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 solution accuracy cannot be improved indefinitely because of the limited computer precision. We propose an innovative method to find the highest attainable accuracy (HAA). To this end, we validate the bound of the highest attainable accuracy for the second-order ordinal differential equations a priori. Based on the FEM method, element degree, we can extrapolate the discretization error when it converges at the analytical rate. As a result, a formula for the highest attainable accuracy is proposed. We apply our method to 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 only using h-refinements.

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.

The accuracy of the numerically obtained solution is influenced by many sources of errors [3]: firstly, errors in the set-up of the models, such as the simplification of the domain and governing equations and the

[*] 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)

approximation of the 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, the iteration error resulting from the artificially controlled tolerance of iterative solvers; finally, the round-off error due to the adoption of finite-precision computer arithmetics, rather than exact arithmetics. One tacitly assumes that most errors are well-balanced and/or negligibly small. 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 [4] are adopted. In this paper, the focus is on the overall discretization error due to truncation and round-off. In particular, we will show that the latter might very well have a significant influence on the overall accuracy and propose a practical strategy to balance both error contributors.

The discretization error strongly depends on the number of degrees of freedom ("DoFs"), denoted by $N_h^{(p)}$, which is a function of the mesh width h and the approximation order p. The truncation error, denoted by E_T, dominates the discretization error only when $N_h^{(p)}$ is not too large, and it decreases with increasing mesh resolution and element degree as it can be expected from finite element theory [5]. Based on this, the commonly used approaches to reduce the truncation error are to reduce the mesh width (h-refinement), increase the approximation order (p-refinement), or apply both strategies simultaneously (hp-refinement) [6]. The round-off error, denoted by E_R, is, however, only negligible for moderately small values of $N_h^{(p)}$ and dominates the overall discretization error if more and more DoFs are employed [7]. Consequently, for a particular approximation order p, by performing h-refinement, the best accuracy is obtained at the break-even point where the discretization error is the smallest. We denote the highest accuracy by $E_\mathrm{min}^{(p)}$ and the optimal number of DoFs by $N_\mathrm{opt}^{(p)}$.

While $N_\mathrm{opt}^{(p)}$ is typically impractically large if low(est)-order approximations are used, it can be very small if high-order approximations are adopted, which are nowadays becoming more and more popular, and make the results more prone to be polluted 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 [8, 9] or take them into account explicitly in the error-estimation procedure [10, 11]. 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 discretization error, for the solution, and its first and second derivative, and propose a practical approach for obtaining $E_\mathrm{min}^{(p)}$. The scope is restricted to one-dimensional model problems, i.e. Poisson, diffusion and Helmholtz equations, for which both the standard finite element method (FEM) and the mixed FEM[12] are considered. To assess the general applicability of the aforementioned approach, the following factors are investigated: the element degree over a wide range, first and second derivative of the solution, type of boundary conditions and method of implementing them, choice and configuration of the linear system solver, order of magnitude of the solution and its derivatives, and equation type.

The paper is organized as follows. The model problem, finite element formulation and numerical implementation are described in Section 2. The general behavior of the discretization error and the approach to predict $E_{\min}^{(p)}$ are discussed in Section 3. Numerical results for determining the offset of the round-off error are shown 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 5.1. The conclusions are drawn in Section 6.

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, when $r(x) = 0$, the diffusion equation is found, and when $r(x) \neq 0$, we obtain the Helmholtz equation. The boundary conditions are $u(x) = g(x)$ on Γ_D and $d(x)u_x = h(x)$ on Γ_N. Here, Γ_D and Γ_N are the boundaries where Dirichlet and Neumann boundary conditions are imposed, respectively. In this paper, for all the equations investigated, the existence of the second derivative is guaranteed in the weak sense, i.e. $u \in H^2(I)$.

