

# MTH 538 NUMERICAL ANALYSIS II

## HOMEWORK 4

Leng-Feng Lee (#2963-2068, llee3@eng.buffalo.edu)  
Advisor: Dr. Mikhail Khenner (mkhenner@nsm.buffalo.edu)  
10 April 2006.

### PROBLEM 1:

Use nonlinear shooting algorithm with  $h = 0.125$  to approximate the solution to the boundary value problem:

$$y'' = 2y^3, \quad -1 < x < 0, \quad y(-1) = \frac{1}{2}, \quad y(0) = \frac{1}{3} \quad (1)$$

Compare the result with the exact solution:  $y(x) = 1/(x+3)$

### SOLUTION:

The nonlinear equation of the following form:

$$y'' = f(x, y, y'), \quad a \leq x \leq b, \quad y(a) = \alpha, \quad y(b) = \beta \quad (2)$$

Can be solve using the nonlinear shooting method by rewriting into the following two initial value problem:

$$y'' = f(x, y(x, t), y'(x, t)), \quad a \leq x \leq b, \quad y(a, t) = \alpha, \quad y'(a, t) = t \quad (3)$$

and:

$$z'' = \frac{\partial f}{\partial y}(x, y, y')z(x, t) + \frac{\partial f}{\partial y'}(x, y, y')z'(x, t), \quad a \leq x \leq b, \quad z(a, t) = 0, \quad y(a, t) = 1 \quad (4)$$

and  $t_k$  is update by using Newton method:

$$t_k = t_{k-1} - \frac{y(b, t_{k-1}) - \beta}{z(b, t_{k-1})} \quad (5)$$

Hence, Equation (1) can be written as:

$$y'' = 2y^3, \quad -1 \leq x \leq 0, \quad y(-1, t) = 1/2, \quad y'(-1, t) = t \quad (6)$$

and

$$z'' = 6zy^2, \quad -1 \leq x \leq 0, \quad z(-1, t) = 0, \quad z'(-1, t) = 1 \quad (7)$$

To solve the ODE Equation of (6) and (7), need to write in first order system and solve it. Define:

$$q_1 = y; \quad q_2 = y'; \quad q_3 = z; \quad q_4 = z' \quad (8)$$

Hence, we have:

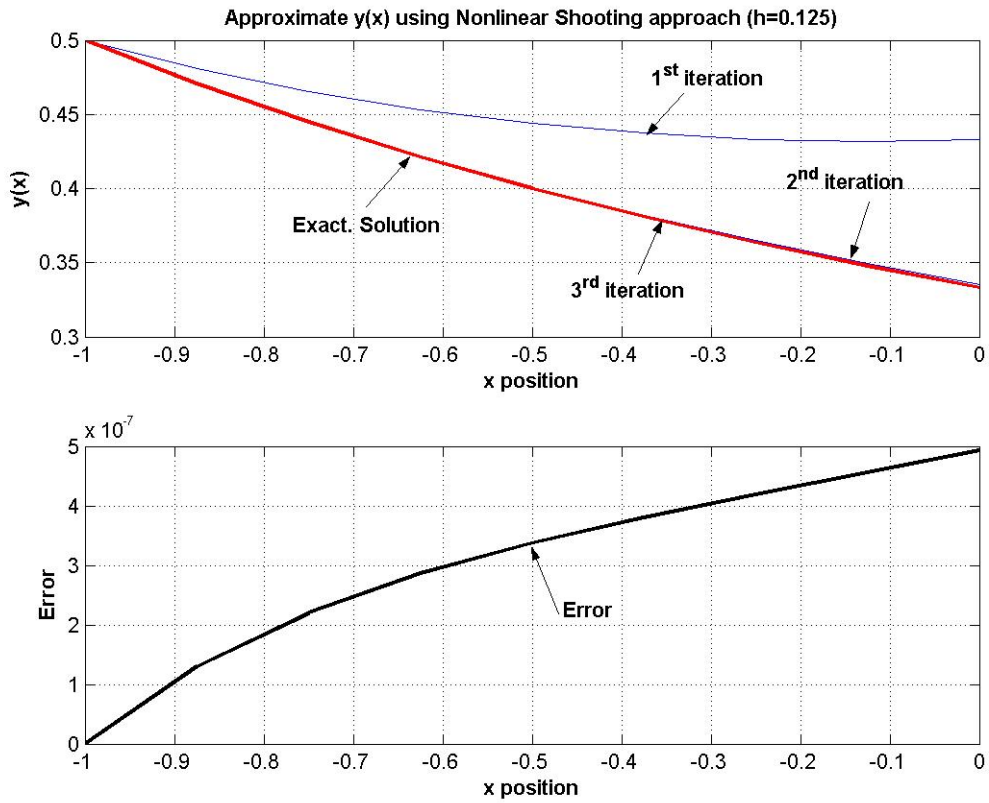
$$\begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \\ \dot{q}_3 \\ \dot{q}_4 \end{bmatrix} = \begin{bmatrix} q_2 \\ 2q_1^3 \\ q_4 \\ 6q_3q_1^2 \end{bmatrix} \quad (9)$$

