A Finite Element Solver for 1D Biharmonic Equations

Candidate Number: 1081225

January 9, 2024

CONTENTS

1 Introduction

1.1 The 1D Biharmonic Problem

The biharmonic equation is a fourth-order differential equation which arises in elasticity theory and plays a vital role in describing the deformation of loaded plates, the stresses in elastic bodies and the solution of the Stokes flow [4, 6]. The equation with homogeneous Dirichlet and Neumann boundary conditions is usually presented as follows

$$\nabla^4 u = f \quad \text{in} \quad \Omega \quad , \quad u = \nabla u \cdot n = 0 \quad \text{on} \quad \Gamma = \partial\Omega \tag{1}$$

where we require Ω to be a bounded domain and $f \in C(\Omega)$. The operator has the form of $\nabla^4 := \frac{\partial^4}{\partial x^4} + 2\frac{\partial^4}{\partial x^2 \partial y^2} + \frac{\partial}{\partial y^4}$ when $u = u(x, y)$ is a function of two independent variables. In this paper, we will mainly focus on the 1D counterpart of this problem, stated as

$$\frac{\mathrm{d}^4}{\mathrm{d}x^4} u = f \quad \text{in} \quad \Omega = (a, b) \quad , \quad u(a) = u(b) = u'(a) = u'(b) = 0 \tag{2}$$

Various methods have been developed to solve the above equations both analytically and numerically, among which the method of finite elements (FEM) has been widely considered in literature due to its advantages in dealing with geometrically complex domains and relaxing regularity requirements imposed on the data as well as the final solution [4].

The primary purpose of this paper is to develop a simple finite element solver for the above 1D biharmonic equation (2). In the first section, we will discuss the variational form as well as the Galerkin approximation of the biharmonic problem. Then, we will turn to the theory of constructing desired function spaces where the solution will be searched. Sections 3 and 4 will be mainly focused on the implementation of the solver. Section 5 will present several numerical results of the solver applied to continuous and discontinuous forcing functions with some simple error analyses. Finally, the conclusion and possible directions for future improvements will be given in Section 6.

Note that in some parts of the subsequent analyses, we will still selectively adopt the 2D formulation of the problem to ensure the universality of the theory. The 1D case is typically used to simplify tedious discussions such as function space constructions and finding physical-to-reference transformations.

1.2 Variational Formulation

The first step of solving a boundary value problem with a finite element method is usually to convert the problem into a variational form (or the weak formulation) and

then check its well-posedness. In 2D cases, the process is started by multiplying the original equation with a test function $v \in \mathbb{V}$

$$\int_\Omega \nabla^4 u \cdot v \, \mathrm{d}\boldsymbol{x} = \int_\Omega f \cdot v \, \mathrm{d}\boldsymbol{x}$$

In order to apply the Lax-Milgram theorem, both u and v should share the same regularity requirements. Thus, it is natural to consider rearranging the derivatives within the integral using integral by parts so that the derivatives can be equally distributed within the new formulation of the problem. To proceed, observe that $\nabla^4 u = \triangle(\triangle u)$ and use the following integration by parts (IBP) formula

$$\int_\Omega v \triangle u \, \mathrm{d}\boldsymbol{x} = -\int_\Omega \nabla v \cdot \nabla u \, \mathrm{d}\boldsymbol{x} + \oint_{\partial\Omega} (v \nabla u) \cdot n \, \mathrm{d}\boldsymbol{s} \tag{IBP}$$

By applying the above formula twice, we have

$$\begin{aligned}\int_\Omega \nabla^4 u \cdot v \, \mathrm{d}\boldsymbol{x} &= \int_\Omega v \triangle(\triangle u) \, \mathrm{d}\boldsymbol{x} = -\int_\Omega \nabla v \cdot \nabla(\triangle u) \, \mathrm{d}\boldsymbol{x} + \oint_{\partial\Omega} v \nabla(\triangle u) \cdot n \, \mathrm{d}\boldsymbol{s} \\ &= \int_\Omega \triangle v \triangle u \, \mathrm{d}\boldsymbol{x} - \oint_{\partial\Omega} \triangle u (\nabla v \cdot n) \, \mathrm{d}\boldsymbol{s} + \oint_{\partial\Omega} v \nabla(\triangle u) \cdot n \, \mathrm{d}\boldsymbol{s}\end{aligned}$$

Since there is no place to encode the boundary conditions weakly in the variational form, we have to impose them strongly in the choice of $\mathbb{V}$. Combining with the fact that the 2nd order (weak) derivatives appear in the variational form for both u and v, $\mathbb{V}$ is therefore determined as

$$\mathbb{V} = H_0^2(\Omega) = \{\ v \in H^2(\Omega)\ :\ v = \nabla v \cdot n = 0 \text{ on } \Gamma = \partial\Omega\ \}$$

The variational form of the 2D biharmonic problem can be stated as : find $u \in H_0^2(\Omega)$ such that

$$\forall\ v \in H_0^2(\Omega)\ :\ \int_\Omega \triangle u \triangle v \, \mathrm{d}\boldsymbol{x} = \int_\Omega f \cdot v \, \mathrm{d}\boldsymbol{x}$$

where the right-hand-side and the left-hand-side of the equation are denoted as $a(u, v)$ and $F(v)$ respectively in the later discussion. Clearly, $a(u, v)$ is a symmetric bilinear form and $F(v)$ is a linear functional defined over $\mathbb{V} = H_0^2(\Omega)$. To apply the Lax-Milgram theorem, it remains to show that $a(u, v)$ is continuous $H_0^2(\Omega)$-coercive and that $F(v)$ is continuous.

1.3 Continuity and Coercivity of the Bilinear Form $a(u, v)$

To show that $a(u, v)$ is continuous $H_0^2(\Omega)$-coercive, we can prove that $a(u, v)$ is, in fact, an inner product over $H_0^2(\Omega)$ [4], which will automatically lead to the desired

result by choosing $C = \alpha = 1$ such that for the induced norm $||\cdot||_{H_0^2(\Omega)} = \sqrt{a(\cdot,\cdot)}$, there is

$$\forall\ u, v \in H_0^2(\Omega)\ :\ |a(u,v)| \leq C\ ||u||_{H_0^2(\Omega)} \cdot ||v||_{H_0^2(\Omega)} \quad \wedge \quad a(v,v) \geq \alpha ||v||^2_{H_0^2(\Omega)}$$

As the symmetry and the linearity of $a(\cdot,\cdot)$ could be directly deduced from the construction of the integral $\int_\Omega \triangle u \triangle v\ \mathrm{d}\boldsymbol{x}$, it remains to verify the positive-definiteness of $a(\cdot,\cdot)$. It is obvious that $a(v,v) = \int_\Omega (\triangle v)^2\ \mathrm{d}\boldsymbol{x} \geq 0$, and when the integral vanishes, $(\triangle v)^2$ must vanish as well. Therefore, $\triangle v = 0$. Note that v_{xx}, v_{yy} are in fact weak derivatives of v, and hence for all test functions $\phi \in C_0^\infty(\Omega)$, we have

$$\int_\Omega \triangle v \cdot \phi\ \mathrm{d}\boldsymbol{x} = \int_\Omega v \cdot \triangle \phi\ \mathrm{d}\boldsymbol{x} \equiv 0$$

To proceed, we need the following weak maximum principle, whose complete version has been thoroughly discussed by Fraenkel in Theorem 2.11 of [5].

Theorem 1.3.1. Weak Maximum Principle
Suppose that Ω is a non-empty open bounded domain belonging to $\mathbb{R}^2$, and $v \in C(\overline{\Omega})$ satisfies $\int_\Omega v \cdot \triangle \phi\ \mathrm{d}\boldsymbol{x} \geq 0$ for all positive test functions $\phi \in C_0^\infty(\Omega)$. Then, v attains its maximum over the boundary $\partial\Omega$.

Since Sobolev's inequality guarantees $H^2(\Omega) \subset C(\Omega)$ (in the sense of equivalence classes) for every Lipschitz domain Ω [4], according to the weak maximum principle, v attains its maximum over the boundary $\partial\Omega$. As $v \in H_0^2(\Omega)$ ensures $v \equiv 0$ on $\partial\Omega$, we therefore conclude that for any $(x,y) \in \Omega$, we have $v(x,y) = 0$. Thus, $a(\cdot,\cdot)$ forms an inner product over the space $H_0^2(\Omega)$, indicating that $a(u,v)$ is indeed a continuous $H_0^2(\Omega)$-coercive symmetric bilinear form.

1.4 Continuity of the Linear Functional $F(v)$

Finally, we show that $F(v)$ is a continuous functional over $H_0^2(\Omega)$ with respect to the norm $||\cdot||_{H_0^2(\Omega)}$. We start by first considering $v^* \in C_0^\infty(\Omega)$

$$\begin{aligned}
|F(v^*)| &= |\langle f, v^* \rangle_{L^2(\Omega)}| \leq ||f||_{L^2(\Omega)} ||v^*||_{L^2(\Omega)} \\
&\leq ||f||_{L^2(\Omega)} \cdot C_\Omega \int_\Omega v_x^{*2} + v_y^{*2}\ \mathrm{d}\boldsymbol{x} \leq ||f||_{L^2(\Omega)} \cdot C_\Omega^2 \int_\Omega v_{xx}^{*2} + v_{yy}^{*2} + 2v_{xy}^{*2}\ \mathrm{d}\boldsymbol{x} \\
&= C_\Omega^2 \cdot ||f||_{L^2(\Omega)} \int_\Omega v_{xx}^{*2} + v_{yy}^{*2} + 2v_{xx}^* v_{yy}^*\ \mathrm{d}\boldsymbol{x} = C_\Omega^2 \cdot ||f||_{L^2(\Omega)} \cdot ||v^*||_{H_0^2(\Omega)}
\end{aligned}$$

Specifically, we applied the **Poincaré inequality** twice for v^* and its first derivatives v_x^*, v_y^* to get the inequalities in line 2. For the first equality in the last line, we have integration by parts

$$\int_\Omega v_{xy}^* v_{xy}^*\ \mathrm{d}\boldsymbol{x} = -\int_\Omega v_{xyy}^* v_x^*\ \mathrm{d}\boldsymbol{x} = -(-\int_\Omega v_{yy}^* v_{xx}^*\ \mathrm{d}\boldsymbol{x}) = \int_\Omega v_{yy}^* v_{xx}^*\ \mathrm{d}\boldsymbol{x}$$

Thus the property holds for all test functions $v^* \in C_0^\infty(\Omega)$. We shall complete our proof for all $v \in H_0^2(\Omega)$ using the following argument : since $H_0^2(\Omega)$ is a completion of $C_0^\infty(\Omega)$ with respect to the metric $d_{H^2(\Omega)}(u,v) = ||u-v||_{H^2(\Omega)}$, which means that $C_0^\infty(\Omega)$ is a dense subset of $H_0^2(\Omega)$, as long as $p(v) = |F(v)|$ and $q(v) = ||v||_{H_0^2(\Omega)}$ are both continuous functions with independent variables from $H_0^2(\Omega)$, in terms of : for all $u \in H_0^2(\Omega)$ there is

$$\forall\ \varepsilon > 0\ :\ \exists\ \delta_u > 0\ \text{ s.t. }\ |p(v) - p(u)| < \varepsilon\ \text{ for all }\ v\ \text{ if }\ d_{H^2(\Omega)}(u,v) < \delta_u$$

over the metric space $(H_0^2(\Omega), d_{H^2(\Omega)})$, we could finish the proof by passing the limits through a sequence of $\{v_n^*\} \subset C_0^\infty(\Omega)$ such that $\lim_n v_n^* = v$ in norm $||\cdot||_{H^2(\Omega)}$. We now start to show that $p(v) = |F(v)|$ and $q(v) = ||v||_{H_0^2(\Omega)}$ are indeed continuous functionals over the desired metric space. For $p(v)$,

$$\begin{aligned}
|p(v) - p(u)| \ &\leq\ |F(v) - F(u)| \ =\ \left|\textstyle\int_\Omega f(v-u)\ \mathrm{d}\boldsymbol{x}\right| \\
&\leq\ ||f(v-u)||_{L^1(\Omega)} \\
\text{Hölder's inequality}\ &\leq\ ||f||_{L^2(\Omega)} \cdot ||v-u||_{L^2(\Omega)} \\
&\leq\ ||f||_{L^2(\Omega)} \cdot ||v-u||_{H^2(\Omega)} \ =\ ||f||_{L^2(\Omega)} \cdot d^2_{H^2(\Omega)}(u,v) \\
&<\ ||f||_{L^2(\Omega)} \cdot \delta_u \ <\ \varepsilon
\end{aligned}$$

