Balancing truncation and round-off errors in FEM: two-dimensional analysis

Abstract

The round-off error is investigated when solving a problem using 2D FEM methods. We consider multiple FEM packages and multiple FEM methods. Different implementation gives different highest achievable accuracy. The round-off error can be well represented by the number of degrees of freedom (DoFs) and a coefficient related to the computer precision. The strategy in [1] for predicting the highest achievable accuracy is extended to 2D cases. The bound of the round-off error is determined by solving a problem with a manufactured solution. Using our strategy the time for obtaining the highest achievable accuracy can be saved ?%.

Keywords: Finite Element Method, Round-off Error, 2D, Accuracy, Efficacy.

1. Introduction

In [1], we observe the round-off error of one-dimensional second-order differential equations has a power-law relation with the number of DoFs, and the coefficient is relatively fixed. Based on that, we furthermore propose a strategy to predict the highest achievable accuracy. The algorithm for realising the strategy allows us to obtain the highest achievable accuracy correctly with a relatively small amount of CPU time. However, the study focuses on one-dimensional cases, and hence there is a lack of the analysis on two-dimensional cases, which is more and more widely used for practical problems. Therefore, the aim of the paper is to investigate the round-off error when solving 2D problems using FEM methods, and extend the strategy in [1] to 2D cases.

The 2D numerical analysis allows the solution to vary both in x and y directions, and hence the physical properties are better retained. In comparison with 1D problems, 2D problems have the following features. First, the problem size of 2D problems increases greatly. For example, using the standard FEM with the grid size h and element degree p, the number of DoFs of 2D problems is the square of that of 1D problems. This increase inevitably entails much more CPU time and hence highlights the importance of the efficacy in solving the problem. Second, the derivatives of 2D problems are more complicated. One is that they contain more components than 1D problems do: the number of components for the first and second derivatives are 2 and 4, respectively, for 2D problems, but 1 for 1D problems; the other is that they have more types: they contain not only pure second derivatives, i.e. u_{xx} and u_{yy}, but also mixed ones, i.e. u_{xy} and u_{yx}, where u is the unknown variable.

The round-off error is deeply related to the limitation in the capability of the IEEE (double) floating-point arithmetic, of which the precision is as high as 1×10^{-16}. Since this arithmetic is a common standard for scientific computing, we may observe its influence on the round-off error in different FEM packages and FEM methods. Therefore, here we investigate two FEM packages: one is deal.II [2], the other one is FEniCS [3]. The former is written in C++ and constructed by quadrilaterals, and the latter is written in Python and constructed by triangles. Different grid types indicate different distributions of degrees of freedom and hence different ways of counting them. In [1], we find the mixed FEM produces higher accuracy than the standard FEM when the number of DoFs is relatively large, using the same element degree; this holds for both the solution and its first and second derivatives. We will demonstrate this for 2D problems using the above two packages.

The paper is organized as follows. The model problem, finite element formulation and numerical implementation are described in Section 2. The approach to predicting the highest achievable accuracy $E_{\min}$ is illustrated in Section 3. The study on the property of the round-off error is given in Section 4. The algorithm for realizing the approach is put forward in Section 6, followed by its validation by a Helmholtz problem in Section 7. The conclusions are drawn in Section 8.

2. Model problem, finite element formulation and numerical implementation

2.1. Model problem

We consider the following two-dimensional second-order differential equation:

$$-\nabla \cdot (D(\mathbf{x})\nabla u) + r(\mathbf{x})u(\mathbf{x}) = f(\mathbf{x}), \qquad \mathbf{x} \in \Omega = [0,\,1] \times [0,\,1], \tag{1}$$

where u denotes the unknown variable, $f(\mathbf{x}) \in L_2(\Omega)$ the prescribed right-hand side, and $D(\mathbf{x})$ and $r(\mathbf{x})$ the coefficient functions. $D(\mathbf{x})$ is assumed to be positive definite. By choosing $D(\mathbf{x})$ as the identity matrix I and $r(\mathbf{x}) = 0$, Eq. (1) reduces to the Poisson equation; for $|D(\mathbf{x})| > 0$, the diffusion equation is found when $r(\mathbf{x}) = 0$, and the Helmholtz equation is found for $r(\mathbf{x}) \neq 0$. The boundary conditions are $u(\mathbf{x}) = g(\mathbf{x})$ on Γ_D and $D(\mathbf{x})\nabla u = \mathbf{h}(\mathbf{x})$ on Γ_N. Here, Γ_D and Γ_N are the boundaries where Dirichlet boundary conditions and Neumann boundary conditions are imposed, respectively. It is important to note that we assume the second derivative exists in the weak sense, i.e. $u \in H^2(\Omega)$ for Eq. (1), even though this is only guaranteed for Poisson equations [4, p. 9].

2.2. Finite element formulation

2.2.1. Preliminaries

Before deriving the weak formulations, we define the following function spaces for a scalar function $t(x, y)$ [5]. First, the space of square integrable functions on Ω [4]:

$$L_2(\Omega) := \left\{ t \Big| \int_\Omega |t|^2 dx = \|t\|^2_{L_2(\Omega)} < +\infty \right\}. \tag{2a}$$

Second, a general form of Eq. (2a) that concerns not only the variable t itself but also its derivatives [4]:

$$H^m(\Omega) := \left\{ t \Big| D^\alpha t \in L_2(\Omega), \qquad \forall |\alpha| \leq m \right\}, \tag{2b}$$

where $m \geq 0$ and

$$D^\alpha t := \frac{\partial^{|\alpha|} t}{\partial x^{\alpha_1} \partial y^{\alpha_2}} \qquad |\alpha| = \alpha_1 + \alpha_2.$$

Obviously, higher-order derivatives are concerned with increasing m. Specifically,

$$\begin{aligned}
H^0(\Omega) &:= L_2(\Omega), \\
H^1(\Omega) &:= \left\{ t \in L_2(\Omega) \Big| \frac{\partial t}{\partial x}, \frac{\partial t}{\partial y} \in L_2(\Omega) \right\}, \text{ and} \\
H^2(\Omega) &:= \left\{ t \in H^1(\Omega) \Big| \frac{\partial^2 t}{\partial^2 x}, \frac{\partial^2 t}{\partial y^2}, \frac{\partial^2 t}{\partial x \partial y} \in L_2(\Omega) \right\}
\end{aligned}$$

For a vector function $\mathbf{t}(x, y)$ [5], we define the following function spaces [6]:

$$H(div, \Omega) := \left\{ \mathbf{t} \in L_2(\Omega, \mathbb{R}^2) \Big| \nabla \cdot \mathbf{t} \in L_2(\Omega) \right\}. \tag{3}$$

For convenience, we introduce three inner products [7]:

$$\langle \mathbf{f}_1, \mathbf{f}_2 \rangle = \int_\Omega \mathbf{f}_1(\mathbf{x}) \cdot \mathbf{f}_2(\mathbf{x}) \, dA, \tag{4a}$$

$$\langle f_1, f_2 \rangle = \int_\Omega f_1(\mathbf{x}) f_2(\mathbf{x}) \, dA, \text{ and} \tag{4b}$$

$$\langle f_1, f_2 \rangle_\Gamma = \int_\Gamma f_1(\mathbf{x}) f_2(\mathbf{x}) ds, \tag{4c}$$

where $\mathbf{f}_1(\mathbf{x})$ and $\mathbf{f}_2(\mathbf{x})$ denote continuous vector-valued functions of length 2, and $f_1(\mathbf{x})$ and $f_2(\mathbf{x})$ denote continuous scalar functions. Γ denotes the boundary of Ω.

2.2.2. The standard FEM

The weak form of Eq. (1) is derived in Appendix A.1. Imposing the Dirichlet boundary conditions strongly, the weak form reads [8]:

> Weak form 1
>
> Find $u \in H^1_D(\Omega)$ such that:
>
> $\langle \nabla\eta,\, D\nabla u\rangle + \langle \eta,\, ru\rangle = \langle \eta,\, f\rangle + \langle \eta,\, \mathbf{h}\cdot\mathbf{n}\rangle_{\Gamma_N}. \qquad \forall \eta \in H^1_{D0}(\Omega),$
>
> with
>
> $$H^1_D(\Omega) = \{t \mid t \in H^1(\Omega),\ t = g \text{ on } \Gamma_D\},$$
> $$H^1_{D0}(\Omega) = \{t \mid t \in H^1(\Omega),\ t = 0 \text{ on } \Gamma_D\}.$$

(5)

η denotes the test function, $\mathbf{n}$ is the unit normal vector; the terms in the right-hand sides consist of information of Neumann boundary conditions which vanishes if no Neumann boundary conditions are prescribed. We approximate u by a linear combination of a finite number of basis functions:

$$u \approx u_h^{(p)} = \sum_{i=1}^{m} u_i \varphi_i^{(p)}. \tag{6}$$

Here, p is the element degree, m is the number of DoFs over the whole domain, which equals $(p \times 2^R + 1)^2$, where R denotes the number of refinements; u_i's are the values of $u_h^{(p)}$ at the DoFs; $\varphi_i^{(p)}$'s are C^0-continuous Lagrange basis functions supported by Gauss-Lobatto points, which feature the Kronecker-delta property, i.e. $\varphi_i^{(p)}(\mathbf{x}_j) = \delta_{ij}$, where $\mathbf{x}_j$ denotes the support point. This type of element will be referred to as Q_p. Note that, the number of DoFs is counted based on problems with real-valued solution, and hence it doubles for problems with complex-valued solution.

Substituting the test functions, the resulting linear system of equations reads

$$AU = F, \tag{7}$$

where A is the stiffness matrix, F the right-hand side and U the numerical solution of u at DoFs.

2.2.3. The mixed FEM

As a first step, we introduce the auxiliary variable

$$\mathbf{v}(\mathbf{x}) = -D(\mathbf{x})\nabla u, \tag{8a}$$

allowing Eq. (1) to be rewritten as

$$-\nabla\cdot\mathbf{v} - r(\mathbf{x})u(\mathbf{x}) = -f(\mathbf{x}). \tag{8b}$$

The weak form of Eq. (1) using the mixed FEM, derived in Appendix A.2, is given by:

Weak form 2

Find $\mathbf{v} \in H_N(div, \Omega)$ and $u \in L^2(\Omega)$ such that:

$$\langle \mathbf{w},\, D^{-1}\mathbf{v} \rangle - \langle \nabla \cdot \mathbf{w},\, u \rangle = -\langle \mathbf{w} \cdot \mathbf{n},\, g \rangle_{\Gamma_D} \qquad \forall w \in H_{N0}(div, \Omega), \tag{9a}$$

$$-\langle q,\, \nabla \cdot \mathbf{v} \rangle - \langle q,\, ru \rangle = -\langle q,\, f \rangle \qquad \forall q \in L^2(\Omega), \tag{9b}$$

with

$$H_N(div, \Omega) = \{\mathbf{t} \mid \mathbf{t} \in H(div, \Omega),\ \mathbf{t} = -\mathbf{h} \text{ on } \Gamma_N\},$$

$$H_{N0}(div, \Omega) = \{\mathbf{t} \mid \mathbf{t} \in H(div, \Omega),\ \mathbf{t} = 0 \text{ on } \Gamma_N\}.$$

In Eq. (9), $\mathbf{w}$ and q denote the test functions of $\mathbf{v}$ and u, respectively, and $\mathbf{n}$ has the same interpretation as before. If the auxiliary variable is taken to be the opposite of the gradient of the solution, i.e.

$$\mathbf{v}(\mathbf{x}) = -\nabla u, \tag{10a}$$

which is utilized in calculating the advection term in [9], then Eq. (1) can be rewritten as

$$-\nabla \cdot (D(\mathbf{x})\mathbf{v}(\mathbf{x})) - r(\mathbf{x})u(\mathbf{x}) = -f(\mathbf{x}). \tag{10b}$$

Using the above representations, the weak form, of which the derivation process is similar to Eq. (9), reads:

Weak form 3

Find $\mathbf{v} \in H_N(div, \Omega)$ and $u \in L^2(\Omega)$ such that:

$$\langle \mathbf{w},\, \mathbf{v} \rangle - \langle \nabla \cdot \mathbf{w},\, u \rangle = -\langle \mathbf{w} \cdot \mathbf{n},\, g \rangle_{\Gamma_D} \qquad \forall w \in H_{N0}(div, \Omega), \tag{11a}$$

$$-\langle q,\, \nabla \cdot (D\mathbf{v}) \rangle - \langle q,\, ru \rangle = -\langle q,\, f \rangle \qquad \forall q \in L^2(\Omega), \tag{11b}$$

with

$$H_N(div, \Omega) = \{\mathbf{t} \mid \mathbf{t} \in H(div, \Omega),\ \mathbf{t} = -D^{-1}\mathbf{h} \text{ on } \Gamma_N\},$$

$$H_{N0}(div, \Omega) = \{\mathbf{t} \mid \mathbf{t} \in H(div, \Omega),\ \mathbf{t} = 0 \text{ on } \Gamma_N\}.$$

$\mathbf{w}$, q and $\mathbf{n}$ have the same meaning as before. We use Eq. (9) if not stated otherwise.

One feature of the 2D mixed FEM is the adoption of non-primitive finite elements for the auxiliary variable [10]. Here, we implement the widely used Raviart-Thomas elements (RT_p) and Brezzi-Douglas-Marini elements (BDM_p) [6], where p still denotes the element degree. The associated elements for u are discontinuous Lagrangian polynomials of degree p (Q_p^{disc}) for the former and discontinuous Legendre elements of degree $p-1$ (P_{p-1}^{disc}) for the latter, where 'disc' denotes the elements for u are discontinuous. The property of these element pairs is introduced below.

RT_p/Q_p^{disc} elemensts. The distribution of degrees of freedom of RT_p elements is referred to [4, Chapter 2]: the number of DoFs is $p+1$ on an edge and $2 \times p \times (p+1)$ on a quad, resulting that the number of DoFs per cell is $2 \times (p+1) \times (p+2)$. Q_p^{disc} elements have the same property as Q_p elements introduced for the standard FEM, except that there are two independent sets of DoFs along the cell interface. Obviously, the number of DoFs per cell is $(p+1)^2$ for Q_p^{disc} elements.

