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

Jie Liu[a,*], Matthias Möller[a], Henk M. Schuttelaars[a]

[a] *Delft Institute of Applied Mathematics*
Delft University of Technology
Van Mourik Broekmanweg 6, 2628 XE Delft, The Netherlands

Abstract

In finite element methods (FEMs), the accuracy of the solution cannot increase indefinitely since the round-off error related to limited computer precision increases when the number of degrees of freedom (DoFs) is large enough. Because a priori information of the highest attainable accuracy is of great interest, we construct an innovative method to obtain the highest attainable accuracy. In this method, the truncation error is extrapolated when it converges at the analytical rate, and the bound of the round-off error follows from a generically valid error estimate, obtained and validated through extensive numerical experiments. The highest attainable accuracy is obtained by minimizing the sum of these two types of errors. We validate this method using a one-dimensional Helmholtz equation in space. It shows that the highest attainable accuracy can be accurately predicted, and the CPU time required is much less compared with that using the successive h-refinement.

Keywords: Finite Element Method (FEM), error estimation, optimal number of degrees of freedom, hp-refinement strategy.

1. Introduction

Many problems in engineering sciences and industry are modelled mathematically by initial-boundary value problems comprising systems of coupled, nonlinear partial and/or ordinary differential equations. These problems often consider complex geometries, with initial and/or boundary conditions that depend on measured data [1]. In some applications, not only the solution, but also its derivatives are of interest [1, 2]. For many problems of practical interest, analytical or semi-analytical solutions are not available, and hence one has to resort to numerical solution methods, such as the finite difference, finite volume, and finite element methods. The latter will be adopted throughout this paper and applied to one-dimensional boundary value problems.

*Corresponding author
Email addresses: j.liu-5@tudelft.nl (Jie Liu), m.moller@tudelft.nl (Matthias Möller), h.m.schuttelaars@tudelft.nl (Henk M. Schuttelaars)

The accuracy of the numerically obtained solution is influenced by many sources of errors [3]: firstly, modelling errors in the set-up of the models, such as the simplification of realistic domains and governing equations and the approximation of initial and boundary conditions; next, truncation errors due to the discretization of the computational domain and the use of basis functions for the function spaces defined on it; then, round-off errors due to the adoption of finite-precision computer arithmetics, rather than exact arithmetics; finally, iteration errors resulting from the artificially controlled tolerance of iterative solvers.

One tacitly assumes that most errors are well-balanced and/or negligibly small. In this paper, the focus is on the truncation error (E_{T}) and the round-off (E_{R}), by considering idealized problems, which does not introduce modelling errors, and using the direct solver, which avoids the introduction of iterative errors. In particular, the round-off error is often ignored based on the argument that it will be 'sufficiently small' if just IEEE-754 double-precision floating-point arithmetics [13] are adopted. Therefore, to improve the accuracy, i.e. to decrease E_{T}, one often reduces the mesh width (h-refinement), increase the approximation order (p-refinement), or apply both strategies simultaneously (hp-refinement) [4, 5].

The common characteristic of these methods is to increase the number of degrees of freedom (DoFs). However, E_{R} increases with the number of DoFs, and dominates the total error if more and more DoFs are employed [6, 7]. While a typically impractically large number of DoFs is required for E_{R} to dominate the total error if low(est)-order approximations are used, the number can be very small if high-order approximations are adopted, which are nowadays becoming more and more popular. This shift to higher order approximations makes the results more prone to be dominated by round-off errors. Despite this alarming observation, to the authors' best knowledge, only very few publications address the impact of accumulated round-off errors on the overall accuracy of the final solution or take them into account explicitly in the error-estimation procedure. The general rule of thumb is still to perform as many h-refinements as possible considering the available computer hardware.

The aim of this paper is to systematically analyze the influence of the round-off error on the total error when using h-refinements for particular approximation order p's. Not only the solution but also its first and second derivatives are investigated for one-dimensional second order model problems, assuming the second derivative exists in the weak sense [8]. Both the standard finite element method (FEM) and the mixed FEM [9] with multiple p's are analyzed. Furthermore, the following factors are considered: types of boundary conditions and methods of implementing them, choices and configurations of the linear system solver, orders of magnitude of the variables and coefficients. Based on the statistics on the evolution of the round-off error, we propose an algorithm to predict the best accuracy E_{min} that occurs when the sum of E_{T} and E_{R} is the smallest, and the corresponding number of DoFs (N_{opt}).

The paper is organized as follows. The model problem, finite element formulation and numerical implementation are described in Section 2. The approach to predicting E_{min} is illustrated in Section 3. The statistics on the evolution of the round-off error are given in Section 4. The algorithm for realizing the

approach is put forward in Section 5, followed by its validation by a Helmholtz problem in Section 6. The conclusions are drawn in Section 7.

2. Model problem, finite element formulation and numerical implementation

2.1. Model problem

Consider the following one-dimensional second-order differential equation:

$$-\left(d(x)u_x\right)_x + r(x)u(x) = f(x), \qquad x \in I = [0,1], \tag{1}$$

with u denoting the unknown variable, which can either be real or complex, $f(x) \in L^2(I)$ a prescribed right-hand side, and $d(x)$ and $r(x)$ continuous coefficient functions. By choosing $d(x) = 1$ and $r(x) = 0$, Eq. (1) reduces to the Poisson equation; for $d(x) > 0$ and not constant, the diffusion equation is found when $r(x) = 0$, and the Helmholtz equation [10] is found when $r(x) \neq 0$. The boundary conditions are $u(x) = g(x)$ on Γ_D and $d(x)u_x = h(x)$ on Γ_N, where Γ_D and Γ_N are the boundaries where Dirichlet and Neumann boundary conditions are imposed, respectively.

2.2. Finite element formulation

For convenience, we introduce two inner products [11]:

$$\langle f_1,\, f_2\rangle = \int_I f_1(x) f_2(x)\, dx, \tag{2a}$$

$$\langle f_1,\, f_2\rangle_\Gamma = f_1(x_0) f_2(x_0). \tag{2b}$$

where $f_1(x)$ and $f_2(x)$ are continuous functions defined on the unit interval I, Γ denotes the boundary of I, and x_0 denotes the value of x on Γ.

2.2.1. 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:

Weak form 1

Find $u \in H^1_D(I)$ such that:

$$\langle \eta_x,\, du_x\rangle + \langle \eta,\, ru\rangle = \langle \eta,\, f\rangle + \langle \eta,\, hn\rangle_{\Gamma_N} \qquad \forall \eta \in H^1_{D0}(I), \tag{3}$$

with

$$H^1_D(I) = \{t \mid t \in H^1(I),\ t = g \text{ on } \Gamma_D\},$$

$$H^1_{D0}(I) = \{t \mid t \in H^1(I),\ t = 0 \text{ on } \Gamma_D\}.$$

Imposing the Dirichlet boundary conditions in the weak sense [12], the weak form reads:

Weak form 2

Find $u \in H^1(I)$ such that:

$$\langle \eta_x,\, du_x \rangle + \langle \eta,\, ru \rangle - \langle \eta,\, du_x n \rangle_{\Gamma_D} + \langle \eta_x,\, un \rangle_{\Gamma_D} - \langle \eta,\, \rho un \rangle_{\Gamma_D} = \langle \eta,\, f \rangle + \langle \eta,\, hn \rangle_{\Gamma_N} + \langle \eta_x,\, gn \rangle_{\Gamma_D} - \langle \eta,\, \rho gn \rangle_{\Gamma_D} \qquad \forall \eta \in H^1(I), \tag{4}$$

where ρ is a positive value that serves as the penalty parameter.