Thus we can always take δ_u to be sufficiently small such that the inequality $|p(v) - p(u)| < \varepsilon$ holds for any v satisfying $d_{H^2(\Omega)}(u,v) < \delta_u$, indicating $p(v) = |F(v)|$ is continuous in the metric space $(H_0^2(\Omega), d_{H^2(\Omega)})$. For $q(v) = ||v||_{H_0^2(\Omega)}$, we first consider $r(v) = q^2(v)$, hence

$$\begin{aligned}
|r(v) - r(u)| \ &=\ \left|\,||v||^2_{H_0^2(\Omega)} - ||u||^2_{H_0^2(\Omega)}\right| \ =\ \left|\textstyle\int_\Omega (\triangle v)^2 - (\triangle u)^2\ \mathrm{d}\boldsymbol{x}\right| \\
&\leq\ ||(\triangle v - \triangle u)(\triangle v + \triangle u)||_{L^1(\Omega)} \\
\text{Hölder's inequality}\ &\leq\ ||\triangle v - \triangle u||_{L^2(\Omega)} \cdot ||\triangle v + \triangle u||_{L^2(\Omega)} \\
\text{Triangular inequality}\ &\leq\ ||\triangle v - \triangle u||_{L^2(\Omega)} \cdot \left[\ ||\triangle v - \triangle u||_{L^2(\Omega)} + 2||\triangle u||_{L^2(\Omega)}\ \right] \\
&=\ ||\triangle v - \triangle u||^2_{L^2(\Omega)} + 2||\triangle v - \triangle u||_{L^2(\Omega)} ||\triangle u||_{L^2(\Omega)}
\end{aligned}$$

Using triangular inequality once again for $||\triangle v - \triangle u||_{L^2(\Omega)}$, we have

$$||\triangle v - \triangle u||_{L^2(\Omega)}\ \leq ||(v-u)_{xx}||_{L^2(\Omega)} + ||(v-u)_{yy}||_{L^2(\Omega)}\ \leq\ d_{H^2(\Omega)}(u,v)$$

Thus

$$\begin{aligned}
|r(v) - r(u)| \ &\leq\ d^2_{H^2(\Omega)}(u,v) + 2d_{H^2(\Omega)}(u,v)||\triangle u||_{L^2(\Omega)} \\
&<\ \delta_u^2 + 2\delta_u ||\triangle u||_{L^2(\Omega)} < \varepsilon
\end{aligned}$$

indicating $r(v) = q^2(v)$ is continuous over $H_0^2(\Omega)$ with respect to the norm $||\cdot||_{H^2(\Omega)}$. Since $\sqrt{\cdot}$ is a continuous function over $\mathbb{R}^+ \cup \{0\}$, $q(v) = \sqrt{r(v)}$ is also continuous

over $H_0^2(\Omega)$. Now taking an arbitrary $v \in H_0^2(\Omega)$ and a sequence $\{v_n^*\} \subset C_0^\infty(\Omega)$ such that $\lim_n v_n^* = v$ in norm $||\cdot||_{H^2(\Omega)}$, we have

$$\begin{aligned}|F(v)| = p\left(\lim_n v_n^*\right) = \lim_n p\left(v_n^*\right) &\leq C_\Omega^2 \cdot ||f||_{L^2(\Omega)} \cdot \lim_n q\left(v_n^*\right) \\ &= C_\Omega^2 \cdot ||f||_{L^2(\Omega)} \cdot q\left(\lim_n v_n^*\right) = C_\Omega^2 \cdot ||f||_{L^2(\Omega)} \cdot ||v||_{H_0^2(\Omega)}\end{aligned}$$

Therefore, $|F(v)| \leq C_\Omega^2 \cdot ||f||_{L^2(\Omega)} \cdot ||v||_{H_0^2(\Omega)}$ holds for arbitrary $v \in H_0^2(\Omega)$. By now, the proof for $F(v)$ being a continuous functional has been completed. According to the Lax-Milgram theorem [4], the variational form of the 2D biharmonic equation is now **well-posed**. Another direct proof of the well-posedness of this problem without stating that $a(\cdot,\cdot)$ induces a norm was provided by Ganesan and Tobiska [6].

For the 1D case, the formulation is similar. A direct integration by parts would provide us with

$$\int_a^b u''''v \, \mathrm{d}x = (vu''' - v'u'')|_a^b + \int_a^b u''v'' \, \mathrm{d}x = \int_a^b fv \, \mathrm{d}x$$

By imposing the boundary conditions strongly we get

$$\mathbb{V} = H_0^2([a,b]) = \{\ v \in H^2([a,b])\ :\ v(a) = v(b) = v'(a) = v'(b) = 0\ \}$$

we can obtain the variational form of finding $u \in H_0^2([a,b])$ such that

$$\forall\ v \in H_0^2([a,b])\ :\ \int_a^b u''v'' \, \mathrm{d}x = \int_a^b f \cdot v \, \mathrm{d}\boldsymbol{x}$$

The proof of the well-posedness for this problem is analogous to the one provided in the 2D case but in a much simpler way.

1.5 Galerkin Approximation

Recall now that the original biharmonic problem (1) has been transformed into the following variational form, i.e., find $u \in H_0^2(\Omega)$ such that

$$\forall\ v \in H_0^2(\Omega)\ :\ \int_\Omega \triangle u \triangle v \, \mathrm{d}\boldsymbol{x} = \int_\Omega f \cdot v \, \mathrm{d}\boldsymbol{x} \tag{3}$$

where the problem is defined over an infinite-dimensional function space $H_0^2(\Omega)$. Nevertheless, it is still a challenging task to solve this problem directly due to the constraints imposed by limited computational resources. One way to tackle this problem is to further restrict the solution to a finite-dimensional (closed) subspace of $\mathbb{V} =$

$H_0^2(\Omega)$, denoted as $\mathbb{V}_h \subset \mathbb{V}$, i.e., find $u_h \in \mathbb{V}_h \subset H_0^2(\Omega)$ such that $\dim(\mathbb{V}_h) = n < \infty$ and

$$\forall\ v_h \in \mathbb{V}_h\ :\ \int_\Omega \triangle u_h \triangle v_h\ \mathrm{d}\boldsymbol{x} = \int_\Omega f \cdot v_h\ \mathrm{d}\boldsymbol{x} \tag{4}$$

Equation (4) is therefore called a **Garlerkin approximation** for the problem (3). For an arbitrary basis of $\mathbb{V}_h$, denoted by $\{\phi_j\}_{j=1}^n$, the solution of the approximation can be expressed as $u_h = \sum_{j=1}^n c_j\phi_j$. To find a solution u_h explicitly, rewrite each v_h as a linear combination of the basis $v_h = \sum_{i=1}^n k_i\phi_i$ and insert it into the variational formulation $a(u_h, v_h) = F(v_h)$

$$\begin{aligned} a(u_h, v_h) = F(v_h) \ &\Rightarrow\ \forall k_i\ :\ a\big(\textstyle\sum_{j=1}^n c_j\phi_j\ ,\ \sum_{i=1}^n k_i\phi_i\big) = F\big(\sum_{i=1}^n k_i\phi_i\big) \\ &\Rightarrow\ \forall k_i\ :\ \textstyle\sum_{i=1}^n k_i a\big(\sum_{j=1}^n c_j\phi_j\ ,\ \phi_i\big) = \sum_{i=1}^n k_i F\big(\phi_i\big) \\ &\Rightarrow\ \forall \phi_i\ :\ a\big(\textstyle\sum_{j=1}^n c_j\phi_j\ ,\ \phi_i\big) = \sum_{j=1}^n c_j\ a\big(\phi_j, \phi_i\big) = F\big(\phi_i\big) \end{aligned}$$

which produces a linear system

$$\mathrm{A}\boldsymbol{c} = \mathrm{F}$$

where $\mathrm{A}_{ij} = a(\phi_j, \phi_i)$, $\boldsymbol{c}_i = c_i$, $\mathrm{F}_i = F(\phi_i)$. Specifically, in our case $\mathrm{A}^T = \mathrm{A}$ is a **real symmetric matrix** with a dimension of $n \times n$ due to the symmetry of the bilinear form $a(u, v) = \int_\Omega \triangle u \triangle v\ \mathrm{d}\boldsymbol{x}$. Furthermore, A is also **positive-definite**, as for all $\boldsymbol{c} \in \mathbb{R}^n \setminus \{\boldsymbol{0}\}$

$$\begin{aligned} \boldsymbol{c}^T\mathrm{A}\boldsymbol{c} &= \textstyle\sum_{i=1}^n \sum_{j=1}^n c_i c_j a(\phi_j, \phi_i) \\ &= a\big(\textstyle\sum_{j=1}^n c_j\phi_j, \sum_{i=1}^n c_i\phi_i\big) = a(v, v) \geq ||v||_{H_0^2(\Omega)} > 0 \end{aligned}$$

The well-posedness of the Galerkin approximation (4) is inherited naturally from the well-posedness of the variational formulation as $\mathbb{V}_h$ is a closed subset of $\mathbb{V}$.

2 Construction of $\mathbb{V}_h$ with Finite Elements

2.1 Finite Elements and Finite Element Spaces

This section will discuss how to construct $\mathbb{V}_h \subset H_0^2(\Omega)$ by exploring appropriate finite element spaces X_h. Before starting, we define finite elements as in [4]

Definition 2.1.1. Finite Element *A finite element is a triple $(K, \mathcal{V}_K, \mathcal{L}_K)$ where $K \subset \mathbb{R}^n$ is a nonempty cell with connected interior and piecewise smooth (Lipschits-continuous) boundary; $\mathcal{V}_K$ is a finite-dimensional function space with $\dim(\mathcal{V}_K) = d_K$ defined on K; $\mathcal{L}_K = \{\ell_i^K\}_{i=1}^{d_K}$ is a basis for the dual space $\mathcal{V}_K^*$, which is also referred to as the degrees of freedom of the element.*

The statement of $\mathcal{V}_K^* = \text{span}(\mathcal{L}_K)$ is equivalent to saying that the basis $\mathcal{L}_K$ is **unisolvent** : $v = 0 \iff \ell_i^K(v) = 0$ for all i such that $1 \leq i \leq d_K$, i.e., the degrees of freedom provide all the information needed to determine a specific function over $\mathcal{V}_K$ uniquely [3, 4]. The basis of $\mathcal{V}_K$ $\{\phi_j^K\}_{j=1}^{d_K}$ uniquely defined by setting $\ell_i^K(\phi_j^K) = \delta_{ij}$ is called a **nodal basis**.

The fact that $\{\phi_j^K\}_{j=1}^{d_K}$ is indeed a basis of $\mathcal{V}_K$ can be verified using the following argument : first, the degrees of freedom determine functions uniquely, indicating that $\{\phi_j^K\}_{j=1}^{d_K}$ are d_K different functions. Second, they must be linearly independent as for all coefficients c_j such that

$$\sum_{j=1}^{d_K} c_j \phi_j^K = 0 \;\Rightarrow\; \ell_i^K(\sum_{j=1}^{d_K} c_j \phi_j^K) = \sum_{j=1}^{d_K} c_j \ell_i^K(\phi_j^K) = c_j = 0$$

i.e., c_j vanishes iff the linear combination of $\{\phi_j^K\}_{j=1}^{d_K}$ is equal to 0. Considering $\dim(\mathcal{V}_K) = d_K$, $\{\phi_j^K\}_{j=1}^{d_K}$ hence forms a basis for $\mathcal{V}_K$.

In general cases, the finite element methods will decompose the target (polytopic) domain Ω into a finite set of cells $\mathcal{M} = \{K_i\}$ called a **mesh**, such that $\overline{\Omega} = \cup_i K_i$, and the none-empty intersections $K_i \cap K_j$ for $i \neq j$ is either a common vertex or a common facet. After meshing, a finite element $(K_i, \mathcal{V}_{K_i}, \mathcal{L}_{K_i})$ is assigned to each of the cell.