Countably, for a unit square refined R times, the number of DoFs for $\mathbf{v}$ reads

$$N_{\mathbf{v}} = \underbrace{(2^R+1) \times 2}_{A_1} \times \underbrace{(p+1) \times 2^R}_{A_2} + \underbrace{4^R}_{B_1} \times \underbrace{2 \times p \times (p+1)}_{B_2}, \tag{12a}$$

where A_1 denotes the number of longest edges, of which the grid size is 1, on the domain, A_2 denotes the number of DoFs on a longest edge, B_1 denotes the total number of smallest cells, and B_2 denotes the number of DoFs inside a smallest cell; the number of DoFs for u reads

$$N_u = 4^R \times (p+1)^2. \tag{12b}$$

Since the composition of RT_p elements is relatively complicated [10], we do not illustrate the explicit approximation of $\mathbf{v}$ here, and refer to Eq. (13b) for the approximation of u:

$$\mathbf{v} \approx \mathbf{v}_h^{(p)} = ?, \tag{13a}$$

$$u \approx u_h^{(p)} = \sum_{j=1}^{(p+1)^2} u_{sj} \psi_j^{(p)} \text{ in cell } s, \text{ for } s = 1, 2, \ldots, t, \tag{13b}$$

where t denotes the total number of cells, u_{sj}'s are values of $u_h^{(p)}$ at the DoFs, and $\psi_j^{(p)}$'s are discontinuous Lagrange polynomials.

$BDM_p/P_{p-1}^{\text{disc}}$ elemensts. We also refer to [4, Chapter 2] for the distribution of DoFs of BDM_p elements: the number of DoFs on an edge is the same with that of RT_p elements, but the number of DoFs on a quad is $p \times (p-1)$ instead, resulting that the number of DoFs per cell is $(p+1) \times (p+2) + 2$. Legendre polynomials used here are L_2-orthogonal and normalized on the reference cell, resulting that the mass matrix is diagonal [11]. The number of DoFs per cell is $p \times (p+1)/2$ for u [11]. Note that, the element degree for u is one order lower than that for $\mathbf{v}$.

For a unit square refined R times, the number of DoFs for $\mathbf{v}$ reads

$$N_{\mathbf{v}} = \underbrace{(2^R+1) \times 2}_{A_1} \times \underbrace{(p+1) \times 2^R}_{A_2} + \underbrace{4^R}_{B_1} \times \underbrace{p \times (p-1)}_{B_3}, \tag{14a}$$

where B_3 denotes the number of DoFs inside a cell for BDM_p elements; the number of DoFs for u reads

$$N_u = 4^R \times \frac{p \times (p+1)}{2}. \tag{14b}$$

The approximation of $\mathbf{v}$ is not shown for the same reason as RT_p elements, and the approximation of u will be added later.

Substituting the test functions, the resulting coupled linear system of equations that has to be solved reads:

$$\begin{bmatrix} M & B \\ B^\top & W \end{bmatrix} \begin{bmatrix} V \\ U \end{bmatrix} = \begin{bmatrix} G \\ H \end{bmatrix}, \tag{15}$$

where the mass matrix M on $H(div, \Omega)$, discrete gradient operator B, its transpose, which is the discrete divergence operator $B^\top$, and the mass matrix W on $L_2(\Omega)$, comprise the left-hand side; G and H are the components of the right-hand side. V and U denote the numerical solution of $\mathbf{v}$ and u at DoFs.

2.3. Numerical implementation

2.3.1. Solution technique

All results are computed in IEEE-754 double precision [12] using the deal.II finite element library [2]. Unless stated otherwise, the computational mesh is obtained by globally refining a single element that covers the unit square Ω, and the Dirichlet boundary conditions are imposed strongly. Since deal.II does not provide native support for complex-valued problems, and hence all components need to be split into their real and imaginary parts.

To compute the occurring integrals, sufficiently accurate Gaussian quadrature formulas are used. To solve the systems of equations, the UMFPACK solver [13], which implements the multi-frontal LU factorization approach, is used unless stated otherwise. This solver results in relatively fast computations of the problems considered in this paper, and prevents the iteration errors of iterative solvers. The derivatives of the numerical solution are computed in the classical finite element manner, e.g. $u_{h,x} = \sum_{i=1}^{m} u_i \varphi_{i,x}$ yields an approximation to u_x using standard FEM.

2.3.2. Error estimation

We measure the error of a variable (var) in the function space that defines it, which basically involves the L_2 norm [14] only. The error is defined as

$$E_h = \|var_h - v\hat{a}r\|_2, \tag{16}$$

where var_h denotes the numerical solution of grid size h, and $v\hat{a}r$ denotes the reference solution. The latter is the exact solution when the exact solution is available and the solution of grid size $h/2$ otherwise [15].

The variable var may contain only one component such as u for both the standard FEM and the mixed FEM, or multiple components, such as ∇u in the standard FEM and $\mathbf{v}$ in the mixed FEM. For the latter, Eq. (16) can be further written as [2, 4]

$$E_h = \sqrt{\sum_{i=1}^{n_c} (\|var_h(i) - v\hat{a}r(i)\|_2)^2}, \tag{17}$$

where n_c denotes the number of components in a variable. That is, we compute the L_2 error of each variable component and then obtain the square root of the sum of their squares.

With respect to the numerical solution, since $u \in H^1(\Omega)$ for the standard FEM, and $u \in L_2(\Omega)$ and $\mathbf{v} \in H(div, \Omega)$ for the mixed FEM, the quantities which can be evaluated using Eq. (16) are up to $\nabla u = [u_x, u_y]^\top$ using the standard FEM, and up to $\nabla \cdot \mathbf{v} = [\partial/\partial x, \partial/\partial y]^\top \cdot [v_1, v_2]^\top = v_{1x} + v_{2y}$ using the mixed FEM. Note that, $\nabla \cdot \mathbf{v}$ only consists of pure second derivatives, i.e. u_{xx} and u_{yy}, when $\mathbf{v} = -\nabla u$, and also consists of mixed second derivatives, i.e. u_{xy} and u_{yx}, when $\mathbf{v} = -D(\mathbf{x})\nabla u$ and $D(\mathbf{x})$ is not diagonal, using the mixed FEM. A summary of the above quantities can be found in Table 1. Note that, $\|\nabla u\|_2$ is also called the H_1 semi-norm of u, and $\|\nabla \cdot \mathbf{v}\|_2$ the $H(div, \Omega)$ semi-norm of $\mathbf{v}$ [2].

Table 1 Quantities of which the L_2 norm exists in theory.

Type of quantities	The standard FEM	The mixed FEM
Solution	u	u
Gradient	∇u	$\mathbf{v}$
Hessian	–	$\nabla \cdot \mathbf{v}$

Consequently, using Eq. (16), we can only compute the error for the quantities in Table 1. Nevertheless, we will compute the error for the second derivative using the standard FEM, as a comparison with its counterpart $\nabla \cdot \mathbf{v}$ using the mixed FEM. The quantity that we are interested is the hessian matrix of u, i.e.

$$\mathbf{H}u = \begin{bmatrix} u_{xx} & u_{xy} \\ u_{yx} & u_{yy} \end{bmatrix},$$

or $\Delta u = u_{xx} + u_{yy}$ when only pure second derivatives are involved. In order to consider the mixed second derivatives when $D(\mathbf{x}) = 1$ for the mixed FEM, we also investigate the gradient of each component of $\mathbf{v}$, i.e.

$$\nabla \mathbf{v} = \begin{bmatrix} v_{1x} & v_{1y} \\ v_{2x} & v_{2y} \end{bmatrix}.$$

Furthermore, we also investigate each component of the aforementioned $\mathbf{H}u$ and $\nabla \mathbf{v}$. To summarize, the quantities of which the error is calculated are shown in Table 2.

Table 2 Quantities of which the error is computed.

Type of quantities	The standard FEM	The mixed FEM
Solution	u	u
Gradient	∇u	$\mathbf{v}$
Hessian	$\mathbf{H}u$, each component of $\mathbf{H}u$, and Δu	$\nabla \cdot \mathbf{v}$ $\nabla \mathbf{v}$ and each component of $\nabla \mathbf{v}$

2.3.3. Convergence of the solution

When the number of DoFs is relatively large, but the round-off error does not exceed the truncation error, the error converges at a fixed rate, known as asymptotic convergence rate, of which the value is one order higher than the approximation order [16]. In practice, the convergence rate in the numerical experiments can be calculated from

$$Q = \log_2 \left(\frac{E_h}{E_{h/2}} \right). \tag{18}$$

3. Sketch of the strategy to predict the highest attainable accuracy

A conceptual sketch of E_h against the number of DoFs (N_h) in a log-log plot can be found in Fig. 1, also see [17].

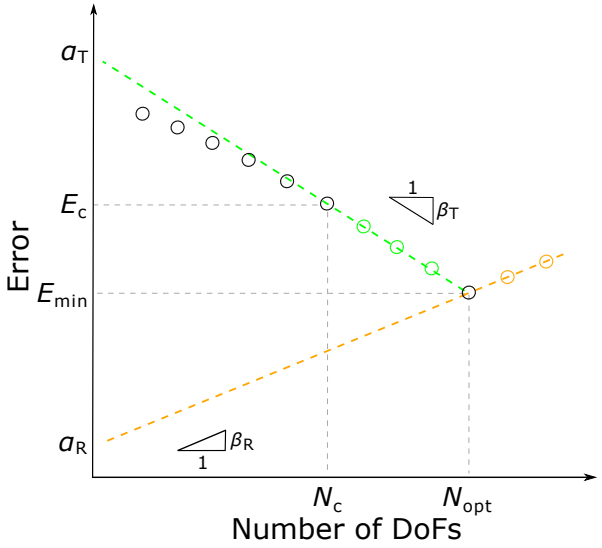


Fig. 1. Conceptual sketch of the error evolution against the number of DoFs.

When the number of DoFs is large enough, the error decreases at an asymptotic order with increasing number of DoFs. The slope of the decreasing part of the error β_T is half of the asymptotic order of convergence for u of the standard FEM and both u and $\mathbf{v}$ of the mixed FEM, cf. Appendix B.

To determine the round-off error, we use the method of manufactured solutions [18, 19, 20].

To make the method independent of u, we require $\|u\|_2$ is the same to that of the real problem.

Based on the round-off error, we determine the coefficients α_R and β_R using the method of least squares.

4. 2D problems

For the benchmark problems, we consider $u = (x-0.5)^2+(x-0.5)(y-0.5)+(y-0.5)^2$. The left and right boundaries are imposed by Dirichlet boundary conditions, and the upper and bottom boundaries are imposed by Neumann boundary conditions. Obviously, u can be exactly approximated when the approximation order is larger than or equal to 2, and the first derivative can be exactly approximated when the approximation order is larger than or equal to 1. We consider $D(\mathbf{x})$'s in Table 3. They are all symmetric: $D(\mathbf{x})$ of Case 1 is an identity matrix, and that of Cases 2–3 varies; $D(\mathbf{x})$ of Case 3 cannot be represented exactly by (Lagrange) polynomials. The property on the positiveness/definiteness can be found in the third column. The resulting L_2 norm of $\mathbf{v}$, denoted by $\|\mathbf{v}\|_2$, and right-hand side f can be found in the last two columns.

Table 3 Various $D(\mathbf{x})$'s.

Case	$D(\mathbf{x})$	Positiveness/ Definiteness	$\|\mathbf{v}\|_2$	f
1	I	Positive definite		
2	$\begin{pmatrix} 1+x+y & xy \\ xy & 1+x+y \end{pmatrix}$			
3	$\begin{pmatrix} e^{-[(x-0.5)^2+(y-0.5)^2]} & xy \\ xy & e^{-[(x-0.5)^2+(y-0.5)^2]} \end{pmatrix}$	Indefinite as a whole		

We are able to implement Q_p, RT_p/Q_p^{disc} and $BDM_p/P_{p-1}^{\text{disc}}$ elements for all the cases, but all results are unavailable now, c.f. Table 4 for an overview.

Table 4 Current status of the results for the benchmark problems in Table 3.

Case		1	2	3
Element type in deal.II	Q_p	✗[1]	✗	✗
	RT_p/Q_p^{disc}	✗	✗	✗
	$BDM_p/P_{p-1}^{\text{disc}}$	✗	✗	✗

[1] Results not available.

5. 1D problems

In this section, we investigate the decrease of the slope of the round-off error β_{R}. We consider 1D problems for which both boundaries are imposed by Dirichlet boundary conditions.

For all the error plots, we use a straight line to approximate the round-off error. To determine this line, we use the relatively larger round-off error, which occurs mostly when $p > 2$. The line color is red for the Poisson problems with the uniform mesh and orange otherwise because the round-off error of the latter might be smaller. For the problems which are not the Poisson problem of the uniform mesh, we also plot the line of the round-off error for it, to have a comparison.

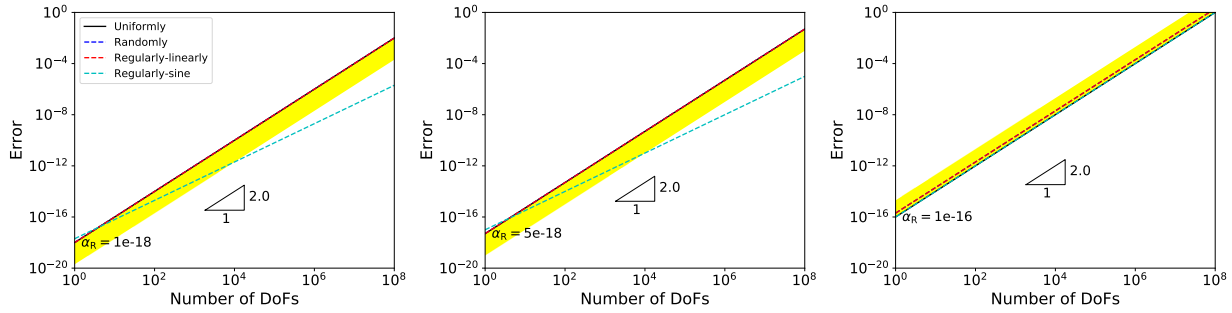
5.1. Poisson problems

We investigate the Poisson problems in Table 5. u is in the Q_p space (made of Lagrange polynomials of order up to p) when $u = (x - 0.5)^2$ and not in Q_p space when $u = e^{-(x-0.5)^2}$; $\|u\|_2$ of the second case (0.92) is about 10 times of that of the first case (0.11). For each problem, we use four kinds of mesh: mesh without distortion, mesh distorted randomly, the first type of regularly distorted mesh and the second type of regularly distorted mesh, c.f. Section Appendix D.1.2 for the settings of the distorted mesh.

