Homework 1 for MATH5311

Yan Bokai
Department of Mathematics

THE HONG KONG
UNIVERSITY OF SCIENCE
AND TECHNOLOGY

October 2, 2022

Problem 1

Solve

$$\begin{cases} u_t = u_{xx}, \\ u(0,t) = u(1,t) = 0, \\ u(x,0) = \begin{cases} 2x & 0 \le x \le \frac{1}{2}, \\ 2-2x & \frac{1}{2} < x \le 1, \end{cases} \end{cases}$$

by the implicit scheme with $\Delta x = 0.05$, $\Delta t = 0.01$ to $t_F = 1$ and compare the results with the result from the explicit scheme.

Solution. For explicit scheme, the iteration formula is

$$U^{n+1} = AU^n,$$

where $U^n = (U_1^n, U_2^n, \cdots, U_{N-1}^n)$,

$$A = \begin{pmatrix} 1-2\nu & \nu & & & \\ \nu & 1-2\nu & \nu & & \\ & \ddots & \ddots & \ddots & \\ & & \nu & 1-2\nu & \nu \\ & & & \nu & 1-2\nu \end{pmatrix},$$

and $\nu = \dfrac{\Delta t}{(\Delta x)^2}$.

For implicit scheme, the iteration formula is

$$BU^{n+1} = U^n,$$

where $U^n = (U_1^n, U_2^n, \cdots, U_{N-1}^n)$, and

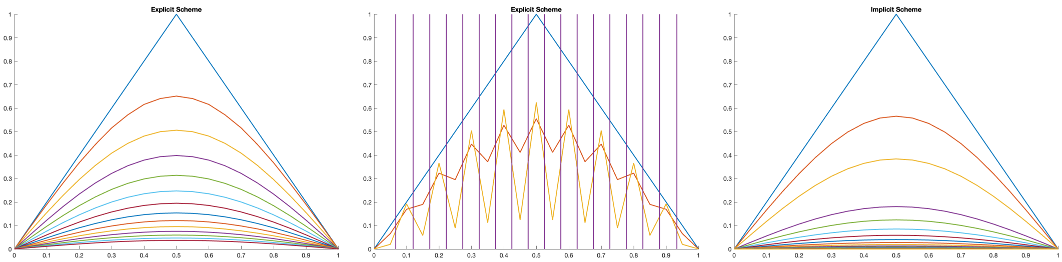
$$B = \begin{pmatrix} 1+2\nu & -\nu & & & \\ -\nu & 1+2\nu & -\nu & & \\ & \ddots & \ddots & \ddots & \\ & & -\nu & 1+2\nu & -\nu \\ & & & -\nu & 1+2\nu \end{pmatrix}.$$

By Cholesky decomposition, we may compute the iteration in two steps,

$$C^T y = U^n,$$
$$CU^{n+1} = y,$$

where C is a bidiagonal matrix, and the complexity of each iteration can be reduced to $O(N)$.

By numerical experiments, we may compare the two schemes. You may refer to the appendix for code details.



(a) (b) (c)

Figure 1: Comparison of the two schemes. (a) Explicit scheme with $\Delta t = 0.0012$. $\nu \leq \frac{1}{2}$, so the scheme is convergent and stable. (b) Explicit scheme with $\Delta t = 0.0013$. $\nu > \frac{1}{2}$ and the extreme fluctuations in the graph show that the scheme is not stable. (c) Implicit scheme with $\Delta t = 0.01$. The graph is consistent with unconditional stability for the implicit scheme.

Problem 2

Show the θ-method is second order in time when $\theta = \frac{1}{2}$.

Solution. Let $T_j^{n+\frac{1}{2}}$ be the truncation error. By definition, $T_j^{n+\frac{1}{2}}$ satisfies

$$\frac{u(x_j, t_{n+1}) - u(x_j, t_n)}{\Delta t} = \frac{u(x_{j+1}, t_n) - 2u(x_j, t_n) + u(x_{j-1}, t_n)}{2(\Delta x)^2} + \frac{u(x_{j+1}, t_{n+1}) - 2u(x_j, t_{n+1}) + u(x_{j-1}, t_{n+1})}{2(\Delta x)^2} + T_j^{n+\frac{1}{2}}$$

We expand all terms at $(x_j, t_{n+\frac{1}{2}})$ and by cancellation,

$$T_j^{n+\frac{1}{2}} = [\frac{1}{24} u_{ttt}(x_j, t_{n+\frac{1}{2}}) + \frac{1}{8} u_{xx}(x_j, t_{n+\frac{1}{2}})](\Delta t)^2 + \frac{1}{12} u_{xxxx}(x_j, t_{n+\frac{1}{2}})(\Delta x)^2 + o((\Delta t)^2) + o((\Delta x)^2).$$

So the θ-method is second order in both time and space for $\theta = \frac{1}{2}$.

Appendix

```
dx = 0.05;
dt = 0.0012;
t_F = 1;
nu = dt/dx^2;
u = [2*(dx:dx:0.5), 2-2*(0.5+dx:dx:1-dx)]';
N = length(u);
A = diag(repmat(1-2*nu, [1, N])) + diag(repmat(nu, [1, N-1]), 1) + diag(repmat(nu,
    [1, N-1]), -1);

t = 0;
figure;
hold on
while t <= t_F
    if mod(t/dt, 20) < 1e-10
        plot(0:dx:1, [0; u; 0], 'linewidth', 1.5);
    end
    u = A*u;
    t = t + dt;
end
title('Explicit Scheme')
ylim([0, 1])
```

Algorithm 1: Code for explicit scheme

```
dx = 0.05;
dt = 0.01;
t_F = 1;
nu = dt/dx^2;
u = [2*(dx:dx:0.5), 2-2*(0.5+dx:dx:1-dx)]';
N = length(u);
A = diag(repmat(1+2*nu, [1, N])) + diag(repmat(-nu, [1, N-1]), 1) + diag(repmat(-
    nu, [1, N-1]), -1);

t = 0;
figure;
hold on
while t <= t_F
    if mod(t/dt, 3) < 1e-10
        plot(0:dx:1, [0; u; 0], 'linewidth', 1.5);
    end
    u = A\u;
    t = t + dt;
end
title('Implicit Scheme')
ylim([0, 1])
```

Algorithm 2: Code for implicit scheme