In both forms, η denotes the test function, n is equal to 1 at $x = 1$, and -1 at $x = 0$; 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:

$$u \approx u_h^{(p)} = \sum_{i=1}^{m} u_i \varphi_i^{(p)}. \tag{5}$$

Here, p is the element degree, m is the number of DoFs, which equals $p \times t + 1$, with t denoting the total number of grid cells; 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. $\varphi_i^{(p)}(x_j) = \delta_{ij}$, with x_j denoting the support point. This type of element will be referred to as P_p. Taking η equal to $\varphi_k^{(p)}$, $k = 1, 2, \ldots, m$, the resulting linear system of equations reads

$$AU = F, \tag{6}$$

where A is the stiffness matrix, F the right-hand side and U the discretized u, i.e. the vector of the coefficients u_i.

2.2.2. The mixed FEM

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

Weak form 3

Find $v \in H_N^1(I)$ and $u \in L^2(I)$ such that:

$$\langle w,\, d^{-1}v \rangle - \langle w_x,\, u \rangle = -\langle w,\, gn \rangle_{\Gamma_D} \qquad \forall w \in H_{N0}^1(I), \tag{7a}$$

$$-\langle q,\, v_x \rangle - \langle q,\, ru \rangle = -\langle q,\, f \rangle \qquad \forall q \in L^2(I), \tag{7b}$$

with

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

$$H_{N0}^1(I) = \{t \mid t \in H^1(I),\ t = 0 \text{ on } \Gamma_N\}.$$

In this form, w and q denote the test functions of v and u, respectively, and n has the same value as before. We approximate v and u by:

$$v \approx v_h^{(p)} = \sum_{i=1}^{n} v_i \varphi_i^{(p)}, \tag{8a}$$

$$u \approx u_h^{(p-1)} = \sum_{j=1}^{p} u_{sj} \psi_j^{(p-1)} \text{ in cell } s, \text{ for } s = 1, 2, \ldots, t, \tag{8b}$$

where n is the number of DoFs for $v_h^{(p)}$, which is equal to $p \times t + 1$, v_i's are the values of $v_h^{(p)}$ at the DoFs, and $\varphi_i^{(p)}$'s are of the same type of basis functions used in Eq. (5); u_{sj} are the values of $u_h^{(p-1)}$ at the DoFs, $\psi_j^{(p-1)}$'s are discontinuous Lagrange basis functions, which means two independent u_{sj}'s have been assigned at the cell interfaces. This pair of elements will be referred to as $P_p/P_{p-1}^{\text{disc}}$. Replacing w and q by $\varphi_k^{(p)}$, $k = 1, 2, \ldots, p \times t + 1$, and $\psi_e^{(p-1)}$, $e = 1, 2, \ldots, p \times t$, respectively, the resulting coupled linear system of equations that has to be solved reads:

$$\begin{bmatrix} M & B \\ B^\top & 0 \end{bmatrix} \begin{bmatrix} V \\ U \end{bmatrix} = \begin{bmatrix} G \\ H \end{bmatrix}, \tag{9}$$

where the mass matrix M, discrete gradient operator B, and its transpose, the discrete divergence operator $B^\top$, comprise the left-hand side; G and H are the components of the right-hand side; V and U are the discretized v and u, i.e. the vectors of the coefficients v_i and u_{sj}, respectively.

For the sake of readability, we will drop the superscript (p) or $(p\text{-}1)$ whenever the approximation order is clear from the context.

2.3. Numerical implementation

2.3.1. Solution technique

All results are computed in IEEE-754 double precision [13] using the deal.II finite element code [14]. Unless stated otherwise, the computational mesh is obtained by globally refining a single element that covers the interval I, and the Dirichlet boundary conditions are imposed strongly. The former means that, when the problem is real valued, the number of DoFs equals $2^R \times p + 1$ using the standard FEM and $2 \times 2^R \times p + 1$ using the mixed FEM, at the Rth refinement; when the problem is complex valued, the above numbers double 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 [15], 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 the iterative solvers. The derivatives of the

numerical solution, which are $u_{h,x}$ and $u_{h,xx}$ in the standard FEM and only $v_{h,x}$ in the mixed FEM, 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. Note that, each differentiation decreases the element degree by one.

2.3.2. Error estimation

For the numerical results var_h, where var can be u, u_x and u_{xx} of the standard FEM, and u, v and v_x of the mixed FEM, the error measured in the L_2 norm is used. It is defined as

$$E_h = \|var_h - var_{\text{exc}}\|_2 \tag{10a}$$

when the exact solution var_{exc} is available, or [16]

$$\widetilde{E_h} = \|var_h - var_{h/2}\|_2 \tag{10b}$$

otherwise, where $var_{h/2}$ is the numerical solution computed on a mesh with grid size $h/2$.

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 discretization error converges at a fixed rate theoretically, of which the value is one order higher than the approximation order [5]. In practice, the convergence rate in the numerical experiments can be calculated from either

$$Q = \log_2 \left(\frac{E_h}{E_{h/2}} \right) \tag{11a}$$

using Eq. (10a), or

$$\widetilde{Q} = \log_2 \left(\frac{\widetilde{E_h}}{\widetilde{E_{h/2}}} \right) \tag{11b}$$

using Eq. (10b).

3. Approach to predicting 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 [17]. When N_h is relatively small ($N_h < N_c$), E_h does not decrease at the aforementioned theoretical order of convergence, and only when N_h is large enough ($N_c \leqslant N_h < N_{\text{opt}}$) this order of convergence is attained. The transition from the first phase, denoted by the black circles, to the second phase, denoted by the green circles, is usually fast [17]. E_h in both phases is controlled by E_{T}, and in the second phase E_h can be represented by

$$E_h \approx E_{\text{T}} = \alpha_{\text{T}} {N_h}^{-\beta_{\text{T}}}. \tag{12}$$

Here α_{T} is the offset, and β_{T} is the slope of the line approximating E_h and equals the theoretical order of convergence, see Appendix B for the proof. Note that, α_{T} can be inverted by using

$$\alpha_{\mathrm{T}} = E_{\mathrm{c}}/{N_{\mathrm{c}}}^{-\beta_{\mathrm{T}}}, \tag{13}$$

at the beginning of the second phase, where E_{c} is the corresponding E_h.

When N_h is increased too much ($N_h \geqslant N_{\mathrm{opt}}$), E_{R} starts to dominate and E_h increases, see the orange circles. At this phase, the slope of the line approximating E_h, denoted by β_{R}, tends to be fixed [7, 18]. β_{R} and the associated offset, denoted by α_{R}, are investigated in detail in Section 4. As will be shown in this section, α_{R} and β_{R} are fixed constants, which allows us to estimate E_h as

$$E_h \approx E_{\mathrm{R}} = \alpha_{\mathrm{R}} {N_h}^{\beta_{\mathrm{R}}}. \tag{14}$$

In summary, the evolution of E_h is described in Table 1.

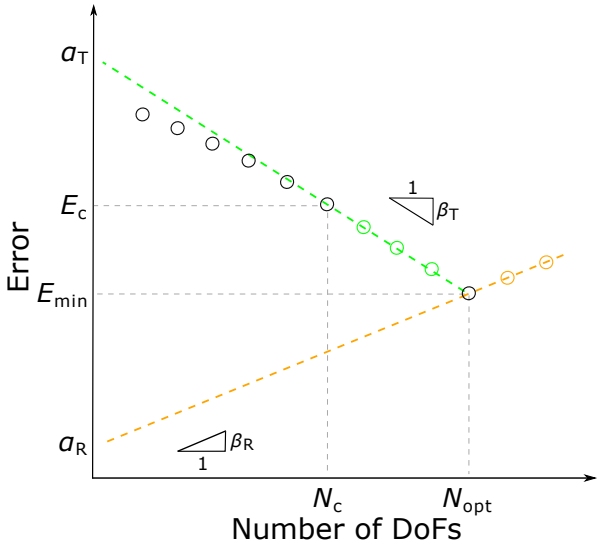


Fig. 1. Conceptual sketch of the error evolution against the number of DoFs.

