FYS3150/FYS4150
Project 1

Magnus Ulimoen
Krister Stræte Karlsen

September 2, 2015

1 Motivation

Solutions of the Poissons equation are a vital component to many aspects physics.

2 a)

Formulation

Given the following ODE with boundary conditions,

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

and the dicretized form following a symmetric Taylor expansion

$$-\frac{v_{i+1} + v_{i-1} - 2v_i}{h^2} = f_i \quad \text{for} \quad i = 1, \dots, n, \quad v_0 = v_n = 0 \tag{2}$$

we are going to show it can be written as system of linear equations of the form:

$$\mathbf{A}\mathbf{v} = \tilde{\mathbf{b}}, \tag{3}$$

Solution

Multipling the discretized equation (2) by h^2 we get:

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

Filling in for i and choosing $\tilde{b_i} = h^2 f_i$ we obtain the following set of equations:

$$\begin{aligned}
2v_1 - v_2 &= \tilde{b_1} \\
-v_1 + 2v_2 - v_3 &= \tilde{b_2} \\
&\vdots \\
-v_{i-1} + 2v_i - v_{i+1} &= \tilde{b_i} \\
&\vdots \\
-v_{n-1} + 2v_n &= \tilde{b_n}
\end{aligned}$$

Now one can easily see that this system of linear equations can written on the form of (3), where

$$\mathbf{A} = \begin{pmatrix} 2 & -1 & 0 & \dots & \dots & 0 \\ -1 & 2 & -1 & 0 & \dots & \dots \\ 0 & -1 & 2 & -1 & 0 & \dots \\ & \dots & \dots & \dots & \dots & \dots \\ 0 & \dots & & -1 & 2 & -1 \\ 0 & \dots & & 0 & -1 & 2 \end{pmatrix}, \quad \mathbf{v} = \begin{pmatrix} v_1 \\ v_2 \\ \dots \\ \dots \\ \dots \\ v_n \end{pmatrix}, \quad \tilde{\mathbf{b}} = \begin{pmatrix} \tilde{b}_1 \\ \tilde{b}_2 \\ \dots \\ \dots \\ \dots \\ \tilde{b}_n \end{pmatrix}.$$

Algorithm for a tridiagonal system of linear equations

We start by looking at the system of equations:

$$
\begin{aligned}
b_1 v_1 + c_1 v_2 &= \tilde{b_1} \qquad (1*)\\
a_2 v_1 + b_2 v_2 + c_3 v_3 &= \tilde{b_2} \qquad (2*)\\
a_3 v_2 + b_3 v_3 + c_3 v_4 &= \tilde{b_3} \qquad (3*)\\
&\vdots\\
a_n v_{n-1} + b_n v_n &= \tilde{b_n} \qquad (n*)
\end{aligned}
$$

If we solve (1*) for v_1 and insert it into (2*) we obtain the following "modified second equation":

$$(b_1 b_2 - a_2 c_1) v_2 + b_1 c_2 v_3 = b_1 \tilde{b_2} - a_2 \tilde{b_1}$$

Now having successfully removed v_1 from the second equation we can go on and solve it for v_2 and insert it into the third equation obtaining:

$$(b_3(b_1 b_2 - a_2 c_1) - a_3 b_1 c_2) v_3 + c_3(b_1 b_2 - a_2 c_1) v_4 = (b_1 b_2 - a_2 c_1)\tilde{b_3} - a_3 b_1 \tilde{b_2} + a_2 a_3 \tilde{b_1}$$

The two modified equations may be written as

$$v_2 = \frac{b_1 \tilde{b_2} - a_2 \tilde{b_1}}{b_1 b_2 - a_2 c_1} - \frac{b_1 c_2}{b_1 b_2 - a_2 c_1} v_3 = \beta_3 + \gamma_3 v_3$$

$$
\begin{aligned}
v_3 =& \frac{(b_1 b_2 - a_2 c_1)\tilde{b_3} - a_3(b_1 \tilde{b_2} - a_2 \tilde{b_1})}{b_3(b_1 b_2 - a_2 c_1) - a_3 b_1 c_2} - \frac{c_3(b_1 b_2 - a_2 c_1)}{b_3(b_1 b_2 - a_2 c_1) - a_3 b_1 c_2} v_4\\
=& \beta_4 + \gamma_4 v_4 = \frac{\tilde{b_3} - a_3 \beta_3}{a_3 \gamma_3 + b_3} + \frac{-c_3}{a_3 \gamma_3 + b3} v_4
\end{aligned}
$$

This prossess can be repeated up untill the last equation. This is the forward substitution step. From the last equation we compute v_n and get all we need to compute v_{n-1}, then v_{n-2}, and so on. This is the backward substitution part of the algorithm. A shrewd reader might see that the coefficiants, β and γ, take a recursive form

$$\beta_{i+1} = \frac{\tilde{b_i} - a_i \beta_i}{a_i \gamma_i + b_i}, \qquad \gamma_{i+1} = \frac{-c_i}{a_i \gamma_i + b_i},$$

and the equation for v_{i-1} reads:

$$v_{i-1} = \beta_i + \gamma_i v_i \tag{4}$$

It follows from (1*) that $\beta_1 = \gamma_1 = 0$. From combining (n*) and (4) we get

$$v_n = \frac{\tilde{b_n} - a_n\beta_n}{a_n\gamma_n + b_n} = \beta_{n+1} + \gamma_{n+1}v_{n+1}, \qquad v_{n+1} = 0.$$

The goal of the seemingly stupid formulation above is to underline the importance of setting $v_{n+1} = 0$ in order to obtain the right formula for v_n.

Having all the nessecary ingredients the algorithm reads as follows.

Algorithm I

$a_i = c_i = -1, \quad i = 1, 2, 3, .., n$
$b_i = 2, \quad i = 1, 2, 3, .., n$
$\tilde{b_i} = h^2 f_i \quad i = 1, 2, 3, .., n$
$\beta_1 = \gamma_1 = 0$
for $i = 1, 2, .., n-1$
    $\beta_{i+1} = \frac{\tilde{b_i} - a_i\beta_i}{a_i\gamma_i + b_i}, \quad \gamma_{i+1} = \frac{-c_i}{a_i\gamma_i + b_i}$

$v_{n+1} = 0$
for $j = n+1, n, .., 1$
    $v_{i-1} = \beta_i + \gamma_i v_i$

This is often refered to as *Thomas Algorithm*, an algorithm for solving tridiagonal systems of linear equations.