For the sake of the global assembly, we require an injective **local-to-global** mapping for each of the cell $\iota_K : \{1, \cdots, d_K\} \mapsto \{1, \cdots, N\}$ so that

$$\iota_K(i) = \iota_{K'}(i') \iff \forall v \in X_n \;:\; \ell_i^K[v|_K] = \ell_{i'}^{K'}[v|_{K'}]$$

Denoting $\ell_{\iota_K(i)}(v) = \ell_i^K[v|_K]$, the **global degrees of freedom** follows as

$$\mathcal{L}_{\text{glob}} = \{\ell_i\}_{i=1}^N = \bigcup_{K \in \mathcal{M}} \{\ell_{\iota_K(i)}\}_{i=1}^{d_K}$$

which forces the matching local degrees of freedom to agree. This eventually leads us to the definition of the corresponding **finite element space** generated by the mesh $\mathcal{M}$, given as

$$X_h = \left\{ \; v \in \mathbb{V} : v|_K \in \mathcal{V}_K \;,\; \forall K \in \mathcal{M} \;\text{ and }\; \ell_i^K[v|_K] = \ell_{i'}^{K'}[v|_{K'}] \;,\; \forall \iota_K(i) = \iota_{K'}(i') \; \right\}$$

In our case, $\mathbb{V} = H_0^2(\Omega)$. When $\mathbb{V}_h := X_h \subset \mathbb{V}$, the approximation of the variational problem posed over $\mathbb{V}$ is called **conforming**. In the next subsection, a 1D Hermite finite element will be considered to form a conforming approximation for the problem we will solve in this paper.

2.2 1D Hermite Finite Elements

Before starting, we first break the target interval $\overline{\Omega} = [a, b]$ into the uniform 1-simplex mesh, i.e., a set of intervals

$$\mathcal{M} = \{[a_i, a_{i+1}]\}_{i=0}^{n-1} \quad \text{where} \quad a_i = a + i(b-a)/n\}$$

The main element utilized by the solver over the above mesh will be the following 1D Hermite element, where the triple $(K, \mathcal{V}_K, \mathcal{L}_K)$ is specified as

$$K = [a, b] \quad , \quad \mathcal{V}_K = \mathcal{P}_3(K) = \text{span}\{1, x, x^2, x^3\}$$
$$\mathcal{L}_K = \{\ \ell_1[v] = v(a)\ ,\ \ell_2[v] = v'(a)\ ,\ \ell_3[v] = v(b)\ ,\ \ell_4[v] = v'(b)\ \}$$

for each of the cell $K_i = [a_i, a_{i+1}] \in \mathcal{M}$. The local degrees of freedom consist of evaluation for both the zeroth and the first derivatives at the endpoints. It is quite straightforward to check that such a finite element is **unisolvent** as the degrees of freedom provide four independent algebraic equations which could uniquely determine a cubic polynomial within the interval.

The local-to-global mapping is just assigning all the same (both the order of derivatives and the evaluation points) evaluations together. Hence, the above kind of discretization eventually offers us $N = 2(n+1)$ degrees of freedom globally. Although it has already been proved that the k-simplex Hermite element is generally only $C^0(\Omega)$-conforming [3], the situation could be slightly mitigated in 1D, where the 1-simplex Hermite element is in fact $C^1(\Omega)$-conforming [4]. According to the Theorem 2.1.2 stated by Ciarlet [3], the $C^1(\Omega)$-conforming of the 1D Hermite element is sufficient to provide a $\mathbb{V}_h \subset H_0^2(\Omega)$ as long as the boundary global degrees of freedom are set to be 0.

2.3 Mapping to the Reference Element

It is a common technique for PDE solvers to simplify their coding tasks via constructing a **reference element** $(\hat{K}, \hat{\mathcal{V}}, \hat{\mathcal{L}})$ for the specified finite element $(K, \mathcal{V}_K, \mathcal{L}_K)$. Via such a construction, computations on different elements within the mesh could be easily projected to the reference element through a predetermined set of diffeomorphisms $\{\mathcal{F}_K\}_{K\in\mathcal{M}}$ such that $\mathcal{F}_K(\hat{K}) = K$, and thereby achieving an integration for the general computational steps.

For each $\mathcal{F}_K \in \{\mathcal{F}_K\}_{K\in\mathcal{M}}$, define the **pull-back** and **push-forward** operations as $\mathcal{F}_K^*[\hat{v}] = \hat{v} \circ \mathcal{F}_K^{-1}$ and $\mathcal{F}_*^K[\ell] = \ell \circ \mathcal{F}_K^*$ respectively, where $\hat{v} \in \hat{\mathcal{V}}$ and $\ell \in \mathcal{L}_K$. Write

$$\mathcal{V}_K^* = \mathcal{F}_K^*[\hat{\mathcal{V}}] = \{\ \hat{v} \circ \mathcal{F}_K^{-1}\ :\ \hat{v} \in \hat{\mathcal{V}}\ \}$$
$$\mathcal{L}_K^* = \{\ \ell[v]\ :\ \ell[v] = \hat{\ell}[v \circ \mathcal{F}_K]\ ,\ \hat{\ell} \in \hat{\mathcal{L}}\ \}$$
$$\mathcal{L}_*^K = \mathcal{F}_*^K[\mathcal{L}_K] = \{\ \ell \circ \mathcal{F}_K^*\ :\ \ell \in \mathcal{L}_K\ \}$$

A simple verification indicates that $(K, \mathcal{V}_K^*, \mathcal{L}_K^*)$ forms a finite elemet over K, i.e, $\mathcal{L}_K^*$ is unisolvent : if $\exists\ v = \hat{v} \circ \mathcal{F}_K^{-1} \in \mathcal{V}_K^*$ such that

$$\forall\ \ell \in \mathcal{L}_K^*\ :\ \ell[v] = \hat{\ell}[\hat{v} \circ \mathcal{F}_K^{-1} \circ \mathcal{F}_K] = \hat{\ell}[\hat{v}] = 0\ \Rightarrow\ \hat{v} \equiv 0\ \Rightarrow\ v \equiv 0$$

Definition 2.3.1. Affine-equivalent Elements *If $\mathcal{F}_K$ is an affine map, then the two elements $(\hat{K}, \hat{\mathcal{V}}, \hat{\mathcal{L}})$ and $(K, \mathcal{V}_K, \mathcal{L}_K)$ are affine-equivalent if $\mathcal{F}_K(\hat{K}) = K$, $\mathcal{V}_K^* = \mathcal{V}_K$ and $\mathcal{L}_*^K = \hat{\mathcal{L}}$ (in the sense of equality of finite sets).*

The definition is presented by [7], indicating that the nodal basis of $(\hat{K}, \hat{\mathcal{V}}, \hat{\mathcal{L}})$ and $(K, \mathcal{V}_K, \mathcal{L}_K)$ should be identical [4]. Using the above definition, it can thus be verified that the 1D Hermite finite elements are **NOT** affine-equivalent : suppose an affine map

$$x = \mathcal{F}_K(\hat{x}) = (\beta - \alpha)\hat{x} + \alpha \quad \text{s.t.} \quad \hat{x} = \mathcal{F}_K^{-1}(x) = \frac{x - \alpha}{\beta - \alpha}$$

is given mapping points $\hat{x} \in \hat{K} = [0, 1]$ to $x \in K = [\alpha, \beta]$. If Hermite elements $(\hat{K}, \hat{\mathcal{V}}, \hat{\mathcal{L}})$ and $(K, \mathcal{V}_K, \mathcal{L}_K)$ are affine-equivalent, it shall have $\mathcal{L}_*^K = \hat{\mathcal{L}}$. However, this is easily violated due to the existence of evaluation for derivatives

$$\mathcal{F}_*^K[\mathcal{L}] = \begin{bmatrix} \hat{v} \circ \mathcal{F}_K^{-1}(\alpha) \\ D_x\hat{v} \circ \mathcal{F}_K^{-1}(\alpha) \\ \hat{v} \circ \mathcal{F}_K^{-1}(\beta) \\ D_x\hat{v} \circ \mathcal{F}_K^{-1}(\beta) \end{bmatrix} = \begin{bmatrix} \hat{v}(0) \\ \frac{\hat{v}'(0)}{\beta-\alpha} \\ \hat{v}(1) \\ \frac{\hat{v}'(1)}{\beta-\alpha} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \frac{1}{\beta-\alpha} & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & \frac{1}{\beta-\alpha} \end{bmatrix} \begin{bmatrix} \hat{v}(0) \\ \hat{v}'(0) \\ \hat{v}(1) \\ \hat{v}'(1) \end{bmatrix} \tag{5}$$

where the second equality is obtained using the chain rule. We may need the following result stated in [7] for further discussion.

Theorem 2.3.1. *For two finite elements $(\hat{K}, \hat{\mathcal{V}}, \hat{\mathcal{L}})$ and $(K, \mathcal{V}_K, \mathcal{L}_K)$ s.t. $\mathcal{F}_K(\hat{K}) = K$ and $\mathcal{V}_K^* = \mathcal{V}_K$ for some affine map $\mathcal{F}_K$, if the nodal sets of degrees of freedom $\hat{\mathcal{L}}$ and $\mathcal{L}_K$ (view as vectors) satisfy $\hat{\mathcal{L}} = V\mathcal{F}_*^K[\mathcal{L}]$ for some matrix V, then the relation between nodal basis $\hat{\boldsymbol{\phi}}$ (for $\hat{\mathcal{V}}$) and $\boldsymbol{\phi}^K$ (for $\mathcal{V}_K$) is specified by*

$$\boldsymbol{\phi}^K = V^T\mathcal{F}_K^*[\hat{\boldsymbol{\phi}}]$$

We finish this subsection by calculating the nodal basis for the reference Hermite element $(\hat{K} = [0, 1], \hat{\mathcal{V}}, \hat{\mathcal{L}})$, which will be used later. Letting the i_{th} basis be formed as

$$\hat{\phi}_i = k_3\hat{x}^3 + k_2\hat{x}^2 + k_1\hat{x} + k_0$$

The coefficients can be determined using the relation of $\hat{\ell}_j[\hat{\phi}_i] = \delta_{ij}$ such that

$$\begin{cases} \hat{\ell}_1[\hat{\phi}_i] = \quad \hat{\phi}_i(0) \quad = \delta_{i1} \\ \hat{\ell}_2[\hat{\phi}_i] = D_{\hat{x}}\hat{\phi}_i(0) = \delta_{i2} \\ \hat{\ell}_3[\hat{\phi}_i] = \quad \hat{\phi}_i(1) \quad = \delta_{i3} \\ \hat{\ell}_4[\hat{\phi}_i] = D_{\hat{x}}\hat{\phi}_i(1) = \delta_{i4} \end{cases} \Rightarrow \begin{cases} k_0 = \delta_{i1} \\ k_1 = \delta_{i2} \\ k_3 + k_2 + k_1 + k_0 = \delta_{i3} \\ 3k_3 + 2k_2 + k_1 = \delta_{i4} \end{cases}$$

Solve the above equations for $i = 1, 2, 3, 4$ gives

$$\hat{\phi}_1(\hat{x}) = 2\hat{x}^3 - 3\hat{x}^2 + 1 \qquad \phi_1^K(x) = 2\left(\tfrac{x-\alpha}{\beta-\alpha}\right)^3 - 3\left(\tfrac{x-\alpha}{\beta-\alpha}\right)^2 + 1 \tag{6}$$

$$\hat{\phi}_2(\hat{x}) = \hat{x}^3 - 2\hat{x}^2 + \hat{x} \qquad \phi_2^K(x) = (\beta-\alpha)\left[\left(\tfrac{x-\alpha}{\beta-\alpha}\right)^3 - 2\left(\tfrac{x-\alpha}{\beta-\alpha}\right)^2 + \left(\tfrac{x-\alpha}{\beta-\alpha}\right)\right] \tag{7}$$

$$\hat{\phi}_3(\hat{x}) = -2\hat{x}^3 + 3\hat{x}^2 \qquad \phi_3^K(x) = -2\left(\tfrac{x-\alpha}{\beta-\alpha}\right)^3 + 3\left(\tfrac{x-\alpha}{\beta-\alpha}\right)^2 \tag{8}$$

$$\hat{\phi}_4(\hat{x}) = \hat{x}^3 - \hat{x}^2 \qquad \phi_4^K(x) = (\beta-\alpha)\left[\left(\tfrac{x-\alpha}{\beta-\alpha}\right)^3 - \left(\tfrac{x-\alpha}{\beta-\alpha}\right)^2\right] \tag{9}$$

By observing that $V = \mathrm{diag}(1,\ \beta-\alpha,\ 1,\ \beta-\alpha)$ in equation (5) and applying the above Theorem 2.3.1 $\boldsymbol{\phi}^K = V^T \mathcal{F}_K^*[\hat{\boldsymbol{\phi}}]$, we could thus obtain the explicit formula (6) – (9) for nodal bases of arbitrary 1-simplex Hermite elements defined on $K = [\alpha, \beta]$. Fig. 2.3.1 gives a sketch for the reference element and its nodal basis.

