Numerical Methods and Scientific Computing
Case Study 4

John B. Geddes

October 7, 2013

Overview

Solving linear systems of equations lies at the heart of many problems in scientific computing. In this tutorial you will develop, implement, and analyse algorithms for solving tri-diagonal linear systems. You will then use your linear system solver to solve the equations generated by a finite-difference approximation to a BVP.

Tri-Diagonal Solver

Develop and implement an algorithm in MATLAB to solve a tri-diagonal system of linear equations using Gaussian Elimination. Use one-dimensional vectors to store the diagonals of the matrix and the right-hand side. Choose a linear system of equations at random to test it on, and compute the run-time as a function of N. Write a brief review of your method, your implementation, and your results using code snippets as evidence.

Finite-Difference Method

Develop and implement an algorithm in MATLAB to use second-order central-differencing to solve the following linear boundary-value problem,

$$v'' + 2xv' - x^2 v = x^2, \; v(0) = 1, v(1) = 0, \tag{1}$$

using your tri-diagonal linear system solver developed earlier. Demonstrate that your algorithm is second-order accurate, and then modify your solver to handle the boundary conditions $v(0) + v'(0) = 1, v'(1) + 0.5v(1) = 0$. Write a brief review of your method, your implementation, and your results using code snippets as evidence.