HOMEWORK 1: FINITE DIFFERENCE METHOD

Due electronically February 13 at 5:00pm

In Group Exercise 0, you created a finite difference solver for the diffusion equation. You will build upon this framework in this homework set to develop and explore a set of finite difference tools. For this assignment, you will need:

- Your solution to Group Exercise 0
- **ftcs.py**: python file where you will implement an FTCS solver
- **btcs.py**: python file where you will implement a BTCS solver
- **sample_in.json**: JSON file containing an example input deck

1. (4 pts) The finite difference method is a very general way to find solutions for the diffusion equation. Of course, the specific solution depends on defining the system of interest by specifying system size, initial and boundary conditions, step sizes in space and time, the diffusion coefficient, and so on. That is a lot of information to include in a function call! So instead, follow the prompt in the **ftcs.py** docstring to read input from an "input deck" given in a JSON file and use your FTCS solver to compute a composition profiles. Turn in your completed **ftcs.py** file with your solutions filled in; there is no need to turn in the input or output files you used.

2. In this problem, you will use your FTCS code to explore diffusion and how it is modeled by explicit finite difference models. To start, consider a thin film with an initial composition profile $c(x,0) = \sin(\pi x/xmax)$, where $0 \leq x \leq xmax$, that is vacuum annealed so that the surface composition $c(0,t) = c(xmax,t) = 0$ at all times. Assume a diffusion coefficient $D = 0.05$. (Note that x and t are in arbitrary units scaled by D.)

   (a) (2 pts) Using FTCS finite differencing, solve for the composition at $t = 0$ and 10. (Be sure to select a stable combination of Δx and Δt.) Does the system reach steady state during the simulation? If not, repeat your simulation for longer times until steady state is reached. Report your estimate for the time required to achieve steady state, and turn in a plot showing the initial and steady state composition profiles.

   (b) (2 pts) As we know, the FTCS scheme is unstable when $r = D\Delta t/\Delta x^2 > 0.5$. For the system in part (a), select Δx and Δt to give $r \sim 0.5$ and adjust $tmax$ to capture the beginning of the instability. Turn in a plot that shows the instability, and report the r value.

   (c) (4 pts) For this system, the exact solution of the diffusion equation is

   $$c(x,t) = \sin\left(\frac{\pi x}{xmax}\right)\exp\left(-\frac{D\pi^2 t}{xmax^2}\right)$$

   For constant value of Δx and at least three values of Δt, calculate the average FTCS error at $t = 10$. Turn in a plot that shows that the FTCS scheme accumulates error in proportion to Δt.

(d) (4 pts) Still using $D = 0.010$, select two other combinations of initial composition profile and constant composition boundary conditions. Turn in plots showing calculated composition profiles at $t = 0$ and 10, and report your initial and boundary conditions.

3. Implicit finite difference methods have one major advantage over explicit ones: They are unconditionally stable. Follow the prompt in the **btcs.py** docstring to implement a backward time central space finite difference solver and demonstrate this advantage.

   (a) (4 pts) Turn in your completed **btcs.py** file with your solutions filled in.

   (b) (4 pts) For the system described in problem 2(b), turn in a plot that shows that the BTCS scheme does not show an instability by $t = 10$. Then, select Δx and Δt to give $r = 2$, and turn in a plot that shows the solution remains stable.

   (c) (2 pts) Choose one of your systems from problem 2(d) and recalculate the composition profile using BTCS. Are the results the same as for FTCS?

   (d) (2 pts) The BTCS scheme is unconditionally stable, and it has the same error accumulation as FTCS. So why would we ever use FTCS? In other words, what is the most significant disadvantage of BTCS relative to FTCS?