2.2. Finite element formulation

For convenience, we introduce the two inner products:

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

$$(g_1(x),\, g_2(x))_\Gamma = g_1(x_0)g_2(x_0). \tag{2b}$$

where $f_1(x)$, $f_2(x)$, $g_1(x)$ and $g_2(x)$ 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:

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

with

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

where n is 1 at $x = 1$, and -1 at $x = 0$. (3)

Next, we approximate the exact solution u_{exc} by a linear combination of a finite number of basis functions:

$$u_{\text{exc}} \approx u_h^{(p)} = \sum_{i=1}^{m} u_i \varphi_i^{(p)}. \tag{4}$$

Here, $\varphi_i^{(p)}$ are C^0-continuous Lagrange basis functions of degree p, denoted as P_p, with Gauss-Lobatto support points x_j, which feature the Kronecker-delta property, i.e. $\varphi_i^{(p)}(x_j) = \delta_{ij}$. The coefficients u_i are the values of $u_h^{(p)}$ at the DoFs, as a direct consequence of the Kronecker-delta property of $\varphi_i^{(p)}$. The number of DoFs of $u_h^{(p)}$, denoted by m, equals $p \times t + 1$, where t is the total number of the grid cells. Finally, taking the test function η equal to $\varphi_k^{(p)},\ k = 1,\, 2,\, \ldots,\, m$, we obtain

$$AU = F, \tag{5}$$

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

2.2.2. The mixed FEM

As a first step, we introduce the auxiliary variable

$$v(x) = -d(x)u_x, \tag{6a}$$

allowing Eq. (1) to be rewritten as

$$-v_x - r(x)u(x) = -f(x). \tag{6b}$$

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. The weak form of Eq. (1) using the mixed FEM, derived in

Appendix A.2, is given by:

Weak form 2

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

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

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

with

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

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

Next, we approximate the exact gradient v_{exc} and the exact solution u_{exc} by a linear combination of a finite number of basis functions:

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

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

where $\varphi_i^{(p)}$ are of the same type of basis functions used in Eq. (4), with coefficients v_i the associated values of $v_h^{(p)}$ at the DoFs; $\psi_j^{(p-1)}$ are discontinuous Lagrange basis functions of degree $p-1$, denoted as P^{disc}_{p-1}, with coefficients $u_{c,j}$ the associated values of $u_h^{(p-1)}$ at the DoFs. This pair of elements will be referred to as $P_p/P^{\text{disc}}_{p-1}$. Since the use of discontinuous basis functions, there are two independent $u_{c,j}$ at cell interfaces. The number of DoFs for $v_h^{(p)}$, denoted by n, equals $p \times t + 1$, and the number of DoFs for $u_h^{(p-1)}$ equals $p \times t$. Finally, replacing the test functions 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, the discrete gradient operator B, and its transpose, the discrete divergence operator $B^\top$, are the components of the discrete left-hand side of Eqs. (7a)–(7b), G and H are the components of the right-hand side, and V and U are the discrete first derivative and solution, i.e. the vectors of the coefficients v_i and u_{cj}, respectively.

For the sake of readability, we will drop the superscript (p), whenever the approximation order is clear from the context.

2.3. Numerical implementation

In what follows, we demonstrate how to obtain the numerical solution for Eq. (1) with specific coefficients and assess its quality. For the latter, both the error, obtained using the analytical solution or the finer numerical solution, and the order of convergence are investigated.

2.3.1. Solution technique

Unless stated otherwise, all results are computed in IEEE-754 double precision [4] using the deal.II finite element code [15] that provides subroutines for creating the computational grid, building and solving the system of equations, and computing the error norms.

The computational mesh is obtained by globally refining a single element that covers the interval I, and the Dirichlet boundary conditions are imposed strongly unless stated otherwise. The former means that, when the solution is real valued, using the standard FEM, the number of DoFs equals $2^{REF} \times p + 1$ at the REFth refinement; using the mixed FEM, the number of DoFs equals $2 \times 2^{REF} \times p + 1$ at the REFth refinement. For complex-valued problems, 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. Furthermore, unless stated otherwise, to solve the matrix equation, the UMFPACK solver [16], which implements the multi-frontal LU factorization approach, is used as it results in relatively fast computations of the problems considered in this paper, and prevents the iteration errors for the iterative solvers.