Table 1 Description of the evolution of E_h.

	1. $N_h < N_{\mathrm{c}}$	2. $N_{\mathrm{c}} \leqslant N_h < N_{\mathrm{opt}}$	3. $N_{\mathrm{opt}} \leqslant N_h$
Feature	Decreasing but not converging at slope β_{T}	Decreasing and converging at slope β_{T}, with the offset α_{T}	Increasing and converging at slope β_{R}, with the offset α_{R}
Dominant error	Truncation error		Round-off error
Formula	-	$E_h \approx E_{\mathrm{T}} = \alpha_{\mathrm{T}} {N_h}^{-\beta_{\mathrm{T}}}$	$E_h \approx E_{\mathrm{R}} = \alpha_{\mathrm{R}} {N_h}^{\beta_{\mathrm{R}}}$

Since the evolution of E_h ($E_{\mathrm{T}} + E_{\mathrm{R}}$) is known after entering the second phase, by solving

$$\frac{d(E_{\mathrm{T}} + E_{\mathrm{R}})}{dN} = 0, \tag{15}$$

we can predict the optimal number of DoFs

$$N_{\text{opt}} = \left(\frac{\alpha_{\text{T}}\beta_{\text{T}}}{\alpha_{\text{R}}\beta_{\text{R}}}\right)^{\frac{1}{\beta_{\text{T}}+\beta_{\text{R}}}}, \tag{16a}$$

and hence, the highest attainable accuracy is given by

$$E_{\text{min}} = \alpha_{\text{T}} {N_{\text{opt}}}^{-\beta_{\text{T}}} + \alpha_{\text{R}} {N_{\text{opt}}}^{\beta_{\text{R}}}. \tag{16b}$$

4. Results

In this section, we assess the general values of α_{R} and β_{R}. We start with a benchmark Poisson equation, for which the influences of solution strategies and boundary conditions are investigated, and then consider general Poisson equations and general diffusion and Helmholtz equations.

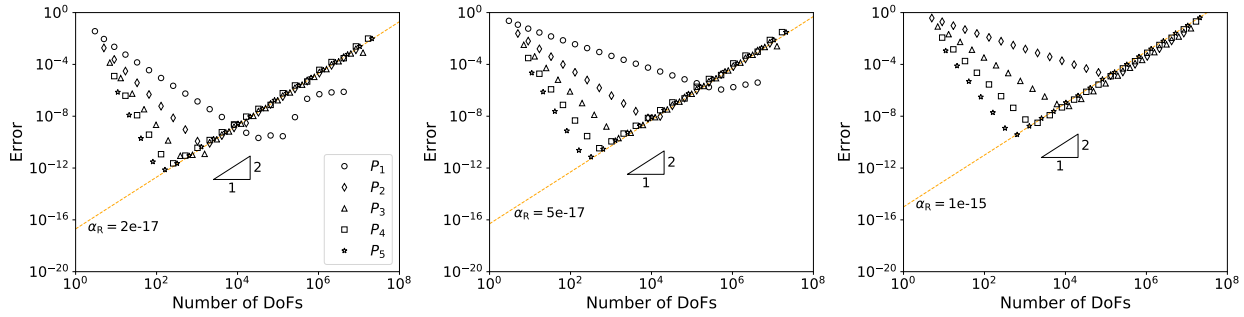
Table 2 Setting of the benchmark Poisson, diffusion and Helmholtz equations.

	"Poisson"	"diffusion"	"Helmholtz"
$d(x)$	1	$1+x$	$(1+i)e^{-x}$
$r(x)$	0	0	$2e^{-x}$
$f(x)$	$-e^{-(x-1/2)^2}\left(4x^2-4x-1\right)$	$-2\pi\cos(2\pi x) + 4\pi^2\sin(2\pi x)(x+1)$	0
$\|f(x)\|_2$	1.60	42.99	0.00
Boundary conditions	$u(0) = e^{-1/4}$	$u(0) = 0$	$u(0) = 1$
	$u(1) = e^{-1/4}$	$u_x(1) = 2\pi$	$u_x(1) = 0$
Analytical solution u_{exc}	$e^{-(x-1/2)^2}$	$\sin(2\pi x)$	$ae^{(1+i)x} + (1-a)e^{-ix}$, $a = 1/((1-i)e^{1+2i}+1)$
$\|u_{\text{exc}}\|_2$	0.92	0.71	1.26

4.1. Benchmark Poisson equation

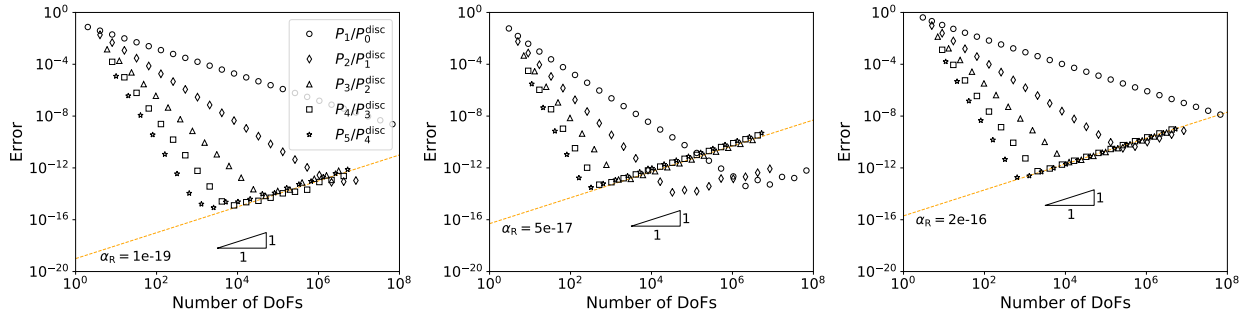
We consider the Poisson equation in Table 2. The error E_h of various variables using the standard FEM and the mixed FEM with p ranging from 1 to 5 are in Fig. 2 and Fig. 3, respectively, in which α_{R} and β_{R} are denoted.

It is found that, for all the variables, the values of α_{R} and β_{R} of different element degrees are the same. The statistics of the former can be found in Fig. 4(a), and the values of the latter, which are only dependent on the FEM method, are 1 using the mixed FEM and 2 using the standard FEM. Notably, α_{R} are of order 10^{-16}, which is as expected when using the double precision, and tend to increase slightly with increasing order of derivative. Furthermore, α_{R} of the mixed FEM is smaller than that of the standard FEM.



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

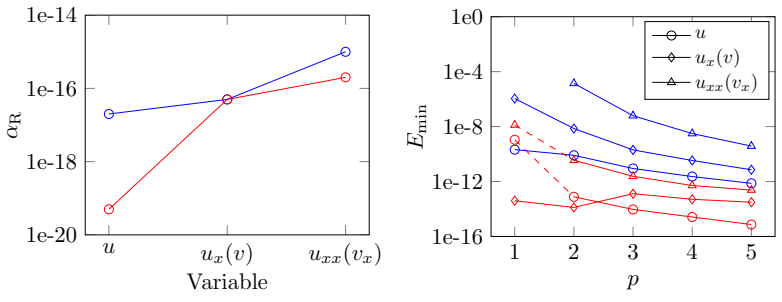
Fig. 2. Absolute errors for the benchmark Poisson equation using the standard FEM.



(a) u **(b)** v **(c)** v_x

Fig. 3. Absolute errors for the benchmark Poisson equation using the mixed FEM.

For larger p, since E_{T} decreases faster, smaller $E_{\min}$ can be obtained, see Fig. 4(b) for the statistics. It also shows that $E_{\min}$ tends to deteriorate by differentiation since E_{T} decreases slower and α_{R} increases slightly. In general, smaller E_{min} can be obtained using the mixed FEM compared to using the standard FEM.