A summary of the resulting β_{R} and α_{R} can be found in Table 5, and an illustration of their contribution to the round-off error can be found in Fig. 2 for $u = (x - 0.5)^2$ and Fig. 3 for $u = e^{-(x-0.5)^2}$. Based on both figures, the round-off error of the distorted mesh basically follows that of the uniform mesh, for which β_{R} is 2.0, and α_{R} is basically proportional to $\|u\|_2$. We notice that when using the third type of mesh, i.e. the distorted mesh of Eq. (D.1b), the round-off error of u and u_x will become smaller: β_{R} decreases to 1.5, and α_{R} basically does not change when $u = (x - 0.5)^2$; β_{R} does not change, and α_{R} decreases when $u = e^{-(x-0.5)^2}$.

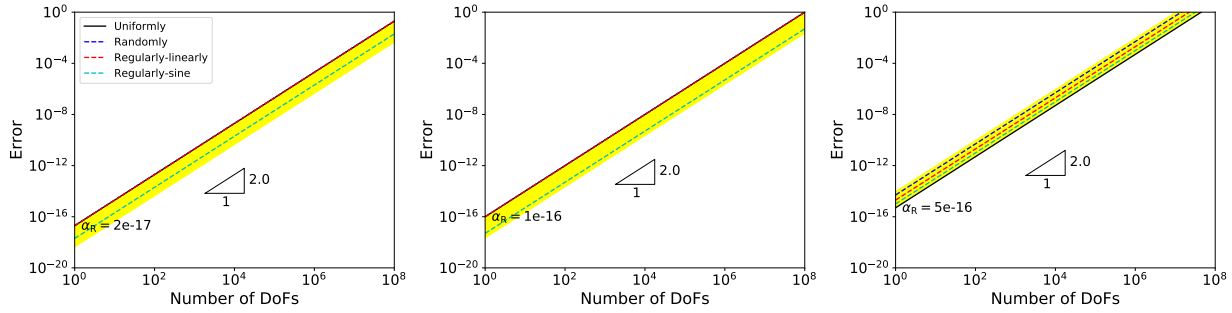
Table 5 Various Poisson problems and the resulting β_{R} and α_{R} in deal.II.

Distortion type	$u = (x - 0.5)^2$		$u = e^{-(x-0.5)^2}$	
	β_{R}	α_{R}	β_{R}	α_{R}
Uniformly	2.0 2.0 2.0	1e-18 5e-18 1e-16	2.0 2.0 2.0	2e-17 1e-16 5e-16
Randomly	2.0 2.0 2.0	1e-18 5e-18 2e-16	2.0 2.0 2.0	2e-17 1e-16 5e-15
Regularly of Eq. (D.1a)	2.0 2.0 2.0	1e-18 5e-18 2e-16	2.0 2.0 2.0	2e-17 1e-16 2e-15
Regularly of Eq. (D.1b)	1.5 1.5 2.0	2e-18 1e-17 1e-16	2.0 2.0 2.0	2e-18 5e-18 1e-15



(a) u **(b)** u_x **(c)** u_{xx}

Fig. 2. Illustration of β_{R} and α_{R} using Q_p elements for the 1D Poisson problem in deal.II.



(a) u **(b)** u_x **(c)** u_{xx}

Fig. 3. Illustration of β_{R} and α_{R} using Q_p elements for the 1D Poisson problem in deal.II.

5.2. Diffusion problems

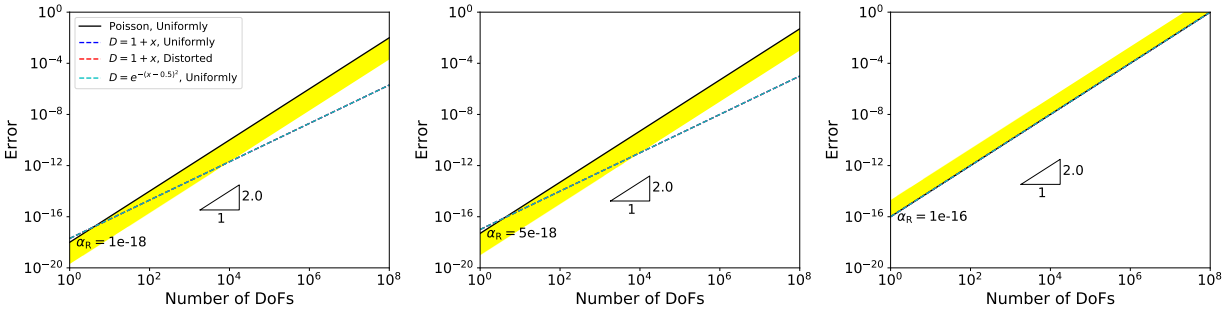
We still investigate the same u's but with various $D(\mathbf{x})$'s illustrated in Table 6. When $D(\mathbf{x}) = 1 + x$, we also use the regularly distorted mesh of Eq. (D.1b). Using the uniform mesh, the error is shown in Fig. D.21–Fig. D.22 when $u = (x - 0.5)^2$ and Fig. D.23–Fig. D.24 when $u = e^{-(x-0.5)^2}$. For the distorted mesh, the error is shown in Fig. D.25 and Fig. D.26, respectively.

An overview of the resulting β_{R} and α_{R} can be found in Table 6, and their contribution to the round-off error can be found in Fig. 4 and Fig. 5, respectively, in which we also plot that of the Poisson problem.

From these figures, in comparison with the Poisson problem, the round-off error of u_{xx} basically does not change, but that of u and u_x basically decreases. We describe the round-off error of u and u_x as follows. With respect to $u = (x - 0.5)^2$, α_{R} basically does not change, and β_{R} decreases to 1.5. With respect to $u = e^{-(x-0.5)^2}$, β_{R} and α_{R} behave differently for different types of $D(\mathbf{x})$. That is, when $D(\mathbf{x}) \in Q_2$, β_{R} remains 2.0, but α_{R} decreases; when $D(\mathbf{x}) \notin Q_2$, β_{R} decreases to 1.5, and α_{R} basically does not change.

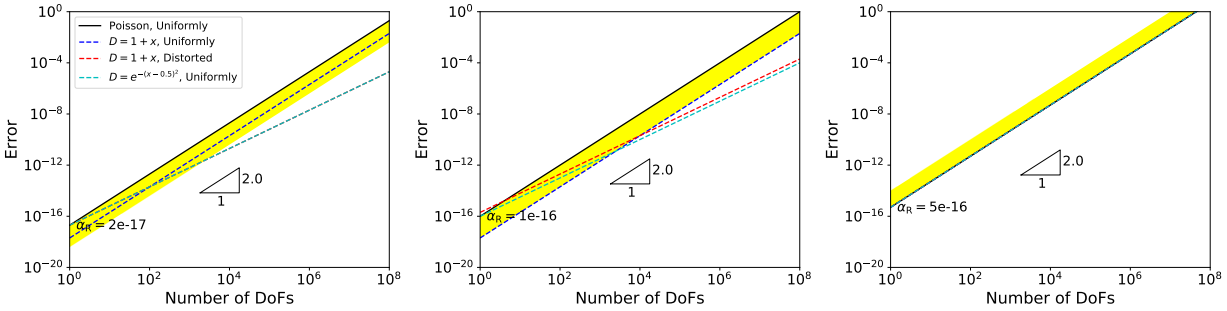
Table 6 Various diffusion problems and the resulting β_{R} and α_{R} in deal.II.

		$u=(x-0.5)^2$		$u=e^{-(x-0.5)^2}$	
		β_{R}	α_{R}	β_{R}	α_{R}
$D(\mathbf{x})=1+x$	Uniformly	1.5 1.5 2.0	2e-18 1e-17 1e-16	2.0 2.0 2.0	2e-18 2e-18 5e-16
	Distorted	1.5 1.5 2.0	2e-18 1e-17 1e-16	1.5 1.5 2.0	2e-17 2e-16 5e-16
$D(\mathbf{x})=e^{-(x-0.5)^2}$	Uniformly	1.5 1.5 2.0	2e-18 1e-17 1e-16	1.5 1.5 2.0	2e-17 1e-16 5e-16



(a) u **(b)** u_x **(c)** u_{xx}

Fig. 4. Illustration of β_{R} and α_{R} using Q_p elements for the 1D diffusion problem in deal.II.



(a) u **(b)** u_x **(c)** u_{xx}

Fig. 5. Illustration of β_{R} and α_{R} using Q_p elements for the 1D diffusion problem in deal.II.

5.3. Helmholtz problems

We investigate the cases shown in Table 7, for which only the uniform mesh is used. The errors for $u=(x-0.5)^2$ are shown in Fig. D.30 – Fig. D.35, and that for $u=e^{-(x-0.5)^2}$ are shown in Fig. D.37– Fig. D.41.

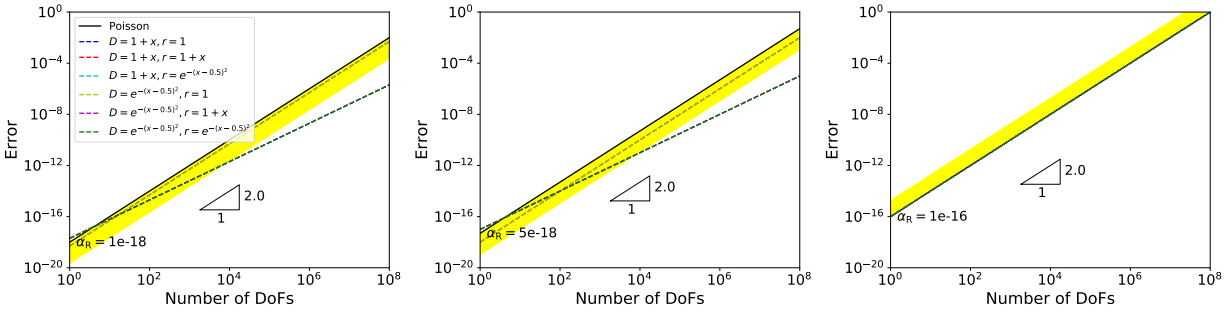
A summary of α_{R} and β_{R} can be found in Table 7, and their contribution to the round-off error is illustrated in Fig. 6 and Fig. 7, respectively, including that of the Poisson problem.

From these figures, the round-off error of u_{xx} is the same with that of the Poisson problem; that of u and u_x is basically smaller than that of the Poisson problem.

We describe the round-off error of u and u_x as follows. β_{R} of u and u_x increases back to 2.0 when $r(\mathbf{x})=1$, and it is still 1.5 when $r(\mathbf{x})$ varies.

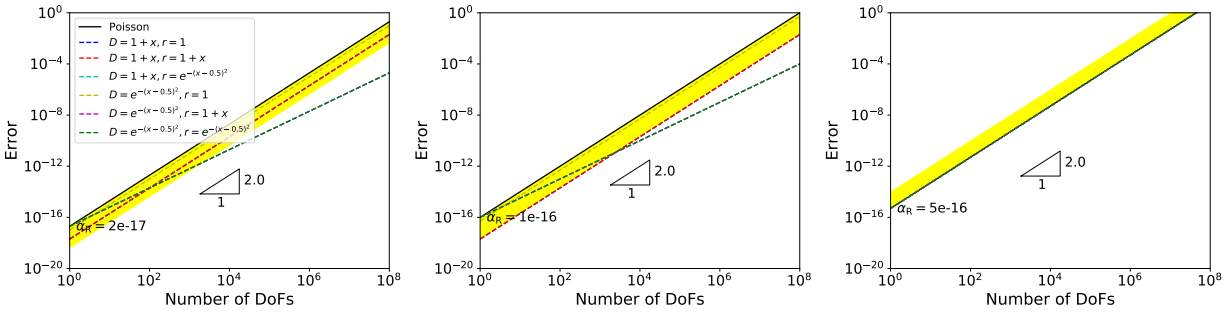
Table 7 Various Helmholtz problems and the resulting β_{R} and α_{R} in deal.II.

	$u=(x-0.5)^2$		$u=e^{-(x-0.5)^2}$		
	β_{R}	α_{R}	β_{R}	α_{R}	
$D(\mathbf{x})=1+x$	2.0 2.0 2.0	5e-19 1e-18 1e-16	2.0 2.0 2.0	2e-18 2e-18 5e-16	$r(\mathbf{x})=1$
	1.5 1.5 2.0	2e-18 1e-17 1e-16	2.0 2.0 2.0	2e-18 2e-18 5e-16	$r(\mathbf{x})=1+x$
	1.5 1.5 2.0	2e-18 1e-17 1e-16	1.5 1.5 2.0	2e-17 1e-16 5e-16	$r(\mathbf{x})=e^{-(x-0.5)^2}$
$D(\mathbf{x})=e^{-(x-0.5)^2}$	2.0 2.0 2.0	5e-19 1e-18 1e-16	2.0 2.0 2.0	1e-17 5e-17 5e-16	$r(\mathbf{x})=1$
	1.5 1.5 2.0	2e-18 1e-17 1e-16	1.5 1.5 2.0	2e-17 1e-16 5e-16	$r(\mathbf{x})=1+x$
	1.5 1.5 2.0	2e-18 1e-17 1e-16	1.5 1.5 2.0	2e-17 1e-16 5e-16	$r(\mathbf{x})=e^{-(x-0.5)^2}$



(a) u **(b)** u_x **(c)** u_{xx}

Fig. 6. Illustration of β_{R} and α_{R} using Q_p elements for the 1D Helmholtz problem in deal.II.



(a) u **(b)** u_x **(c)** u_{xx}

Fig. 7. Illustration of β_{R} and α_{R} using Q_p elements for the 1D Helmholtz problem in deal.II.

Concluding Section 5, the round-off error of u_{xx} basically does not change; the round-off error of u and u_x may be smaller than that of the Poisson problem with the uniform mesh because of the mesh distortion or varying D or r. To cover wider scenarios, we propose a strip area with one bound being the line approximating the round-off error of the Poisson problem with the uniform mesh. The another bound is defined by multiplying α_{R} of the above line by a factor of 0.02 for u and u_x and by a factor of 20 for u_{xx}.

6. Algorithm

Based on the validation experiments from the previous section, we introduce a novel a posteriori algorithm for determining $E_{\min}$ and its associated N_{opt} for the solution and its first and second derivative without performing brute-force mesh refinement. We call the algorithm *DoFinder*.

In *DoFinder*, we define the following coefficients and use them in the steps given below.

– a minimal number of h-refinements before carrying out '*NORMALIZATION*' and '*PREDICTION*', denoted by $R_{\min}$, with the following default values:

$$R_{\min} = \begin{cases} 9 - p & \text{for } p < 6, \\ 4 & \text{otherwise.} \end{cases} \tag{19}$$

We choose this parameter mainly because the error might increase, or decrease faster than the asymptotic order of convergence for coarse refinements, especially for lower-order elements.

