FEM Homework Report

Yue Wu*

September 26, 2022

1 Introduction

Make a program to solve the two-point boundary value problem with purely Neumann boundary condition:

$$\begin{cases} -u'' = f, & 0 < x < 1 \\ u'(0) = u'(1) = 0 \end{cases} \tag{1}$$

Use an equidistant mesh and a piecewise linear polynomial space V_h as the finite element space. Use $f(x) = -12x^2 + 12x - 2$, $u(x) = x^2(x-1)^2$ to test your program, and compute the following errors:

$$\|u - u_h\|_{L^2[0,1]}, \quad \|u - u_h\|_{H^1[0,1]} \tag{2}$$

2 Method

2.1 The Galerkin approximation

We use the following equidistant mesh:

$$x_j = jh,\ j = 0, 1, \cdots, N, \quad I_j = [x_{j-1}, x_j], \quad h_j = x_j - x_{j-1}, \quad h = \max_j h_j \tag{3}$$

and we use the following finite element space:

$$V_h = \left\{ v \in C[0,1] \middle| v|_{I_j} \in \mathcal{P}^1(I_j), j = 1, \cdots, N \right\} \tag{4}$$

The Galerkin approximation to the problem using approximation space V_h is given by the following:

$$\begin{aligned} &\text{find } u_h \in V_h, \text{ such that:} \\ &\forall v_h \in V_h,\ \left(u_h', v_h'\right) = (f, v_h) + u'(1)v(1) - u'(0)v(0) = (f, v_h) \end{aligned} \tag{5}$$

*School of the Gifted Young, University of Science and Technology of China, Hefei, Anhui, China. E-mail: pilotjohnwu@mail.ustc.edu.cn

Since the boundary condition is purely Neumann, it is obvious that the solution space is an affine space. Using the Fredholm theory for second-order elliptic equations, we need to modify the problem into the following one with an Lagrange multiplier:

$$\begin{aligned} &\text{find } u_h \in V_h \text{ and } \lambda \in \mathbb{R} \text{ such that:} \\ &\forall v_h \in V_h, \begin{cases} (u_h', v_h') + (\lambda, v_h) = (f, v_h) \\ (1, u_h) = c \end{cases} \end{aligned} \tag{6}$$

To approximate integrals of non-polynomial functions **in computing the right-hand-side term $\boldsymbol{f}$**, we apply the 2-point Gauss-Lobatto quadrature rule **in both cases**.

To approximate integrals of non-polynomial functions **in computing the errors**, we apply the 2-point Gauss-Legendre quadrature rule and the 2-point Gauss-Lobatto quadrature rule **respectively**.

Here, we give the 2-point Gauss-Legendre quadrature rule and the 3-point Gauss-Lobatto quadrature rule:

$$\int_{-1}^{1} f(x)\mathrm{d}x \approx f(-\frac{1}{\sqrt{3}}) + f(\frac{1}{\sqrt{3}}) \tag{7}$$

$$\int_{-1}^{1} f(x)\mathrm{d}x \approx f(-1) + f(1) \tag{8}$$

2.2 The global basis and matrix

The global (nodal) basis for the finite element space V_h is given by:

$$\phi_0(x) = \begin{cases} \frac{x_0 - x}{h_1}, & x \in I_1 \\ 0, & \text{otherwise} \end{cases} \tag{9a}$$

$$\phi_j(x) = \begin{cases} \frac{x - x_{j-1}}{h_j}, & x \in I_j \\ \frac{x_j - x}{h_{j+1}}, & x \in I_{j+1} \\ 0, & \text{otherwise} \end{cases}, \quad j = 1, \cdots, N-1 \tag{9b}$$

$$\phi_N(x) = \begin{cases} \frac{x - x_{N-1}}{h_N}, & x \in I_N \\ 0, & \text{otherwise} \end{cases} \tag{9c}$$

Using global basis, each function $v_h \in V_h$ has the following nodal representation:

$$v_h(x) = \sum_{k=1}^{2N-1} v_h(x_{\frac{k}{2}})\phi_k(x) \tag{10}$$

In this manner, the global stiffness matrix is $\boldsymbol{A} = \left(\left(\phi_i', \phi_j'\right)\right)_{(N+1)\times(N+1)}$ with entries $A_{i,j} = \left(\phi_{i-1}', \phi_{j-1}'\right)$, $i, j = 1, \cdots, N+1$.

Lemma 1. *The matrix $\boldsymbol{A}$ is symmetric and positive semi-definite. The null space of $\boldsymbol{A}$ is precisely spanned by $\mathbf{1}$ (the vector with unit entries).*

Proof. Since we are using nodal representations, it is straight-forward to check that $\boldsymbol{A}\mathbf{1} = \mathbf{0}$. We next show that the null space ;cannot be larger.

