Math 317 Assignment 4

Due in class: November 29th, 2016

Instructions: Submit a hard copy of your solution with your name and student number. (**No name = zero grade!**) You must include all relevant program code, electronic output and explanations of your results. Write your own codes and comment them. Late assignment will not be graded and will receive a grade of zero.

1. Consider the boundary value problem,

$$\begin{cases} -u''(x) + 3u'(x) - 2u(x) = f(x) & \text{on } [0,1], \\ u(0) = \alpha, \\ u(1) = \beta. \end{cases} \tag{1}$$

(a) (10 marks) Design a finite difference method to solve the B.V.P. with the local truncation error of $\mathcal{O}(h^2)$. State the entries of your resulting linear system in the form $A_h \vec{u}_h = \vec{f}_h$.

Solution: We are looking for a finite difference method with local truncation error of $\mathcal{O}(h^2)$. Thus we will consider the following approximations for u' and u'':

$$u'(x) = \frac{u(x+h) - u(x-h)}{h} + \mathcal{O}(h^2)$$
$$u''(x) = \frac{u(x+h) - 2u(x) + u(x-h)}{h^2} + \mathcal{O}(h^2)$$

Plugging these expressions in the equation and ignoring the error term leads to

$$-\frac{u(x+h) - 2u(x) + u(x-h)}{h^2} + 3\frac{u(x+h) - u(x-h)}{h} - 2u(x) \approx f(x).$$

Taking $x = x_j := a + jh$ where $h = (b-a)/N$ for $j = 1, \dots, N-1$ in the above equation leads to

$$-\frac{u_{j+1} - 2u_j + u_{j-1}}{h^2} + 3\frac{u_{j+1} - u_{j-1}}{2h} - 2u_j = f(x_j)$$

where we are using u_j to denote the approximation of $u(x_j)$. The above equation can be rewritten as

$$\left(-1 - \frac{3}{2}h\right) u_{j-1} + (2 - 2h^2)u_j + \left(-1 + \frac{3}{2}h\right) u_{j+1} = h^2 f(x_j)$$

for $j = 1, \dots, n$.
This results in a linear system of the form $A_h u_h = f_h$ where

$$A_h = \begin{bmatrix} 2-2h^2 & -1+\frac{3}{2}h & 0 & \cdots & 0 \\ -1-\frac{3}{2}h & 2-2h^2 & -1+\frac{3}{2}h & \ddots & \vdots \\ 0 & \ddots & \ddots & \ddots & 0 \\ \vdots & \ddots & \ddots & \ddots & -1+\frac{3}{2}h \\ 0 & \cdots & 0 & -1-\frac{3}{2}h & 2-2h^2 \end{bmatrix},$$

$$u_h = \begin{bmatrix} u_1 \\ \vdots \\ u_{N-1} \end{bmatrix}, \quad \text{and} \quad f_h = \begin{bmatrix} h^2 f(x_1) + \left(1 + \frac{3}{2}h\right)\alpha \\ h^2 f(x_2) \\ \vdots \\ h^2 f(x_{N-2}) \\ h^2 f(x_{N-1}) + \left(1 - \frac{3}{2}h\right)\right)\beta \end{bmatrix}.$$

Note that A_h is a $(N-1) \times (N-1)$ matrix, and u_h and f_h are column vectors with $N-1$ entries.

(b) (5 marks) Find the exact solution to (1) for $\alpha = 1, \beta = 1 + e$ and $f(x) = 3 - 2x$.

Solution: We first look for a particular solution of

$$-u''(x) + 3u'(x) - 2u(x) = 3 - 2x$$

It's easy to see that $u_p(x) = x$ is such a solution. (Alternatively, given that the right hand side is a 1^{st} order polynomial, we could propose $u_p(x) = Ax + B$ and solve for $A, B \in \mathbb{R}$.)
Now, we look for the general solution to the homogeneous equation, i.e.,

$$-u''(x) + 3u'(x) - 2u(x) = 0.$$

The characteristic polynomial is given by

$$-\lambda^2 + 3\lambda - 2 = -(\lambda - 1)(\lambda - 2).$$

The solutions are then given by

$$u_h(x) = c_1 e^x + c_2 e^{2x}$$

for $c_1, c_2 \in \mathbb{R}$.
Thus the solutions of

$$-u''(x) + 3u'(x) - 2u(x) = 3 - 2x$$

are of the form

$$u(x) = u_p(x) + u_h(x) = x + c_1 e^x + c_2 e^{2x}$$

for $c_1, c_2 \in \mathbb{R}$. Imposing now the boundary conditions $u(0) = 1$ and $u(1) = 1 + e$ we get $c_1 = 1$ and $c_2 = 0$.
The exact solution is then

$$u(x) = x + e^x.$$

(c) (20 marks) For the B.V.P. from part (b), perform a convergence analysis of your method (measure the error in the l_2 norm) and deduce the convergence rate. In order to do so write a program to approximate the solution of (1) using the finite difference method define in (a) for $h = 2^{-1}, \dots 2^{-8}$, plot log(error) versus log(h) and deduce the convergence rate by estimating the slope in the loglog plot.