(a) α_{R} **(b)** $E_{\min}$

Fig. 4. Statistics on α_{R} and $E_{\min}$ of the benchmark Poisson equation. The blue color denotes results using the standard FEM and the red color denotes results using the mixed FEM.

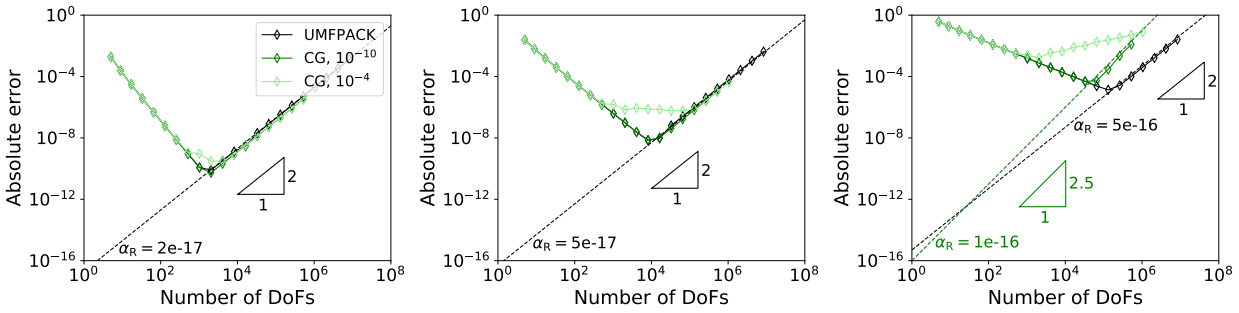
In Sections 4.1.1 – 4.1.2, the sensitivity of the above results will be investigate, using P_2 elements for the

standard FEM and P_4/P_3^{disc} elements for the mixed FEM.

4.1.1. Solution strategy

In this section, we investigate the influence of the solution strategy on the accuracy of the numerical solution. In particular, we compare the outcome when applying the direct solver UMFPACK with that of using the iterative Conjugate Gradient (CG) method [20], which can be applied since the system matrix A in Eq. (6) is symmetric and positive definite. The tolerance of the CG solver is set to be the product of a parameter, denoted by tol_{prm}, and the L_2 norm of the discrete right-hand side $\|F\|_2$. When the L_2 norm of the residual, i.e. $\|F - Au\|_2$ in Eq. (6), is smaller than the tolerance, the iteration is stopped. For the mixed FEM, we additionally investigate the impact of using a segregated solution approach based on the Schur complement instead of a fully coupled approach.

The standard FEM. The CG solver is stopped once $\|F-Au\|_2 \leqslant tol_{prm}\|F\|_2$, with $tol_{prm} = 10^{-10}$ and 10^{-4}, respectively. The absolute errors for u, u_x and u_{xx} using the CG solver are shown in Fig. 5, in comparison with that using the direct solver UMFPACK.



(a) Solution **(b)** First derivative **(c)** Second derivative

Fig. 5. Comparison of the errors using the CG solver and the UMFPACK solver.

When tol_{prm} is adequately small, i.e. $tol_{prm} = 10^{-10}$, the round-off error for the solution and the first derivative using the CG solver is the same with that using the UMFPACK solver; the round-off error for the second derivative using the CG solver increases faster than that using the UMFPACK solver. When tol_{prm} is too large, i.e. $tol_{prm} = 10^{-4}$, the error contribution due to the iterative solver dominates both truncation and round-off errors.

The mixed FEM. Since the resulting matrix Eq. (9) is indefinite, a widely used alternative is to decouple the fully coupled monolithic approach

$$B^\top M^{-1} B U = B^\top M^{-1} G - H, \tag{17a}$$

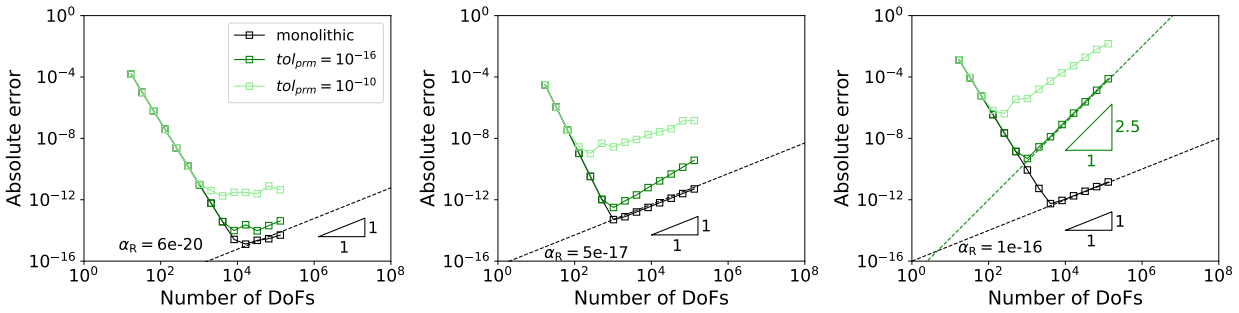
$$M V = G - B U \tag{17b}$$

and solve both equations in segregated manner, i.e. Eq. (17a) is solved in the first place to obtain U, and then it is substituted into Eq. (17b) to obtain V.

Eq. (17a) involves the term $M^{-1}G$ in the right-hand side, which is computed by solving the auxiliary linear system $MY = G$ by using either the UMFPACK or the CG solver. The same options are available for solving Eq. (17b).

The difficulty in solving Eq. (17a) lies in not assembling the Schur complement matrix explicitly since it comprises M^{-1}. The CG solver only makes use of matrix-vector products of the form $(B^\top M^{-1} B)W$, which can be computed by the following three-step algorithm: $X = BW$, $MY = X$ and $Z = B^\top Y$. As before, the linear system $MY = X$ can be solved by the UMFPACK or the CG solver.

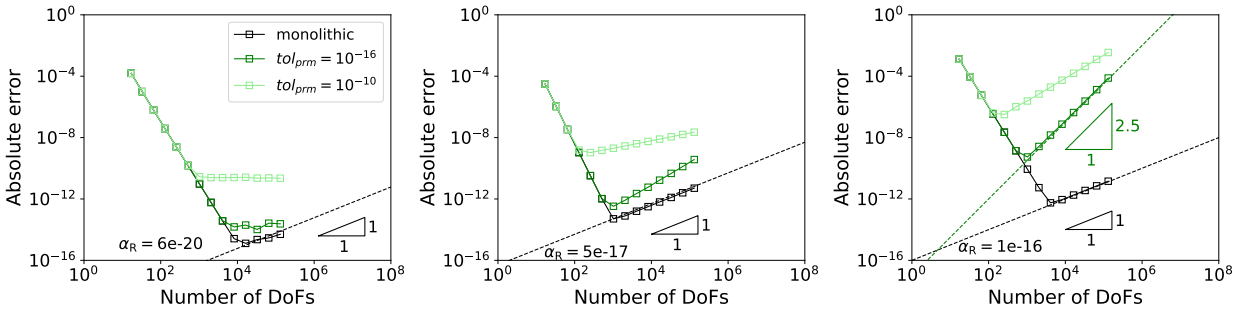
We first investigate the influence of tol_{prm} of the CG solver on the accuracy of the solutions when the left-hand side is $B^\top M^{-1} B$. In this case, the UMFPACK solver is used to solve the matrix equations when the left-hand side is M. For tol_{prm} being 10^{-16} and 10^{-10}, the results are shown in Fig. 6, in comparison with that obtained from solving the monolithic Eq. (9) directly using the UMFPACK solver. It shows that, for the problem at hand, the monolithic solution approach yields by far the most accurate solution and derivative values. The round-off error for v_x increases fastest using the Schur complement approach even though tol_{prm} is sufficiently small, i.e. $tol_{prm} = 10^{-16}$, which makes the highest attainable accuracy much lower. When tol_{prm} is less strict, i.e. $tol_{prm} = 10^{-10}$, the iteration error dominates the total error instead of the round-off error.