Equation (9) is solved using fixed time-step ode solver, which uses forth order Runge-Kutta algorithm, with initial condition  $\tilde{q}_0 = [1/2 \quad t_k \quad 0 \quad 1]$  and fixed step of  $h = 0.125$ .

Where  $t_k$  is updated using Equation (5). The result is shown as follow:

iterations	$t_k$	$y(b, t_k)$	$y(b, t_k) - \beta$	< Tolerance?
0	$t_0 = \frac{(\beta - \alpha)}{(b - a)} = -0.1667$	$y_N = 0.4327$	0.994	No
1	$t_1 = t_0 - \frac{y(b, t_0) - \beta}{z(b, t_0)}$ $= -0.1667 - \frac{0.4327 - 1/3}{1.2138}$ $= -0.2485$	$y_N = 0.3351$	0.0017	No
2	$t_2 = t_1 - \frac{y(b, t_1) - \beta}{z(b, t_1)}$ $= -0.2485 - \frac{0.3351 - 1/3}{1.1729}$ $= -0.2500$	$y_N = 0.3333$	$4.9337 \times 10^{-7}$	YES

The solution is shown in Figure 1.



**Figure 1: Solution for problem 1, plot of 3 iterations of shooting method along with the exact solution. The error between the exact solution and the one obtained from nonlinear shooting method also included. The tolerance for termination is  $1 \times 10^{-3}$ .**

The error is in the order of  $10^{-7}$ , obtained in three iterations.

**PROBLEM 2.**

Use nonlinear finite-difference method with  $h = 0.125$  to approximate the solution to the boundary value problem:

$$y'' = 2y^3, \quad -1 < x < 0, \quad y(-1) = \frac{1}{2}, \quad y(0) = \frac{1}{3} \quad (10)$$

Compare the result with the exact solution:  $y(x) = 1/(x+3)$

The nonlinear second-order boundary-value problem of the form:

$$y'' = f(x, y, y'), \quad a \leq x \leq b, \quad y(a) = \alpha, \quad y(b) = \beta \quad (11)$$

can be solved using similar approach as the linear finite difference method (so called Nonlinear finite difference method). However, since the resulting system of equation is not linear, a iterative process was then implemented to determine the solution.