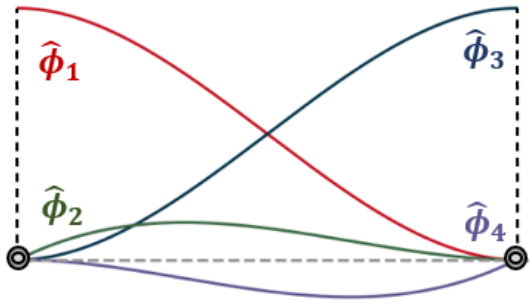

Fig. 2.3.1. Reference Element and Nodal Basis

2.4 2D Conforming Finite Elements

Constructing a reasonable $\mathbb{V}_h$ tends to be a far more complicated task in higher dimensions. When considering 2D biharmonic equations, a conforming approximation may be constructed using triangular elements such as quintic Argyris triangles, Bell elements, and Hsieh-Clough-Tocher triangles, etc [4, 6, 7]. Nevertheless, these elements usually involve a large number of degrees of freedom. For instance, 2D Argyris elements require 21 local degrees of freedom. Worse still, these elements usually do not possess affine equivalence due to a requirement for evaluating normal derivatives [3, 6, 7]. An exception is given by Bogner–Fox–Schmit rectangles, which is both $C^1(\Omega)$-conforming and affine equivalent [3, 6]; yet such rectangular elements usually tend to be less flexible for discretizing domains compared to triangular ones.

3 Local and Global Assembly

3.1 The Assembly Algorithm

Recall that the strategy for discretization at the beginning of Section 2.2 provides us with a mesh $\mathcal{M} = \{[a_i, a_{i+1}]\}_{i=0}^{n-1}$ composed of n intervals, each equipped with an equivalent length of $h = (b-a)/n$ and a 1D Hermite element. At each node a_i, 2 global degrees of freedom $\ell_{i_1}, \ell_{i_2} \in \{\ell_j\}_{j=1}^{N}$ are assigned, evaluating the zeroth and the first derivatives of a given function. Furthermore, the set of local-to-global maps $\{\iota_K\}_{K\in\mathcal{M}}$ helps to form a set of global nodal basis $\{\phi_j\}_{j=1}^{N}$ by "sticking" together two nodal bases $\phi_p^{K_{i-1}}, \phi_q^{K_i}$ from adjacent elements K_{i-1}, K_i with matching global degrees of freedom, i.e., $\iota_{K_{i-1}}(p) = \iota_{K_i}(q) = i_k$, (see Fig. 3.1.1.). We shall use the following notation in the coming analyses to clarify the differences between local and global degrees of freedom (resp., nodal basis)

$$\mathcal{L}_{\text{glob}} = \{\ \ell_j\ \}_{j=1}^{N} \quad , \quad \mathbb{V}_h = \text{span}\{\ \phi_j\ \}_{j=1}^{N}$$

where $N = 2(n+1)$ in our case.

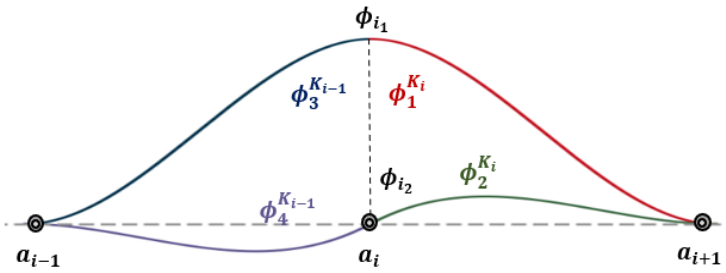

Fig. 3.1.1. Formation of Global Basis

In order to solve the 1D version of the variational problem (4), namely, find $u_h \in \mathbb{V}_h \subset H_0^2(\Omega)$ such that $\dim(\mathbb{V}_h) = 2(n+1) < \infty$ and

$$\forall\ v_h \in \mathbb{V}_h\ :\ a(u_h, v_h) = \int_a^b u_h'' v_h''\ \mathrm{d}x = \int_a^b f \cdot v_h\ \mathrm{d}x = F(v_h) \tag{10}$$

we need to first compute the global stiffness matrix A where $\mathrm{A}_{ij} = a(\phi_j, \phi_i)$ and the global load vector F where $\mathrm{F}_i = F(\phi_i)$. While one may try to compute these quantities directly over the whole domain, a more efficient and neat way is provided by the **assembly algorithm**, where we only locally visit the interval once and form the global stiffness matrix (resp., the global load vector) by adding the local contributions of all relevant basis functions (local stiffness matrix of the interval) together. The implementation is given in Algorithm 1.

Algorithm 1: Assembly Algorithm

Initialize: $\mathrm{A}\ ,\ \mathrm{F} \longleftarrow O_{N\times N}\ ,\ O_{N\times 1}$

for $K \in \mathcal{M}$ **do**

  $\mathrm{A}_K\ ,\ \mathrm{F}_K \longleftarrow O_{d_K\times d_K}\ ,\ O_{d_K\times 1}$

  $\boldsymbol{\iota}_K \longleftarrow [\iota_K(1)\ ,\ \cdots\ ,\ \iota_K(d_K)]$

  for $i = 1 : d_K$ **do**

    $\mathrm{F}_K[i] \longleftarrow F(\phi_i^K)$

    for $j = 1 : d_K$ **do**

      $\mathrm{A}_K[i,j] \longleftarrow a(\phi_j^K, \phi_i^K)$

  $\mathrm{F}[\boldsymbol{\iota}_K] \longleftarrow \mathrm{F}[\boldsymbol{\iota}_K] + \mathrm{F}_K[i]$

  $\mathrm{A}[\boldsymbol{\iota}_K, \boldsymbol{\iota}_K] \longleftarrow \mathrm{A}[\boldsymbol{\iota}_K, \boldsymbol{\iota}_K] + \mathrm{A}_K$

Here N and d_K denote the numbers of global and local degrees of freedom respectively. The above algorithm requires a total number of $|\mathcal{M}|d_K^2$ local evaluations for the bilinear form $a(\cdot,\cdot)$ if all finite elements are identical. We shall be able to reduce this number by approximately half by utilizing the symmetry property of the bilinear form with the modification in Algorithm 2 below. In our problem where $|\mathcal{M}| = 2(n+1), d_K \equiv 4$, this means the number of evaluations will be reduced from $O(32n)$ to $O(16n)$.

Algorithm 2: Modification for Symmetric Bilinear form

for $K \in \mathcal{M}$ **do**

  ...

  for $i = 1 : d_K$ **do**

    ...

    for $j = i : d_K$ **do**

      $\mathrm{A}_K[i,j] \longleftarrow a(\phi_j^K, \phi_i^K)$

  ...

  $\mathrm{A}_K \longleftarrow \mathrm{A}_K + \mathrm{A}_K^T - \mathrm{diag}(\mathrm{A}_K)$

  $\mathrm{A}[\boldsymbol{\iota}_K, \boldsymbol{\iota}_K] \longleftarrow \mathrm{A}[\boldsymbol{\iota}_K, \boldsymbol{\iota}_K] + \mathrm{A}_K$

3.2 Evaluation of Local Stiffness Matrices

Three main functions need to be implemented to apply Algorithm 2. First, one has to evaluate the bilinear form $a(\phi_j^K, \phi_i^K)$ in order to get the local stiffness matrix

A_K. Then, quadrature rules should be used to calculate $F(\phi_i^K)$ inside the local load vector F_K. Finally, local-to-global maps ι_K must be specified so that all the local contributions can be added correctly. In this section, we will introduce how to evaluate the local bilinear form via the reference element constructed in Section 2.3. We will stick to our previous 1D formulation of biharmonic equations with Hermite elements. The reference element is restricted within the domain of $\hat{K} = [0, 1]$ and the element map for $K = [\alpha, \beta]$ is given by an affine transformation

$$x = \mathcal{F}_K(\hat{x}) = (\beta - \alpha)\hat{x} + \alpha \quad \text{s.t.} \quad \hat{x} = \mathcal{F}_K^{-1}(x) = \frac{x - \alpha}{\beta - \alpha}$$

Let $h = \beta - \alpha$ be the length of the target interval where the element is located, and $\hat{\boldsymbol{\phi}}, \boldsymbol{\phi}^K$ be ordered basis functions of the reference and target elements formed as column vectors. Using the transformation theorem 2.3.1, we have

$$\begin{aligned}\mathrm{A}_K &= \int_K \frac{\mathrm{d}^2}{\mathrm{d}x^2}\boldsymbol{\phi}^K(x) \cdot \frac{\mathrm{d}^2}{\mathrm{d}x^2}[\boldsymbol{\phi}^K(x)]^T \ \mathrm{d}x = \int_K \frac{\mathrm{d}^2}{\mathrm{d}x^2}V^T\hat{\boldsymbol{\phi}}(\hat{x}) \cdot \frac{\mathrm{d}^2}{\mathrm{d}x^2}[\hat{\boldsymbol{\phi}}(\hat{x})]^T V \ \mathrm{d}x \\ &= V^T \left[\int_K \frac{\mathrm{d}^2}{\mathrm{d}x^2}\hat{\boldsymbol{\phi}}(\hat{x}) \cdot \frac{\mathrm{d}^2}{\mathrm{d}x^2}[\hat{\boldsymbol{\phi}}(\hat{x})]^T \ \mathrm{d}x\right] V\end{aligned}$$

The last line is valid due to the fact that $V = \mathrm{diag}(1,\ \beta - \alpha,\ 1,\ \beta - \alpha)$ is just a constant matrix independent of the variable x. We should proceed to calculate the derivatives of $\frac{\mathrm{d}^2}{\mathrm{d}x^2}\hat{\boldsymbol{\phi}}(\hat{x})$ with the chain rule

$$\begin{aligned}\frac{\mathrm{d}^2}{\mathrm{d}x^2}\hat{\boldsymbol{\phi}}(\hat{x}) &= \frac{\mathrm{d}}{\mathrm{d}x}\left[\frac{\mathrm{d}}{\mathrm{d}\hat{x}}\hat{\boldsymbol{\phi}}(\hat{x})\frac{\mathrm{d}\hat{x}}{\mathrm{d}x}\right] = \frac{\mathrm{d}}{\mathrm{d}x}\left[\frac{\mathrm{d}}{\mathrm{d}\hat{x}}\hat{\boldsymbol{\phi}}(\hat{x})\right]\frac{\mathrm{d}\hat{x}}{\mathrm{d}x} + \cancel{\frac{\mathrm{d}}{\mathrm{d}\hat{x}}\hat{\boldsymbol{\phi}}(\hat{x})\frac{\mathrm{d}^2\hat{x}}{\mathrm{d}x^2}} \\ &= \frac{\mathrm{d}^2}{\mathrm{d}\hat{x}^2}\hat{\boldsymbol{\phi}}(\hat{x})\left(\frac{\mathrm{d}\hat{x}}{\mathrm{d}x}\right)^2 = \frac{1}{h^2}\hat{\boldsymbol{\phi}}''(\hat{x})\end{aligned}$$

The second term in the second equality vanishes because the affine transformation we choose here is just a linear map. Finally, we apply changing of variables for integrals

$$\mathrm{A}_K = \frac{1}{h^4}\ V^T\left[\int_0^1 \hat{\boldsymbol{\phi}}''(\hat{x}) \cdot [\hat{\boldsymbol{\phi}}''(\hat{x})]^T \ \left(\frac{\mathrm{d}}{\mathrm{d}x}\hat{x}\right)^{-1} \mathrm{d}\hat{x}\right] V \tag{11}$$

$$= \frac{1}{h^3}\ V^T\left[\int_0^1 \hat{\boldsymbol{\phi}}''(\hat{x}) \cdot [\hat{\boldsymbol{\phi}}''(\hat{x})]^T \mathrm{d}\hat{x}\right] V \tag{12}$$

Using formula $(6) - (9)$ for the nodal basis of the reference element, we have $\hat{\boldsymbol{\phi}}''(\hat{x}) = [\ 12\hat{x} - 6\ ,\ 6\hat{x} - 4\ ,\ -12\hat{x} + 6\ ,\ 6\hat{x} - 2\ ]^T$. Plugging this into Equation (12) gives us the final expression for A_K (the calculation of the integrals is implemented by Python, see Appendix A.2.1)

$$\mathrm{A}_K = \frac{1}{h^3}\begin{bmatrix} 12 & 6h & -12 & 6h \\ 6h & 4h^2 & -6h & 2h^2 \\ -12 & -6h & 12 & -6h \\ 6h & 2h^2 & -6h & 4h^2 \end{bmatrix} \tag{13}$$

We can also apply this formula to non-uniform meshes by replacing h with h_i, which denotes the length of the interval where the i^{th} element is located.

