APPM/MATH 4660 Homework #5
DUE 04/04/08

April 7, 2008

Feel free to work in groups, but your final code and your final writeup must be your own work. Please also hand in a copy of both your code as well as the output.

1. Consider the boundary value problem in Section 11.2, problem 3a:

$$\begin{cases} y'' &= y^3 - yy' \\ 1 \leq & x \leq 2 \\ y(1) &= \frac{1}{2} \\ y(2) &= \frac{1}{3} \end{cases} \tag{1}$$

with actual solution $y(x) = \frac{1}{x+1}$. Use the nonlinear shooting method to solve this boundary value problem, with a stopping tolerance of 10^{-4} for the Newton iterations. Recall that you'll have to solve (1) as an ODE (guessing $y'(1)$), as well as the sensitivity equations:

$$\begin{cases} z'' &= (3y(x)^2 - y'(x))z - y(x)z' \\ 1 \leq & x \leq 2 \\ z(1) &= 0 \\ z'(1) &= 1 \end{cases} \tag{2}$$

While you can solve for $y(x)$ first, and then interpolate for use in solving (2), it's probably easier to solve them simultaneously as one big system.

NOTES:

(a) In my lecture, I denoted the guess for $y'(1)$ as m, because we're used to m being "slope". The book uses t for reasons unclear to me.

(b) As far as solving the IVP, you are free to use whichever approximation scheme you desire. That is, feel free to use the RK method you wrote last semester, **ode45** from Matlab, **NDSolve** from Mathematica, etc.

2. Consider the BVP in Section 11.3, problem #3b (Yes, this is the same equation as in hw #4) with actual solution

$$\begin{aligned}
y(x) &= C_1 x^{-m} + C_2 x^{-p} + \frac{3}{2} + \ln(x) \\
m &= \frac{1}{2}(3-\sqrt{17}) \approx -0.56155 \\
p &= \frac{1}{2}(3+\sqrt{17}) \approx 3.5616 \\
C_1 &= -2 - C_2 \approx -0.95647 \\
C_2 &= \frac{2^{1-m} - \frac{3}{2}}{-2^{-m} + 2^{-p}} \approx -1.0435
\end{aligned}$$

(a) Write code that takes the discretization number N as an argument and then uses the Linear Finite Difference Method (with centered differences) to solve for an approximate solution. NOTE: You'll be solving a tri-diagonal linear system $A^N y^N = b^N$ where the vector y^N is an approximation to the function y at the gridpoints x_i. Similarly to question 1, this problem is about BVP's, not about linear systems, i.e., choose whichever method is easiest for you to implement in solving $A^N y^N = b^N$, such as your Gaussian elimination code from 4650, the "\" operator in Matlab, **LinearSolve** in Mathematica, **gsl_linalg_solve_tridiag** in the GSL libraries, etc.

(b) Solve for a solution using $N = 2, 8, 32, 128$.

(c) Compare to the actual solution using the ∞-norm. What order convergence do you observe?