Equation (11), when solve using centered-difference method, a  $N \times N$  nonlinear system can be obtained as follow:

$$\begin{aligned} 2w_1 + w_2 + h^2 f\left(x_1, w_1, \frac{w_2 - \alpha}{2h}\right) - \alpha &= 0, \\ -w_1 + 2w_2 - w_3 + h^2 f\left(x_2, w_2, \frac{w_3 - \alpha}{2h}\right) &= 0, \\ &\vdots \\ -w_{N-2} + 2w_{N-1} - w_N + h^2 f\left(x_{N-1}, w_{N-1}, \frac{w_N - w_{N-2}}{2h}\right) &= 0, \\ -w_{N-1} + 2w_N + h^2 f\left(x_N, w_N, \frac{\beta - w_{N-1}}{2h}\right) - \beta &= 0 \end{aligned} \quad (12)$$

Newton's method for nonlinear system is used to approximate the solution to this system. The solution is iterated and will converge to the solution of the system provided that the initial approximation of the solution is close to the solution and the Jacobian matrix for the system is nonsingular. The Jacobian of the system of Equation (12) is tridiagonal with the entry expressed as:

$$J(w_1, \dots, w_N)_{ij} = \begin{cases} -1 + \frac{h}{2} f_{y'} \left( x_i, w_i, \frac{w_{i+1} - w_{i-1}}{2h} \right), & \text{for } i = j-1 \text{ and } j = 2, \dots, N, \\ 2 + h^2 f_y \left( x_i, w_i, \frac{w_{i+1} - w_{i-1}}{2h} \right), & \text{for } i = j \text{ and } j = 1, \dots, N, \\ -1 - \frac{h}{2} f_{y'} \left( x_i, w_i, \frac{w_{i+1} - w_{i-1}}{2h} \right), & \text{for } i = j+1 \text{ and } j = 1, \dots, N-1, \end{cases} \quad (13)$$

where  $w_0 = \alpha$  and  $w_{N+1} = \beta$ .

Newton method then required that at each iteration, the  $N \times N$  nonlinear system be solved for  $v_1, v_2, \dots, v_N$ , since:

$$w_i^k = w_i^{k-1} + v_i, \quad \text{for each } i = 1, 2, \dots, N \quad (14)$$

Equation (10), compare to Equation (11), we have:

$$f(x, y, y') = 2y^3 \quad (15)$$

The partial derivative of  $f(x, w, w')$  with respect to  $y$  is given by:

$$f_y(x, y, y') = 6y^2 \quad (16)$$

The partial derivative of  $f(x, w, w')$  with respect to  $y'$  is given by:

$$f_{y'}(x, y, y') = 0 \quad (17)$$

This nonlinear finite difference method was then coded in MATLAB, using zeros as the initial guess. Similarly, the solutions were obtained using  $h = 0.125$ , and the error was computed. In addition, the solutions were then compared with those obtained from problem 1.

#### **RESULT:**

The nonlinear finite difference method for solving this problem is coded in MATLAB. The tolerance used in the iteration process is  $1 \times 10^{-3}$ .

