University of Saskatchewan
Department of Mathematics and Statistics

Numerical Analysis I
(MATH 314)
Instructor: Dr. Raymond J. Spiteri

ASSIGNMENT 02
Due: 10:00 a.m. Tuesday, October 22, 2013

1. [**20 marks**]

   (a) When deriving the trapezoidal method, we replaced $\dot{\mathbf{y}}(t_{n-1/2})$ in

   $$\frac{\mathbf{y}(t_n) - \mathbf{y}(t_{n-1})}{\Delta t_n} = \dot{\mathbf{y}}(t_{n-1/2}) + \mathcal{O}((\Delta t_n)^2)$$

   by an average

   $$\dot{\mathbf{y}}(t_{n-1/2}) = \frac{1}{2}(\dot{\mathbf{y}}(t_{n-1}) + \dot{\mathbf{y}}(t_n))$$

   and then substituted for $\dot{\mathbf{y}}$ from the ODE

   $$\dot{\mathbf{y}} = \mathbf{f}(t, \mathbf{y}).$$

   If instead we use the ODE first to write $\dot{\mathbf{y}}(t_{n-1/2}) = \mathbf{f}(t_{n-1/2}, \mathbf{y}_{n-1/2})$ and then replace $\mathbf{y}_{n-1/2}$ with an average

   $$\mathbf{y}_{n-1/2} = \frac{1}{2}(\mathbf{y}_{n-1} + \mathbf{y}_n),$$

   we obtain an important method called the *implicit midpoint method*:

   $$\mathbf{y}_n = \mathbf{y}_{n-1} + \Delta t_n \mathbf{f}\left(t_{n-1/2}, \frac{1}{2}(\mathbf{y}_{n-1} + \mathbf{y}_n)\right).$$

   Show that this method is symmetric, second-order, and A-stable. How does it relate to the trapezoidal method for the linear constant-coefficient ODE $\dot{\mathbf{y}} = \mathbf{A}\mathbf{y}$?

   (b) Show that one step of the trapezoidal method can be viewed as half a step of Forward Euler followed by half a step of Backward Euler.

   (c) Show that one step of the implicit midpoint method can be viewed as half a step of Backward Euler followed by half a step of Forward Euler.

2. [**20 marks**] The following classical example from astronomy provides us with strong motivation to olve ODEs with error control.

   Consider a planet with (normalized) mass $\mu = 0.012277471$, a sun with mass $\hat{\mu} = 1 - \mu$, and a moon with negligible mass moving in a two-dimensional plane. The motion of the moon is governed by the equations

$$\begin{aligned}
\ddot{u}_1 &= u_1 + 2\dot{u}_2 - \hat{\mu}\frac{u_1+\mu}{r_1} - \mu\frac{u_1-\hat{\mu}}{r_2},\\
\ddot{u}_2 &= u_2 - 2\dot{u}_1 - \hat{\mu}\frac{u_2}{r_1} - \mu\frac{u_2}{r_2},\\
r_1 &= [(u_1+\mu)^2 + u_2^2]^{3/2},\\
r_2 &= [(u_1-\hat{\mu})^2 + u_2^2]^{3/2}.
\end{aligned}$$

Starting with the initial conditions

$$\begin{aligned}
&u_1(0) = 0.994, \qquad u_2(0) = 0, \qquad \dot{u}_1(0) = 0,\\
&\dot{u}_2(0) = -2.00158510637908252240537862224,
\end{aligned}$$

the orbit is *periodic* with period $T < 17.1$. Note that $r_1 = 0$ at $(u_1, u_2) = (-\mu, 0)$ and $r_2 = 0$ at $(u_1, u_2) = (\hat{\mu}, 0)$, so we need to be careful when the orbit comes close to these points!

Solve this equation using MATLAB's `ode45`. Set the quantities `AbsTol` and `RelTol` equal to 10^{-6}. Report on the number of successful and rejected time steps. (*Hint:* Type `help ode45` at the MATLAB prompt and look at how to set options with `odeset`.)

Now use the classical fourth-order Runge–Kutta method with a *constant* stepsize to integrate this problem on $[0, 17.1]$ using 100, 1000, 10000, 20000 steps. Plot the orbit u_2 vs. u_1 in each case. How many uniform steps are needed before the orbit appears to be *qualitatively* correct? What percentage of adaptive steps relative to constant steps was required to get the same (or better!) accuracy?