The derivatives, which are $u_{h,x}$ and $u_{h,xx}$ in the standard FEM and only $u_{h,xx}$ in the mixed FEM, are computed in the classical finite element manner, e.g. $u_{h,x}^{(p-1)} = \sum_{i=1}^{m} u_i \varphi_{i,x}^{(p)}$ yields an approximation to u_x using standard FEM.

2.3.2. Error estimation

For the numerical results var_h, where var can be u, u_x and u_{xx}, the discretization 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 [17]

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

otherwise, where $var_{h/2}$ is the numerical solution computed on a mesh of grid size $h/2$. Furthermore, we compute the order of convergence from either $\log_2\left(\frac{E_h}{E_{h/2}}\right)$ or $\log_2\left(\frac{\widetilde{E_h}}{\widetilde{E_{h/2}}}\right)$, for which the theoretical value is one order higher than the approximation order[5].

3. Approach to finding the optimal number of DoFs

3.1. Theoretical evolution of the discretization error

The conceptual sketch of E_h against N_h in the log-log axes can be found in Fig. 1, which is applicable to different variables for both the standard and mixed FEMs. When N_h is relatively small, E_h may not decrease at the aforementioned theoretical order of convergence, denoted by the black circles, but it basically does when N_h is relatively large, denoted by the green circles. During these two phases, E_h is controlled by the truncation error E_{T}, and it can be represented by

$$E_h \approx E_{\mathrm{T}} = \alpha_{\mathrm{T}} {N_h}^{-\beta_{\mathrm{T}}}, \tag{11}$$

in the latter phase, where α_{T} is the offset, and β_{T} the slope of the line approximating E_h.

For the increase part of E_h, denoted by the orange circles, it is controlled by the round-off error E_{R}. Since the slope for the line approximating it tends to be fixed[7, 18], it can be represented by

$$E_h \approx E_{\mathrm{R}} = \alpha_{\mathrm{R}} {N_h}^{\beta_{\mathrm{R}}}, \tag{12}$$

where α_{R} is the offset and β_{R} the slope of the line approximating E_h. Moreover, since the values of the two constants are given or formulized in section 4, E_{R} can be determined a priori.

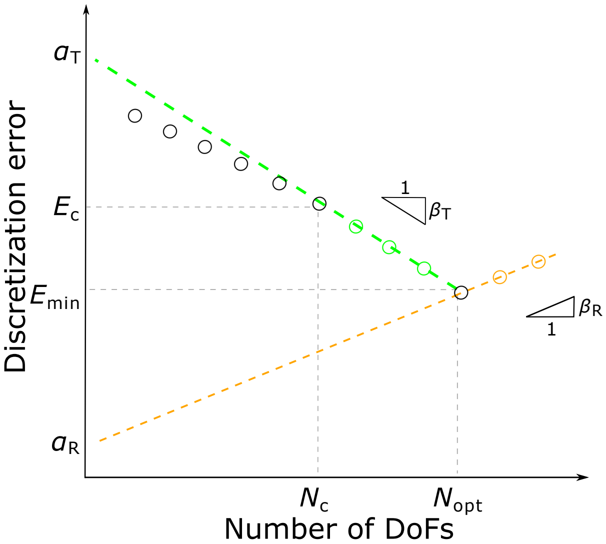


Fig. 1. Conceptual sketch of the discretization error against the number of DoFs.

