Math 5620 Lecture Notes Day 7.

So far, we have a simple test problem:

$$\begin{cases} u''(x) = f(x) \\ u(0) = \alpha \\ u(1) = \beta \end{cases}$$

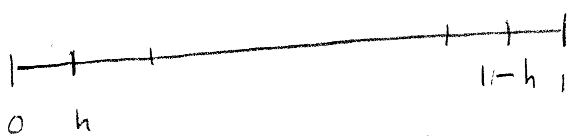
2pt BVP

$$\Downarrow$$

$$u''(x) \approx \frac{u(x-h) - 2u(x) + u(x+h)}{h^2}$$

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

$$\begin{cases} u(0) = \alpha \\ u(1) = 1 \end{cases}$$



To compute an approximation of u at h we need

$$u''(h) \approx \frac{u(0) - 2u(h) + u(2h)}{h^2}$$

- we know $u(0) = \alpha$
- we don't know $u(2h)$.

To compute $u(2h)$

$$u''(2h) \approx \frac{u(h) - 2u(2h) + u(3h)}{h^2}$$

- We don't know any of these values explicitly.

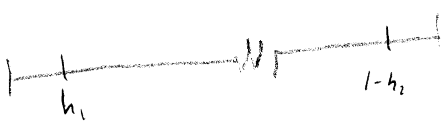
We can approach the same idea from the other boundary $u(1) = \beta$ is given, and we can write

$$u''(1-h) \approx \frac{u(1-2h) - 2u(1-h) + u(1)}{h^2}$$

The analogous problem exists.

- We know $u(1)$
- We don't know $u(1-2h)$ and $u(1-h)$.

One last concern exists



If we are not careful we will not match out in the middle of our domain.

If we use

$$h = \frac{1-0}{m+1} = \frac{b-a}{m+1}$$

we will be able to avoid this minor issue

$$\Downarrow$$

Define points where an approximation is to be computed via

$$h = \frac{1}{m+1} \Rightarrow x_j = j \times h$$

$\Downarrow$

- We know $u(x_0) = u(0) = \alpha$ and $u(x_{m+1}) = u(1) = \beta$
- We don't know $u(x_j) = u(jh) = ?$

$\Downarrow$

Define some notation

$$u(x_j) \approx U_j \quad \Rightarrow \quad u''(x_j) \approx \frac{U_{j-1} - 2U_j + U_{j+1}}{h^2} \qquad j = 1, 2, \ldots, m$$

where

$$\frac{U_{j-1} - 2U_j + U_{j+1}}{h^2} = f(x_j) = f_j$$

$\Downarrow$

The matrix equation

$$AU = F$$

$$U = \begin{bmatrix} U_1 \\ U_2 \\ \vdots \\ U_m \end{bmatrix}$$

$$A = \frac{1}{h^2}\begin{bmatrix} -2 & 1 & & 0 \\ 1 & -2 & \ddots & \\ & \ddots & \ddots & 1 \\ 0 & & 1 & -2 \end{bmatrix} \qquad F = \begin{bmatrix} f(x_1) - \alpha/h^2 \\ f(x_2) \\ \vdots \\ f(x_{m-1}) \\ f(x_m) - \beta/h^2 \end{bmatrix}$$

This is a fairly easy system of equations to solve

Let's do some coding: Thomas Alg.

$$A = \begin{bmatrix} a_{11} & a_{12} & 0 & \cdots & & 0 \\ a_{21} & a_{22} & a_{23} & 0 & \cdots & 0 \\ 0 & a_{32} & a_{33} & a_{34} & 0 \cdots & 0 \\ & & \ddots & \ddots & \ddots & \\ & 0 & & \ddots & \ddots & a_{n-1,n} \\ & & & & a_{m,m-1} & a_{m,n} \end{bmatrix}$$

Note $A^T = A$ in this case.

Full routine $\underline{Ax=b}$

```
for k = 1, m-1
    for i = k+1, m
        factor = a_{i,k} / a_{k,k}
        for j = k+1, m
            a_{i,j} = a_{i,j} - factor * a_{kj}
        end
        F_i = F_i - factor * b_k
    end
end
```

This can be truncated for efficiency.

```
for k = 1, m-1
    factor = a_{k+1,k} / a_{kk}
    a_{k+1,k+1} = a_{k+1,k+1} - factor * a_{k,k+1}
    b_{k+1} = b_{k+1} - factor * b_k
end
```

This cuts the computational effort by a bunch

Back substitution

$$A = \begin{bmatrix} a'_{11} & a'_{12} & 0 & \cdots & & 0 \\ 0 & a'_{22} & a'_{23} & 0 & \cdots & 0 \\ 0 & 0 & a'_{33} & a'_{34} & & \\ & & & & & a'_{m-1,m} \\ & & & & & a'_{m,m} \end{bmatrix} \qquad b = \begin{bmatrix} b'_1 \\ b'_2 \\ \vdots \\ \\ b'_m \end{bmatrix}$$

So

$$u_m = b'_m / a'_{m,m}$$

$$u_{m-1} = (b'_{m-1} - a'_{m-1,m} * u_m) / a'_{m-1,m-1}$$

$$u_{m-2} = (b'_{m-2} - a'_{m-2,m-1} \cdot u_{m-1}) / a'_{m-2,m-2}$$

$$\vdots$$

$$u_k = (b'_k - a'_{k,k+1} \, u_{k+1}) / a'_{kk}$$

$$\vdots$$

$$u_1 = (b'_1 - a'_{1,1+1} \cdot u_2) / a'_{11}$$

The code:

```
u_m = b'_m / a'_mm
for k = m-1,1
    u_k = (b'_k - a'_k,k+1 · u_k+1) / a'_kk
end
```

Storage considerations. Too many zeros! We can get by using arrays / vectors for this example.

Define 3 vectors:

$$\begin{cases} ad - A\text{ diagonal} \\ al1 - \text{first sub diagonal} \\ as1 - \text{first super diagonal} \end{cases}$$

$$\begin{cases} ad[i] = -2/h^2 \\ as(1) = 1/h^2 \\ al(i) = 1/h^2 \end{cases}$$

"$\underline{b[i] = f(x_i)}$"

So, with this we can do the following

```
for k=1,m-1          hmm
    factor = al[k]/ad[k]
    ad[k+1][k+1] = ad[k+1][k+1] - factor * as[k]
    b[k+1] = b[k+1] - factor * b[k]
end
u[m] = b[m] / ad[m]
for k=m-1, 1
    u[k] = ( b[k] - as[k] * u[k+1] ) / ad[k]
end
```