3. **[30 marks]**

   (a) Give the Butcher array representation of the following Runge–Kutta method:

$$\begin{aligned}
\mathbf{K}_1 &= \mathbf{f}(t_{n-1}, \mathbf{y}_{n-1})\\
\mathbf{K}_2 &= \mathbf{f}(t_{n-1} + \Delta t_n/2, \mathbf{y}_{n-1} + \Delta t_n \mathbf{K}_1/2)\\
\mathbf{K}_3 &= \mathbf{f}(t_{n-1} + \Delta t_n, \mathbf{y}_{n-1} - \Delta t_n \mathbf{K}_1 + 2\Delta t_n \mathbf{K}_2)\\
\mathbf{y}_n &= \mathbf{y}_{n-1} + \Delta t_n(\mathbf{K}_1 + 4\mathbf{K}_2 + \mathbf{K}_3)/6.
\end{aligned}$$

   By considering the exact error at $t = 1$ in solving the initial-value problem

$$\dot{y} = -2ty^2, \qquad y(0) = 1,$$

   for constant step-sizes $\Delta t_n = \Delta t = 0.2,\ 0.1,\ 0.05$, estimate the order of the method. Note that the exact solution is $y(t) = \frac{1}{t^2+1}$.

(b) According to Butcher's rooted tree theory, we have the following order conditions for a Runge–Kutta method with Butcher tableau $\begin{array}{c|c} \mathbf{c} & \mathbf{A} \\ \hline & \mathbf{b}^T \end{array}$ are

$$\begin{aligned} &\text{Order 1:} \quad \mathbf{b}^T\mathbf{e} = 1. \\ &\text{Order 2:} \quad \mathbf{b}^T\mathbf{c} = \frac{1}{2}. \\ &\text{Order 3:} \quad \mathbf{b}^T\mathbf{c}^2 = \frac{1}{3}, \quad \mathbf{b}^T\mathbf{A}\mathbf{c} = \frac{1}{6}. \end{aligned}$$

where $\mathbf{e} = (1, 1, \ldots, 1)^T$.

Prove that the method from part (a) is third-order accurate. This will involve checking that all of the order conditions up to and including order 3 are satisfied as well as checking that at least one order condition of order 4 is not satisfied.

4. **[30 marks]**

(a) Show that when we apply the implicit midpoint method to the linear, *variable-coefficient* problem

$$\dot{y} = \lambda(t)y,$$

(instead of the test equation), the condition of absolute stability

$$|y_n| \leq |y_{n-1}|$$

still holds. In this case we say the implicit midpoint method is *AN-stable*[1]. Show that the trapezoidal method is *not* AN-stable.

(b) For the autonomous system $\dot{\mathbf{y}} = \mathbf{f}(\mathbf{y})$, show that N constant steps of size Δt of the trapezoidal method is the same as starting with half a step of Forward Euler, continuing with $N-1$ steps of the implicit midpoint method, and finishing with half a step of Backward Euler.

We say therefore that the trapezoidal and implicit midpoint methods are *dynamically equivalent*: for Δt sufficiently small, their behaviours are similar (independent of N) even over very long times $t_f = N\Delta t \gg 1$.

(c) What is the stability function of the following explicit Runge–Kutta method?

$$\begin{array}{c|cccc} 0 & 0 & 0 & 0 & 0 \\ \frac{1}{2} & \frac{1}{2} & 0 & 0 & 0 \\ 1 & \frac{1}{2} & \frac{1}{2} & 0 & 0 \\ \frac{1}{2} & \frac{1}{6} & \frac{1}{6} & \frac{1}{6} & 0 \\ \hline & \frac{1}{6} & \frac{1}{6} & \frac{1}{6} & \frac{1}{2} \end{array}$$

Using MATLAB, plot the method's region of absolute stability.
Hint: Type `help contour`.

[1]The N is usually understood to stand for *nonautonomous*.

5. **[0 marks]** Provide a plan for your project. The plan should include a well-defined topic, goals for the project, and timelines for achieving the goals and writing the interim and final reports. After review of the plan by the instructor, a meeting will be set up to provide feedback on your project and the plan for its completion.