3.3 Evaluation of Local Load Vectors

In this section, we will describe how to evaluate the local load vectors. A similar transformation as in the previous section could easily provide us with the following expression for the local load vector

$$\mathrm{F}_K = hV^T \int_0^1 \hat{f}_K(\hat{x})\hat{\boldsymbol{\phi}}(\hat{x})\ \mathrm{d}\hat{x} \tag{14}$$

where $\hat{f}_K(\hat{x}) := f \circ \mathcal{F}_K(\hat{x}) = f[(\beta-\alpha)\hat{x}+\alpha]$. The integral $\int_0^1 \hat{f}_K(\hat{x})\hat{\boldsymbol{\phi}}(\hat{x})\ \mathrm{d}\hat{x}$ for an arbitrary forcing function $f \in L^2([\alpha,\beta])$ will be approximated by quadrature rules. Since our elements are equipped with bases from $\mathcal{P}_3$, it is necessary to choose quadrature rules with an algebraic degree of precision $m \geq 3$, meaning that the rule will be exact for polynomials of degree m or less.

The above requirement can be easily achieved with a minimum of 2-point evaluation with the **Gauss-Legendre** rule, whose degree of precision will be $2n-1$ for n-node evaluations, or, more efficiently, with a 3-point evaluation with the **Gauss-Lobatto** rule, which allows a repetitive use of end-point evaluations and has a degree of precision of $2n-3$ for n fixed nodes [11]. The general formula for these quadrature rules applied to our unit reference interval will be

$$\int_0^1 \hat{f}_K(\hat{x})\hat{\boldsymbol{\phi}}(\hat{x})\ \mathrm{d}\hat{x} \approx \frac{1}{2}\sum_{j=1}^{n} w_j \hat{f}_K(\tfrac{\tilde{x}_j+1}{2})\hat{\boldsymbol{\phi}}(\tfrac{\tilde{x}_j+1}{2}) = \hat{\Phi}\boldsymbol{y}_K \tag{15}$$

where w_j and $\tilde{x}_j$ are fixed parameters which are listed in Appendix A.1.1 [11], and

$$\hat{\Phi}_{ij} = \hat{\boldsymbol{\phi}}_i(\tfrac{\tilde{x}_j+1}{2}) \quad ; \quad (\boldsymbol{y}_K)_j = \tfrac{1}{2} w_j \hat{f}_K(\tfrac{\tilde{x}_j+1}{2}) \tag{16}$$

An approximation for the local load vector is thus given by

$$\mathrm{F}_K \approx hV^T \hat{\Phi}\boldsymbol{y}_K \tag{17}$$

Note that we can always evaluate $\hat{\Phi}$ in advance for a specified quadrature rule. One may also use composite rules to further improve the accuracy of the estimate, which will be given as

$$\int_0^1 \hat{f}_K(\hat{x})\hat{\boldsymbol{\phi}}(\hat{x})\ \mathrm{d}\hat{x} \approx \sum_{I_i \in P([0,1])} \sum_{j=1}^{n} \frac{l_i}{2} w_j \hat{f}_K(x_{ij})\hat{\boldsymbol{\phi}}(x_{ij}) \quad \text{s.t.} \quad x_{ij} = \frac{1}{2} l_i \tilde{x}_j + x_i^* \tag{18}$$

where $P([0,1])$ is a partition for the unit reference interval $[0,1]$; l_i, x_i^* is the length and the middle point for the i^{th} interval $I_i \in P([0,1])$.

4 Further Details for Solver Implementation

4.1 Elements Labeling and Local-to-Global Maps

Labelling elements and constructing local-to-global maps in 2D or higher dimensions can be quite a tedious task [2]. Fortunately, this could be simplified when the dimension is restricted to 1D. As we are coding with Python, where the indexes all start from 0, we shall construct the following local-to-global maps for each $K_i = [a_i, a_{i+1}]$, $i = 0 \cdots (n-1)$

$$\forall\ j = 1 \cdots 4\ :\ \iota_{K_i}(j) = 2i + (j-1) \tag{19}$$

The global degrees of freedom will be labelled as $\{0, 1, 2, 3, \cdots, 2n-1, 2n, 2n+1\}$.

4.2 Boundary Conditions

4.2.1 Homogeneous Boundary Conditions

One final step before solving the assembled system $\mathrm{A}\boldsymbol{c} = \mathrm{F}$ will be imposing the homogeneous boundary conditions to get the modified system $\mathrm{A}_0\boldsymbol{c} = \mathrm{F}_0$. Recall that in our case, the boundary conditions are enforced by setting all boundary degrees of freedom $\ell_0, \ell_1, \ell_{2n}, \ell_{2n+1}$ to 0, which will cause the corresponding global bases $\phi_0, \phi_1, \phi_{2n}, \phi_{2n+1}$ to vanish as they were originally determined as the nodal bases concerning the above degrees of freedom. Hence, given A, F, the modified system should be

$$\begin{bmatrix} O & O & O \\ O & \mathrm{A}_{\text{int}} & O \\ O & O & O \end{bmatrix} \begin{bmatrix} O \\ \boldsymbol{c}_{\text{int}} \\ O \end{bmatrix} = \begin{bmatrix} O \\ \mathrm{F}_{\text{int}} \\ O \end{bmatrix} \tag{20}$$

where $\mathrm{A}_{\text{int}} = \mathrm{A}[2 : 2n-1, 2 : 2n-1]$, $\boldsymbol{c}_{\text{int}} = \boldsymbol{c}[2 : 2n-1]$ and $\mathrm{F}_{\text{int}} = \mathrm{F}[2 : 2n-1]$. We should therefore solve $\mathrm{A}_{\text{int}}\boldsymbol{c}_{\text{int}} = \mathrm{F}_{\text{int}}$ instead of the original system $\mathrm{A}\boldsymbol{c} = \mathrm{F}$.

4.2.2 Non-homogeneous Boundary Conditions

After finding a numerical approximation for the boundary value problem (2) with homogeneous boundary conditions, we can obtain a solution for its counterpart with non-homogeneous conditions (21) analytically via a decomposition technique.

$$\frac{\mathrm{d}^4}{\mathrm{d}x^4}u = f \ \text{ in } \ \Omega = (a, b) \ \ , \ \ u(x)|_{a,b} = BC_1(x) \quad u'|_{a,b} = BC_2(x) \tag{21}$$

To do so, we decompose the solution into two sub-problems defined over $\Omega = (a, b)$ and of the form

$$\frac{\mathrm{d}^4}{\mathrm{d}x^4}u_f = f \quad \text{s.t.} \ \ u(x)|_{a,b} = 0 \qquad\qquad u'|_{a,b} = 0 \tag{22}$$

$$\frac{\mathrm{d}^4}{\mathrm{d}x^4}u_{\text{BC}} = 0 \ \ \text{s.t.} \ \ u(x)|_{a,b} = BC_1(x) \quad u'|_{a,b} = BC_2(x) \tag{23}$$

and the general solution will be formulated as $u = u_f + u_{\text{BC}}$. The first equation involving u_f can be solved with our 1D finite element solver, leaving the sub-problem (23) just as a boundary value problem of a normal homogeneous linear equation with constant coefficients. Using basic ODE theories, the solution is given by $u_{\text{BC}}(x) = C_0 + C_1 x + C_2 x^2 + C_3 x^3$ where the coefficients can be uniquely determined through boundary conditions by solving the following 4 by 4 system

$$\begin{bmatrix} 1 & a & a^2 & a^3 \\ 1 & b & b^2 & b^3 \\ 0 & 1 & 2a & 3a^2 \\ 0 & 1 & 2b & 3b^2 \end{bmatrix} \begin{bmatrix} C_0 \\ C_1 \\ C_2 \\ C_3 \end{bmatrix} = \begin{bmatrix} BC_{1a} \\ BC_{1b} \\ BC_{2a} \\ BC_{2b} \end{bmatrix} \tag{24}$$

The coefficient matrix has a determinant equal to $-(b-a)^4$, and thus is always invertible as long as $b \neq a$. The inverse of this matrix can be calculated directly using the Gaussian Elimination $[\ A \mid I\ ] \sim [\ I \mid A^{-1}\ ]$ (see Appendix A.1.2).

5 Numerical Results and Error Analysis

5.1 Error Estimates

The error analysis introduced in this part will be based on the following theorem in the book written by Braess [1] on page 79 Theorem 6.4, stated as below

Theorem 5.1.1. Interpolation Error for Triangular Elements
Suppose the domain $\Omega \subset \mathbb{R}^n$ is discretized with a shape-regular triangulation, where all triangles are equipped with a C^0-conforming element consisting of piece-wise polynomials of degree $t-1$. Then there exists a constant c s.t.

$$||u - \mathcal{I}_h u||_{H^m(\Omega)} \leq c h^{t-m} |u|_{H^t(\Omega)} \quad \text{for} \quad u \in H^t(\Omega) \quad \text{and} \quad 0 \leq m \leq t$$

where $\mathcal{I}_h$ denotes the global interpolation operator and h is the mesh size.

In our case, all elements are defined over an interval; thus the requirement for shape regularity must be met as

$$\sup_h \max_{K \in \mathcal{M}_h} \frac{h_K}{\rho_K} \equiv 1$$

where h is the mesh size, K is a cell in mesh $\mathcal{M}_h$ and (ρ_K) h_K are (incircle) diameter of the cell K. Since 1D Hermite elements are also C^0, in fact, C^1-conforming, the above theorem is ready for use with $t = 4$. Combined with the **quasi-optimality property** [4], which states that for coercive problems, the Galerkin approximation over function space $\mathbb{V}_h \subset \mathbb{V}$ has the following inequality

$$||u - u_h||_{\mathbb{V}} \leq \frac{C}{\alpha} \inf_{v_h \in \mathbb{V}_h} ||u - v_h||_{\mathbb{V}} \leq \frac{C}{\alpha} ||u - \mathcal{I}_h u||_{\mathbb{V}}$$

for some norm $||\cdot||_{\mathbb{V}}$ equipped by $\mathbb{V}$ where C, α are continuity and coercivity constant respectively. According to our previous formulation for the biharmonic problems (3) and (4), we have $C = \alpha = 1$ and $\mathbb{V} = H_0^2(\Omega) \subset H^2(\Omega)$, indicating for $u \in H^4(\Omega)$, it is expected to have the following error bounds

$$||u - u_h||_{L^2(\Omega)} \leq c_0 h^4 |u|_{H^4(\Omega)} \sim O(h^4) \tag{25}$$

$$||u - u_h||_{H^1(\Omega)} \leq c_1 h^3 |u|_{H^4(\Omega)} \sim O(h^3) \tag{26}$$

$$||u - u_h||_{H^2(\Omega)} \leq c_2 h^2 |u|_{H^4(\Omega)} \sim O(h^2) \tag{27}$$

As it is enough to ask for $u, u', u'', u''' \in H^1(\Omega)$ to ensure $u \in H^4(\Omega)$, the above error bounds are valid for all piece-wise smooth u_s such that $u, u', u'', u''' \in C(\Omega)$ are continuous.