(a) Solution **(b)** First derivative **(c)** Second derivative

Fig. 6. Influence of the CG solver on the accuracy when the left-hand side is the Schur complement using the mixed FEM.

Next, we investigate the influence of tol_{prm} of the CG solver when the left-hand side is M. In this case, the CG solver with tol_{prm} being 10^{-16} is used to solve the matrix equation with the left-hand side being $B^\top M^{-1} B$. For tol_{prm} being 10^{-16} and 10^{-10}, the results are shown in Fig. 7, in comparison with that obtained from solving the monolithic Eq. (9) directly using the UMFPACK solver. It also shows that, when the tolerance is less strict, i.e. $tol_{prm} = 10^{-10}$, the iteration error dominates the total error before the round-off error.



(a) Solution **(b)** First derivative **(c)** Second derivative

Fig. 7. Influence of the CG solver on the accuracy when the left-hand side is M using the mixed FEM.

In summary, for the standard FEM, the CG solver gives the same accuracy for u and u_x as the UMFPACK solver when tol_{prm} is strict enough, while the UMFPACK solver is recommended for computing u_{xx}; for the mixed FEM, the accuracy for all the three variables is the highest when using the UMFPACK solver to solve the monolithic Eq. (9) directly. Moreover, the application of the CG solver on both the standard and mixed FEM methods shows that less strict values for tol_{prm} introduce iteration errors.

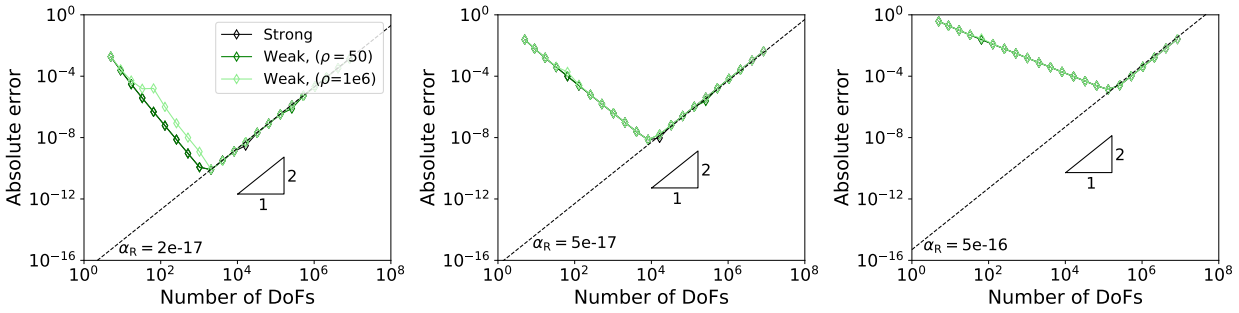
4.1.2. Boundary condition

In this section, two aspects of the influence of the boundary conditions on the round-off error are investigated: first the method of implementing the Dirichlet boundary conditions, and secondly types of boundary conditions.

For the first aspect, using Weak form 2 for $\rho = 50$ and 10^6, the discretization errors are depicted in Fig. 8, in comparison with that using Weak form 1. As can be seen, both weak and strong imposition of the Dirichlet boundary condition yield the same trend line for the round-off error for the solution and its derivatives, and the magnitude of the penalty parameter in the weak imposition makes no difference. In addition, small penalty parameters might lead to larger truncation errors for u, but the difference diminishes when the penalty parameter is large enough.

To construct the problem for the second aspect, the Dirichlet boundary condition at the left boundary ($x = 0$) is kept while the Dirichlet boundary condition at the right boundary ($x = 1$) has been replaced by the Neumann boundary condition $u_x(1) = -e^{-1/4}$, leading to the same solution and derivative profiles.

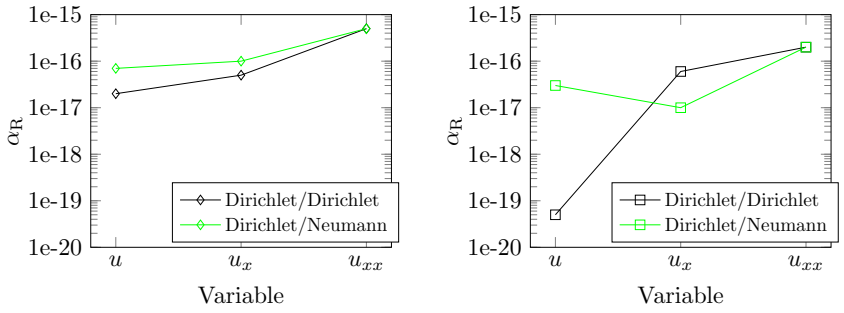
The standard FEM. Using the standard FEM, the offsets α_{R} for the two types of boundary conditions are depicted in Fig. 9(a). For the Dirichlet/Neumann boundary condition, the offsets α_{R} for u and u_x are slightly larger than that for the Dirichlet/Dirichlet boundary condition by a factor of 3.5 and 2, respectively. The offsets α_{R} for u_{xx} are identical for the two types of boundary conditions.



(a) Solution **(b)** First derivative **(c)** Second derivative

Fig. 8. Comparison of the errors for imposing the Dirichlet boundary condition strongly and weakly.

The mixed FEM. Using the mixed FEM, the offsets α_{R} for the two types of boundary conditions are depicted in Fig. 9(b). As can be seen, the type of boundary conditions plays a more important role for α_{R} for the solution than α_{R} for other variables.



(a) The standard FEM **(b)** The mixed FEM

Fig. 9. Comparison of the errors for imposing Dirichlet/Dirichlet and Dirichlet/Neumann boundary conditions.

In summary, α_{R} are relatively independent of the variations in the type of boundary conditions and the method Dirichlet boundary conditions are implemented, which is an important prerequisite for our a posteriori refinement strategy to be applicable for a wide range of problems.

To conclude the sections on sensitivity analysis, the factors that cannot be mitigated are the tolerances for the iterative linear solver, that can be mitigated are the order of magnitude, and that are relatively irrelevant are the boundary conditions.

In Sections 4.2 – 4.3, where the influences of $u(x)$, $d(x)$ and $r(x)$ are investigated, we only consider the Dirichlet boundary conditions, and use P_2 elements for the standard FEM and P_4/P_3^{disc} elements for the mixed FEM.

4.2. General Poisson equation

In this section, we will again consider the Poisson equation, but now focus on the influence of the order of magnitude of the solution and right-hand side on α_{R}. To cover a wide range of scenarios, we choose the cases shown in Table 3. Each case contains a coefficient $c_i,\ i = 1, 2, \ldots, 5$, which is varied over several orders of magnitude so that the L_2 norm of the solution, denoted by $\|u\|_2$, and the L_2 norm of the right-hand side, denoted by $\|f\|_2$, extend over a wide range of magnitudes. Fig. 10 gives an overview of the distribution of $\|u\|_2$ and $\|f\|_2$.

Table 3 Setting of the Poisson equation with different right-hand sides.