3.2. Strategy to find the optimal number of DoFs

When E_h starts to decrease at the analytical rate, for which E_h and N_h read E_{c} and N_{c}, respectively, α_{T} can be inverted by using

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

After this point, both the development of E_{T} and E_{R} are known. Obviously, N_{opt} occurs when $E_{\mathrm{T}}+E_{\mathrm{R}}$ is the smallest. By solving

$$\frac{d(E_{\mathrm{T}}+E_{\mathrm{R}})}{dN}=0, \tag{14}$$

we can predict

$$N_{\mathrm{opt}}=\left(\frac{\alpha_{\mathrm{T}}\beta_{\mathrm{T}}}{\alpha_{\mathrm{R}}\beta_{\mathrm{R}}}\right)^{\frac{1}{\beta_{\mathrm{T}}+\beta_{\mathrm{R}}}}, \tag{15a}$$

and hence, the highest attainable accuracy

$$E_{\mathrm{min}}=\alpha_{\mathrm{T}}{N_{\mathrm{opt}}}^{-\beta_{\mathrm{T}}}+\alpha_{\mathrm{R}}{N_{\mathrm{opt}}}^{\beta_{\mathrm{R}}}. \tag{15b}$$

4. Determination of the error constants in Fig. 1

To determine the constants α_{R} and β_{R} in Fig. 1, we first investigate three benchmark equations, followed by a wide range of second-order differential equations: Poisson equations with various $u(x)$ and $f(x)$ and several representative diffusion and Helmholtz equations. Finally, we analyse the influence of the solution strategy and boundary condition.

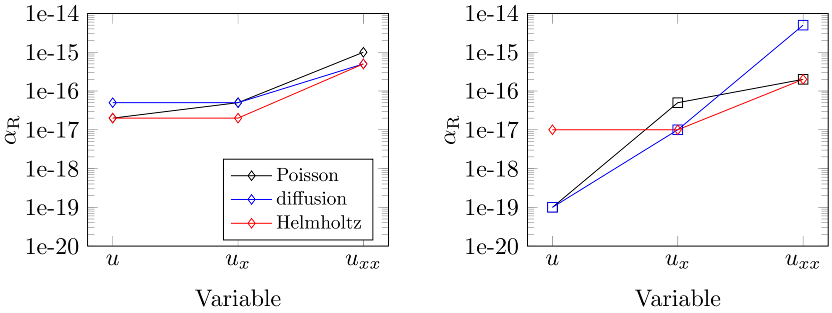
4.1. Benchmark equations

The benchmark equations are shown in Table 1, for which the element degree ranges from 1 to 5.

Table 1 Benchmark 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
$\lVert f(x)\rVert_2$	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}	$e^{-(x-1/2)^2}$	$\sin(2\pi x)$	$ae^{(1+i)x}+(1-a)e^{-ix}$, $a=1/((1-i)e^{1+2i}+1)$
$\lVert u_{\mathrm{exc}}\rVert_2$	0.92	0.71	1.26

For all the element degrees, α_{R} and β_{R} for one particular variable are basically the same. The values of the former are shown in Fig. 2, which are as expected when using the double precision[13]. The values of the latter are 2 using the standard FEM and 1 using the mixed FEM, which are taken as constants if not stated otherwise.



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

Fig. 2. α_{R} for the benchmark equations.

In addition, as that shown in Fig. 1, β_{T} can be reached fast: for the benchmark Poisson equation, when the approximation order is 3, the development of the order of convergence is shown in Table 2.

Table 2 Evolution of order of convergence for the benchmark Poisson equation.