5.2 Continuous Forcing Functions

Since we are studying just 1D biharmonic equations, known solutions can be easily constructed via analytical techniques such as variation of parameters, etc. There is also another way to construct test solutions, which is simply to find an existing function and take its 4^{th} derivative. There is no need to deliberately construct examples with homogeneous boundary conditions, as the solver will automatically fit these conditions using the technique mentioned in Section 4.2.2. The examples we used are listed in Table 5.2.1

Index	Forcing Function	True Solution	Target Interval
i	f	u_{true}	$(\ a\ ,\ b\ )$
1	$\frac{1}{\sqrt{2\pi}}(x^4 - 6x^2 + 3)e^{-\frac{x^2}{2}}$	$\frac{1}{\sqrt{2\pi}}e^{-\frac{x^2}{2}}$	$(\ -\pi\ ,\ \pi\ )$
2	$\cos x \sin^2 x$	$\frac{7}{27}\sin^2 x \cos x + \frac{20}{81}\cos^3 x$	$(\ -\pi\ ,\ \pi\ )$
3	$e^x \sin x$	$-\frac{1}{4}e^x \sin x$	$(\ -\pi\ ,\ \pi\ )$
4	$4(4x^4 - 3)\cos x^2 + 48x^2 \sin x^2$	$\cos x^2$	$(-\sqrt{3\pi}\ ,\ \sqrt{3\pi})$

Table 5.2.1. Known Solutions for Continuous Forcing Functions

The above solutions u are all in C^∞ and their errors can be estimated with inequalities $(25) - (26)$. The numerical results have been summarized in Fig. 5.2.1 (see Appendix A.3.1 for a clear version with extra description). The integrals in the load vectors were calculated using the 4-point Gauss-Legendre rule with a degree of precision equal to 7.

The results are mostly in line with the prediction: the error converges at $O(h^4)$ when applying the L^2-norm, while is 1 order slower when turning to the H^1-norm and gets even worse for the H^2-norm with $O(h^2)$.

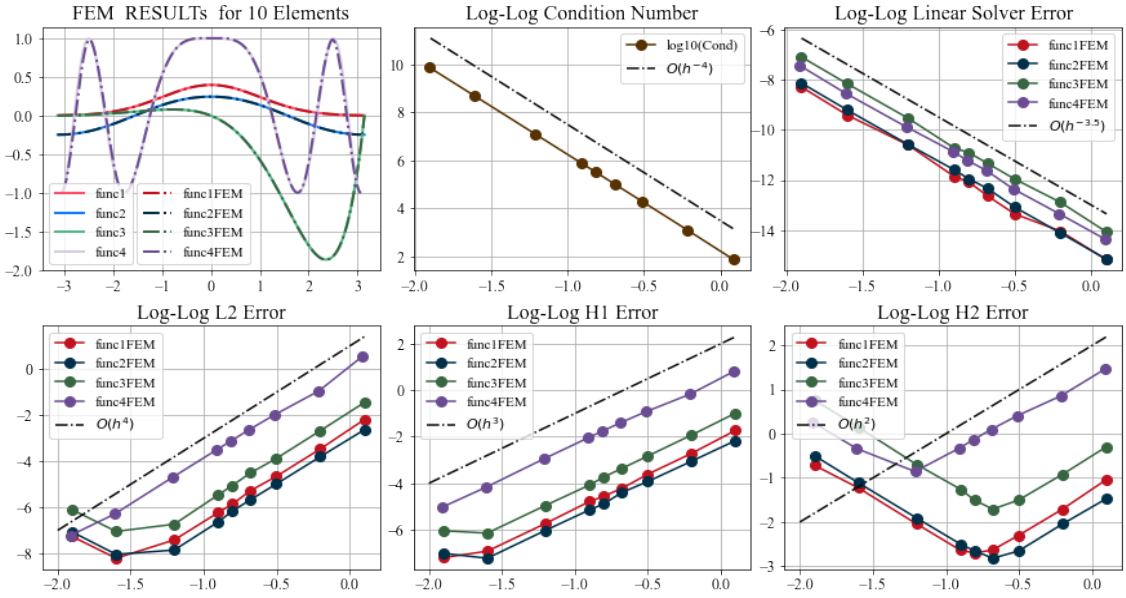

Fig. 5.2.1. Numerical Results for Continuous Forcing Functions

However, as the meshes are getting refined, the error starts to increase counter-intuitively. This is somewhat normal as the stiffness matrices are becoming increasingly ill-conditioned for small h : when $h \approx 0.1$, the condition number of the matrix almost reaches 10^6. The rate of this growing condition number is about $O(h^{-4})$ according to the experiment and may lead to a decreasing accuracy (measured with $||\mathrm{A}\tilde{\boldsymbol{c}} - \mathrm{F}||_2$) for solving the assembled system $\mathrm{A}\boldsymbol{c} = \mathrm{F}$ as is shown in the above plot $(1, 3)$ of Fig. 5.2.1. The effect of the quadrature rule may not play a very significant role here in these examples, as the adaptive **quad** method from **Scipy** with absolute tolerance *epsabs* $= 1.49 \times 10^{-15}$ provides a solution with only minor differences (see Appendix A.3.1 for comparison).

5.3 Discontinuous Forcing Functions

As was mentioned in the introduction, the finite element method allows us to reduce the regularity requirements for the original problem. While in the original formulation, the forcing term f is required to be continuous, the variational formulation makes it possible to seek solutions while f is only in $L^2(\Omega)$. For simplicity, we consider examples when f is piece-wise constant, i.e., $f = fr \cdot \mathbb{I}_{[a,c)} + fl \cdot \mathbb{I}_{[c,b]}$ where $c \in [a, b] = \Omega$ and fr, fl are constants. The true solutions can therefore be constructed with $u_{\text{true}} = u^-_{\text{true}} \cdot \mathbb{I}_{[a,c)} + u^+_{\text{true}} \cdot \mathbb{I}_{[c,b]}$, where

$$u^-_{\text{true}} = \sum_{i=0}^{4} A_i x^i \quad \text{and} \quad u^+_{\text{true}} = \sum_{i=0}^{4} B_i x^i$$

The 10 unknowns $A_0, \cdots, A_4$ and $B_0, \cdots, B_4$ can be obtained by solving the differential equation piece-wisely in $[a, c)$ and $[c, b]$, combined with 4 additional continuity

requirements $\frac{\mathrm{d}^k}{\mathrm{d}x^k} u^-_{\text{true}}(c) = \frac{\mathrm{d}^k}{\mathrm{d}x^k} u^-_{\text{true}}(c)$ for $k = 0 \cdots 3$. The same method can be applied to construct more complex known solutions where $fl := fl(x)$ and $fr := fr(x)$ are continuous functions over the restricted interval such that $fl(c) \neq fr(c)$. The examples we used are listed in Table 5.2.1

Index	Forcing Function	BC 1		BC 2	
i	f	$u(a)$	$u(b)$	$u'(a)$	$u'(b)$
1	$2 \cdot \mathbb{I}_{[-1,0)} - 2 \cdot \mathbb{I}_{[0,1]}$	0	0	0	0
2	$\mathbb{I}_{[-1,0)} + 2 \cdot \mathbb{I}_{[0,1]}$	$\frac{1}{2}$	$\frac{1}{2}$	0	0
3	$-\mathbb{I}_{[-1,0)} - 2 \cdot \mathbb{I}_{[0,1]}$	0	0	1	-1
4	$3 \cdot \mathbb{I}_{[-1,0)} + 4 \cdot \mathbb{I}_{[0,1]}$	$\frac{1}{2}$	$-\frac{1}{2}$	1	-1

Table 5.2.1. Examples for Discontinuous Forcing Functions

The explicit form of the true solutions can be seen in Appendix A.1.3. The numerical results have been summarized in the following Fig. 5.3.1 (see Appendix A.3.2 for a clear version and a comparison result between different quadrature methods).

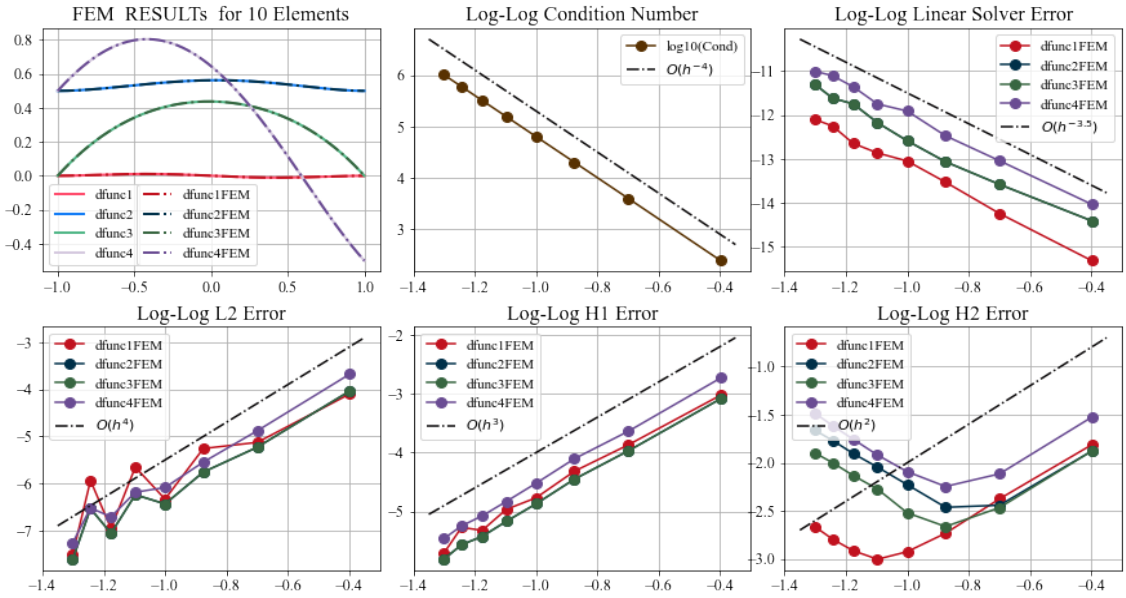

Fig. 5.3.1. Numerical Results for Discontinuous Forcing Functions

Again, the true solutions in our examples are just piece-wise polynomials and thus must be contained within the space of $H^4(\Omega)$. We can apply the same error estimates as in the previous section and find that the L^2, H^1, H^2 error follow a convergence rate of order $O(h^4)$, $O(h^3)$ and $O(h^2)$ respectively. A similar issue also occurs in solving ill-conditioned systems $\mathrm{A}\boldsymbol{c} = \mathrm{F}$ where the condition number $\kappa_2(\mathrm{A}) \sim O(h^{-4})$.

6 Conclusion

This paper mainly discusses the implementation of a finite element solver for the 1D biharmonic problem. The solver was originally derived from problems with homogeneous boundary conditions, but extra adjustments were made in Section 4.2.2 to let it also support non-homogeneous BCs. The correctness of the solver was tested in Section 5 to further validate the theory we used : for all test examples, the solver can produce a relatively reasonable solution using a mesh containing only 10 elements regardless of the continuity of the force function $f \in L^2(\Omega)$. The errors it generates were also mostly in line with the theory mentioned in the literature.

Our experiment further proves the effectiveness of the finite element method in solving higher-order equations, and also indicates its broad application in the fields of engineering and applied mathematics. Yet, there are still several aspects which we have not included in this article due to the length restriction. First, as was shown in the previous section, the solver is faced with the problem of tackling extremely ill-conditioned systems for refined mesh size h. The problem can be alleviated using preconditioning techniques to transform the ill-conditioned stiffness matrix into a well-conditioned one, such as using the direct-iterative method proposed by Wilson [10] and further expanded by Shi [9].

Second, another source of error may be related to the selection of quadrature rules used to evaluate the load vectors. In our experiment, the 4-point Gauss-Legendre rule was applied to solve the problem; however, it might be intuitively straightforward that the general error may be further reduced by using rules of a higher precision or by using composite rules instead of simple ones. The relationship between the general error of the FEM and the precision of the quadrature rules used may be another field worth further investigation.