– the allowed maximum N_h : 10^8, denoted by $N_{\max}$.

– a stopping criterion c_s for seeking the L_2 norm of the dependent variable, of which the value is 0.001 by default. We choose this parameter because the analytical solution does not exist for most practical problems.

– a relaxation coefficient c_r for seeking the asymptotic order of convergence, with the following default values:

$$c_r = \begin{cases} 0.9 & \text{for } p < 4, \\ 0.7 & \text{for } 4 \leqslant p < 10, \\ 0.5 & \text{otherwise.} \end{cases} \tag{20}$$

The procedure of *DoFinder* consists of four steps, which are explained below:

Step-1. '*INPUT*'. In this step, the custom input shown in the Table 8 has to be provided.

Table 8 Custom input of *DoFinder*.

Type	Item
Problem	• the problem to be solved
	• variables of which the accuracy is of interest
FEM	• standard or mixed formulation
	• an ordered array of element degrees $\{p_{\min}, \ldots, p_{\max}\}$

Step-2. '*NORMALIZATION*'. The function of this step is to find the L_2 norm of the dependent variable, in which elements of degree $p_{\min}$ are used. The specific procedure can be found in Algorithm 1.

Algorithm 1: NORMALIZATION

while $N_h < N_{\max}$ **do**
  if $\left|\frac{\|var_h\|_2 - \|var_{2h}\|_2}{\|var_h\|_2}\right| < c_s$ **then**
    $\|var\|_2 \leftarrow \|var_h\|_2$;
    break;
  else
    $h \leftarrow h/2$;
    calculate $\|var_h\|_2$;
  end
end

Step-3. '*Round-off error determination*' In this step, we use the method of manufactured solutions for determining α_{R} and β_{R}. The process is as follows.

Step-4. '*PREDICTION*'. This step finds $E_{\min}$ for each *var* and p of interest, as illustrated in Fig. 1. The procedure for carrying out this step can be found in Algorithm 2.

Algorithm 2: PREDICTION

while $N_h < N_{\max}$ **and** $\widetilde{E_h} > E_{\mathrm{R}}$ **do**
  $\widetilde{Q} \leftarrow \log_2\left(\widetilde{E_{2h}}/\widetilde{E_h}\right)$;
  if $\widetilde{Q} \geqslant \beta_{\mathrm{T}} \times c_r$ **then**
    $N_{\mathrm{c}} \leftarrow N_h$;
    $E_{\mathrm{c}} \leftarrow \widetilde{E_h}$;
    $\alpha_{\mathrm{T}} \leftarrow E_{\mathrm{c}}/{N_{\mathrm{c}}}^{-\beta_{\mathrm{T}}}$;
    $N_{\mathrm{opt}} \leftarrow \left(\frac{\alpha_{\mathrm{T}}\beta_{\mathrm{T}}}{\alpha_{\mathrm{R}}\beta_{\mathrm{R}}}\right)^{\frac{1}{\beta_{\mathrm{R}}+\beta_{\mathrm{T}}}}$;
    $E_{\min} \leftarrow \alpha_{\mathrm{T}}{N_{\mathrm{opt}}}^{-\beta_{\mathrm{T}}} + \alpha_{\mathrm{R}}{N_{\mathrm{opt}}}^{\beta_{\mathrm{R}}}$;
  else
    $h \leftarrow h/2$;
    calculate $\widetilde{E_h}$;
  end
end

Step-5. '*OUTPUT*'. In this step, we output $E_{\min}$, N_{opt}, etc., obtained from *Step-3*.

7. Application

7.1. 1D problems

We investigate the following problem:

$$((0.01+x)(1.01-x)u_x)_x - (0.01i)u(x) = 1.0, \qquad x \in I = [0,1], \tag{21}$$

with homogeneous Dirichlet and Neumann boundary conditions imposed as follows: $u(0) = 0$ and $u_x(1) = 0$. Both the standard FEM and the mixed FEM are investigated, with the element degree p taken in $\{1, 2, \ldots, 5\}$. Variables u, u_x and u_{xx} using the standard FEM and u, v and v_x using the mixed FEM are investigated.

7.2. 2D problems

We investigate the problem of Eq. (1) with the following settings:

$$D = \begin{bmatrix} 1.45\text{e-}4 - 7.56\text{e-}10i & 1.21\text{e-}9 - 8.91\text{e-}15i \\ -1.21\text{e-}9 + 8.91\text{e-}15i & 1.45\text{e-}4 - 7.56\text{e-}10i \end{bmatrix},$$

$r = 1.40\text{e-}4i$, and $g = 1.35$ on the left boundary, i.e. $x = 0$, and $\mathbf{h} \cdot \mathbf{n} = 0$ on other boundaries. That is, the left boundary is imposed by Dirichlet boundary conditions, and the other boundaries by Neumman boundary conditions.

8. Conclusion

We investigate the dependence of the round-off error on the number of DoFs for the two-dimensional case, and extend our strategy for predicting the highest achievable accuracy to the two-dimensional case. It shows that the round-off error also exists for 2D problems. The round-off error increases according to a power-law function with the number of DoFs. The mixed FEM gives higher accuracy than the standard FEM does. The above behaviour of the round-off error exists not only on deal.II, but also on FEniCS. The round-off error is approximated by consulting the round-off error of a problem of which the solution is manufactured to be exactly approximated by lower-order polynomials. Our algorithm is capable of evaluating the highest achievable accuracy of different packages and FEM methods.

Appendix A. Derivation of the weak form

Appendix A.1. The standard FEM

Multiplying Eq. (1) by a test function $\eta \in H^1(\Omega)$, and integrate it over Ω yield

$$\langle \eta, -\nabla \cdot (D\nabla u) + ru \rangle = \langle \eta, f \rangle. \tag{A.1}$$

By applying Gauss's theorem, we obtain

$$\langle \nabla\eta,\, D\nabla u\rangle + \langle \eta,\, ru\rangle = \langle \eta,\, f\rangle + \langle \eta,\, D\nabla u\cdot \mathbf{n}\rangle_{\Gamma}. \tag{A.2}$$

Taking $\eta = 0$ on Γ_D, and substituting the natural boundary condition $D\nabla u = \mathbf{h}$ on Γ_N, we obtain Eq. (5).

Appendix A.2. The mixed FEM

Multiplying Eq. (8a) by a test function of $\mathbf{v}$, i.e. $\mathbf{w} \in H_{N0}(div, \Omega)$, and integrate it over Ω yield

$$\langle D^{-1}\mathbf{v} + \nabla u, \mathbf{w}\rangle = 0. \tag{A.3a}$$

Applying Gauss's theorem to Eq. (A.3a), it becomes

$$\langle \mathbf{w},\, D^{-1}\mathbf{v}\rangle - \langle \nabla\cdot\mathbf{w},\, u\rangle = -\langle \mathbf{w}\cdot\mathbf{n},\, u\rangle_{\Gamma}. \tag{A.3b}$$

Unlike the standard FEM, for the mixed FEM, the essential boundary conditions are imposed on Γ_N, and the natural boundary conditions on Γ_D. By taking $\mathbf{w} = \mathbf{0}$ on Γ_N, and substituting the natural boundary conditions, i.e. $u = g$ on Γ_D, we obtain Eq. (9a).

Multiplying Eq. (8b) by a test function of u, i.e. $q \in L_2(\Omega)$, and integrating it over Ω yield

$$-\langle q,\, \nabla\cdot\mathbf{v}\rangle - \langle q,\, ru\rangle = -\langle q,\, f\rangle, \tag{A.4}$$

which results in Eq. (9b).

Appendix B. Determination of $\boldsymbol{\beta_{\mathrm{T}}}$

The standard FEM. For the grid size h and element degree p, the number of DoFs

$$N_h = ((1/h)\times p + 1)^2. \tag{B.1}$$

Therefore,

$$h = \frac{p}{\sqrt{N_h} - 1}. \tag{B.2}$$

Since the error of the solution [16]

$$E_h \leqslant Ch^{p+1}, \tag{B.3}$$

substituting Eq. (B.2) into Eq. (B.3), we obtain

$$E_h \leqslant C_1 \times (\sqrt{N_h} - 1)^{-(p+1)}, \tag{B.4}$$

where $C_1 = Cp^{p+1}$. Therefore, the slope β_{T} of the solution is about $(p+1)/2$.

The mixed FEM with RT_p/Q_p^{disc} elements. For the grid size h and element degree p, the number of DoFs of the gradient

$$\begin{aligned} N_h^{\text{grad}} &= (1/h) \times (p+1) \times (2 \times (1/h - 1) + 4) + (1/h^2) \times D_{RT}(p) \\ &\approx (1/h)^2 \times (2 \times (p+1) + D_{RT}(p)). \end{aligned} \tag{B.5}$$

Therefore,

$$h \approx \frac{p+1}{\sqrt{N_h^{\text{grad}}/2}}. \tag{B.6}$$

Substituting Eq. (B.6) into Eq. (B.3), we have

$$E_h \leqslant C_2 \times \left(\sqrt{N_h^{\text{grad}}}\right)^{-(p+1)}, \tag{B.7}$$

where $C_2 = C \times ((p+1)/\sqrt{2})^{p+1}$. Therefore, the slope β_{T} of the gradient is about $(p+1)/2$.

The number of DoFs of the solution

$$N_h^{\text{soln}} = ((1/h) \times (p+1))^2. \tag{B.8}$$

Therefore,

$$h = \frac{p+1}{\sqrt{N_h^{\text{soln}}}}. \tag{B.9}$$

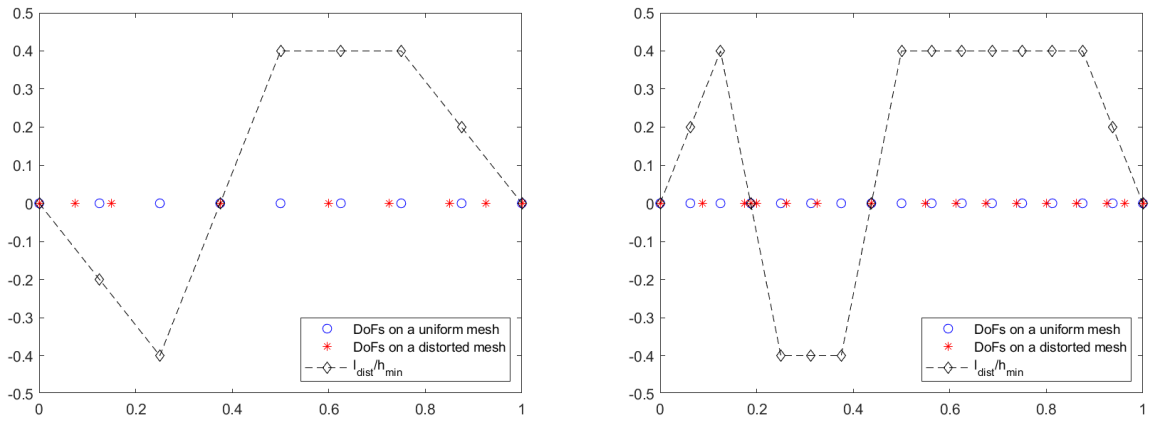
Substituting Eq. (B.9) into Eq. (B.3), we have

$$E_h \leqslant C_3 \times \left(\sqrt{N_h^{\text{soln}}}\right)^{-(p+1)}, \tag{B.10}$$

where $C_3 = C \times (p+1)^{p+1}$. Therefore, the slope β_{T} of the solution is about $(p+1)/2$.

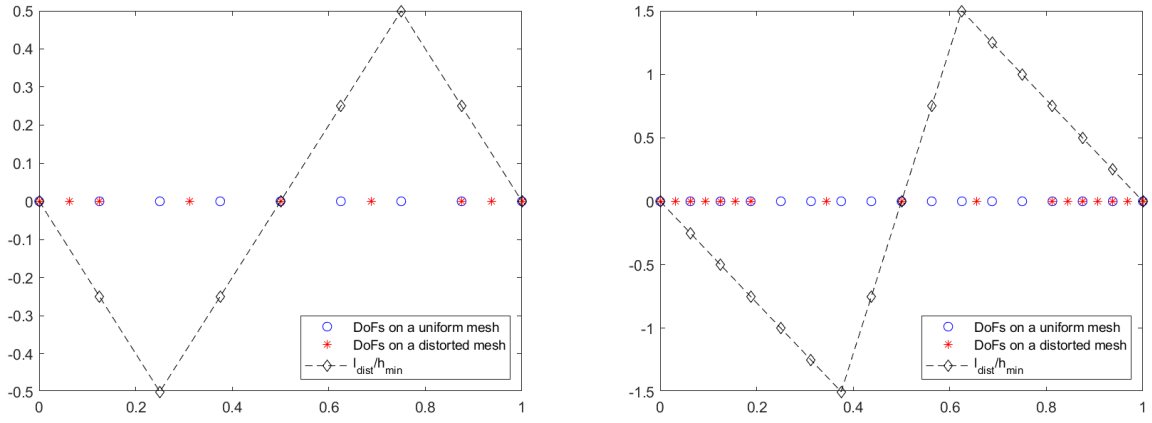
The mixed FEM with $BDM_p/P_{p-1}^{\text{disc}}$ elements. For this type of elements, we just need to replace $D_{RT}(p)$ in Eq. (B.5) by $D_{BDM}(p)$.

Appendix C. Illustration of mesh distortion



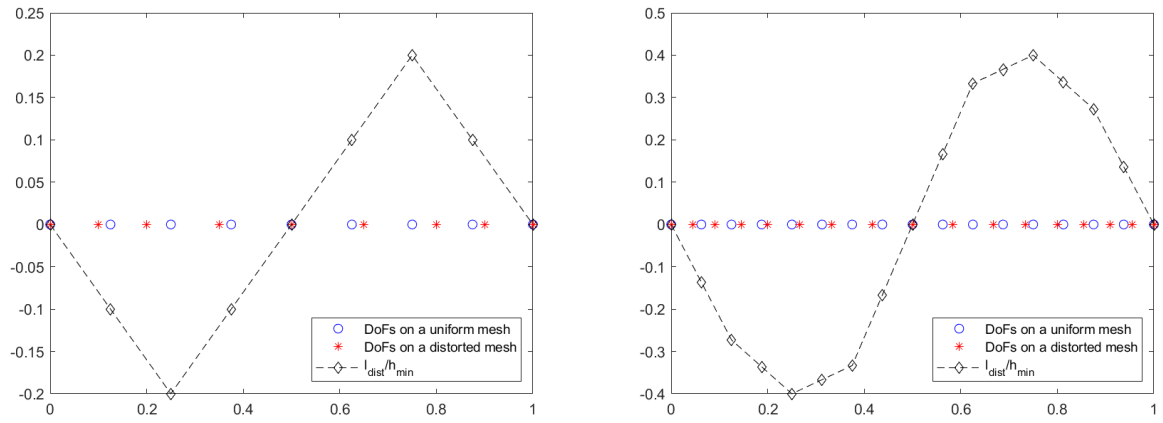
(a) $R = 2$ **(b)** $R = 3$

Fig. C.8. Comparison of the distribution of DoFs between the uniform mesh and the randomly distorted mesh when $p = 2$.



