Computational PDEs, Fall 2021
Basic Methods for Advection-Diffusion Equations

Aleksandar Donev
Courant Institute, NYU, donev@courant.nyu.edu

August 10th, 2021
Due Sunday **September 19th 2021**

This and most of the other homeworks will be in 1D and therefore very simple and quick to do computationally in MATLAB. You could easily solve them with a very fine grid (say 1024 points) (or very small time step size), but you will learn nothing from that: Every convergent method will solve a smooth problem very accurately if you make the space-time grid fine enough. However, the goal here is to learn about the **issues** that will also appear in **2D and 3D**, where it would be very challenging indeed to do a simulation with a 1024^3 grid! So please think carefully about how you choose the grid size. The goal is always to make the grid size and time step size as large as possible while controlling the error, so we can solve large problems over a long time in 3D. So **explore** how much you can **reduce the resolution until problems appear**, and try to explain/understand when problems appear and why. The goal here is to think about how the theory in class connects to this specific problem, not to solve the (trivial!) problem itself.

Also important: Please do **not** use any specific form of input functions I give in your solution. For example, the initial condition below is $[\sin(\pi x)]^{100}$. Do **not** use this specific form in your solution. For example, do not give an exact analytical solution to the PDE for this specific form. These homeworks are exercises in **numerical analysis** and not PDE analysis. Do everything numerically and write your code in a way that one can easily change the specific functions appearing in the problem specification.

Please make an effort to write good code. Again, think about how easily this code could be modified to solve a similar but different problem (e.g., if there were a source or "reaction" term on the r.h.s. of the PDE, if the grid were not uniform, if the time step size were variable, etc.); a modular well-designed code will be reusable, will not have repeated code, and will be well-documented and easy to read.

1 Advection-Diffusion in Periodic Domains

Consider numerically solving the advection-diffusion equation

$$u_t + (a(x)u)_x = (d(x)u_x)_x \,,$$

on the periodic domain $0 \le x < 1$, for a *smooth* initial condition

$$u(x,0) = [\sin(\pi x)]^p \,,$$

on a uniform grid [Note: You can use the fact the initial condition is "smooth" if desired, but not its specific form, that is, in your code one should be able to change a line or two to change the initial condition.], where p is an exponent.

In your solution, use the *best method* you (already!) know of and you can write code for. I am purposely vague about the precise meaning of "best," and want you to explain why the method is "best" or at least "good." It is very important to note that whatever "best" means it is specific to the particular parameters, that is, what a good method is depends on the values of the advection velocity $a(x)$, the diffusion coefficient $d(x)$, *and* the initial condition.

Important: You will all come back to (pieces of) this homework later and improve upon your solution until it is "good enough," so the effort spent on it will pay out in the longer term.

1.1 Constant coefficients

First, consider the case $p = 100$, $a = 1$, and $d = 0.001$ being constants, and compute numerically $u(x, t = 1)$. How accurate (i.e., how many digits are correct) is your solution? Does your method work (well) for $d = 0$?

For this part, you can use the fact that the coefficients are constant, or, just use the method from part 1.2; think of this part as a question posed to you on an oral qualifying exam.

1.2 Variable coefficients

Now keep $p = 100$ but make the advection speed and the diffusion coefficient non-constant, though again "smooth",

$$a(x) = a_0 \left(\frac{3}{4} - \frac{1}{4} \sin(4\pi x) \right).$$

$$d(x) = d_0 \left(2 + \cos(2\pi x) \right).$$

By default, use $a_0 = 1$ and $d_0 = 0.001$, but it is important to vary d_0 to vary the relative importance of advection versus diffusion. It is advised to validate/test the advection and diffusion pieces *separately*.

1. Write down the "best" spatio-temporal discretization you know of. Explain what you expect the order of accuracy of the method to be when you refine in space only, in time only, and when you refine *both* space and time together (explain how you would do that).
2. Validate your code in some way (e.g., by solving problem 1.1 using the new method/code, or, if you know how, using a manufactured analytical solution).
3. Refine the resolution in *both space and time* (i.e., in space-time, not space or time separately) to empirically estimate the overall or space-time order of convergence. Indicate what value of the exponent p and spatio-temporal grid parameters you used and why.
4. Solve the advection-diffusion equation for $p = 100$ up to time $t = 1$ again, and if you can indicate how accurate (you think) your solution is.
5. (Optional) Investigate (empirically, not analytically, for now) the stability of your scheme — for example, is it limited in stability by both advection and diffusion or only advection? That is, how big do you think your time step size can be before the scheme becomes unstable.

1.3 Boundary conditions (optional)

Do this part only if you are comfortable with it; everyone will come back to this again as a separate homework. For this part it is OK to take constant coefficients, $a = 1$ and $d = \text{const}$.

Solve the PDE with boundary conditions

$$\begin{aligned} u(0,t) &= \sin^p(-\pi t) \\ u_x(1,t) &= 0 \quad \text{if} \quad d > 0. \end{aligned}$$

Observe that if $d = 0$ the exact solution here is $u(0,t) = \sin^p(\pi(x - t))$, which is the same as for periodic conditions. You can use any value(s) of d and p you wish, but if you do this part be curious and explore a bit to understand the limitations of the method of you chose.