If $\boldsymbol{u} \notin \text{Span}\{\mathbf{1}\}$, the polynomial $u_h \in V_h$ corresponding to $\boldsymbol{u} \in \mathbb{R}^{N+1}$ is non-constant, thus $\boldsymbol{u}^T\boldsymbol{A}\boldsymbol{u} = a(u_h, u_h) > 0$. □

2.3 The local basis and matrix

The local (nodal) shape functions within a reference element $I = [0, 1]$ are:

$$\psi^0(x) = 1 - x \tag{11a}$$
$$\psi^1(x) = x \tag{11b}$$

The local basis functions for an element I_j are:

$$\psi_j^{(k)} = \psi^{(k)}(\xi_j(x)), \quad k = 0, 1 \quad j = 1, \cdots, N \tag{12}$$

with the affine transformation which maps I_j to $[0, 1]$:

$$\xi_j(x) = \frac{x - x_{j-1}}{h_j} \tag{13}$$

The local stiffness matrix $\boldsymbol{K}_j \in \mathbb{R}^{(N+1)\times(N+1)}$ is:

$$\boldsymbol{K}_j = \text{diag}\left(\mathbf{0}_{(j-1)}, h_j^{-1}\begin{pmatrix} 1 & -1 \\ -1 & 1 \end{pmatrix}, \mathbf{0}_{N-j}\right), \quad j = 1, \cdots, N \tag{14}$$

and the local right-hand-side $\boldsymbol{F}_j \in \mathbb{R}^{2N-1}$ is:

$$\boldsymbol{F}_j = \begin{pmatrix} \mathbf{0}_{j-1} \\ \int_{I_j} f(x)\psi_j^0(x)\mathrm{d}x \\ \int_{I_j} f(x)\psi_j^1(x)\mathrm{d}x \\ \mathbf{0}_{N-j} \end{pmatrix}, \quad j = 1, \cdots, N \tag{15}$$

With these local components, the global matrix and RHS is

$$\boldsymbol{A} = \sum_{j=1}^{N} \boldsymbol{K}_j \tag{16a}$$
$$\boldsymbol{F} = \sum_{j=1}^{N} \boldsymbol{F}_j \tag{16b}$$

and we use point values of u_h as the variable: $\boldsymbol{u} = (u_h(x_0), \cdots, u_h(x_N))^T$.

2.4 Solving the saddle-point system

Since we used a Lagrange method for the purely Neumann problem, we need to solve the saddle-point system:

$$\begin{pmatrix} \boldsymbol{A} & \boldsymbol{B} \\ \boldsymbol{B}^T & 0 \end{pmatrix} \begin{pmatrix} \boldsymbol{u} \\ \lambda \end{pmatrix} = \begin{pmatrix} \boldsymbol{F} \\ c \end{pmatrix} \tag{17}$$

where $\boldsymbol{B} \in \mathbb{R}^{N+1}$ is given by:

$$\boldsymbol{B} = \sum_{j=1}^{N} \boldsymbol{B}_j = \sum_{j=1}^{N} \begin{pmatrix} \mathbf{0}_{j-1} \\ \int_{I_j} \psi_j^0(x)\mathrm{d}x \\ \int_{I_j} \psi_j^1(x)\mathrm{d}x \\ \mathbf{0}_{N-j} \end{pmatrix} \tag{18}$$

Theorem 1. *The saddle-point system above has a unique solution.*

Proof. We only need to show that when $\boldsymbol{F} = \mathbf{0}$ and $c = 0$, the above system only admits zero solution.

When $\boldsymbol{B}^T\boldsymbol{u} = 0$, we have $\boldsymbol{u} = \boldsymbol{B}_\perp \boldsymbol{v}$, where $\boldsymbol{B}_\perp \in \mathbb{R}^{(N+1)\times N}$ is orthogonal to $\boldsymbol{B}$, of rank N. Then, solving $\boldsymbol{A}\boldsymbol{u} + \lambda\boldsymbol{B} = \boldsymbol{F}$ is equivalent to minimizing $\frac{1}{2}\boldsymbol{u}^T\boldsymbol{A}\boldsymbol{u} - \boldsymbol{u}^T\boldsymbol{F}$, which is also to minimize $\frac{1}{2}\boldsymbol{v}^T\boldsymbol{B}^T\boldsymbol{A}\boldsymbol{B}^T\boldsymbol{v} - \boldsymbol{v}^T\boldsymbol{B}^T\boldsymbol{F}$. Using the null space of $\boldsymbol{A}$, we know that $\boldsymbol{B}^T\boldsymbol{A}\boldsymbol{B}$ is symmetric positive definite, thus the system is well-posed. □

3 Results

We use $N = 10, 20, 40, 80$ equidistant elements for computation. By running the scripts in `main.m`, the outputs from the `matlab` program we coded are given as following:

N	L^2 error	order	H^1 error	order
10	7.7742e-04	-	2.7681e-02	-
20	1.8839e-04	2.0450	1.3151e-02	1.0738
40	4.6714e-05	2.0118	6.4853e-03	1.0199
80	1.1654e-05	2.0030	3.2313e-03	1.0051

Table 1: Table of error and order using Gauss-Legendre rule

N	L^2 error	order	H^1 error	order
10	1.5627e-03	-	4.6907e-02	-
20	3.7730e-04	2.0503	2.2641e-02	1.0509
40	9.3460e-05	2.0133	1.1216e-02	1.0134
80	2.3311e-05	2.0034	5.5946e-03	1.0034

Table 2: Table of error and order using Gauss-Lobatto rule

Remark 1. *In the test, since the exact solution* $u(x)$ *satisfies* $\int_0^1 u(x)\mathrm{d}x = \frac{1}{30}$*, we choose* $c = \frac{1}{30}$ *in order to make the error checking procedure easier.*

We should emphasize that the value of c *will not influence the solution, substituting* c *with* $c + \Delta c$ *will only output* $u(x) + \Delta c/m([0,1])$ *instead of* $u(x)$*.*

4 Discussion

From this experiement, we verified the theoretical 2-nd order convergence under the L^2 norm and 1-st order convergence under the H^1 norm for piecewise quadratic conforming FEM numerically. In the algorithm, we used the 2-point Gauss-Lobatto quadrature rule to approximate integration of non-polynomial functions in the right-hand-side. This does not cause the convergence order to degenerate since the quadrature is 2-th order accurate.