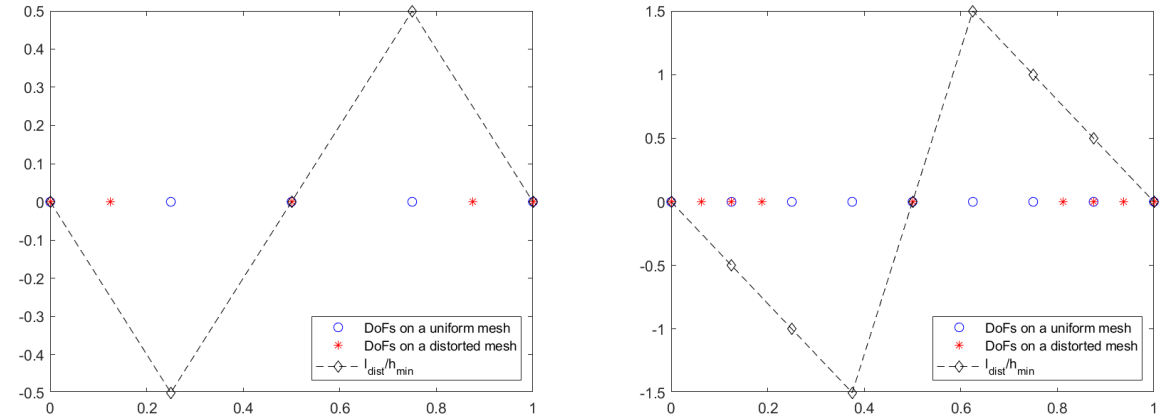
(a) $R = 2$ **(b)** $R = 3$

Fig. C.9. Comparison of the distribution of DoFs between the uniform mesh and the regularly distorted mesh of Eq. (D.1a) when $p = 2$.



(a) $R = 2$ **(b)** $R = 3$

Fig. C.10. Comparison of the distribution of DoFs between the uniform mesh and the regularly distorted mesh of Eq. (D.1b) when $p = 2$.



(a) $R = 2$ **(b)** $R = 3$

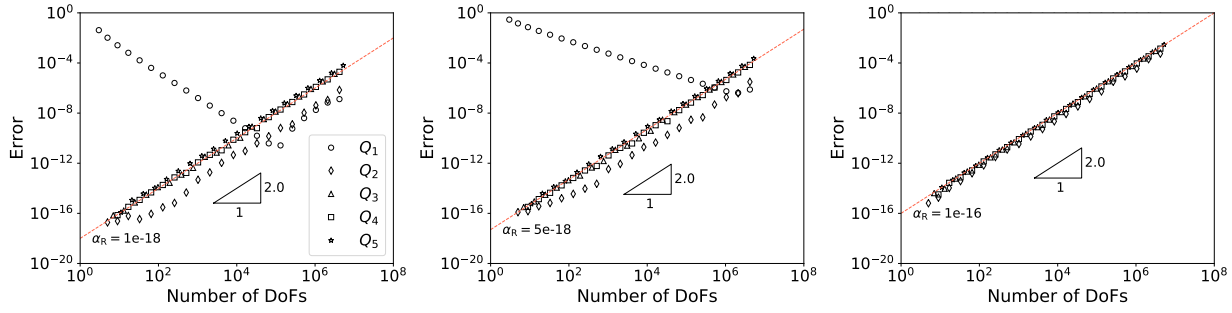
Fig. C.11. Comparison of the distribution of DoFs between the uniform mesh and the regularly distorted mesh of Eq. (D.1a) when $p = 1$.

Appendix D. Error plotting

Appendix D.1. Poisson problems

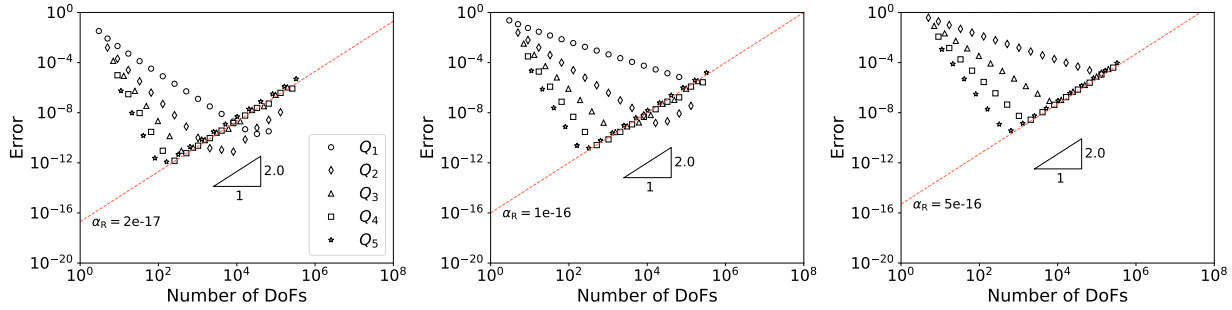
Appendix D.1.1. Results of the uniform mesh

Using the Uniform mesh, the error is shown in Fig. D.12 and Fig. D.13, respectively.



(a) u **(b)** u_x **(c)** u_{xx}

Fig. D.12. Errors using Q_p elements for the 1D Poisson problem with $u = (x - 0.5)^2$ in deal.II.



(a) u **(b)** u_x **(c)** u_{xx}

Fig. D.13. Errors using Q_p elements for the 1D Poisson problem with $u = e^{-(x-0.5)^2}$ in deal.II.

Appendix D.1.2. Influence of mesh distortion

In this section, we investigate the influence of mesh distortion. We use two methods for creating the distorted meshes. The first one is the function GridTools::distort_random(factor, triangulation); the second one is the function GridTools::transform(function, triangulation).

For the former, the argument "factor" is defined by $c_f = \frac{l_{\text{dist}}}{h_{\min}}$, where l_{dist} denotes the maximum absolute distortion length around a vertex, and $h_{\min}$ the minimal adjacent cell length. In this paper, c_f is taken to be 0.4. For the latter, functions are taken to distort the mesh. We use two functions:

$$y = \begin{cases} \frac{y}{2.0}, & \text{if } y < 0.5 \\ \frac{y+1.0}{2.0}, & y > 0.5 \end{cases} \tag{D.1a}$$

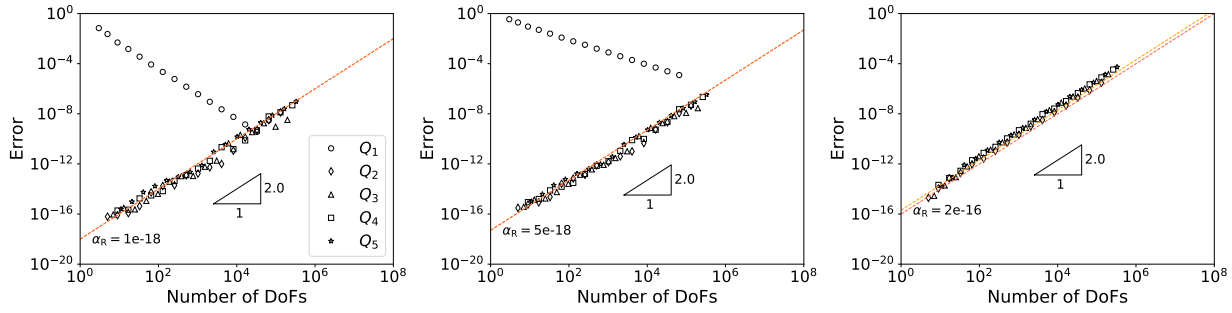
and

$$y = \begin{cases} \frac{y}{1.5-y}, & \text{if } y < 0.5 \\ 1.0 - \frac{1.0-y}{0.5+y}, & y > 0.5 \end{cases} \tag{D.1b}$$

For the above distortion schemes, a comparison of the DoFs distribution between the uniform mesh and

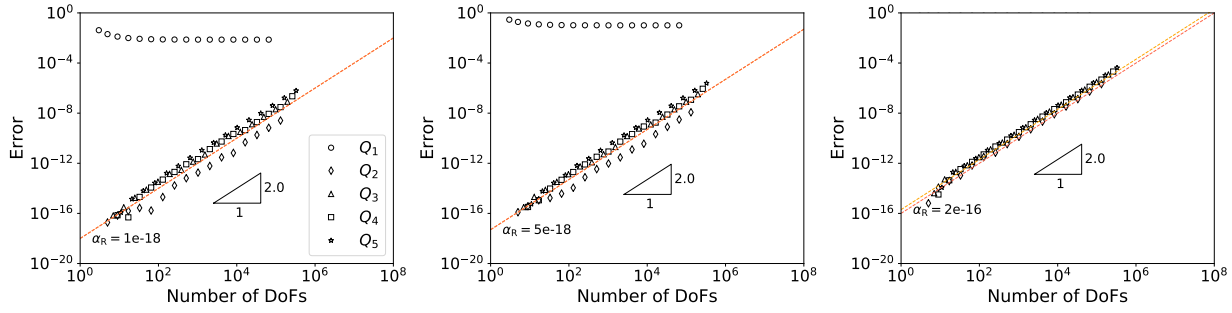
the distorted mesh, including the actual distortion factor, can be found in Fig. C.8–Fig. C.10, respectively, in which $p = 2$, and $R = 2$ and 3 if not stated otherwise.

Using different kinds of mesh, the error for the problem with $u = (x - 0.5)^2$ is shown in Fig. D.14–Fig. D.16; that for the problem with $u = e^{-(x-0.5)^2}$ is shown in Fig. D.17–Fig. D.19, in which the ordering of the figure caption is in blue. We note that for the first kind of regularly distorted mesh, the solution might not converge as expected because of the ill-posedness of the structure of the mesh, e.g. there is only one DoF, which is at $x = 0.5$, between $x = 0.2$ and $x = 0.8$ when $p = 1$, c.f. Fig. C.11.



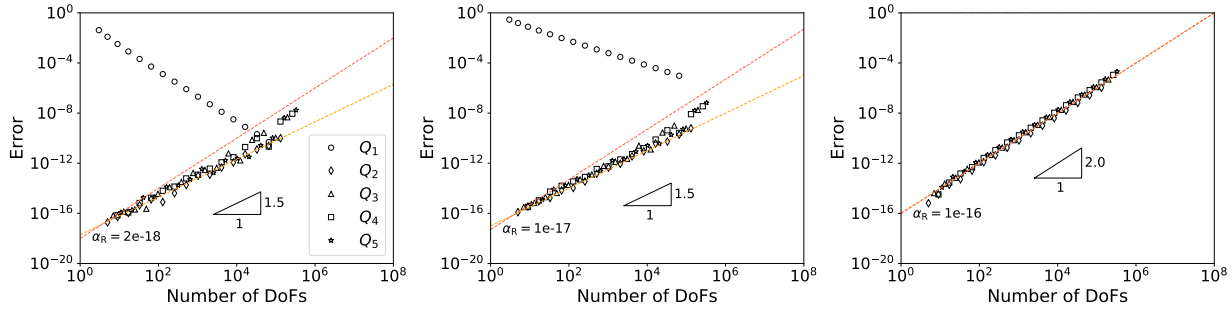
(a) u **(b)** u_x **(c)** u_{xx}

Fig. D.14. Errors using Q_p elements for the 1D Poisson problem with $u = (x - 0.5)^2$ with the randomly distorted mesh in deal.II.



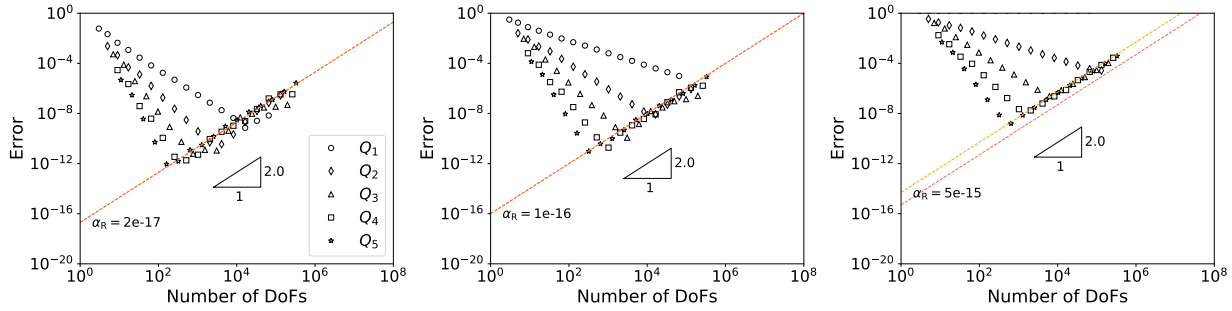
(a) u **(b)** u_x **(c)** u_{xx}

Fig. D.15. Errors using Q_p elements for the 1D Poisson problem with $u = (x - 0.5)^2$ with the regularly distorted mesh of Eq. (D.1a) in deal.II.



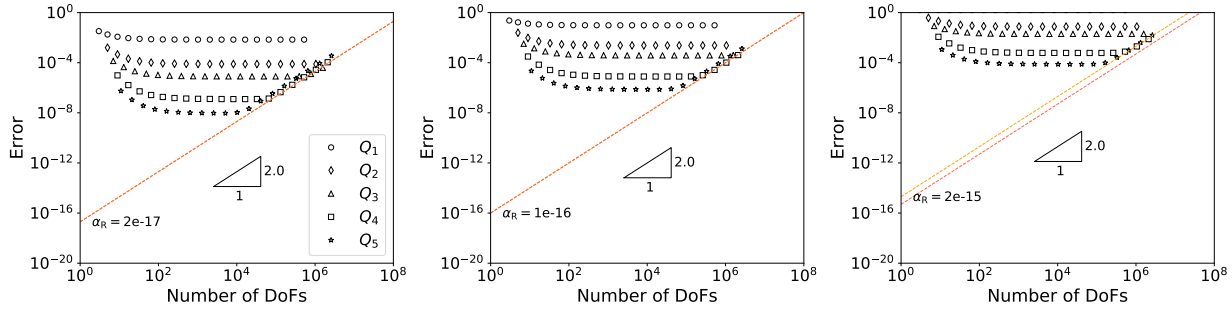
(a) u **(b)** u_x **(c)** u_{xx}

Fig. D.16. Errors using Q_p elements for the 1D Poisson problem with $u = (x - 0.5)^2$ with the regularly distorted mesh of Eq. (D.1b) in deal.II.



