ME614 Spring 2017 - Homework 2

Numerical Stability, Advection & Diffusion

(Due March 10, 2017)

Please prepare this homework in the folder `/homework2/` of your git repository and submit it following the instructions in the last section of Homework 0. The main script should be named `homework.py` and placed in the `/homework2/code/` folder. The submitted code needs to run and create all of the required plots in the folder `/homework2/report/figures/` at once. The use of LaTeX for your report is strongly recommended (but not required) and you can start with a template from http://www.latextemplates.com/. Discussions and sharing of ideas are encouraged but individually prepared submissions (codes, figures, written reports, etc.) are required. A plagiarism detection algorithm will be run against all codes submitted. Do NOT include in your git repository files that are not required for homework submission (e.g. the syllabus, zip files with python libraries, other PDFs, sample python sessions etc). Follow exactly the suggested folder structure of your homework git repository as all of the grading tools on the instructor's end are scripted. Points will be deducted from late submissions at a rate of 20% of the overall homework value per day late. Homeworks are due at 11:59 PM of the due date.

Problem 1

In ideal arithmetics the calculation

$$\begin{aligned}\widehat{\pi}^* &= \pi \times \underbrace{\pi \times \pi \times \pi \times \quad \ldots \quad \times \pi}_{n-\text{times}} \\ \widehat{\pi} &= \widehat{\pi}^* \times \underbrace{\frac{1}{\pi} \times \frac{1}{\pi} \times \frac{1}{\pi} \times \quad \ldots \quad \times \frac{1}{\pi}}_{n-\text{times}}\end{aligned}$$

yields $\pi = \widehat{\pi}$ for any value of n. This is not the case for finite-precision arithmetics, such as the ones performed by a CPU. Try performing the same calculation on your computer by changing the floating-point format of all of the variables involved to half (`float16`), single (`float32`) and double (`float64`) precisions. After evaluating $\widehat{\pi}$ for different precisions, convert it to double precision and then evaluate the round-off error $\epsilon = |\pi - \widehat{\pi}|$ (in double precision).

Perform the following task(s):

(a) Plot the roundoff error versus n for $n \in [1, 2, \ldots, n_{\max}]$ where $n_{\max}$ is the maximum number of multiplications allowed by your machine, at a given precision, before overflow occurs. Results will be messy but revealing.

[10%]

Problem 2

Discretize the linear advection/diffusion partial differential equation

$$\frac{\partial u}{\partial t} + c\frac{\partial u}{\partial x} - \alpha\frac{\partial^2 u}{\partial x^2} = 0 \tag{1}$$

where $c > 0, \alpha > 0$, on a *periodic* domain, with a uniform grid of N points and length L. Adopt a second-order central discretization scheme for the diffusive term at all times (that's the harmless one). For the linear advection term, prepare the following three discretizations: central second-order, upwind first-order and upwind second-order. Also, combine the aforementioned spatial discretization strategies for the convective term with two time advancement strategies: explicit Euler, and Crank-Nicholson. For each of the six (6) combinations, prepare the circulant sparse matrices **A** and **B** that allow to cast (1) in the discrete form

$$\mathbf{A}\,\mathbf{u}^{n+1} = \mathbf{B}\,\mathbf{u}^{n} \tag{2}$$

Perform the following task(s):

(a) Verify that

$$u(x,t) = c_1 e^{-\omega_1^2\,\alpha\,t}\sin\left[\omega_1\,(x - c\,t) - \gamma_1\right] - c_2 e^{-\omega_2^2\,\alpha\,t}\cos\left[\omega_2\,(x - c\,t) - \gamma_2\right] \tag{3}$$

satisfies (1).

[5%]

(b) Using all of the six (6) discretization methods, integrate (1) in the time up to $T_f = (\omega_2^2\,\alpha)^{-1}$ using the discrete form (2) and from the initial conditions

$$u(x,0) = c_1\sin(\omega_1\,x - \gamma_1) - c_2\cos(\omega_2\,x - \gamma_2) \tag{4}$$

where $\omega_1 = 2\pi/L$ and $\omega_2 = 2\,m\,\pi/L$ and the constants $\gamma_1, \gamma_2, c_1, c_2, m, L$ can take on arbitrary (non-zero) values of your choice. Plot the numerical solution at time $t = T_f$ obtained with $N = 10, 25, 50, 100$ points comparing one of the best performing discretization methods with the least performing one. Make sure to include on the same plot the analytical solution sampled on the finest grid (as a reference).

[10%]

(c) For all of the aforementioned spatial discretizations but for the Crank-Nicholson time-advancement method only, plot the root-mean-square error at $t = T_f$ versus the number of grid points (for a constant, very small Δt) and versus Δt (for a constant, very small Δx). Verify that the resulting order of accuracy (both in time and space) is the expected one. Use the initial conditions (4) and analytical solution (3).

[20%]

(d) Plot the `spy` of **A** and **B** for $N = 10$ for all of the 6 spatial/temporal discretization strategies.

[5%]

(e) For explicit Euler only, plot flooded iso-contours of the spectral radius (between levels $[0, 2]$) of the transition matrix $\mathbf{T} = \mathbf{A}^{-1}\mathbf{B}$ in the C_α vs C_c plane where

$$C_c = \frac{\Delta t\,c}{\Delta x}, \quad C_\alpha = \frac{\Delta t\,\alpha}{\Delta x^2}$$

are the advection and diffusion-based Courant-Friedrichs-Lewy (CFL) numbers, respectively. What you will have found, this way, is the region of computational instability/stability: C_α and C_c such that the spectral radius is above/below 1. This is the region where round-off error is (not) expected to cause your calculation to blow up.

[15%]

Problem 3

In the *non-periodic* domain $x \in [0, L]$, discretize the advection/diffusion partial differential equation

$$\frac{\partial u}{\partial t} + c\frac{\partial u}{\partial x} - \alpha\frac{\partial^2 u}{\partial x^2} = -\beta\, u \tag{5}$$

where $\beta > 0$, with boundary conditions $u(0,t) = a\cos(\omega\, t)$ and $\frac{du}{dx}\Big|_{x=L} = 0$. Adopt second-order central discretization for the diffusive terms and second-order upwind for the convective terms on a *non-uniform* grid with points clustered at $x = 0$. Do not collocate grid points right at the boundary but use the ghost-cell method. Start with uniform initial conditions

$$u(x,0) = 0.0 \tag{6}$$

and integrate in time with a Crank-Nicholson scheme.

Perform the following task(s):

(a) Acquire some familiarity with the physics of the problem by playing with the parameters $c(> 0), \alpha, \beta, \omega, a$ showcasing the effects of each of these parameters on the solution. For one combination of these parameters, plot the transient as the solution evolves from the initial conditions to the final steady state.

[10%]

(b) The "steady state" solution of (5), with the given boundary conditions, should form a Stokes boundary layer on the left end of the domain with thickness, δ, which is function of all the parameters of the problem,

$$\delta = f(c, \alpha, \beta, L, a, \omega). \tag{7}$$

By adopting the Π-theorem, derive the functional dependency among the various Π groups that can be generated from (7) and with the use of the numerical simulations collect some data points to verify the derived scaling laws.

[25%]