Finally, the solver mainly uses uniform meshes while constructing discretizations. Uniform meshes are easy to generate, yet they may not be the optimal choice for all forcing terms f. Within regions where f is well-behaved, it may be advantageous to generate wider cells to increase efficiency. Alternatively, a finer discretization may be needed otherwise. A remedy is to adopt the adaptive FEM, where cell sizes are controlled locally to further improve the accuracy of the solution [8].

7 References

[1] D. Braess. "Finite Elements: Theory, Fast Solvers, and Applications in Solid Mechanics". In: 3rd ed. Cambridge University Press, 2007. Chap. Conforming Finite Elements, p. 79. DOI: 10.1017/CBO9780511618635.

[2] J. Chessa. *Programing the Finite Element Method with Matlab*. Purdue University. 2002. URL: https://www.math.purdue.edu/~caiz/math615/matlab_fem.pdf (visited on 12/08/2023).

[3] P.G. Ciarlet. *The Finite Element Method for Elliptic Problems*. Classics in Applied Mathematics. Society for Industrial and Applied Mathematics, 2002. ISBN: 9780898715149. URL: https://books.google.co.uk/books?id=1PF-WSONl9IC (visited on 12/04/2023).

[4] P.E. Farrell. *Lecture Notes for C6.4 Finite Element Methods for PDEs*. University of Oxford. 2021. URL: https://people.maths.ox.ac.uk/farrellp/femvideos/notes.pdf (visited on 12/04/2023).

[5] L.E. Fraenkel. "An Introduction to Maximum Principles and Symmetry in Elliptic Problems". In: Cambridge Tracts in Mathematics. Cambridge: Cambridge University Press, 2000. Chap. Some Maximum Principles for Elliptic Equations, pp. 39–86. DOI: 10.1017/CBO9780511569203.

[6] S. Ganesan and L. Tobiska. "Finite Elements: Theory and Algorithms". In: Cambridge IISc Series. Cambridge University Press, 2017. Chap. Biharmonic Equation, pp. 59–86. DOI: 10.1017/9781108235013.006.

[7] R.C. Kirby. "A General Approach to Transforming Finite Elements". en. In: *The SMAI Journal of computational mathematics* 4 (2018), pp. 197–224. DOI: 10.5802/smai-jcm.33. URL: https://smai-jcm.centre-mersenne.org/articles/10.5802/smai-jcm.33/ (visited on 12/10/2023).

[8] L.Y. Li and P. Bettess. "Adaptive Finite Element Methods: A Review". In: *Applied Mechanics Reviews* 50.10 (Oct. 1997), pp. 581–591. ISSN: 0003-6900. DOI: 10.1115/1.3101670.

[9] G. Shi. "Direct-iterative Solution of Ill-Conditioned Finite Element Stiffness Matrices". In: *International Journal for Numerical Methods in Engineering* 18.2 (1982), pp. 181–194. DOI: https://doi.org/10.1002/nme.1620180204. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/nme.1620180204 (visited on 12/13/2023).

[10] E.L. Wilson. "Solution of Sparse Stiffness Matrices for Structural Systems". In: *Sparse Matrix Proceedings 1978*. SIAM Englewood Cliffs, NJ. 1979, pp. 1–24.

[11] A.C. Yew. *Numerical Integration: Gaussian Quadrature Rules*. Brown University. 2011. URL: `https://www.dam.brown.edu/people/alcyew/handouts/GLquad.pdf` (visited on 12/08/2023).

A Appendix

A.1 Implementation Notes

A.1.1 Quadrature Rules

Coefficients for simple quadrature rules applied in the solver are listed in Table A.1.1.

Nodes Number	Gauss Legendre		Gauss Lobatto	
n	$\tilde{x}_j$	w_j	$\tilde{x}_j$	w_j
2	$-\sqrt{1/3}$	1	-1	1
	$\sqrt{1/3}$	1	1	1
3	$-\sqrt{3/5}$	5/9	-1	1/3
	0	8/9	0	4/3
	$\sqrt{3/5}$	5/9	1	1/3
4	$-\sqrt{(15+2\sqrt{30})/35}$	$(18-\sqrt{30})/36$	-1	1/6
	$-\sqrt{(15-2\sqrt{30})/35}$	$(18+\sqrt{30})/36$	$-\sqrt{1/5}$	5/6
	$\sqrt{(15-2\sqrt{30})/35}$	$(18+\sqrt{30})/36$	$\sqrt{1/5}$	5/6
	$\sqrt{(15+2\sqrt{30})/35}$	$(18-\sqrt{30})/36$	-1	1/6
5	*	*	-1	1 /10
	*	*	$-\sqrt{3/7}$	49/90
	*	*	0	32/45
	*	*	$\sqrt{3/7}$	49/90
	*	*	1	1 /10

Table A.1.1. Quadrature Rule Coefficients

A.1.2 Fitting for Non-homogeneous Boundary Conditions

The inverse of the coefficient matrix in (24) is

$$\frac{1}{(b-a)^3}\begin{bmatrix} b^2(b-3a) & a^2(3b-a) & -ab^2(b-a) & -a^2b(b-a) \\ 6ab & -6ab & b(2a+b)(b-a) & a(a+2b)(b-a) \\ -3(b+a) & 3(b+a) & -(a+2b)(b-a) & -(2a+b)(b-a) \\ 2 & -2 & b-a & b-a \end{bmatrix}$$

A.1.3 True Solution for Discontinuous Forcing Functions

The coefficients $A_0 = BC_1(a)$, $B_0 = BC_1(b)$, $A_1 = BC_2(a)$, $B_1 = BC_2(b)$ and $A_4 = fr/24$, $B_4 = fl/24$ is determined by the equation. Setting $c = (a+b)/2$ and $l = (b-a)/2$. Then the rest coefficients can be obtained by solving the following system led by the continuity requirements

$$\begin{bmatrix} l^2 & l^3 & -l^2 & l^3 \\ 2l & 3l^2 & 2l & -3l^2 \\ 2 & 6l & -2 & 6l \\ 0 & 1 & 0 & -1 \end{bmatrix} \begin{bmatrix} A_2 \\ A_3 \\ B_2 \\ B_3 \end{bmatrix} = \begin{bmatrix} (B_0 - A_0) - l(B_1 + A_1) + l^4(B_4 - A_4) \\ (B_1 - A_1) - 4l^3(B_4 + A_4) \\ 12l^2(B_4 - A_4) \\ -4l(B_4 + A_4) \end{bmatrix}$$

The results are listed as follows

Index	Function	Coefficients				
i	$u^-(u^+)$	$A_0(B_0)$	$A_1(B_1)$	$A_2(B_2)$	$A_3(B_3)$	$A_4(B_4)$
1	u^-	0	0	1/8	-5/24	1/12
	u^+	0	0	-1/8	-5/24	-1/12
2	u^-	1/2	0	7/32	-19/96	1/24
	u^+	1/2	0	9/32	29/96	1/12
3	u^-	0	1	-23/32	19/96	-1/24
	u^+	0	-1	-25/32	-29/96	-1/12
4	u^-	1/2	1	-67/96	-9/32	1/8
	u^+	-1/2	-1	83/96	85/96	1/6

Table A.1.3. True Solutions for Samples in Table 5.2.1

A.2 Additional Codes

A.2.1 Expression for Local Stiffness Matrices

```
# Calculte local stiffness matrix
import sympy as sp
x, h = sp.symbols('x h')
D2phi = sp.Matrix([12*x - 6, 6*x - 4, -12*x + 6, 6*x - 2])

V = sp.diag(1, h, 1, h)
basis_product = D2phi * D2phi .transpose()
integral_matrix = basis_product.applyfunc(lambda elem: sp.
    integrate(elem, x))

integral_at_1 = integral_matrix.subs(x, 1)
integral_at_0 = integral_matrix.subs(x, 0)
result = V.T * (integral_at_1 - integral_at_0) * V
```

A.3 Figures and Effects of Quadrature Rules

A.3.1 Comparison Results for Continuous Forcing Functions

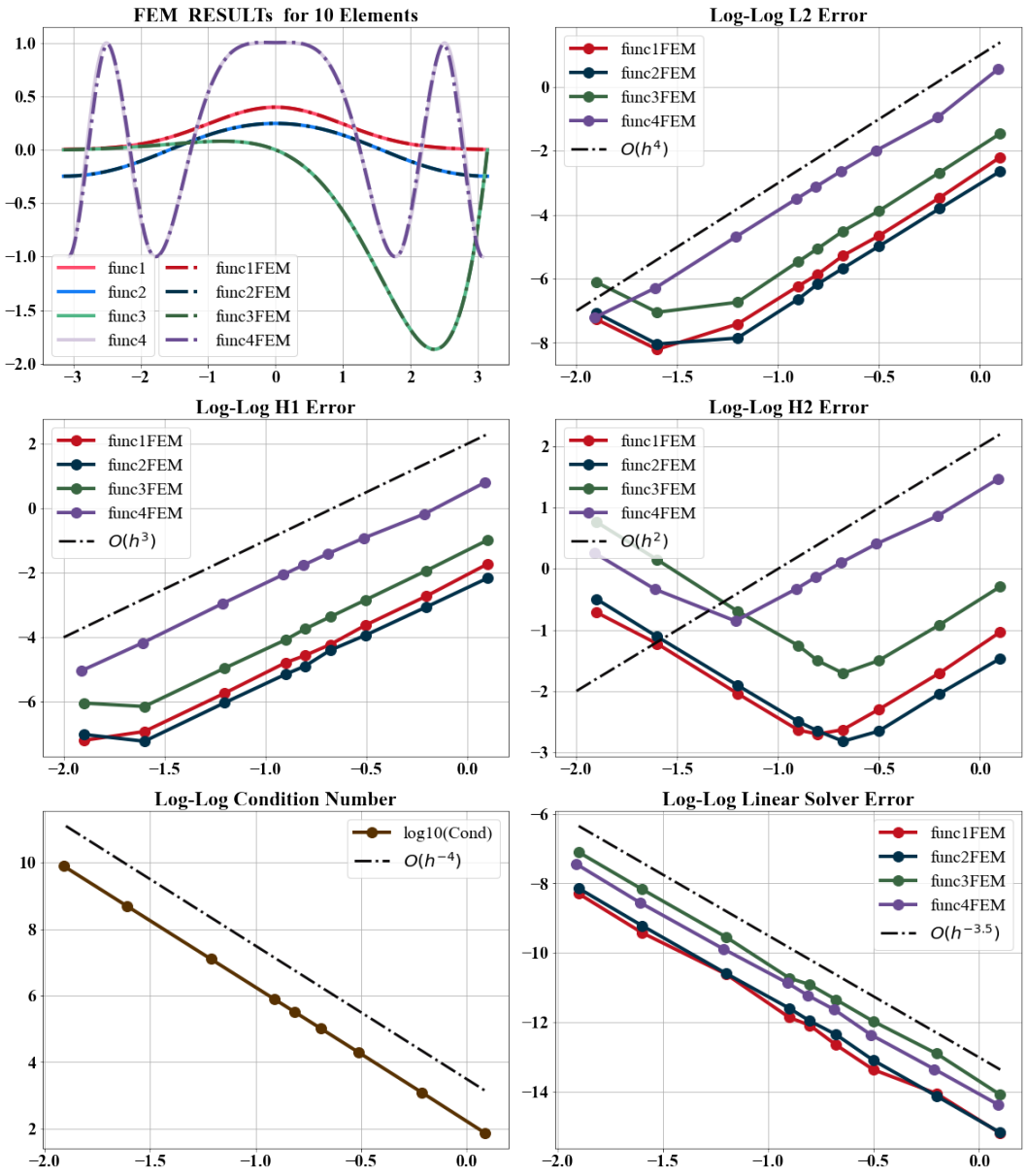

Fig. A.3.1.1. Results for Continuous Forcing Functions with GaussLeg4

The x-axis for the first plot is just the independent variable of the solution $u(x)$; The x-axes for all other 5 plots are identically set as $\log_{10} h$, where h is the mesh size. The above results Fig. A.3.1.1. are obtained using 4-point Gauss-Legendre quadrature.