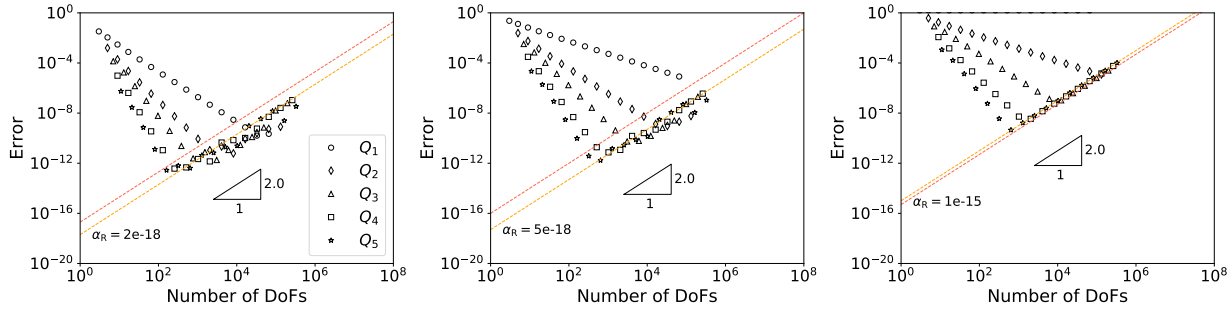
(a) u **(b)** u_x **(c)** u_{xx}

Fig. D.17. Errors using Q_p elements for the 1D Poisson problem with $u = e^{-(x-0.5)^2}$ with the randomly distorted mesh in deal.II.



(a) u **(b)** u_x **(c)** u_{xx}

Fig. D.18. Errors using Q_p elements for the 1D Poisson problem with $u = e^{-(x-0.5)^2}$ with the regularly distorted mesh of Eq. (D.1a) in deal.II.



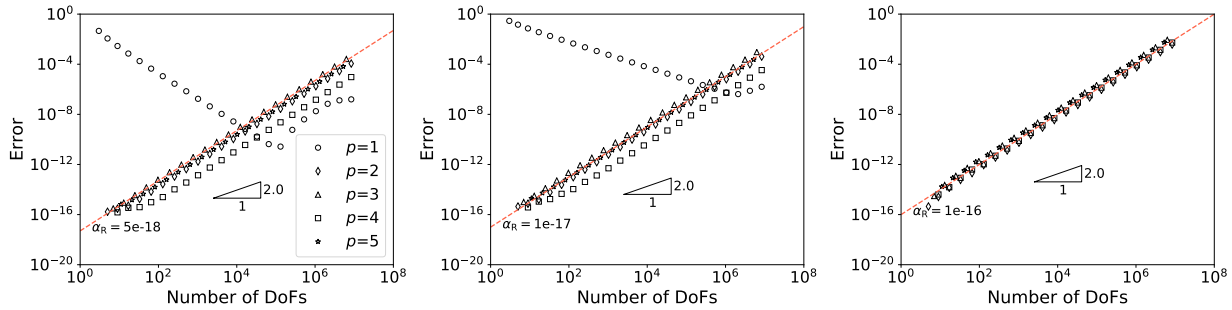
(a) u **(b)** u_x **(c)** u_{xx}

Fig. D.19. Errors using Q_p elements for the 1D Poisson problem with $u = e^{-(x-0.5)^2}$ with the regularly distorted mesh of Eq. (D.1b) in deal.II.

Appendix D.1.3. Influence of FEM packages

MATLAB. Using MATLAB for the problem, the error is shown ...

FEniCS. Using FEniCS for $u = (x - 0.5)^2$, the error is shown in Fig. D.20. Note that, the ordering of the figure caption is in cyan when the problem is solved using FEniCS. We see the error is basically the same with that using deal.II.



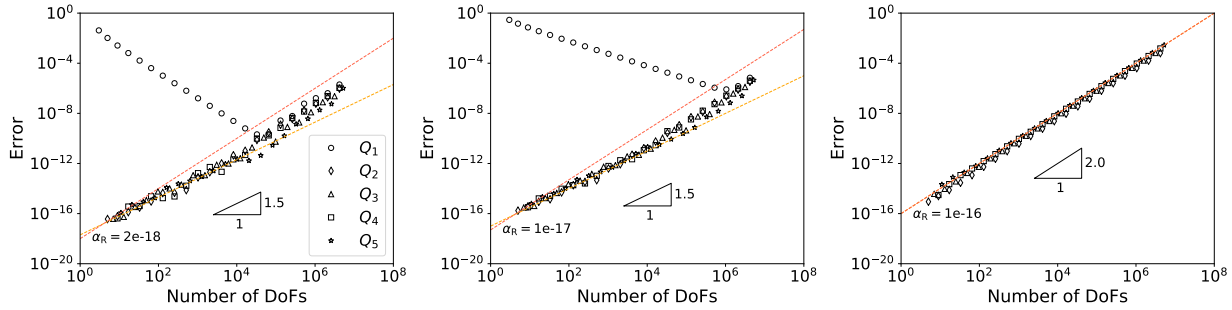
(a) u **(b)** u_x **(c)** u_{xx}

Fig. D.20. Errors using Q_p elements for the 1D Poisson problem with $u = (x - 0.5)^2$ in FEniCS.

Appendix D.2. Diffusion problems

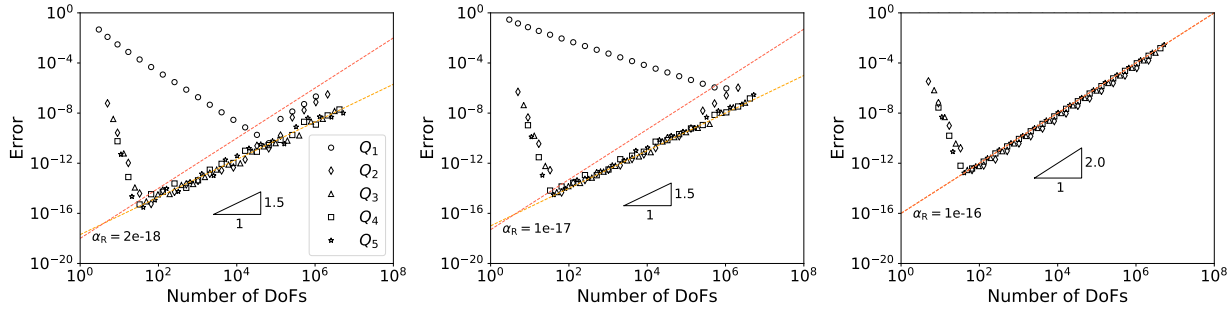
Appendix D.2.1. Results of the uniform mesh

$u = (x - 0.5)^2$. In this paragraph, we show the error for the problem with $u = (x - 0.5)^2$.



(a) u **(b)** u_x **(c)** u_{xx}

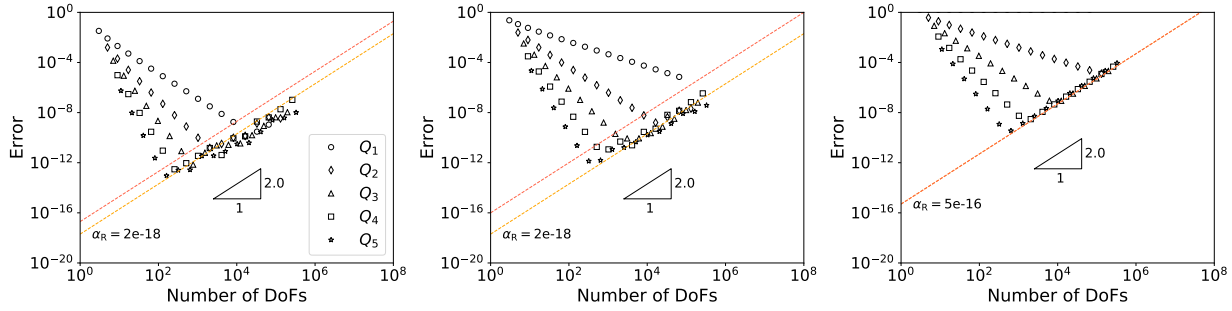
Fig. D.21. Errors using Q_p elements for the 1D diffusion problem with $u = (x - 0.5)^2$ and $D(\mathbf{x}) = 1 + x$ in deal.II.



(a) u **(b)** u_x **(c)** u_{xx}

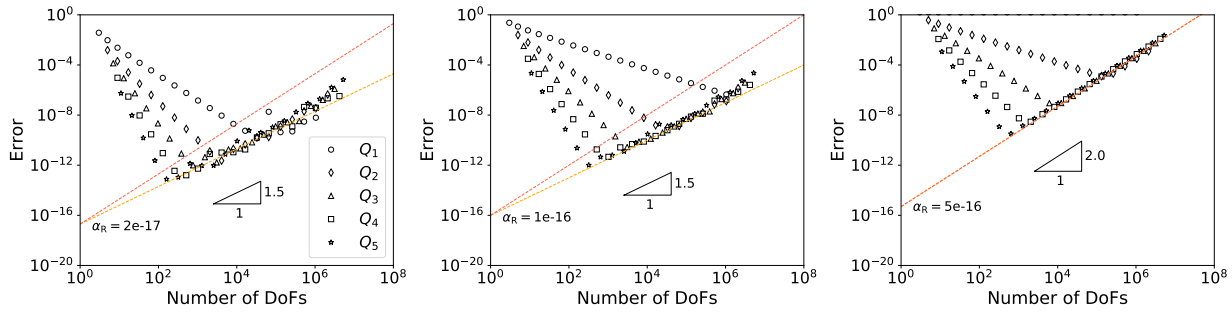
Fig. D.22. Errors using Q_p elements for the 1D diffusion problem with $u = (x - 0.5)^2$ and $D(\mathbf{x}) = e^{-(x-0.5)^2}$ in deal.II.

$u = e^{-(x-0.5)^2}$. In this paragraph, we show the error for the problem with $u = e^{-(x-0.5)^2}$.



(a) u **(b)** u_x **(c)** u_{xx}

Fig. D.23. Errors using Q_p elements for the 1D diffusion problem with $u = e^{-(x-0.5)^2}$ and $D(\mathbf{x}) = 1 + x$ in deal.II.



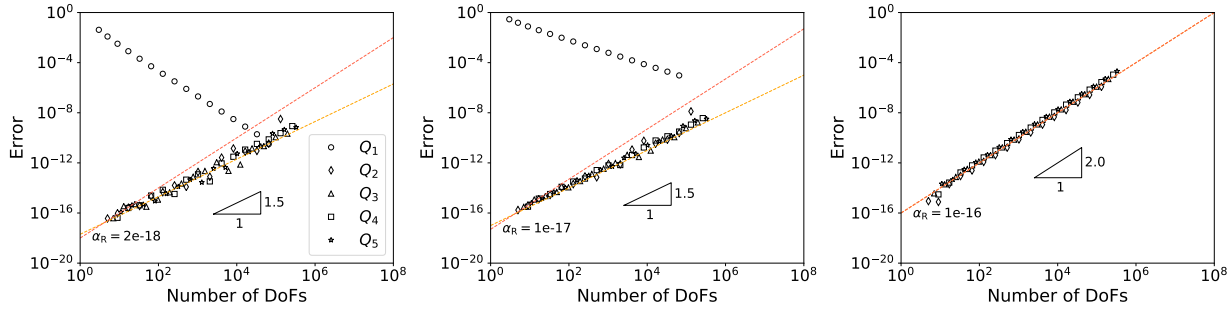
(a) u **(b)** u_x **(c)** u_{xx}

Fig. D.24. Errors using Q_p elements for the 1D diffusion problem with $u = e^{-(x-0.5)^2}$ and $D(\mathbf{x}) = e^{-(x-0.5)^2}$ in deal.II.

Appendix D.2.2. Influence of mesh distortion

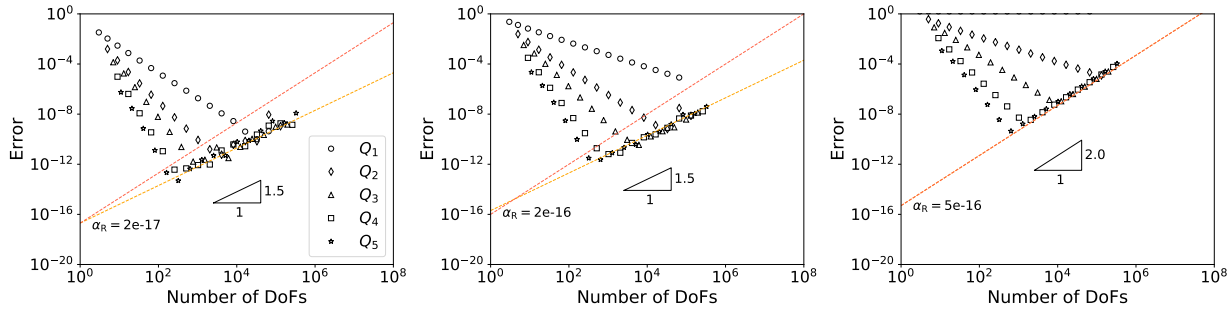
In this section, we investigate the influence of mesh distortion on the diffusion problem.

As can be seen, for u_{xx}, the distorted mesh basically has no influence on the round-off error; for u and u_x, when β_{R} is already 1.5, the mesh distortion has basically no influence on the round-off error, and when β_{R} is 2.0, the mesh distortion would decrease β_{R} to 1.5.



(a) u **(b)** u_x **(c)** u_{xx}

Fig. D.25. Errors using Q_p elements for the 1D diffusion problem with $u = (x - 0.5)^2$ and $D(\mathbf{x}) = 1 + x$ with the regularly distorted mesh of Eq. (D.1b) in deal.II.