(a) The standard FEM

# of	Order of convergence		
refinements	u	u_x	u_{xx}
2	3.97	4.02	3.87
3	3.99	4.00	3.98
4	4.00	4.00	4.00

(b) The mixed FEM

# of	Order of convergence		
refinements	u	u_x	u_{xx}
2	3.96	4.02	3.86
3	3.99	4.00	3.98
4	4.00	4.00	4.00

4.2. General second-order differential equations

First, to cover a wide range of $\|u_{\mathrm{exc}}\|_2$, together with $\|f\|_2$, we investigate the cases shown in Table 3, for which the distribution of $\|u_{\mathrm{exc}}\|_2$ and $\|f\|_2$ can be found in Fig. 3. Second, we investigate various $d(x)$ shown in Table 4 for the diffusion equations with $u = e^{-(x-1/2)^2}$. Lastly, we consider $r(x)$ equal to the first five cases in Table 4 for the Helmholtz equations with $u = e^{-(x-1/2)^2}$ and $d(x) = 1$. Specifically, we restrict ourselves to P_2 and P_4/P_3^{disc} elements, and only Dirichlet boundary conditions are considered.

Table 3 Settings of the Poisson equations with various $\|u_{\text{exc}}\|_2$ and $\|f\|_2$.

Case	$f(x,c)$	$u_{\text{exc}}(x,c)$	c
1	$\sin(2\pi cx)$	$(2\pi c)^{-2}\sin(2\pi cx)$	1e-2, 1e-1, 1e0, 1e1, 1e2
2	$(2\pi c)\sin(2\pi cx)$	$(2\pi c)^{-1}\sin(2\pi cx)$	
3	$\sin(2\pi cx)+1$	$(2\pi c)^{-2}\sin(2\pi cx)-\frac{x^2}{2}$	1e-4, 1e-2, 1e0, 1e2, 1e4
4	$-e^{-c(x-1/2)^2}\cdot$ $\left(4c^2(x-1/2)^2-2c\right)$	$e^{-c(x-1/2)^2}$	

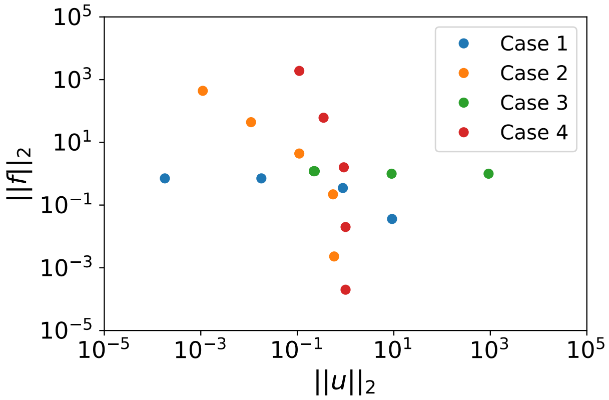


Fig. 3. Distribution of $\|u_{\text{exc}}\|_2$ and $\|f\|_2$ for the Poisson equations in Table 3.

Table 4 Various $d(x)$ for the diffusion equations.

#	$d(x)$	$\|d(x)\|_2$	#	$d(x)$	$\|d(x)\|_2$
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			

For all the cases, we found the relations shown in Table 5.

Table 5 Relation between α_{R} and a variety of L_2 norms for the second-order differential equations.

	The standard FEM	The mixed FEM
u	2e-17 $\times \|u\|_2$	1e-18 $\times \|u\|_2$
$d(x)u_x$	5e-17 $\times \|u\|_2$	1e-16 $\times \|d(x)\|_2 \|u\|_2$
$(d(x)u_x)_x$	5e-16 $\times \|u\|_2$	5e-16 $\times \|d(x)u_x\|_2$

4.3. Sensitivity analysis

We focus on the benchmark Poisson equation, and the approximation order for each variable is 3.

4.3.1. Solution strategy

The alternative solution method is the iterative Conjugate Gradient (CG) method [19], which is applied when the left-hand side is symmetric and positive definite. In the standard FEM, it can be applied directly, while in the mixed FEM, since the left-hand side of Eq. (9) is indefinite, it is applied after segregating Eq. (9) based on the Schur complement. 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.

The standard FEM. The evolution of discretization error using the CG solver for $tol_{prm} = 10^{-10}$ and 10^{-6} is shown in Fig. 4(a), in comparison with that using the UMFPACK solver.