Case	$f(x)$	Boundary conditions		$u(x)$
		$u(0)$	$u(1)$	
1	$\sin(2\pi c_1 x)$	0	$(2\pi c_1)^{-2}\sin(2\pi c_1)$	$(2\pi c_1)^{-2}\sin(2\pi c_1 x)$
2	$-e^{-c_2(x-1/2)^2}\cdot$ $\left(4{c_2}^2(x-1/2)^2 - 2c_2\right)$	$e^{-c_2/4}$	$e^{-c_2/4}$	$e^{-c_2(x-1/2)^2}$
3	$\sin(2\pi c_3 x) + 1$	0	$(2\pi c_3)^{-2}\sin(2\pi c_3) - \frac{1}{2}$	$(2\pi c_3)^{-2}\sin(2\pi c_3 x) - \frac{x^2}{2}$
4	$(2\pi c_4)\sin(2\pi c_4 x)$	0	$(2\pi c_4)^{-1}\sin(2\pi c_4)$	$(2\pi c_4)^{-1}\sin(2\pi c_4 x)$
5	0	0	${c_5}^{-1}$	${c_5}^{-1}x$

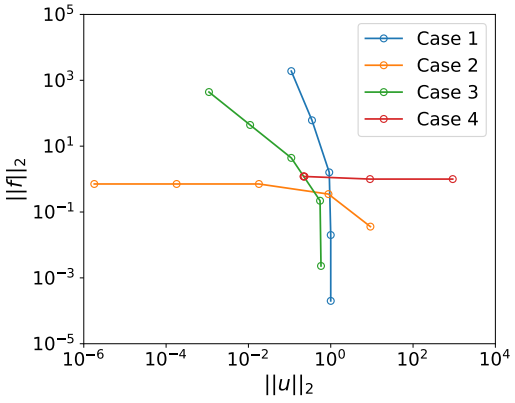
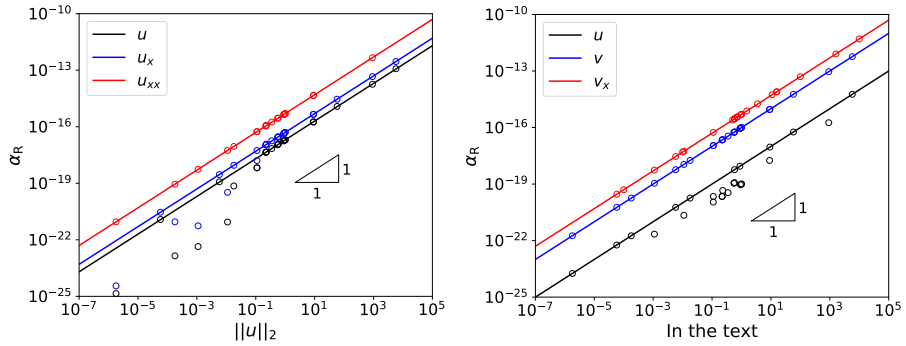


Fig. 10. Distribution of $\|u\|_2$ and $\|f\|_2$ of the Poisson equations in Table 3.

For these Poisson equations, the error basically evolves according to that shown in Fig. 1. To summarize, β_{R} is 2 using the standard FEM and 1 using the mixed FEM. α_{R} is shown in Fig. 11. In Fig. 11(a), the ratio is 2e-17, 5e-17 and 5e-16 for u, u_x and u_{xx}, respectively. In Fig. 11(b), of which the x axis is $\|u\|_2$ for u and v, and $\|v\|_2$ for v_x, the ratio is 1e-18, 1e-16 and 5e-16 for u, v and v_x, respectively. Therefore, α_{R} can be expressed as the product of a constant and an unknown that is shown in Table 4.



(a) The standard FEM **(b)** The mixed FEM

Fig. 11. α_{R} for the influence of $u(x)$.

Table 4 α_{R} in terms of the product of a constant and an unknown for the one-dimensional second order Poisson equations.

(a) The standard FEM

	Constant	Unknown
u	2e-17	$\|u\|_2$
u_x	5e-17	
u_{xx}	5e-16	

(b) The mixed FEM

	Constant	Unknown
u	1e-18	$\|u\|_2$
v	1e-16	
v_x	5e-16	$\|v\|_2$

Furthermore, to make α_{R} independent of the unknowns, we propose the scaling schemes that is shown in Table C.9. Note that, two schemes are required for the mixed FEM: M_1 for u and v_x, and M_2 for v. These schemes are generally able to recover the constants in Table 4. In what follows, the scaling scheme is used only if the scaling factor is out of [0.5, 2].

4.3. General diffusion and Helmholtz equation

We consider the diffusion and Helmholtz equations with $u(x)$ the same as the benchmark Poisson equation. We take $d(x)$ in Table 5 for the former, and $d(x) = 1$ and $r(x)$ in Table 6 for the latter.

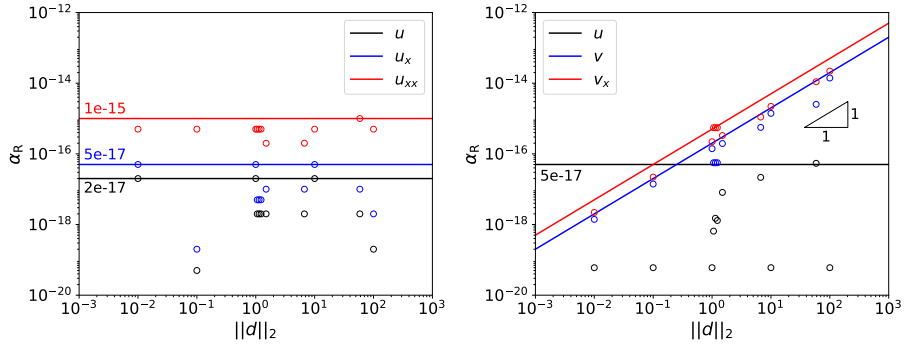
For these two types of equations, the error also evolves according to that shown in Fig. 1 basically. β_{R} is 2 using the standard FEM and 1 using the mixed FEM. α_{R} is shown in Fig. 12 and Fig. 13, respectively. In Fig. 12(b), the ratio is 2e-16 and 5e-16 for v and v_x, respectively. As can be seen, for the diffusion equations, α_{R} is linearly proportional to $\|d\|_2$ for v and v_x using the mixed FEM, while it is relatively independent of $\|d\|_2$ in other scenarios; for the Helmholtz equations, α_{R} is independent of $r(x)$. Therefore, we amend α_{R} in Table 4 to be that shown in Table 7.

Table 5 Various $d(x)$ for the diffusion equations.

Order	$d(x)$	$\|d\|_2$	Order	$d(x)$	$\|d\|_2$
1	0.01	0.01	7	$1+\sin(10x)$	1.14
2	0.1	0.1	8	$1+\sin(100x)$	1.06
3	1	1	9	$1+x$	1.5
4	10	10	10	$1+10x$	6.7
5	100	100	11	$1+100x$	58.6
6	$1+\sin(x)$	1.23			

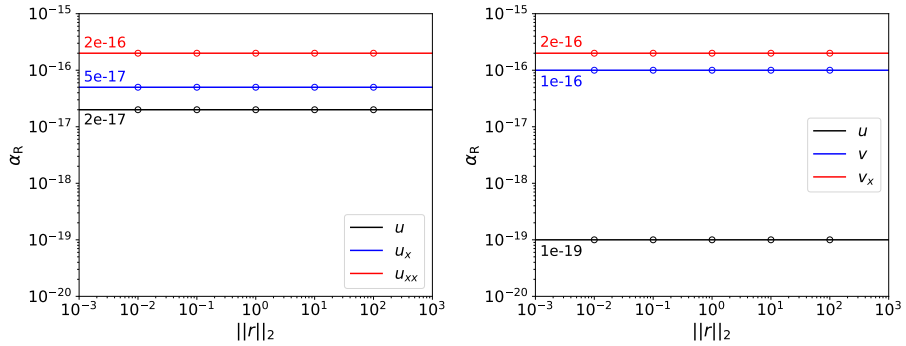
Table 6 Various $r(x)$ for the Helmholtz equations.

