FYS3150 - PROJECT 1 - AUTUMN 2015

Mari Dahl Eggen

10. september 2015

Introduction

Many important differential equations in the Sciences can be written as linear second-order differential equations

$$\frac{d^2y}{dx^2} + k^2(x)y = f(x),$$

where f is normally called the inhomogeneous term and k^2 is a real function.
A classical equation from electromagnetism is Poisson's equation. The electrostatic potential Φ is generated by a localized charge distribution $\rho(\mathbf{r})$. In three dimensions it reads

$$\nabla^2\Phi = -4\pi\rho(\mathbf{r}).$$

With a spherically symmetric Φ and $\rho(\mathbf{r})$ the equations simplifies to a one-dimensional equation in r, namely

$$\frac{1}{r^2}\frac{d}{dr}\left(r^2\frac{d\Phi}{dr}\right) = -4\pi\rho(r),$$

which can be rewritten via a substitution $\Phi(r) = \phi(r)/r$ as

$$\frac{d^2\phi}{dr^2} = -4\pi r\rho(r).$$

The inhomogeneous term f or source term is given by the charge distribution ρ multiplied by r and the constant -4π.
We will rewrite this equation by letting $\phi \rightarrow u$ and $r \rightarrow x$. The general one-dimensional Poisson equation reads then

$$-u''(x) = f(x).$$

In this project we will solve the one-dimensional Poissson equation with Dirichlet boundary conditions by rewriting it as a set of linear equations.
To be more explicit we will solve the equation

$$-u''(x) = f(x), \quad x \in (0,1), \quad u(0) = u(1) = 0.$$

and we define the discretized approximation to u as v_i with grid points $x_i = ih$ in the interval from $x_0 = 0$ to $x_{n+1} = 1$. The step length or spacing is defined as $h = 1/(n+1)$. We have then the boundary conditions $v_0 = v_{n+1} = 0$. We approximate the second derivative of u with

$$-\frac{v_{i+1} + v_{i-1} - 2v_i}{h^2} = f_i \quad \text{for } i = 1, \ldots, n, \tag{1}$$

where $f_i = f(x_i)$.

a)

We will now show that Equation (1) can be written as a linear set of equations of the form

$$\mathbf{A}\mathbf{v} = \mathbf{b}, \tag{2}$$

where $\mathbf{A}$ is the $n \times n$ tridiagonal matrix

$$\mathbf{A} = \begin{pmatrix} 2 & -1 & 0 & \dots & \dots & 0 \\ -1 & 2 & -1 & 0 & \dots & \vdots \\ 0 & -1 & 2 & -1 & 0 & \vdots \\ \vdots & \dots & \dots & \dots & \dots & \vdots \\ 0 & \dots & \dots & -1 & 2 & -1 \\ 0 & \dots & \dots & 0 & -1 & 2 \end{pmatrix}$$

and $b_i = h^2 f_i$. We starts with Equation (2), and shows that it can be rewritten to the expression in Equation (1).

$$\begin{pmatrix} 2 & -1 & 0 & \dots & \dots & 0 \\ -1 & 2 & -1 & 0 & \dots & \vdots \\ 0 & -1 & 2 & -1 & 0 & \vdots \\ \vdots & \dots & \dots & \dots & \dots & \vdots \\ 0 & \dots & \dots & -1 & 2 & -1 \\ 0 & \dots & \dots & 0 & -1 & 2 \end{pmatrix} \begin{pmatrix} v1 \\ \vdots \\ v_{i-1} \\ v_i \\ v_{i+1} \\ \vdots \\ v_n \end{pmatrix} = h^2 \begin{pmatrix} f1 \\ \vdots \\ f_{i-1} \\ f_i \\ f_{i+1} \\ \vdots \\ f_n \end{pmatrix}$$

$$\Downarrow$$

$$2v_1 - v_2 = h^2 f_1$$

$$-v_1 + 2v_2 - v_3 = h^2 f_2$$

$$-v_2 + 2v_3 - v_4 = h^2 f_3$$

$$\vdots$$

$$-v_{i-1} + 2v_i - v_{i+1} = h^2 f_i$$

$$\Downarrow$$

$$-\frac{v_{i+1} + v_{i-1} - 2v_i}{h^2} = f_i$$

In this case the source term is given by $f(x) = 100e^{-10x}$, and the same interval and boundary conditions as above is used. For this $f(x)$ the above differential equation has a closed-form solution given by $u(x) = 1 - (1 - e^{-10})x - e^{-10x}$. To test if this is the right solution we insert $u(x)$ in the Poisson equation $-u''(x) = f(x)$.

$$u'(x) = -(1 - e^{-10})x - e^{-10x}$$

$$\Downarrow$$

$$-u''(x) = 100e^{-10x} = f(x)$$

b)

We can rewrite our matrix $\mathbf{A}$ in terms of one-dimensional vectors a, b, c of length $1 : n$. Our linear equation reads

$$\mathbf{A} = \begin{pmatrix} b_1 & c_1 & 0 & \dots & \dots & \dots \\ a_2 & b_2 & c_2 & \dots & \dots & \dots \\ & a_3 & b_3 & c_3 & \dots & \dots \\ & \dots & \dots & \dots & \dots & \dots \\ & & & a_{n-2} & b_{n-1} & c_{n-1} \\ & & & & a_n & b_n \end{pmatrix} \begin{pmatrix} v_1 \\ v_2 \\ \dots \\ \dots \\ \dots \\ v_n \end{pmatrix} = \begin{pmatrix} \tilde{b}_1 \\ \tilde{b}_2 \\ \dots \\ \dots \\ \dots \\ \tilde{b}_n \end{pmatrix}. \tag{3}$$

A tridiagonal matrix is a special form of banded matrix where all the elements are zero except for those on and immediately above and below the leading diagonal. The above tridiagonal system can be written as

$$a_i v_{i-1} + b_i v_i + c_i v_{i+1} = \tilde{b}_i, \tag{4}$$

for $i = 1, 2, \dots, n$. The algorithm for solving this set of equations is rather simple and requires two steps only, a decomposition and forward substitution and finally a backward substitution.
Your first task is to set up the algorithm for solving this set of linear equations. Find also the precise number of floating point operations needed to solve the above equations. Compare this with standard Gaussian elimination and LU decomposition. Then you should code the above algorithm and solve the problem for matrices of the size 10 × 10, 100 × 100 and 1000 × 1000. That means that you choose $n = 10$, $n = 100$ and $n = 1000$ grid points.
Compare your results (make plots) with the closed-form solution for the different number of grid points in the interval $x \in (0, 1)$. The different number of grid points corresponds to different step lengths h.