Numerical PDEs Homework #2

In this worksheet, we'll compare efficiency in different time stepping algorithms.

1 The Heat Equation

Find $f(x, y, t)$, the appropriate Dirichlet boundary conditions, and the appropriate initial condition, so that the solution to the forced heat equation

$$u_t = u_{xx} + u_{yy} + f(x, y, t),\ t > 0,\ x, y \in [0, 2\pi] \times [0, 2\pi] \tag{1}$$

is $u(x, y, t) = \cos(10t)\sin(x)\sin(y)$. Use finite differences with $N = 64$ grid points in each direction to discretize in space, and the following time stepping schemes to discretize in time

1. Forward Euler
2. Backward Euler
3. Crank–Nicolson
4. Explicit Midpoint (RK2)
5. BDF2
6. Extra credit: RK45 (please use an open implementations e.g ode45, it's very annoying to implement)

2 Problem 1:

Solve the heat equation up to $T = 2\pi(seconds)$ using different time step sizes Δt (except for RK45, let's just trust that), and show that the relative error

$$Err = \frac{\|u_{numerical} - u_{exact}\|}{\|u_{exact}\|}$$

scales like $\mathcal{O}(\Delta t)$ for both Forward/Backward Euler methods, and scales like $\mathcal{O}(\Delta t^2)$ for the rest. **Note: a common mistake here is an 'off by one error' where you're actually calculating the solution at $t = 2\pi - \Delta t$ or $t = 2\pi + \Delta t$. This can happen if you start or end the time loop at the wrong point. Also using a value of Δt that does not evenly divide $T = 2\pi$ will cause problems.**

3 Problem 2:

Using each of the above methods, find the largest value of Δt that you can get away with in order to meet the following error thresholds (the largest value of Δt that guarantees your numerical error is less that e.g 10^{-2})

1. $Err \leq 5 \times 10^{-2}$
2. $Err \leq 1 \times 10^{-3}$
3. $Err \leq 5 \times 10^{-6}$

At each error threshold, calculate the total cost of each method using the following metric

$$\text{Cost} = \Big(\text{cost of Cholesky factorization}\Big) + $$
$$(\#\text{ of timesteps}) \times \Big([\#\text{ of sparse matrix mult.s per step}] \times [\text{cost of sparse mult.}] + $$
$$[\#\text{ of sparse matrix solves per step}] \times [\text{cost of sparse solve}] \Big)$$

where

$$\text{cost of sparse mult.} \approx 5 \times N$$
$$\text{cost of sparse solve} \approx N \times \log(N)$$
$$\text{cost of Cholesky factorization} \approx N^{3/2},$$

and $N = 64^2$ is the system size. **Which method is cheapest at each error threshold?**

Note: the costs of matrix multiplication, factorization, and factorized solves that I've given above are specific to this problem. In problems with nastier matrices, these costs can be much worse.