Order	$r(x)$	$\|r\|_2$
1	0.01	0.01
2	0.1	0.1
3	1	1
4	10	10
5	100	100



(a) The standard FEM **(b)** The mixed FEM

Fig. 12. α_{R} for the general diffusion equations.



(a) The standard FEM **(b)** The mixed FEM

Fig. 13. α_{R} for the general Helmholtz equations.

Table 7 α_{R} in terms of the product of a constant and an unknown for the one-dimensional second order differential equations.

(a) The standard FEM

	Constant	Unknown
u	2e-17	$\|u\|_2$
u_x	5e-17	
u_{xx}	1e-15	

(b) The mixed FEM

	Constant	Unknown
u	2e-17	$\|u\|_2$
v	2e-16 $\times\|d\|_2$	
v_x	1e-15	$\|v\|_2$

5. A posteriori algorithm for finding the optimal number of degrees of freedom

Based on the validation experiments from the previous section, we introduce a novel a posteriori algorithm for determining E_{min} for the solution and its first and second derivative without performing the brute-force mesh refinement. Table 8 gives the default settings and the required custom input of the algorithm.

Furthermore, we use the following coefficients in the algorithm:

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

$$REF_{\mathrm{min}} = \begin{cases} 9-p & \text{for p} < 6, \\ 4 & \text{otherwise.} \end{cases} \tag{18}$$

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

– a stopping criterion c_s for seeking the scaling factor $\|var_{\mathrm{exc}}\|_2$ in Table C.9, its 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 theoretical order of convergence, with the following default

Table 8 Settings of the algorithm.

Item	Default	Custom
Problem	-	• the differential equation to be solved • its associated boundary conditions
Grid	• initial number of vertices: 2 • the vertices are equidistant	-
FEM	• the maximum N_h, denoted by $N_{\max}$, : 10^8 • Dirichlet boundary conditions are imposed strongly	• standard or mixed formulation • an ordered array of element degrees $\{p_{\min}, \ldots, p_{\max}\}$
Computer precision	IEEE-754 double precision	-
Solver	UMFPACK	-
var	-	• chosen from $\{u,\ u_x,\ u_{xx}\}$ • error tolerance tol_{var}

values:

$$c_r = \begin{cases} 0.9 & \text{for p} < 4, \\ 0.7 & \text{for } 4 \leqslant \text{p} < 10, \\ 0.5 & \text{otherwise.} \end{cases} \tag{19}$$

– the offset α_{R}, see Table 4 for the default values.

The procedure of our algorithm consists of four steps, which are explained below:

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

Step-2. '*NORMALIZATION*'. The function of this step is to find the scaling factor to normalize problems of different orders of magnitude for the variable. The specific procedure can be found in Algorithm 1, where elements of degree $p_{\min}$ are used.

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**
    $\|var_{\text{exc}}\|_2 \leftarrow \|var_h\|_2$;
    break;
  else
    $h \leftarrow h/2$;
    calculate $\|var_h\|_2$ using Eq. (10a) without scaling;
  end
end

Step-3. '*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 $\widetilde{E_h} > E_{\text{R}}$ **and** $N_h < N_{\max}$ **do**
  $\widetilde{Q} \leftarrow \log_2\left(\widetilde{E_{2h}}/\widetilde{E_h}\right)$;
  if $\widetilde{Q} \geqslant \beta_{\text{T}} \times c_r$ **then**
    $N_{\text{c}} \leftarrow N_h$;
    $E_{\text{c}} \leftarrow \widetilde{E_h}$;
    $\alpha_{\text{T}} \leftarrow E_{\text{c}}/{N_{\text{c}}}^{-\beta_{\text{T}}}$;
    $N_{\text{opt}} \leftarrow \left(\frac{\alpha_{\text{T}}\beta_{\text{T}}}{\alpha_{\text{R}}\beta_{\text{R}}}\right)^{\frac{1}{\beta_{\text{R}}+\beta_{\text{T}}}}$;
    $E_{\min} \leftarrow \alpha_{\text{T}}{N_{\text{opt}}}^{-\beta_{\text{T}}} + \alpha_{\text{R}}{N_{\text{opt}}}^{\beta_{\text{R}}}$;
  else
    $h \leftarrow h/2$;
    calculate $\widetilde{E_h}$ using Eq. (10b) with proper scaling schemes;
  end
end

Step-4. '*OUTPUT*'. In this step, we output $E_{\min}$ obtained from *Step*-3.

6. Validation

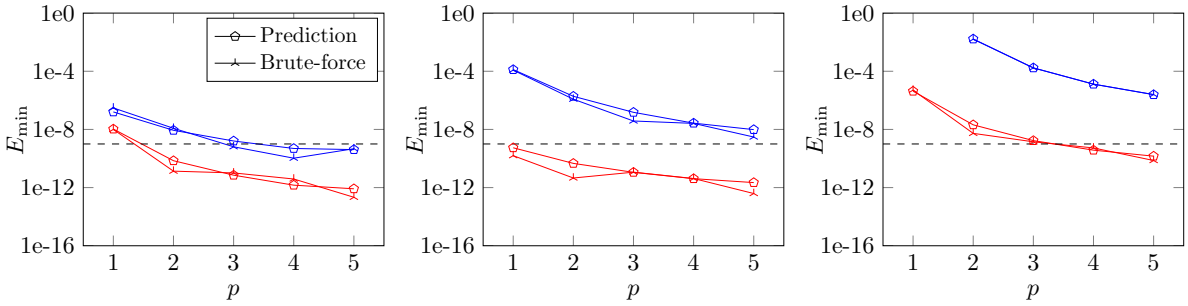
In what follows, we validate the strategy discussed in Section 3 by using the following Helmholtz problem:

$$\left((0.01+x)(1.01-x)u_x\right)_x - (0.01i)u(x) = 1.0, \qquad x \in I = (0,1), \tag{20}$$

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, and the element degree p has a range of $\{1, 2, \ldots, 5\}$. Variables u, u_x and u_{xx} are all investigated, for which tol_{var} is set to be 10^{-9}.

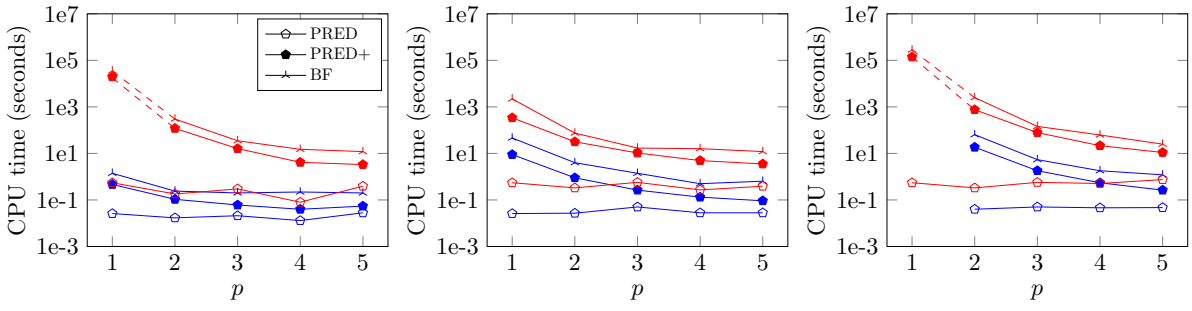
Using the prediction approach and the brute-force approach, $E_{\min}$ are compared in Fig. 14. As can be seen, $E_{\min}$ can be predicted correctly.



(a) Solution **(b)** First derivative **(c)** Second derivative

Fig. 14. Comparison of $E_{\min}$ for Eq. (20) using the algorithm and the brute-force refinement. The blue color denotes the standard FEM, and the red color denotes the mixed FEM.