The following results Fig. A.3.1.2. are obtained by replacing the built-in quadrature rule with the adaptive quadrature function **quad** from Python **Scipy** with an absolute tolerance of $epsabs = 1.49 \times 10^{-15}$

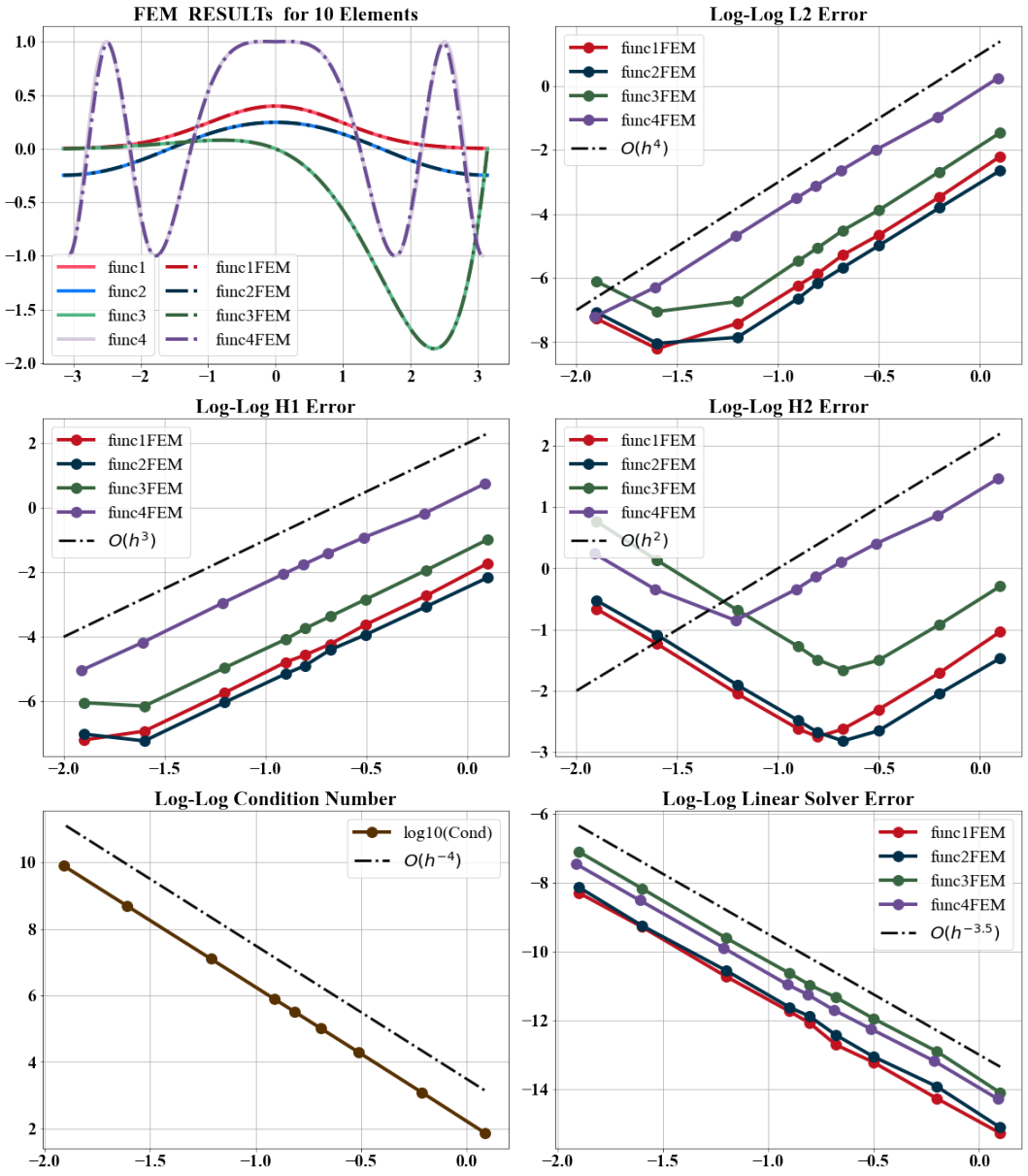

Fig. A.3.1.2. Results for Continuous Forcing Functions with Scipy quad

A.3.2 Comparison Results for Discontinuous Forcing Functions

The following results Fig. A.3.2.1. are obtained using 4-point Gauss-Legendre quadrature.

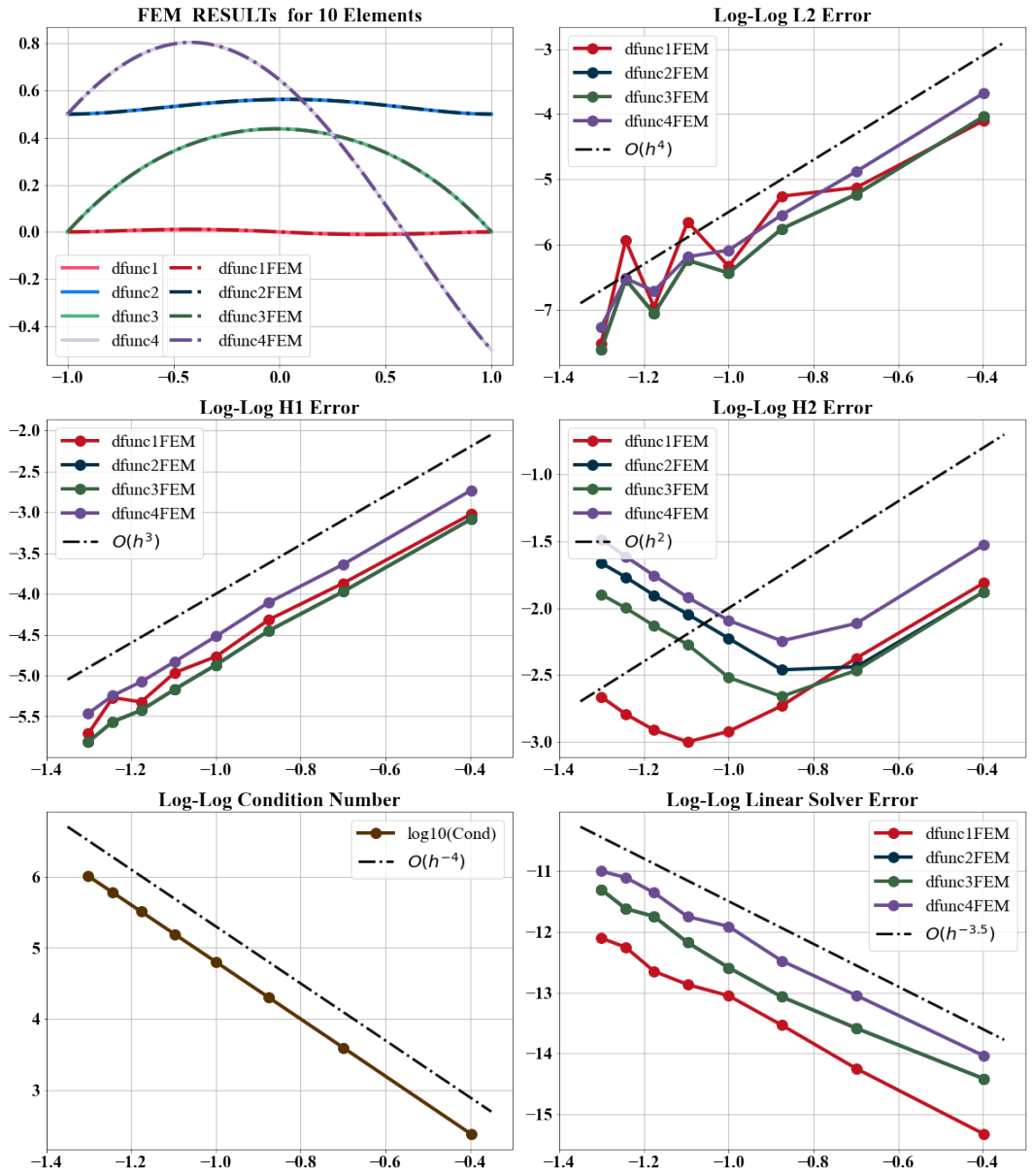

Fig. A.3.2.1. Results for Discontinuous Forcing Functions with GaussLeg4

The results Fig. A.3.2.2. in the next page are obtained by replacing the built-in quadrature rule with the adaptive quadrature function **quad** from Python **Scipy** with an absolute tolerance of $epsabs = 1.49 \times 10^{-15}$.

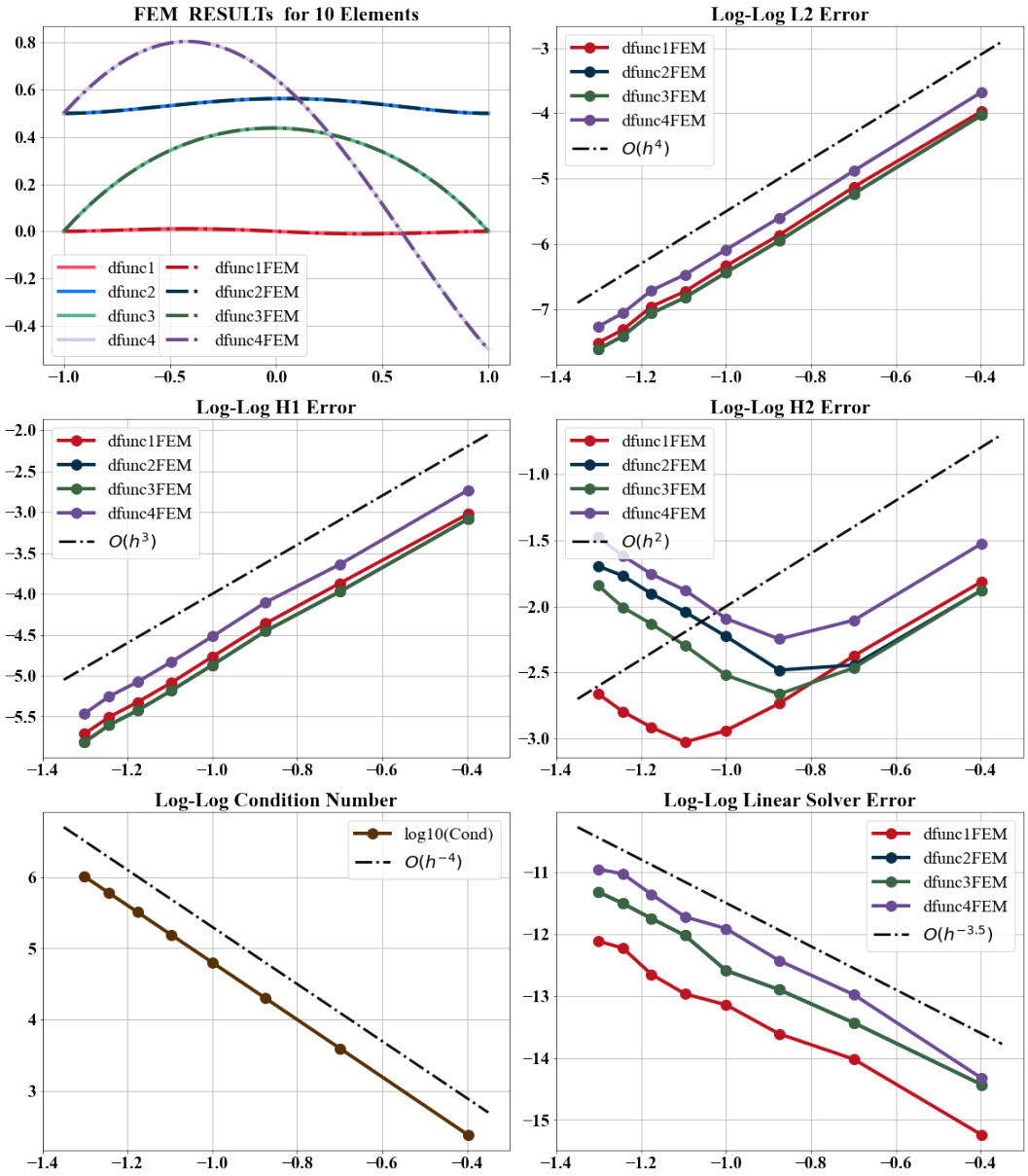

Fig. A.3.2.1. Results for Discontinuous Forcing Functions with Scipy quad