The results obtained using the nonlinear finite difference method are plotted in

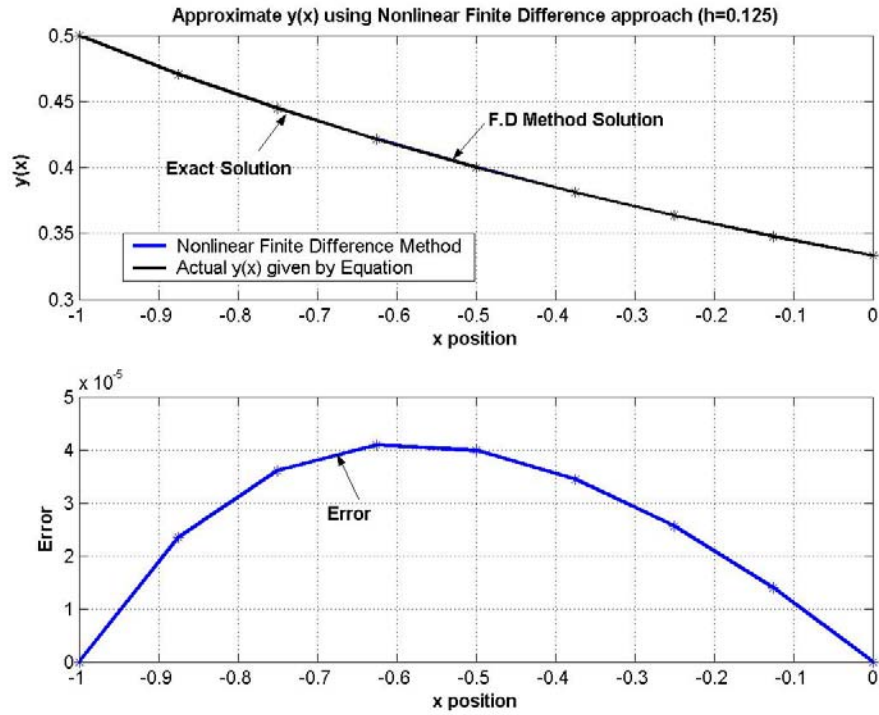


Figure 2: Plot of solution obtained from finite-difference method and the exact solution. The error between the two is also plotted.

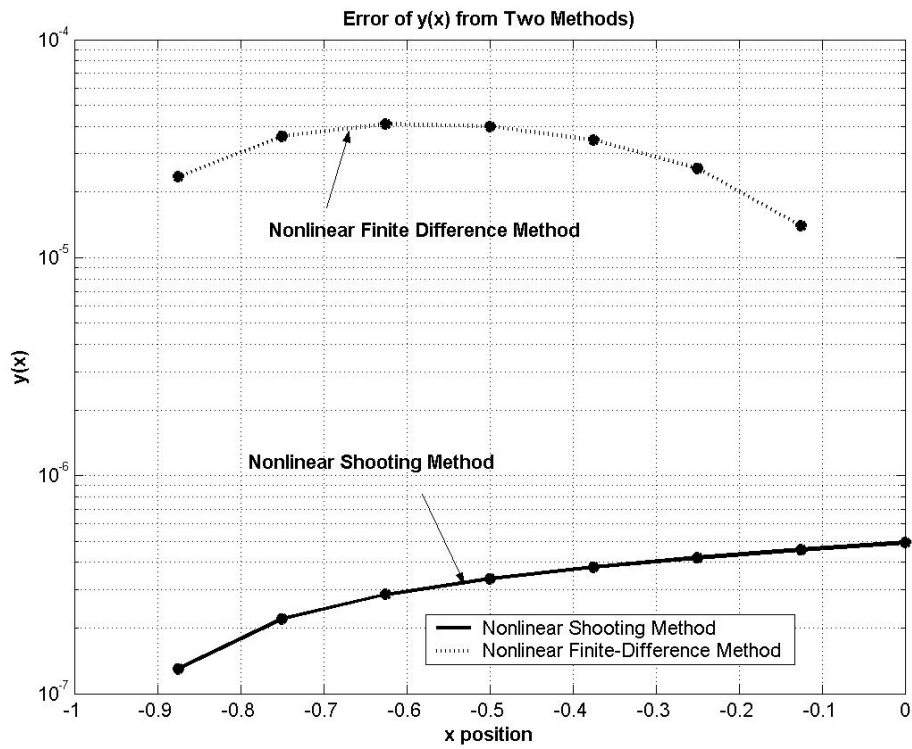


Figure 3: Comparison of Error from Nonlinear Shooting Method and Nonlinear Finite-Difference Method.

## **CONCLUSION:**

By using Nonlinear Shooting Method (with Newton method for updating  $t$ ), we obtained a solution within  $1 \times 10^{-3}$  tolerance in 3 iterations. Similarly, we obtain the solution with the same tolerance value in 3 iterations with Nonlinear Finite-Difference method. However, the maximum error between the exact solution and the approximated solution is in the order of  $10^{-7}$  for nonlinear shooting method, and is  $10^{-5}$  for finite-difference method. Hence, for this problem, nonlinear shooting method gives a more accurate solution using a same termination tolerance value.

The code for this two problems is attached for reference.