(a) u **(b)** u_x **(c)** u_{xx}

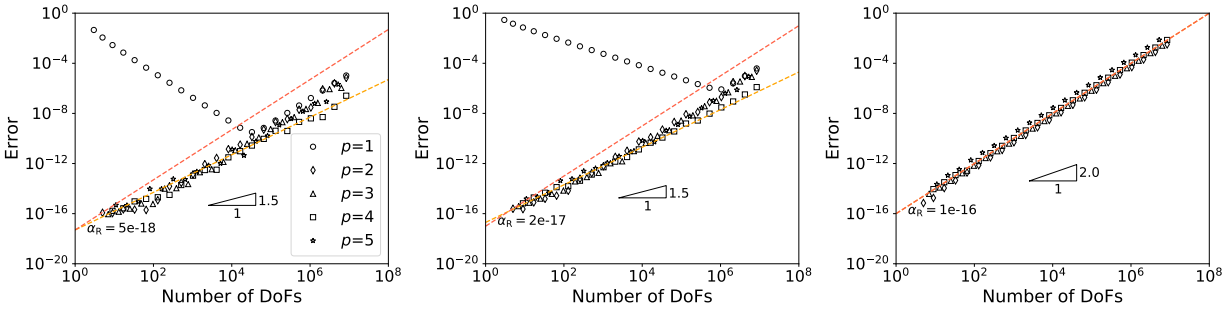
Fig. D.26. Errors using Q_p elements for the 1D diffusion problem with $u = e^{-(x-0.5)^2}$ and $D(\mathbf{x}) = 1 + x$ with the regularly distorted mesh of Eq. (D.1b) in deal.II.

Appendix D.2.3. Influence of FEM packages

Using FEniCS for different diffusion problems in Table D.9, the error is shown in Fig. D.27–Fig. D.29. A summary of β_{R} and α_{R} can be found in Table D.9. As can be seen, the values of α_{R} and β_{R} are basically the same with that using deal.II. However, when $D(\mathbf{x}) \notin Q_2$, the truncation error may decrease not as fast as that when using deal.II, c.f. Fig. D.28 and Fig. D.29.

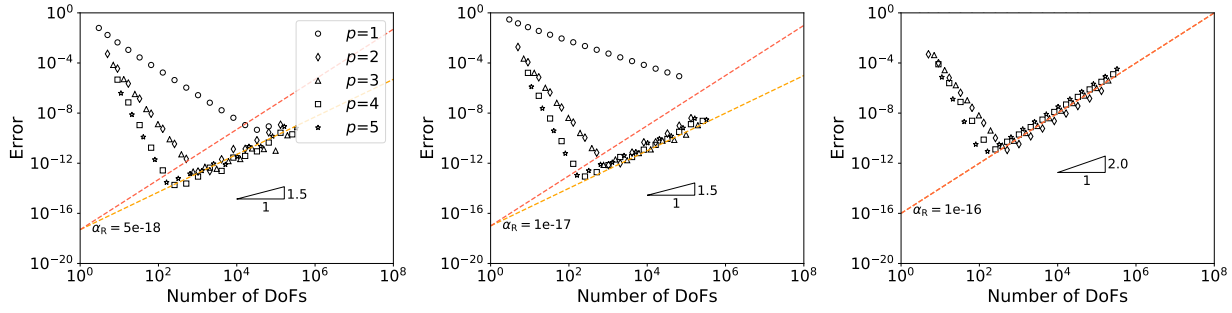
Table D.9 Various diffusion problems and the resulting β_{R} and α_{R} in FEniCS.

	$u=(x-0.5)^2$		$u=e^{-(x-0.5)^2}$	
	β_{R}	α_{R}	β_{R}	α_{R}
$D(\mathbf{x})=1+x$	(1.5→2.0) (1.5→2.0) 2.0	5e-18 2e-17 1e-16		
$D(\mathbf{x})=e^{-(x-0.5)^2}$	1.5 1.5 2.0	5e-18 1e-17 1e-16		
$D(\mathbf{x})=0.5+\cos^2 x$	1.5 1.5 2.0	5e-18 1e-17 1e-16		



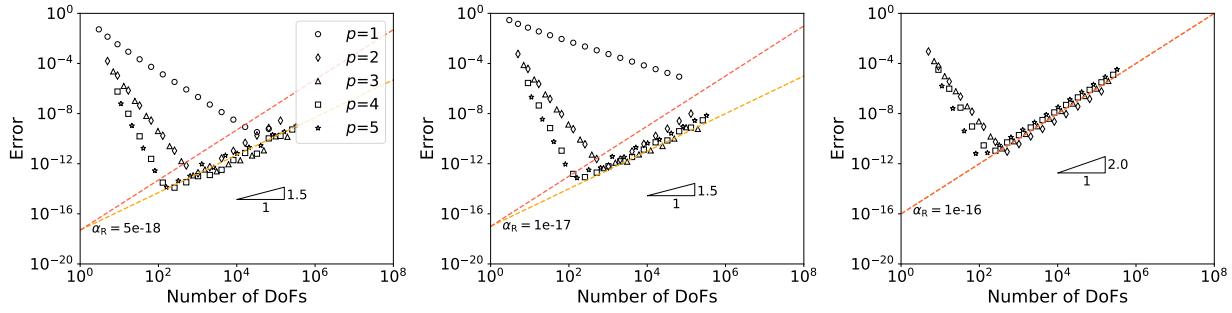
(a) u **(b)** u_x **(c)** u_{xx}

Fig. D.27. Errors using Q_p elements for the 1D diffusion problem with $u=(x-0.5)^2$ and $D(\mathbf{x})=1+x$ in FEniCS.



(a) u **(b)** u_x **(c)** u_{xx}

Fig. D.28. Errors using Q_p elements for the 1D diffusion problem with $u = (x - 0.5)^2$ and $D(\mathbf{x}) = e^{-(x-0.5)^2}$ in FEniCS.

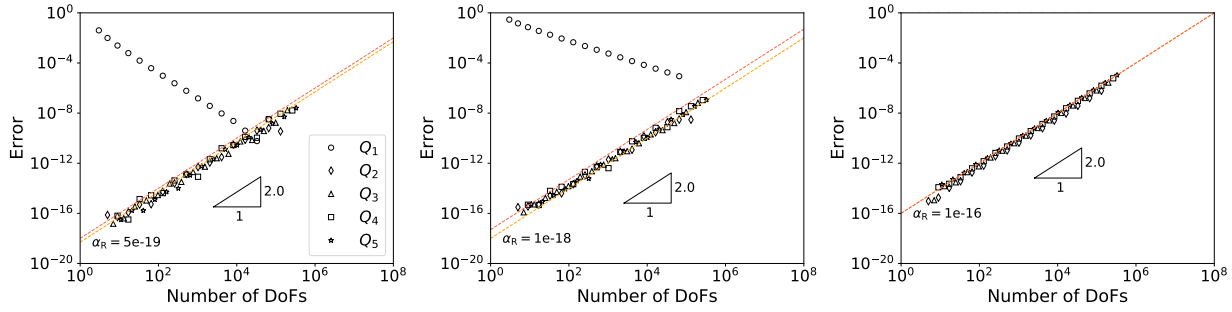


(a) u **(b)** u_x **(c)** u_{xx}

Fig. D.29. Errors using Q_p elements for the 1D diffusion problem with $u = (x - 0.5)^2$ and $D(\mathbf{x}) = 0.5 + \cos^2 x$ in FEniCS.

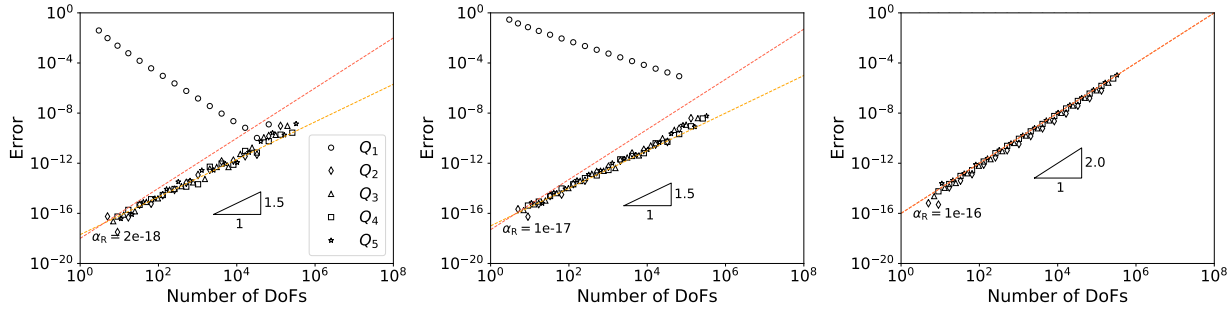
Appendix D.3. Helmholtz problems

$u = (x - 0.5)^2$. In this paragraph, we show the error for the problem with $u = (x - 0.5)^2$.



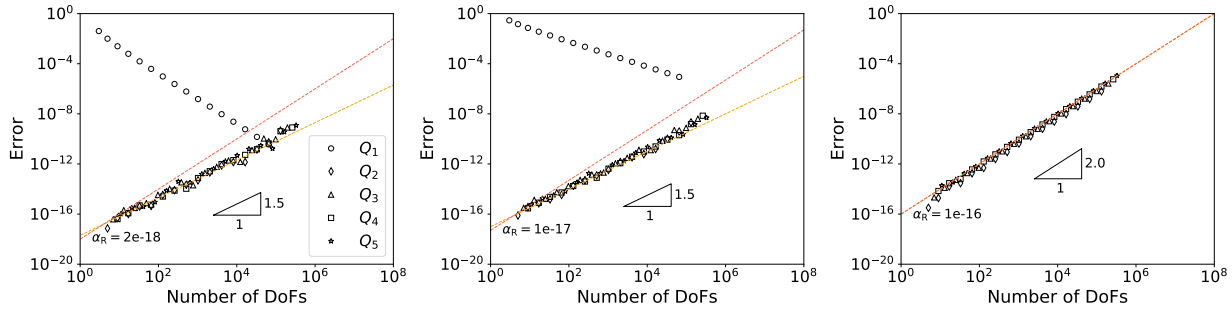
(a) u **(b)** u_x **(c)** u_{xx}

Fig. D.30. Errors using Q_p elements for the 1D Helmholtz problem with $u = (x-0.5)^2$, $D(\mathbf{x}) = 1+x$ and $r(\mathbf{x}) = 1$ in deal.II.



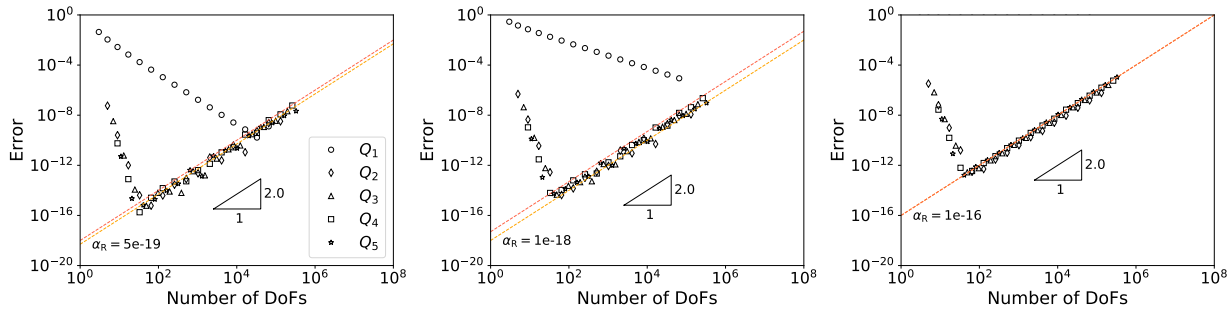
(a) u **(b)** u_x **(c)** u_{xx}

Fig. D.31. Errors using Q_p elements for the 1D Helmholtz problem with $u = (x - 0.5)^2$, $D(\mathbf{x}) = 1 + x$ and $r(\mathbf{x}) = 1 + x$ in deal.II.



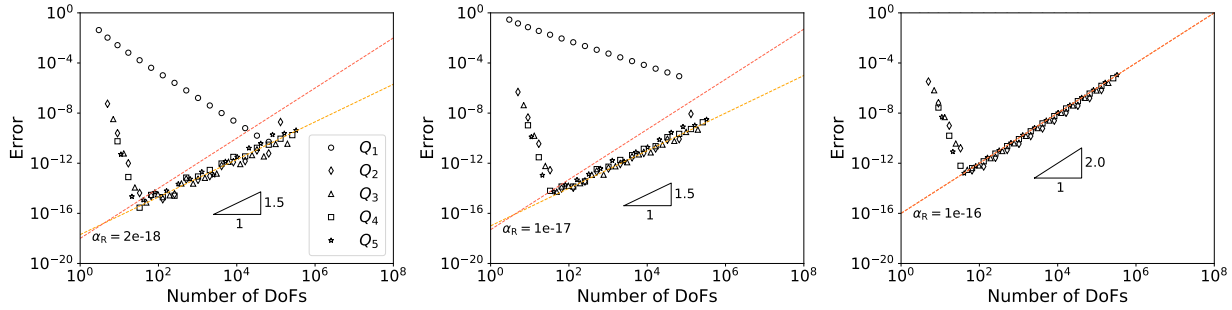
(a) u **(b)** u_x **(c)** u_{xx}

Fig. D.32. Errors using Q_p elements for the 1D Helmholtz problem with $u = (x - 0.5)^2$, $D(\mathbf{x}) = 1 + x$ and $r(\mathbf{x}) = e^{-(x-0.5)^2}$ in deal.II.



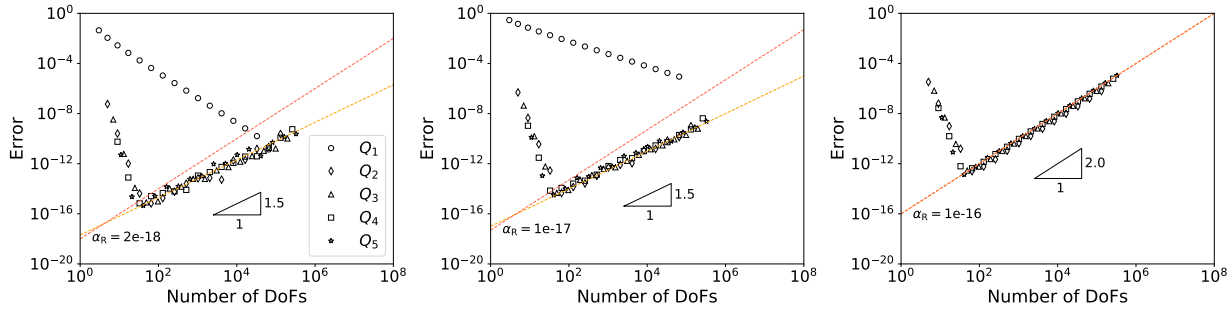
(a) u **(b)** u_x **(c)** u_{xx}

Fig. D.33. Errors using Q_p elements for the 1D Helmholtz problem with $u = (x - 0.5)^2$, $D(\mathbf{x}) = e^{-(x-0.5)^2}$ and $r(\mathbf{x}) = 1$ in deal.II.