Solution: To perform the convergence analysis we solve the problem with $h \in \{2^{-1}, \dots, 2^{-8}\}$. For each h, we obtain a vector $u_h = [u_1 \dots u_{N-1}]^T$ and the corresponding error vector e_h given by

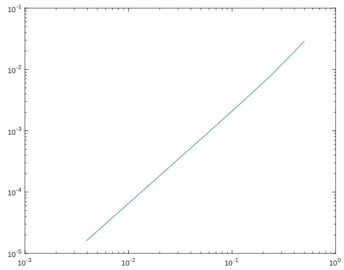
$$e_h = [u_1 - u(x_1) \dots u_{N-1} - u(x_{N-1})]^T$$

where u is the exact solution obtained in b). Measuring the error in the l_2 norm then leads to to computing

$$error = \sqrt{(u_1 - u(x_1))^2 + \dots + (u_{N-1} - u(x_{N-1}))^2}.$$

Now we plot $\log(error)$ versus $\log(h)$ and the slope corresponds to the convergence rate. In Figure **??** we can see the plot obtained. The slope is approximately 3/2 which agrees with the theory since as we saw in class for the Poisson equation we had

$$||u - u_h||_2 = \mathcal{O}(h^{3/2}).$$



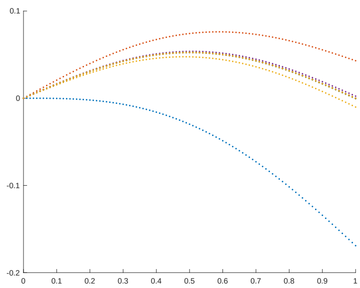
Loglog plot of the error versus h.

2. (35 marks) Consider the prescribed curvature problem which arises in modelling beam deflection:

$$\begin{cases} -\frac{u''(x)}{(1+u'(x)^2)^{\frac{3}{2}}} = 2x(1-x) \quad \text{on } [0,1], \\ u(0) = 0, \\ u(1) = 0. \end{cases}$$

Solve the nonlinear B.V.P. using the shooting method with the Newton's method. Choose initial slope of $s_0 = 0$ and use a stopping tolerance of $\epsilon = 10^{-10}$. Use the forward Euler method to solve the associated I.V.P. with $N = 100$. Plot the successive solutions of different slopes. How many iterations of the Newton method is needed to reach the tolerance?

Solution: The plot with the successive solutions of different slopes can be seen in the figure below. A total of 4 iterations were needed to reach the tolerance.



Plot of the successive solutions of different slopes.

3. Consider the linear system,

$$\begin{aligned} 3x_1 - x_2 \quad\quad &= 2 \\ -x_1 + 3x_2 - x_3 &= 2 \\ -x_2 + 3x_3 &= -1 \end{aligned} \qquad \text{with } \boldsymbol{x}_0 = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}.$$

(a) (5 marks) Find the exact solution by Gaussian elimination.

Solution: Using Gaussian elimination we get

$$\left[\begin{array}{ccc|c} 3 & -1 & 0 & 2 \\ -1 & 3 & -1 & 2 \\ 0 & -1 & 3 & -1 \end{array}\right] \sim \left[\begin{array}{ccc|c} 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{array}\right]$$

and so the exact solution is $(x_1, x_2, x_3) = (1, 1, 0)$.

(b) (5 marks) Compute by hand the first iteration of the Jacobi method and the Gauss-Seidel method.

Solution: We have

$$A = \begin{bmatrix} 3 & -1 & 0 \\ -1 & 3 & -1 \\ 0 & -1 & 3 \end{bmatrix} \quad \text{and} \quad b = \begin{bmatrix} 2 \\ 2 \\ -1 \end{bmatrix}.$$

Hence

$$x_J^{(1)} = D^{-1}\left(b - (L+U)x_J^{(0)}\right) = \begin{bmatrix} \frac{2}{3} \\ 1 \\ -\frac{1}{3} \end{bmatrix} \quad \text{and} \quad x_{GS}^{(1)} = (L+D)^{-1}\left(b - Ux_{GS}^{(0)}\right) = \begin{bmatrix} \frac{2}{3} \\ \frac{8}{9} \\ -\frac{1}{27} \end{bmatrix},$$

where

$$x_J^{(0)} = x_{GS}^{(0)} = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}.$$

(c) (20 marks) Compare the number of iterations required to reach a successive relative error of 10^{-10} in the l_2 norm for the method of Richardson ($\omega = 0.2$), Jacobi, Gauss-Seidel and SOR ($\theta = 1.1$).

Solution: As stopping criteria we used $||x^{(k+1)} - x^{(k)}||/||x^{(k)}|| < 10^{-10}$. The Richardson method took 57 iterations. The Jacobi method took 32 iterations. The Gauss-Seidel method took 16 iterations. The SOR method took 13 iterations.