Suggested solution MAT-MEK 4270/9270
8/12 - 2023

1a) The Galerkin method is to find $u_N \in V_N$ such that

$$(u-u_N, v)_{L^2(\Omega)} = 0 \,, \quad \forall v \in V_N$$

1b) Using $v = \psi_i(x)$ and $u_N = \sum_{j=0}^{N} \hat{u}_j \psi_j(x)$, the Galerkin method is

$$\left(u(x) - \sum_{j=0}^{N} \hat{u}_j \psi_j(x), \psi_i(x)\right)_{L^2(\Omega)} = 0 \,,$$

for $i \in \{0,1,\ldots,N\}$. On integral form

$$\sum_{j=0}^{N} \int_\Omega \psi_j(x)\psi_i(x)\,dx\, \hat{u}_j = \int_\Omega u(x)\psi_i(x)\,dx \,.$$

Using now $\psi_i(x) = P_i(\underline{X}(x))$ and a change of variables

$$\sum_{j=0}^{N} \int_{-1}^{1} P_j(\underline{X}) P_i(\underline{X}) \frac{dx}{d\underline{X}} d\underline{X}\, \hat{u}_j = \int_{-1}^{1} u(x(\underline{X}))\, P_i(\underline{X}) \frac{dx}{d\underline{X}} d\underline{X}$$

1b)

$$\sum_{j=0}^{N} (P_j, P_i)_{L^2([-1,1])} \hat{u}_j = (u(x(\underline{x})), P_i)_{L^2([-1,1])}$$

Using $(P_j, P_i)_{L^2([-1,1])} = \frac{2}{2i+1} \delta_{ij}$ and solving for $\hat{u}$

$$\underline{\underline{\hat{u}_i = \frac{2i+1}{2} (u(x(\underline{x})), P_i)_{L^2([-1,1])}}}$$

1c) With $x = \frac{\underline{x}+1}{2}$

$$u(x) = x(1-x) = \frac{\underline{x}+1}{2}\left(1 - \left(\frac{\underline{x}+1}{2}\right)\right)$$

$$= \frac{1}{4}(1-\underline{x}^2)$$

The coefficients are ($N=2$ since $u \in \mathbb{P}_2$)

$$\hat{u}_0 = \frac{1}{2}\left(\frac{1}{4}(1-\underline{x}^2), 1\right)_{L^2([-1,1])}$$

$$= \frac{1}{8}\left[\underline{x} - \frac{1}{3}\underline{x}^3\right]_{-1}^{1} = \frac{1}{8}\left(1 - \frac{1}{3} - \left(-1 + \frac{1}{3}\right)\right)$$

$$= \underline{\frac{1}{6}}$$

1c) $\hat{u}_1 = 0$ since $u(x)$ is even.

$$\hat{u}_2 = \frac{2\cdot 2+1}{2}\left(\frac{1}{4}(1-\underline{x}^2), \frac{1}{2}(3\underline{x}^2-1)\right)_{L^2([-1,1])}$$

$$= \frac{5}{16}\int_{-1}^{1}(1-\underline{x}^2)(3\underline{x}^2-1)\,d\underline{x} = \frac{5}{16}\int_{-1}^{1} -3x^4+4x^2-1\,d\underline{x}$$

$$= \frac{5}{16}\left[-\frac{3}{5}x^5+\frac{4}{3}x^3-x\right]_{-1}^{1} = -\frac{5}{16}\,\frac{8}{15} = -\underline{\frac{1}{6}}$$

$$\underline{\underline{\hat{u} = \left\{\frac{1}{6}, 0, -\frac{1}{6}\right\}}}$$

2a) The Galerkin method is to find $u_N \in V_N$ such that

$$-(u_N', v') + \alpha(u_N', v) + 3(u_N, v) = (f, v), \ \forall v \in V_N$$

The basis functions are chosen such that $\varphi_i(\pm 1) = 0 \ \forall\, i = 0, 1, \ldots, N$.

2b) Using $v = \varphi_i$ and $u_N = \sum_{j=0}^{N} \hat{u}_j \varphi_j$

$$\sum_{j=0}^{N} \left[-(\varphi_j', \varphi_i') + \alpha(\varphi_j', \varphi_i) + \beta(\varphi_j, \varphi_i) \right] \hat{u}_j = (f, \varphi_i)$$

for $i \in 0, 1, \ldots, N$.

Specific bases:

$$\left\{ P_i - P_{i+2} \right\}_{i=0}^{N} \quad \text{or} \quad \left\{ \sin\left((i+1)\pi \frac{(x+1)}{2}\right) \right\}_{i=0}^{N}$$

2c)

$$\tilde{a}^{(e)}_{rs} = \frac{h}{2} \int_{-1}^{1} l_r(\underline{x}) \, l_s(\underline{x}) \, d\underline{x}$$

$$\tilde{S}^{(e)}_{rs} = \frac{2}{h} \int_{-1}^{1} l_r'(\underline{x}) \, l_s'(\underline{x}) \, d\underline{x}$$

$$\tilde{A}^{(e)} = \frac{h}{3} \begin{bmatrix} 1 & \frac{1}{2} \\ \frac{1}{2} & 1 \end{bmatrix}, \quad \tilde{S}^{(e)} = \frac{1}{h} \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix}$$

2d) Use the weak form from 2a

$$-(u_N', v') + 3(u_N, v) = (f, v) - \left[u_N' v\right]_{x=-1}^{x=1}$$

Insert for boundary conditions to get

$$* \quad -(u_N', v') + 3(u_N, v) = (f, v) - \left(b v(1) - a v(-1)\right)$$

Assemble linear problem using $u_N = \sum_{j=0}^{N} \hat{u}_j \varphi_j$ and $v = \varphi_i$. The only nonzero boundary contributions are for $i = 0$ and $i = N$, but all follows naturally. (* is all that is required for full score).

