i)$ as function of x, for different times t_i.

Plot the time evolution $\phi(x_p, t)$ at a specific location, for example $x_p = L/2$. Compare with the theoretical solution (in practice the infinite sum converges quickly):

$$\phi(x,t) = \phi_w + \frac{2}{\pi}(\phi_0 - \phi_w) \sum_{n=1}^{\infty} \frac{1-(-1)^n}{n} e^{-(\Gamma/\rho)\mu_n^2 t} \sin(\mu_n x), \quad \mu_n = \frac{n\pi}{L}.$$

Plot the spatio-temporal evolution $\phi(x, t)$ as function of both x and t. (For 2D plots you can use Matlab functions such as `surf`, `contour` or `imagesc`.)

Play with Δx and Δt and check whether the different temporal schemes are stable, and whether the solution has nonphysical oscillations.

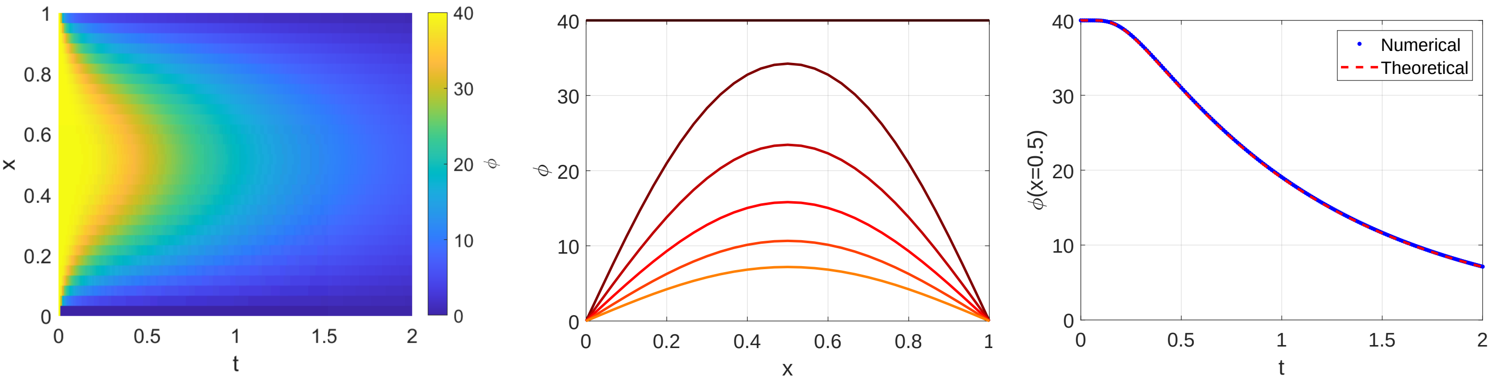

Figure 1: Pure diffusion. Left: Space-time plot of $\phi(x, t)$. Middle: snapshots $\phi(x)$ at different time instants. Right: time evolution of $\phi(x = 0.5, t)$.

2 Convection-diffusion

Here $u > 0$, and $\Gamma = 0$ (pure convection) or $\Gamma = 0.01$ kg/(m.s) (convection-diffusion).

Start the simulation with a Gaussian initial condition $\phi(x, t = 0) \propto \exp(-(x - x_0)^2/\sigma^2)$ with center x_0 and width σ of your choice.

Again, plot several snapshots, the temporal evolution at a specific location, and the spatio-temporal evolution. Try different values of Δx and Δt.

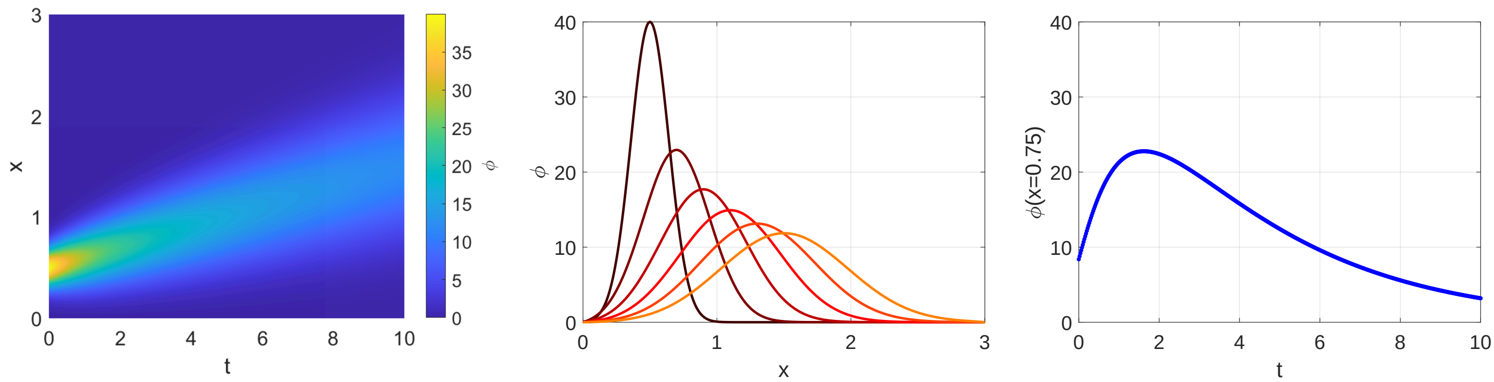

Figure 2: Convection-diffusion. Left: Space-time plot of $\phi(x, t)$. Middle: snapshots $\phi(x)$ at different time instants. Right: time evolution of $\phi(x = 0.75, t)$.