The CPU time required by the prediction approach (PRED) and the brute-force approach (BF) is shown in Fig. 15. Next to time PRED, and the computation time for the optimal grid (PRED+) using the prediction approach is also given. As can be seen, both time BF and time PRED+ decrease with increasing element degree. Time PRED+ is much smaller compared to time BF, see Fig. 16 for the percentage of the CPU time saved by PRED+, which shows a saving of the CPU time basically more than 60% and 40% for the standard FEM and the mixed FEM, respectively. Last but not least, time PRED is negligible compared to time PRED+.



(a) Solution **(b)** First derivative **(c)** Second derivative

Fig. 15. Comparison of the CPU time to obtain $E_{\min}$ for Eq. (20) using the algorithm and the brute-force refinement. The blue color denotes the standard FEM, and the red color denotes the mixed FEM.

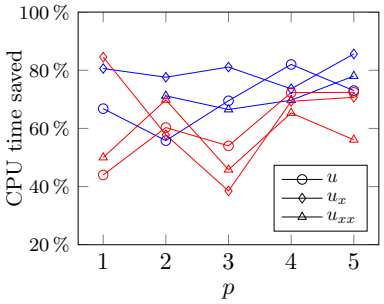


Fig. 16. Percentage of CPU time saved using the algorithm. The blue color denotes the standard FEM, and the red color denotes the mixed FEM.

Furthermore, the dashed line indicating the desired error tolerance in Fig. 14 cannot be reached using the standard FEM, whereas it can be reached using the mixed FEM with P_4/P_3^{disc} or betters. When using P_4/P_3^{disc}, N_{opt} for u, u_x and u_{xx} are predicted to be 6042, 9812 and 123486, respectively.

7. Conclusions

A novel approach is presented to predict the highest attainable accuracy for second-order ordinary differential equations using the finite element methods. In contrast to the brute-force approach, which uses successive h-refinements, this approach uses only a few coarse grid refinements. This approach is viable for the solution and its first and second derivative, for both the standard FEM and the mixed FEM, and different element degrees. The algorithm for implementing the approach shows that the highest attainable accuracy can be accurately predicted and the CPU time is significantly reduced. To compute the solution of the highest attainable accuracy using our approach, the CPU time can be saved more than 60% for the standard FEM and 40% for the mixed FEM.

Future research will focus on the validation of the approach for 2D second-order problems, where the influence of the linear system solver, local mesh refinement and boundary conditions might be significantly different from 1D problems.

Appendix A. Derivation of the weak form

Appendix A.1. The standard FEM

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

$$\langle \eta, -(du_x)_x + ru \rangle = \langle \eta, f \rangle. \tag{A.1}$$

By applying Gauss's theorem, we obtain

$$\langle \eta_x, du_x \rangle + \langle \eta, ru \rangle = \langle \eta, f \rangle + \langle \eta, du_x n \rangle_{\Gamma_N}, \tag{A.2}$$

which gives that shown in Eq. (3). Adding auxiliary terms to the above equation renders Eq. (4).

Appendix A.2. The mixed FEM

As a first step, we introduce the auxiliary variable

$$v(x) = -d(x)u_x, \tag{A.3a}$$

allowing Eq. (1) to be rewritten as

$$-v_x - r(x)u(x) = -f(x). \tag{A.3b}$$

Multifply Eq. (A.3a) by a test function of v, i.e. $w \in H^1_{N0}(I)$, and integrate it over I yield

$$\langle d^{-1}v + u_x, w\rangle = 0. \tag{A.4a}$$

Applying Gauss's theorem to Eq. (A.4a), it becomes

$$\langle w,\, d^{-1}v\rangle - \langle w_x,\, u\rangle = -\langle w,\, gn\rangle_{\Gamma_D}. \tag{A.4b}$$

Multiply Eq. (A.3b) by a test function of u, i.e. $q \in L^2(I)$, and integrate it over I yield

$$-\langle q,\, v_x\rangle + \langle q,\, ru\rangle = \langle q,\, f\rangle. \tag{A.5}$$

Eq. (A.4b) and Eq. (A.5) result in those shown in Eq. (7).

Appendix B. Proof of the slope of the decrease of the error

Here we give the proof for the standard FEM. The process for the mixed FEM is similar.

For the grid size h and element degree p, the number of DoFs

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

Therefore,

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

Since the error [5]

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

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

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

where $C_1 = Cp^{p+1}$. Therefore, the slope is $\beta_{\mathrm{T}} = p + 1$

Appendix C. Scaling schemes

Table C.9 System of equations using various scaling schemes.

	Scheme	Left-hand side	Solution	Right-hand side
The standard FEM	S	A	$\frac{1}{\|u\|_2}U$	$\frac{1}{\|u\|_2}F$
The mixed FEM	M_1	$\begin{bmatrix} M & \frac{\|u\|_2}{\|v\|_2}B \\ B^T & 0 \end{bmatrix}$	$\begin{bmatrix} \frac{1}{\|v\|_2}V \\ \frac{1}{\|u\|_2}U \end{bmatrix}$	$\frac{1}{\|v\|_2}\begin{bmatrix} G \\ H \end{bmatrix}$
	M_2	$\begin{bmatrix} M & B \\ B^T & 0 \end{bmatrix}$	$\frac{1}{\|u\|_2}\begin{bmatrix} V \\ U \end{bmatrix}$	$\frac{1}{\|u\|_2}\begin{bmatrix} G \\ H \end{bmatrix}$

References

[1] Mohit Kumar, Henk M. Schuttelaars, Pieter C. Roos, and Matthias Möller. Three-dimensional semi-idealized model for tidal motion in tidal estuaries. *Ocean Dynamics*, 66(1):99–118, 2016.

[2] GF Carey. Derivative calculation from finite element solutions. *Computer Methods in Applied Mechanics and Engineering*, 35(1):1–14, 1982.

[3] Joel H Ferziger and Milovan Peric. *Computational methods for fluid dynamics*. Springer Science & Business Media, 2012.

[4] B Guo and I Babuška. The hp version of the finite element method. *Computational Mechanics*, 1(1):21–41, 1986.

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

[6] Julen Alvarez-Aramberri, David Pardo, Maciej Paszynski, Nathan Collier, Lisandro Dalcin, and Victor M Calo. On round-off error for adaptive finite element methods. *Procedia Computer Science*, 9:1474–1483, 2012.

[7] Ivo Babuska and Gustaf Söderlind. On roundoff error growth in elliptic problems. *ACM Transactions on Mathematical Software*, 44(3):1–22, 2018.

[8] Jindrich Necas. *Direct methods in the theory of elliptic equations*. Springer Science & Business Media, 2011.

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

[10] Richard Haberman. *Applied partial differential equations with Fourier series and boundary value problems*. Pearson Higher Ed, 2012.

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

[12] Jouni Freund and Rolf Stenberg. On weakly imposed boundary conditions for second order problems. In *Proceedings of the Ninth Int. Conf. Finite Elements in Fluids*, pages 327–336. Venice, 1995.

[13] 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.

[14] Giovanni Alzetta, Daniel Arndt, Wolfgang Bangerth, Vishal Boddu, Benjamin Brands, Denis Davydov, Rene Gassmöller, Timo Heister, Luca Heltai, Katharina Kormann, et al. The deal.II library, version 9.0. *Journal of Numerical Mathematics*, 26(4):173–183, 2018.

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

[16] Olof Runborg. Lecture notes in numerical solutions of differential equations (dn2255): Verifying numerical convergence rates, 2012.

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

[18] Meshing considerations for linear static problems. https://www.comsol.com/blogs/meshing-considerations-linear-static-problems/. Accessed: 2019-12-9.

[19] W Kahan. Floating-point tricks to solve boundary-value problems faster. *University of California@ Berkeley*, 2013.

[20] Theo Ginsburg. The conjugate gradient method. *Numer. Math.*, 5(1):191–200, December 1963.