(a) u **(b)** u_x **(c)** u_{xx}

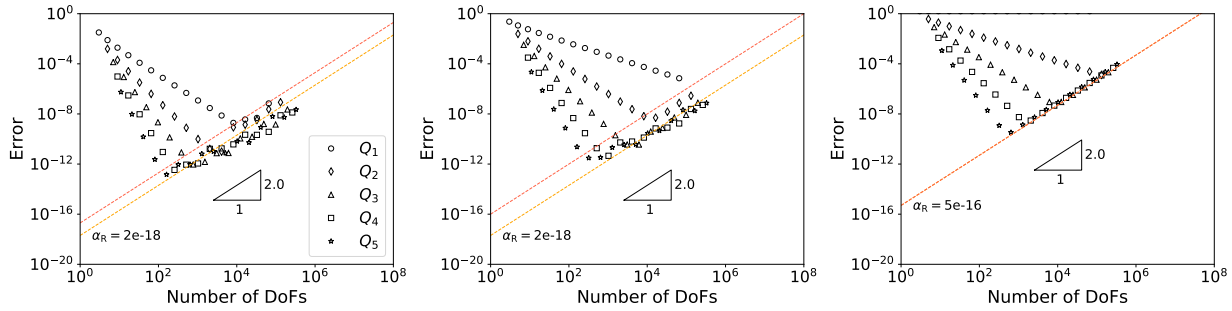
Fig. D.34. Errors using Q_p elements for the 1D Helmholtz problem with $u = (x-0.5)^2$, $D(\mathbf{x}) = e^{-(x-0.5)^2}$ and $r(\mathbf{x}) = 1+x$ in deal.II.



(a) u **(b)** u_x **(c)** u_{xx}

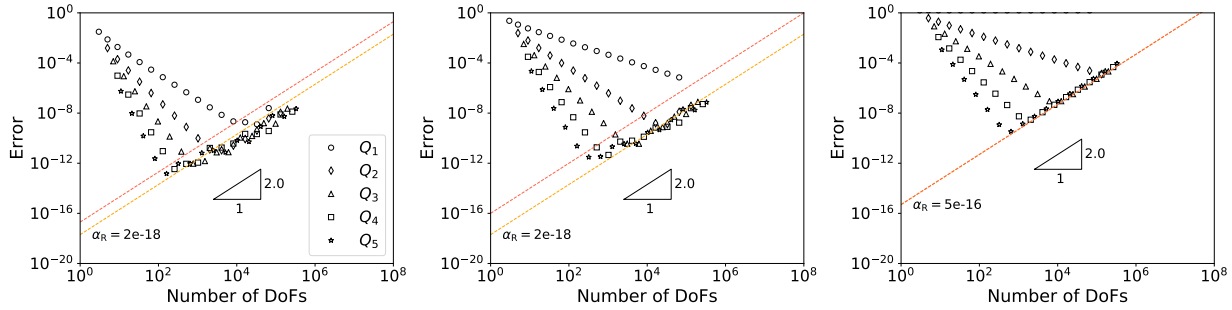
Fig. D.35. Errors using Q_p elements for the 1D Helmholtz problem with $u = (x-0.5)^2$, $D(\mathbf{x}) = e^{-(x-0.5)^2}$ and $r(\mathbf{x}) = e^{-(x-0.5)^2}$ in deal.II.

$u = e^{-(x-0.5)^2}$. In this paragraph, we show the error for the problem with $u = e^{-(x-0.5)^2}$.



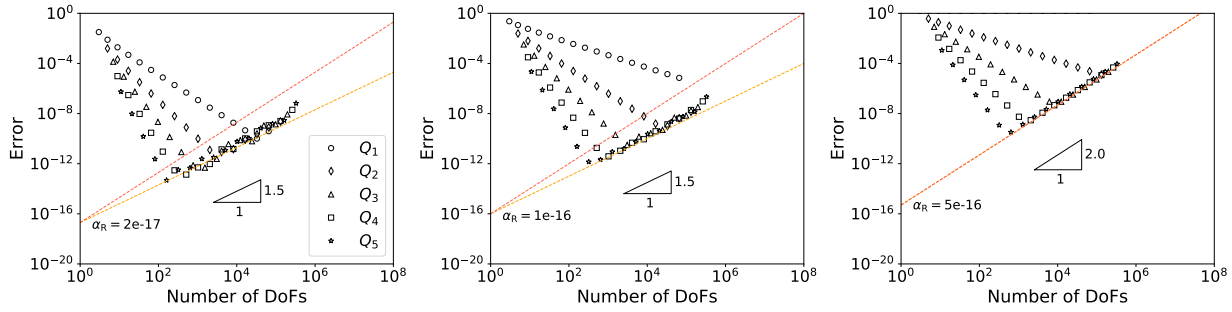
(a) u **(b)** u_x **(c)** u_{xx}

Fig. D.36. Errors using Q_p elements for the 1D Helmholtz problem with $u = e^{-(x-0.5)^2}$, $D(\mathbf{x}) = 1+x$ and $r(\mathbf{x}) = 1$ in deal.II.



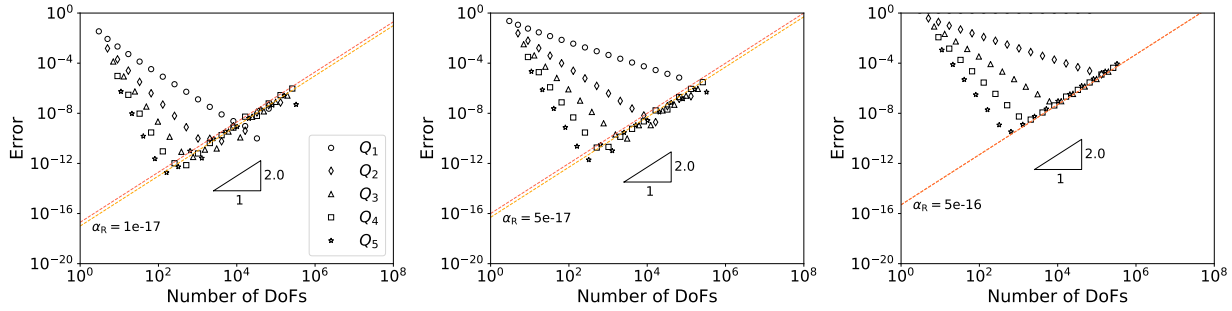
(a) u **(b)** u_x **(c)** u_{xx}

Fig. D.37. Errors using Q_p elements for the 1D Helmholtz problem with $u = e^{-(x-0.5)^2}$, $D(\mathbf{x}) = 1 + x$ and $r(\mathbf{x}) = 1 + x$ in deal.II.



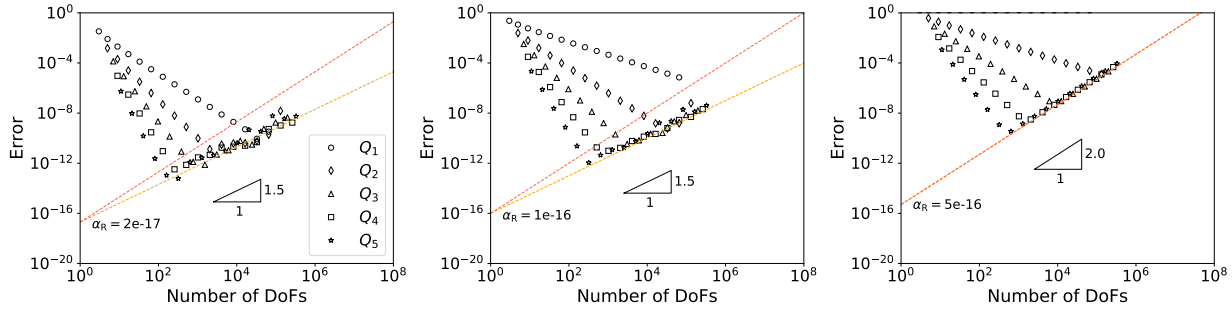
(a) u **(b)** u_x **(c)** u_{xx}

Fig. D.38. Errors using Q_p elements for the 1D Helmholtz problem with $u = e^{-(x-0.5)^2}$, $D(\mathbf{x}) = 1 + x$ and $r(\mathbf{x}) = e^{-(x-0.5)^2}$ in deal.II.



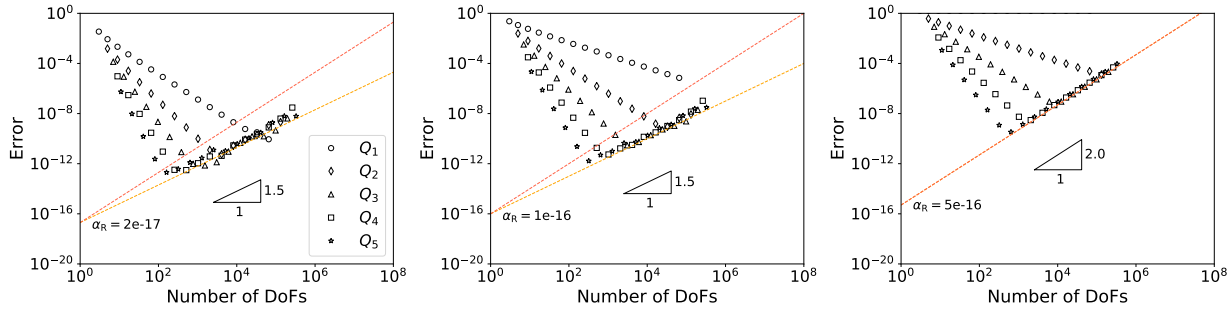
(a) u **(b)** u_x **(c)** u_{xx}

Fig. D.39. Errors using Q_p elements for the 1D Helmholtz problem with $u = e^{-(x-0.5)^2}$, $D(\mathbf{x}) = e^{-(x-0.5)^2}$ and $r(\mathbf{x}) = 1$ in deal.II.



(a) u **(b)** u_x **(c)** u_{xx}

Fig. D.40. Errors using Q_p elements for the 1D Helmholtz problem with $u = e^{-(x-0.5)^2}$, $D(\mathbf{x}) = e^{-(x-0.5)^2}$ and $r(\mathbf{x}) = 1+x$ in deal.II.



(a) u **(b)** u_x **(c)** u_{xx}

Fig. D.41. Errors using Q_p elements for the 1D Helmholtz problem with $u = e^{-(x-0.5)^2}$, $D(\mathbf{x}) = e^{-(x-0.5)^2}$ and $r(\mathbf{x}) = e^{-(x-0.5)^2}$ in deal.II.

References

[1] Jie Liu, Matthias Möller, and Henk M Schuttelaars. Balancing truncation and round-off errors in fem: One-dimensional analysis. *Journal of Computational and Applied Mathematics*, 386:113219, 2021.

[2] Wolfgang Bangerth, Ralf Hartmann, and Guido Kanschat. deal. ii—a general-purpose object-oriented finite element library. *ACM Transactions on Mathematical Software (TOMS)*, 33(4):24, 2007.

[3] Martin Alnæs, Jan Blechta, Johan Hake, August Johansson, Benjamin Kehlet, Anders Logg, Chris Richardson, Johannes Ring, Marie E Rognes, and Garth N Wells. The fenics project version 1.5. *Archive of Numerical Software*, 3(100), 2015.

[4] Daniele Boffi, Franco Brezzi, Michel Fortin, et al. *Mixed finite element methods and applications*, volume 44. Springer, 2013.

[5] Gopal Menon. What is a scalar function?, 2018. [Online; accessed 14-January-2021].

[6] Marie E Rognes, Robert C Kirby, and Anders Logg. Efficient assembly of h(div) and h(curl) conforming finite elements. *SIAM Journal on Scientific Computing*, 31(6):4130–4151, 2010.

[7] Seymour Lipschutz and Marc Lipson. *Linear Algebra: Schaum's Outlines*. McGraw-Hill, 2009.

[8] Zhangxin Chen. *Finite element methods and their applications*. Springer Science & Business Media, 2005.

[9] Mohit Kumar, Henk M Schuttelaars, and Pieter C Roos. Three-dimensional semi-idealized model for estuarine turbidity maxima in tidally dominated estuaries. *Ocean Modelling*, 113:1–21, 2017.

[10] Raviart thomas elements in deal.ii, 2020. [Online; accessed 7-Jan-2020].

[11] Dgp elements in deal.ii, 2020. [Online; accessed 17-November-2020].

[12] Dan Zuras, Mike Cowlishaw, Alex Aiken, Matthew Applegate, David Bailey, Steve Bass, Dileep Bhandarkar, Mahesh Bhat, David Bindel, Sylvie Boldo, et al. IEEE standard for floating-point arithmetic. *IEEE Std 754-2008*, pages 1–70, 2008.

[13] Timothy A Davis. Algorithm 832: UMFPACK V4.3 – an unsymmetric-pattern multifrontal method. *ACM Transactions on Mathematical Software (TOMS)*, 30(2):196–199, 2004.

[14] Stan Z. Li and Anil Jain, editors. *L2 norm*, pages 883–883. Springer US, Boston, MA, 2009.

[15] Olof Runborg. *Verifying Numerical Convergence Rates*. KTH Computer Science and Communication, 2012.

[16] Mark S Gockenbach. *Understanding and implementing the finite element method*, volume 97. Siam, 2006.

[17] John Charles Butcher. *Numerical methods for ordinary differential equations*. John Wiley & Sons, 2016.

[18] Kambiz Salari and Patrick Knupp. Code verification by the method of manufactured solutions. Technical report, Sandia National Labs., Albuquerque, NM (US); Sandia National Labs ..., 2000.

[19] Patrick J Roache. Code verification by the method of manufactured solutions. *J. Fluids Eng.*, 124(1):4–10, 2002.

[20] Michael H Gfrerer and Martin Schanz. Code verification examples based on the method of manufactured solutions for kirchhoff–love and reissner–mindlin shell analysis. *Engineering with Computers*, 34(4):775–785, 2018.