

CS 624: Numerical Solution of Differential Equations
Spring 2004
Problem Set 2

Handed out: Mon., Feb. 16.

Due: Wed., Feb. 25 in lecture.

1. Exercise 1.8.1 of the text. Restrict attention to consistent, D-stable explicit LMS formulas. [Hints: (1) First, argue that any D-stable consistent LMS has at least one nonzero β_j . This is established by arguing that $C_0 = 0 \Rightarrow \rho(1) = 0$, then D-stability $\Rightarrow \rho'(1) \neq 0$, then $\rho'(1) \neq 0, C_1 = 0 \Rightarrow$ not all β_j 's are zero. (2) If $p(z) = z^s + a_{s-1}z^{s-1} + \dots + a_0$ is a monic polynomial, then for each i , a_{s-i} is the sum of all possible i -fold products of the roots, multiplied by a sign factor. In other words, if the roots of p are z_1, \dots, z_s , then

$$a_{s-i} = (-1)^i \sum_{\{j_1, \dots, j_i\} \subset \{1, \dots, s\}} z_{j_1} \cdots z_{j_i}.$$

2. In lecture it was argued that if all the real parts of A are negative in the linear constant-coefficient ODE $d\mathbf{u}/dt = A\mathbf{u}$, then $\mathbf{u}(t) \rightarrow \mathbf{0}$ as $t \rightarrow \infty$. Construct a 2×2 linear IVP $d\mathbf{u}/dt = A(t)\mathbf{u}$ (i.e., linear with nonconstant coefficients) such that all eigenvalues of $A(t)$ are negative real numbers bounded away from 0 for all t (i.e., all eigenvalues of $A(t)$ for all t are less than or equal to some constant λ_{\max} , and $\lambda_{\max} < 0$) yet $\mathbf{u} \rightarrow \infty$ as $t \rightarrow \infty$. Verify that your construction has an unbounded solution by computing trajectories with e.g. `ode45` in Matlab. (A proof of unboundedness is not required.)

[Hint: first consider ODEs of the form $\mathbf{u}' = T\mathbf{u}$ where T is 2×2 upper triangular. Note that you can easily specify the eigenvalues of T . Find a T with negative eigenvalues and initial conditions such that the solution \mathbf{u} blows up for awhile, and then dies off. Next consider $\mathbf{u}' = Q(t)TQ(t)^T\mathbf{u}$, where $Q(t)$ is 2×2 orthogonal parametrized as $Q(t) = [\cos \theta(t), \sin \theta(t); -\sin \theta(t), \cos \theta(t)]$. Figure out a way to define $\theta(t)$ so that it moves mass from one entry of \mathbf{u} to the other at exactly the right t -values back and forth to cause unbounded growth.]

3. An IVP can change from being stiff to nonstiff (or vice versa) as the solution evolves. Consider, e.g., the system $u' = -u; v' = -v/u$ with initial conditions $u(0) = v(0) = 1$. Write a paragraph or two in which you analyze whether this system is stiff and why. Then try the Matlab IVP solver `ode23` on it. Integrate it out to $t = 10$ and plot both components using a log-scale for the y-axis (i.e., use the `semilogy` function). This solver `ode23` is not intended for stiff problems and exhibits some pathology on this problem.
4. Consider the reactions taking place in a homogeneous solution described by the chemical formulas $A \leftrightarrow B + C$ and $C \leftrightarrow D$, where A, B, C, D are chemical species. There

are four reactions here (namely $A \rightarrow B + C$, $B + C \rightarrow A$, $C \rightarrow D$ and $D \rightarrow C$); let the four rate constants be m_1, m_2, m_3, m_4 respectively. Let $\alpha, \beta, \gamma, \delta$ be the concentrations of A, B, C, D as functions of t . Then the equations governing this system are

$$\begin{aligned}\alpha' &= -m_1\alpha + m_2\beta\gamma, \\ \beta' &= m_1\alpha - m_2\beta\gamma, \\ \gamma' &= m_1\alpha - m_2\beta\gamma - m_3\gamma + m_4\delta, \\ \delta' &= m_3\gamma - m_4\delta.\end{aligned}$$

Suppose the initial conditions are $a^0 = 1, \beta^0 = \gamma^0 = \delta^0 = 0$. Suppose the rate constants are $m_1 = 1, m_2 = .5, m_3 = 100000, m_4 = .1$. This means that A is slowly converted to B and C which are even more slowly converted back, C is very rapidly converted to D , and D is slowly converted back to C . So at the end of the reaction, one would expect most of the product to be B and D .

Integrate the above equations out to $t = 6$ in Matlab using `ode23` and `ode15s`. Note that `ode15s` is intended for stiff problems like this one. For `ode15s`, experiment with both numerical Jacobians (the default) and user-specified Jacobians. For user-specified Jacobians, you have to write code to provide the Jacobian. The help-files on these functions will explain how they work. Turn in plots of the concentrations of B and D for the various methods.

Hand in listings of all m-files, at least two interesting plots, and a couple of paragraphs describing your experience with these methods.