FYS4150: Project 1

Ingrid A V Holm

September 8, 2016

a)

We want to solve the one-dimensional Poisson equation with Dirichlet boundary conditions:

$$-u''(x) = f(x),\ \ x \in (0,1),\ \ u(0) = u(1) = 0 \tag{1}$$

We approximate the second derivative using the 3 point formula:

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

For $i = 1, 2, ..., n$, where $f_i = f(x_i)$. We rewrite this equation:

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

Setting in $i = 1, 2, 3$ we start to see a pattern:

$$-v_0 + 2v_1 - v_2 = h^2 f_1$$

Since $v_0 = 0$:

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

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

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

We can write this as a matrix working on a vector:

$$
\begin{pmatrix}
2 & -1 & 0 & ... & ... & 0 \\
-1 & 2 & -1 & 0 & ... & ... \\
0 & -1 & 2 & -1 & 2 & ... \\
 & ... & ... & ... & ... & ... \\
0 & ... & & -1 & 2 & -1 \\
0 & ... & & 0 & -1 & 2
\end{pmatrix}
\begin{pmatrix}
v_1 \\ v_2 \\ ... \\ ... \\ ... \\ v_n
\end{pmatrix}
=
\begin{pmatrix}
\tilde{b}_1 \\ ... \\ ... \\ ... \\ \tilde{b}_n
\end{pmatrix}
$$

$$\mathbf{Av} = \mathbf{b}$$

We assume a source term and closed-form solution:

$$f(x) = 100e^{-10x} \tag{3}$$

$$u(x) = 1 - (1 - e^{-10})x - e^{-10x} \tag{4}$$

Check that this satisfies the Poisson equation:

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

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

b)

We rewrite the set of equations in terms of a tridiagonal matrix:

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

In index notation this becomes:

$$a_i v_{i-1} + b_i v_i + c_1 v_{i+1} = \tilde{b}_i \tag{5}$$

With $a_1 = c_n = 0$. The algorithm for solving this set of equations consists of two steps; a decomposition and forward substitution followed by a backward substitution:

Step 1:

$$\beta = b_1$$
$$\gamma = vector[n]$$
$$v_1 = \frac{\tilde{b}_1}{b_1}$$
$$\text{for } (j = 2; j \leq n; j = j + 1):$$
$$\quad \gamma_j = \frac{c_{j-1}}{\beta}$$
$$\quad \beta = b_j - a_j * \gamma_j$$
$$\quad v_j = \frac{(\tilde{b}_j - a_j \cdot v_{j-1})}{\beta}$$

Step 2:

$$\text{for } (j = n - 1; j \geq 1; j = j - 1):$$
$$\quad v_j = \gamma_{j+1} \cdot v_{j+1}$$