Chapter 8 Exercises

From: *Finite Difference Methods for Ordinary and Partial Differential Equations*
by R. J. LeVeque, SIAM, 2007. `http://www.amath.washington.edu/~rjl/fdmbook`

Exercise 8.1 *(stability region of TR-BDF2)*

Use `makeplotS.m` from Chapter 7 to plot the stability region for the TR-BDF2 method (8.6). Observe that the method is L-stable.

Exercise 8.2 *(Stiff decay process)*

The mfile `decay1.m` uses `ode113` to solve the linear system of ODEs arising from the decay process

$$A \overset{K_1}{\to} B \overset{K_2}{\to} C \tag{Ex8.2a}$$

where $u_1 = [A],\ u_2 = [B]$, and $u_3 = [C]$, using $K_1 = 1,\ K_2 = 2$, and initial data $u_1(0) = 1,\ u_2(0) = 0$, and $u_3(0) = 0$.

(a) Use `decaytest.m` to determine how many function evaluations are used for four different choices of `tol`.

(b) Now consider the decay process

$$A \overset{K_1}{\to} D \overset{K_3}{\to} B \overset{K_2}{\to} C \tag{Ex8.2b}$$

Modify the m-file `decay1.m` to solve this system by adding $u_4 = [D]$ and using the initial data $u_4 = 0$. Test your modified program with a modest value of K_3, e.g., $K_3 = 3$, to make sure it gives reasonable results and produces a plot of all 4 components of u.

(c) Suppose K_3 is much larger than K_1 and K_2 in (Ex8.2b). Then as A is converted to D, it decays almost instantly into C. In this case we would expect that $u_4(t)$ will always be very small (though nonzero for $t > 0$) while $u_j(t)$ for $j = 1,\ 2,\ 3$ will be nearly identical to what would be obtained by solving (Ex8.2a) with the same reaction rates K_1 and K_2. Test this out by using $K_3 = 1000$ and solving (Ex8.2b). (Using your modified m-file with `ode113` and set `tol=1e-6`).

(d) Test `ode113` with $K_3 = 1000$ and the four tolerances used in `decaytest.m`. You should observe two things:

(i) The number of function evaluations requires is much larger than when solving (Ex8.2a), even though the solution is essentially the same,

(ii) The number of function evaluations doesn't change much as the tolerance is reduced.

Explain these two observations.

(e) Plot the computed solution from part (d) with `tol = 1e-2` and `tol = 1e-4` and comment on what you observe.

(f) Test your modified system with three different values of $K_3 = 500,\ 1000$ and 2000. In each case use `tol = 1e-6`. You should observe that the number of function evaluations needed grows linearly with K_3. Explain why you would expect this to be true (rather than being roughly constant, or growing at some other rate such as quadratic in K_3). About how many function evaluations would be required if $K_3 = 10^7$?

(g) Repeat part (f) using `ode15s` in place of `ode113`. Explain why the number of function evaluations is much smaller and now roughly constant for large K_3. Also try $K_3 = 10^7$.

Exercise 8.3 *(Stability region of RKC methods)*

Use the m-file `plotSrkc.m` to plot the stability region for the second-order accurate s-stage Runge-Kutta-Chebyshev methods for $r = 3,\ 6$ with damping parameter $\epsilon = 0.05$ and compare the size of these regions to those shown for the first-order accurate RKC methods in Figures 8.7 and 8.8.

Exercise 8.4 *(Implicit midpoint method)*

Consider the implicit Runge-Kutta method

$$\begin{aligned} U^* &= U^n + \frac{k}{2} f(U^*, t_n + k/2), \\ U^{n+1} &= U^n + k f(U^*, t_n + k/2). \end{aligned} \tag{Ex8.4a}$$

The first step is Backward Euler to determine an approximation to the value at the midpoint in time and the second step is the midpoint method using this value.

(a) Determine the order of accuracy of this method.

(b) Determine the stability region.

(c) Is this method A-stable? Is it L-stable?

Exercise 8.5 *(The θ-method)*

Consider the so-called θ-method for $u'(t) = f(u(t), t)$,

$$U^{n+1} = U^n + k((1-\theta) f(U^n, t_n) + \theta f(U^{n+1}, t_{n+1})), \tag{Ex8.5a}$$

where θ is a fixed parameter. Note that $\theta = 0,\ 1/2,\ 1$ all give familiar methods.

(a) Show that this method is A-stable for $\theta \geq 1/2$.

(b) Plot the stability region $\mathcal{S}$ for $\theta = 0,\ 1/4,\ 1/2,\ 3/4,\ 1$ and comment on how the stability region will look for other values of θ.