Assignment #7

Due Monday, 10 April 2023, at the start of class

Please read textbook[1] Chapters 6 and 7. Within this material we are de-emphasizing the discussion of multistep methods, so full understanding of sections 5.9, 6.4, 7.3, and 7.6.1 is not expected. However, *actually reading* these Chapters is going to be important to success on this and later Assignments.

Problem P30. Consider the "θ-methods" for $u' = f(t,u)$, namely

$$U^{n+1} = U^n + k\Big[(1-\theta)f(t_n, U^n) + \theta f(t_{n+1}, U^{n+1})\Big],$$

where $0 \le \theta \le 1$ is a fixed parameter. I did all parts of this problem by hand.

a) Cases $\theta = 0, 1/2, 1$ are all familiar methods. Name them.

b) Find the (absolute) stability regions for $\theta = 0, 1/4, 1/2, 3/4, 1$. (*Hint.* Write the complex number $z = k\lambda$ as $z = x + iy$. Find the circles!)

c) Show that the θ-methods are A-stable for $\theta \ge 1/2$.

Problem P31. Consider this Runge-Kutta method, a one-step and implicit interpretation of the multistep midpoint method:

$$\begin{aligned} U^* &= U^n + \frac{k}{2} f\big(t_n + k/2, U^*\big), \\ U^{n+1} &= U^n + k f\big(t_n + k/2, U^*\big). \end{aligned}$$

The first stage is backward Euler to determine an approximation to the value at the midpoint in time. The second stage is a midpoint method using this value.

a) Determine the order of accuracy of this method. That is, compute the truncation error accurately enough to know the power p in $\tau = O(k^p)$.

b) Determine the stability region. Is this method A-stable? Is it L-stable?

Problem P32. Subsection 5.9.4 explains why the explicit trapezoid rule (5.53) is sometimes called a *predictor-corrector* method. A full Euler step is used to estimate ("predict") the new solution value, and then a formerly-implicit method, the original trapezoid method, is used to actually take the step, which "corrects" the prediction. This problem attempts to construct a predictor-corrector method with higher order.

a) On page 132 (section 5.9) there is a list of explicit Adams-Bashforth and implicit Adams-Moulton multistep methods. Suppose we cut a step into two parts, with stage steps $k/2$. From U^n suppose we take a forward Euler step of length $k/2$ to

[1] R. J. LeVeque, *Finite Difference Methods for Ordinary and Partial Diff. Eqns.*, SIAM Press 2007

give U^* at $t_n + k/2$. Then we use U^n and U^* in the second-order ("2-step") Adams-Bashforth scheme to give $U^\dagger$ at $t_n + k = t_{n+1}$. The value $U^\dagger$ is the predicted value at t_{n+1}. Then we use the second-order ("2-step") Adams-Moulton scheme, using known values $U^n, U^*, U^\dagger$ on the right-hand side, as the corrector to give U^{n+1}. Write down these formulas. (You don't need to implement this scheme, but your formulas should make it clear how to do so.)

b) The scheme you constructed in part **(a)** is a one-step, multi-stage Runge-Kutta scheme. Write down its Butcher tableau. (*Hint.* Be careful! You really can write the scheme in form (5.34), where $Y_1 = U^n$, $Y_2 = U^*$, $Y_3 = U^\dagger$.)

c) Following what is described at the beginning of section 7.6.2, apply the scheme in part **(a)** to the test equation $u' = \lambda u$ and write the scheme as $U^{n+1} = R(z)U^n$ for $z = \lambda k$. What is $R(z)$? From the form of $R(z)$, what is the order of the truncation error? (That is, what is p in $\tau = O(k^p)$?) Would you recommend this method?

d) Generate a plot of the stability region of the method; see the advice in section 7.6.2 on how to do this.

Problem P33. For a famously stiff problem, consider the heat PDE

$$u_t = u_{xx} \tag{1}$$

Here $u(t, x)$ is the temperature in a rod of length one ($0 \le x \le 1$) and we set boundary temperatures to zero ($u(t, 0) = 0$ and $u(t, 1) = 0$). For an initial temperature distribution we set one part hotter than the rest:

$$u(0, x) = \begin{cases} 1, & 0.25 < x < 0.5, \\ 0, & \text{otherwise.} \end{cases}$$

Suppose we seek $u(1, x)$, i.e. we set $t_f = 1$.

We apply the *method of lines* (MOL) to (1). That is, we discretize the spatial (x) derivatives using the notation from Chapter 2. Specifically, use $m + 1$ subintervals, let $h = 1/(m + 1)$, and let $x_j = jh$ for $j = 0, 1, 2, \ldots, m + 1$. Now $U_j(t) \approx u(t, x_j)$. By eliminating unknowns $U_0 = 0$ and $U_{m+1} = 0$, and keeping the time derivatives as ordinary derivatives, from (1) we get a linear ODE system of dimension m,

$$U(t)' = AU(t) \tag{2}$$

where $U(t) \in \mathbb{R}^m$ and A is *exactly* the matrix in the textbook's equation (2.10). For a given m, note $U(0)$ is computed from the above formula for $u(0, x)$.

a) Implement both forward and backward Euler on (2). For BE, store A using `sparse` storage and solve the equation using backslash or another linear solver which will automatically detect that the matrix is tridiagonal and solve it efficiently.

b) Now consider the $m = 100$ case. For BE, compute and show the solution using $N = 100$ time steps. For FE, $N = 100$ will generate extraordinary explosion. (Confirm this but don't show it.) Determine the largest-possible absolutely-stable time step k

from the eigenvalues of A and the stability region of FE. Finally, compare the computational costs of the two runs by counting floating-point multiplications.[2] You will conclude that an implicit is indeed effective in this case.

[2]For an $m \times m$ tridiagonal matrix A, Av costs $3m$ multiplications while $A\backslash v$ costs $5m$ multiplications.