W.D. Henshaw **Math 6840: Problem Set 2** **Due:** Thurs. Feb 7, 2019.

**General note: For computer exercises please provide listings of all computer programs that were used to generate the results. Please present your results in a clear and neat manner.**

**1**. (20 points) Convergence proof. Consider the initial-boundary-value problem (IBVP) for the heat equation,

$$
\begin{aligned}
&u_t = \kappa u_{xx}, && a < x < b, && 0 \le t \le t_{\text{final}}, \\
&u(a,t) = g_a(t), && u(b,t) = g_b(t), && t > 0 \\
&u(x,0) = u_0(x), && && a < x < b.
\end{aligned}
$$

(a) Carefully write out the BE-CD2 scheme (backward-Euler in time and second-order accurate central difference in space) for this IBVP. Be sure to specify where each equation holds.

(b) Determine the equations satisfied by the error, $e_i^n = u(x_i,t^n) - U_i^n$, (including boundary and initial conditions). Also determine the leading order term in the truncation error, $\tau_i^n$ (express this in terms of spatial derivatives of $u$ at time $t^n$).

(c) In class we proved the convergence of the FE-CD2 scheme. Repeat this proof but for the BE-CD2 scheme. What conditions on $r = \kappa\Delta t/\Delta x^2$ are required in the proof?

**2**. (20 points) Mixed boundary conditions and ghost points (centered). Consider the initial-boundary-value problem (IBVP) for the heat equation, with mixed boundary conditions

$$
\begin{aligned}
&u_t = \kappa u_{xx} + f(x,t), && && a < x < b, \quad 0 \le t \le t_{\text{final}}, \\
&\alpha_a u(a,t) + \beta_a u_x(a,t) = g_a(t), && \alpha_b u(b,t) + \beta_b u_x(b,t) = g_b(t), && t > 0 \\
&u(x,0) = u_0(x), && && a < x < b,
\end{aligned}
$$

with $a = 0$, $b = 1$, $\kappa = 0.1$, $\alpha_a = 0.5$, $\beta_a = -1$, $\alpha_b = 0.25$, and $\beta_b = 2$.

(a) Write down the AB2-CD2 scheme to solve these equations that uses the second-order accurate Adams-Bashforth scheme in time and second-order accurate centered differences in space. For the ODE $y' = f(y,t)$ the AB2 scheme

$$
Y^{n+1} = Y^n + \Delta t\Big(\frac{3}{2}f(Y^n,t^n) - \frac{1}{2}f(Y^{n-1},t^{n-1})\Big).
$$

This is a three-level scheme in time and needs a starting guess, use the exact solution, $u^e(x,t)$, at $t = -\Delta t$ to get started. Use the AB2-CD2 scheme for the interior equations and apply this scheme on boundary and interior points, $i = 0,1,2,\ldots,N_x$. Use a centered approximation to the mixed boundary conditions using one ghost-point on the left and right. For example, on the left use

$$
\alpha_a U_0^n + \beta_a D_0 U_0^n = g_a(t^n).
$$

These equations will determine the values at the ghost points. Be sure to carefully write the boundary and initial conditions and indicate where each equation is applied in index space.

(b) Write a Matlab code to solve the scheme from part (a).

Test the code in parts (c) and (d) below using a mesh refinement study as follows. Solve the problem for $N_x = 10 \times 2^m$, $m = 1, 2, 3, 4, 5$ using a time-step that satisfies $\kappa\Delta t/\Delta x^2 = 1/4$ (adjust this time-step, as shown in class, to exactly reach $t = t_{\text{final}}$ using $N_t$ time-steps). Print $m$, $N_x$, $\Delta t$ and the max-norm errors $E_m$ at $t = 1$, for each m, along with the ratios $E_{m-1}/E_m$. Use the print statements (or similar):

```
fprintf('Heat: %s, %s%d, %s: t=%4.2f: Nx=%3d Nt=%5d dt=%9.3e maxErr=%8.2e',ts,bcOption,bcOrder,...
        ms,t,Nx,Nt,dt,errMax(m));
if( m==1 ) fprintf('\n'); else fprintf(' ratio=%8.2e\n',errMax(m-1)/errMax(m)); end;
```

In my code `ts='AB2'`, `bcOption='centered' or 'one-sided'`, `bcOrder=1,2` and `ms='trig' or 'poly'`.