Evaluate $u_N(x)$:

2d) Find the element that x belongs to. (An expression is not required)

$$e = \frac{x+1}{h} \quad \text{(floor division)}$$

Find

$$\underline{\underline{u_N(x) = \sum_{r=0}^{1} \hat{u}_{e+r}\, l_r(\underline{X}(x))}}$$

3a) $\Delta t = \frac{T}{N_t}$, where N_t is the number of time steps.

$\Delta x = \frac{L}{N}$, wher N is the number of intervals. There are $N+1$ mesh points.

$$u_j^n = u(x_j, t_n)$$

3a) To specify $u(x,0) = I(x)$, use interpolation

$$\underline{\underline{u_j^0 = u(x_j, 0) = I(x_j)\,, \quad j = 0, 1, \ldots, N}}$$

To specify $\dfrac{\partial u(x,0)}{\partial t} = 0$, use

$$\frac{\partial u(x,0)}{\partial t} \approx \frac{u^1 - u^{-1}}{2\Delta t} = 0 \Rightarrow u^1 = u^{-1}.$$

Here $u^n = \{u_j^n\}_{j=0}^{N}$.

Use Eq. (13) at $t=0$ to specify u^1:

$$\frac{u^1 - 2u^0 + u^{-1}}{\Delta t^2} = \frac{c^2}{\Delta x^2} D^{(2)} u^0$$

$$\underline{\underline{u^1 = u^0 + \frac{c^2 \Delta t^2}{2\Delta x^2} D^{(2)} u^0}}$$

Here $D^{(2)} = \begin{bmatrix} \ddots & \ddots & \ddots & \\ & 1 & -2 & 1 \\ & \ddots & \ddots & \ddots \end{bmatrix} \in \mathbb{R}^{N+1 \times N+1}$

3a) Suitable boundary conditions must be applied for

$$u_0^1 \text{ and } u_N^1$$

3b) With Dirichlet boundary conditions

$u_0^n = u_N^n = 0$ for all $n = 0, 1, \ldots, N_t$

This may be incorporated into $D^{(2)}$

$$D^{(2)} = \begin{bmatrix} 0 & & & & \\ 1 & -2 & 1 & & \\ & \ddots & \ddots & \ddots & \\ & & 1 & -2 & 1 \\ & & & & 0 \end{bmatrix}$$

The algorithm is:

Set $u_j^0 = I(x_j)$, $j = 0, 1, \ldots, N$

Find $u^1 = u^0 + \left(\frac{c\Delta t}{\Delta x}\right)^2 \frac{1}{2} D^{(2)} u^0$

For $n = 1, 2, \ldots, N_t - 1$ find

$$u^{n+1} = 2u^n - u^{n-1} + \left(\frac{c\Delta t}{\Delta x}\right)^2 D^{(2)} u^n$$

3c) With Neumann boundary conditions

$$\frac{\partial u(0,t)}{\partial x} = \frac{\partial u(L,t)}{\partial x} = 0$$

we have for left boundary

$$\frac{\partial u(0,t_n)}{\partial x} \approx \frac{u_1^n - u_{-1}^n}{2\Delta x} = 0 \Rightarrow u_1^n = u_{-1}^n$$

For right boundary:

$$\frac{\partial u(L,t_n)}{\partial x} \approx \frac{u_{N+1}^n - u_{N-1}^n}{2\Delta x} = 0 \Rightarrow u_{N+1}^n = u_{N-1}^n$$

Use Eq. (12) for $j=0$:

$$u_0^{n+1} = 2u_0^n - u_0^{n-1} + \frac{c^2\Delta t^2}{\Delta x^2}\left(u_1^n - 2u_0^n + u_{-1}^n\right)$$

$$u_0^{n+1} = 2u_0^n - u_0^{n-1} + \frac{c^2\Delta t^2}{\Delta x^2}\left(2u_1^n - 2u_0^n\right)$$

Use Eq. (12) for $j=N$:

$$u_N^{n+1} = 2u_N^n - u_N^{n-1} + \frac{c^2\Delta t^2}{\Delta x^2}\left(-2u_N^n + 2u_{N-1}^n\right)$$

3c) Implementation can modify $D^{(2)}$

$$D^{(2)} = \begin{bmatrix} -2 & 2 & 0 & & & \\ 1 & -2 & 1 & & & \\ & \ddots & \ddots & \ddots & & \\ & & & 1 & -2 & 1 \\ & & & 0 & 2 & -2 \end{bmatrix}$$

The complete solution algorithm is exactly like in 3b, only with a modified $D^{(2)}$.