(c) Test your code using the polynomial manufactured solution,

$$u^e(x,t) = (b_0 + b_1 t + b_2 t^2)\,(c_0 + c_1 x + c_2 x^2),$$

for $b_0 = .5$, $b_1 = .7$, $b_2 = .9$ and $c_0 = 1$, $c_1 = .8$, $c_2 = .6$. Perform a grid refinement study and confirm the the scheme is exact (up to round-off errors).

(d) Repeat (c) but for the trigonometric manufactured solution

$$u^e(x,t) = \cos(k_t t)\,\cos(k_x x),$$

with $k_t = \pi$ and $k_x = 7\pi$. Perform a grid refinement study and confirm that the ratio of the errors approaches 4.
In addition, for this case, plot, on a single graph, the errors at the final time for m=1,2,3 (i.e. all errors on the same graph).

**3**. (20 points) Mixed boundary conditions (one-sided).

Repeat the previous question but now using some one-sided approximations in the boundary conditions.

(a) Use a second-order one-sided approximation to the boundary condition as given in class. This uses the following approximation on the left-side:

$$\frac{\partial u}{\partial x}(0,t^n) \approx \frac{-3U_0^n + 4U_1^n - U_2^n}{2\Delta x}$$

Use the one-sided approximations to determine the values on the boundary. Use third-order extrapolation to assign the values in the ghost-points (i.e. solving for $U_{-1}^n$ from $D_+^3 U_{-1}^n = 0$).
Write down the AB2-CD2 discrete approximation (including interior, boundaries and initial conditions) and perform the tests and plots from parts (c) and (d) above for this approximation.

(b) Use a first-order accurate one-sided approximation to the boundary condition. On the left side use $D_+$ as the approximation to $\partial_x u$:

$$\frac{\partial u}{\partial x}(0,t^n) \approx D_+ U_0^n$$

while on the right use $D_-$. Use the one-sided approximations to determine the values on the boundary. Use third-order extrapolation to assign the values in the ghost-points.

Write down the AB2-CD2 discrete approximation (including interior, boundaries and initial conditions) and perform the tests and plots from parts (c) and (d) above for this approximation. In this case, however, the solution may not be second-order accurate.

**4**. (20 points) (Implicit time-stepping) Consider the initial-boundary-value problem (IBVP) for the *modified* heat-equation,

$$\begin{aligned} &u_t = \kappa u_{xx} - \alpha\, u, && a < x < b, \quad 0 \le t \le t_{\text{final}}, \\ &u(a,t) = 0, \quad u(b,t) = 0, && t > 0 \\ &u(x,0) = \sin(k\pi x), && a < x < b, \end{aligned}$$

with $a = 0$, $b = 1$, $\alpha = .1$, and $k$ an integer.

(a) Find the exact solution which is of the form $u(x,t) = \hat{u}(t)\, \sin(k\pi x)$.

(b) Write down the theta-CD2 scheme that uses the theta-method in time and second-order central differences in space. Be sure to specify the initial conditions and boundary conditions and indicate where each equation is applied in index space.

(c) Find the leading order term in the equation-truncation error (ETE) for the theta-CD2 scheme. (Hint: the scheme should be second-order accurate in time when $\theta = 1/2$.)

(d) Write a Matlab code to solve modified heat-equation IBVP using the theta-CD2 scheme. Use values $k = 4$, $\kappa = .05$, $\alpha = .1$ and $t_{\text{final}} = 0.5$.
Provide results for three-cases

1. $\theta = 0.5$ (Crank-Nicolson).
2. $\theta = 1.0$ (Backward-Euler).
3. $\theta = .25$ (Partially implicit).

For each case, solve the problem for $N_x = 10 \times 2^m$, $m = 1, 2, 3, 4$, using a time-step that satisfies $\Delta t = \Delta x$ (adjust this time-step, as shown in class, to exactly reach $t = t_{\text{final}}$ using $N_t$ time-steps). Print $m$, $N_x$, $N_t$, $\Delta t$ and the max-norm errors $E_m$ at $t = 1$, for each m, along with the ratios $E_{m-1}/E_m$. Use the print statements:

```
fprintf('theta=%3.1f: t=%10.4e: Nx=%3d Nt=%4d dt=%9.3e maxErr=%8.2e',theta,t,Nx,Nt,dt,errMax(m));
if( m==1 ) fprintf('\n'); else fprintf(' ratio=%8.2e\n',errMax(m-1)/errMax(m)); end;
```

For $m = 1$, plot the computed and exact solution at $t = 1$ on the same graph. On a second graph, plot the errors for m=1,2,3 (i.e. all errors on the same graph). In each case, label the axes and provide a title and legend. Confirm the expected order of accuracy for cases 1 and 2. What happens for case 3?