When tol_{prm} is adequately small, i.e. $tol_{prm} = 10^{-10}$, the CG solver produces basically the same error with the UMFPACK solver except for the second derivative, for which the round-off error increases faster when using the CG solver. When tol_{prm} is too large, i.e. $tol_{prm} = 10^{-6}$, the error contribution due to the iterative solver dominates the discretization error. Furthermore, it is also found that the errors of higher-order elements are more easier to be affected by larger tol_{prm}.

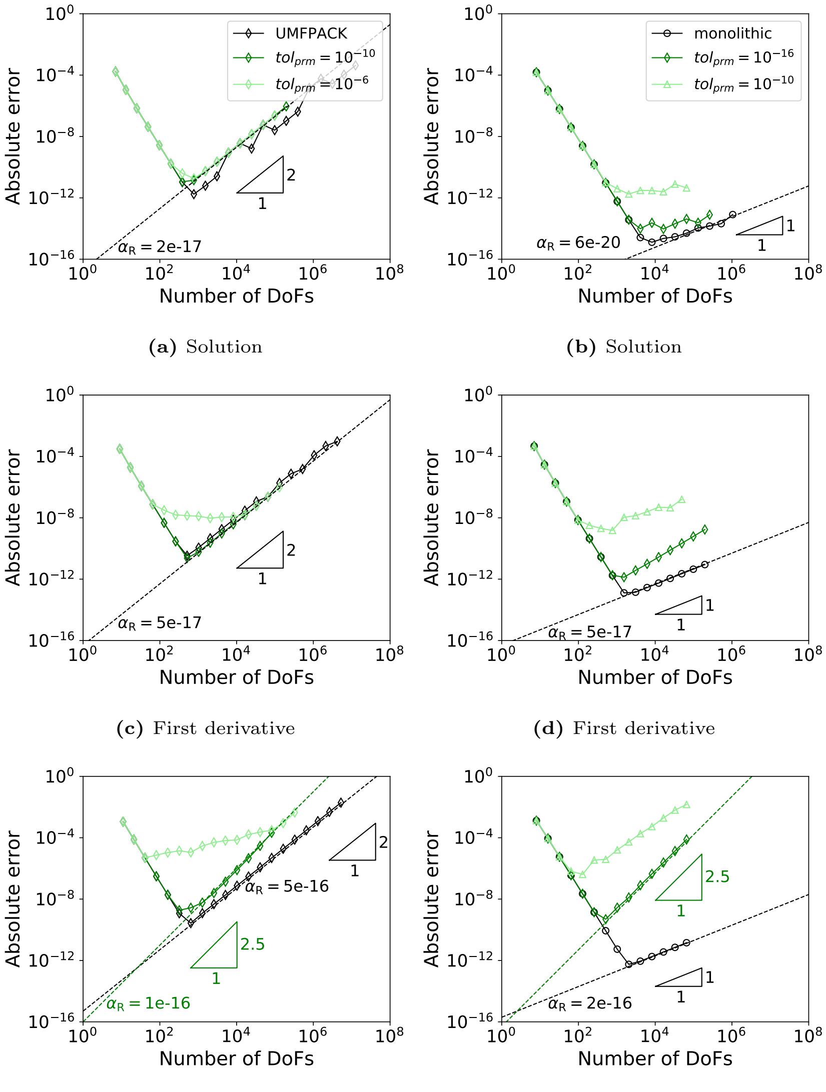
The mixed FEM. The resulting system of equations after segregating Eq. (9) reads

$$B^{\top} M^{-1} B U = B^{\top} M^{-1} G - H, \tag{16a}$$

$$M V = G - B U, \tag{16b}$$

for which Eq. (16a) is solved in the first place to obtain U, which is then substituted into Eq. (16b) to obtain V.

We first investigate the influence of tol_{prm} of the CG solver on the solution accuracy when the left-hand side is $B^{\top} M^{-1} B$ (Schur complement). In this case, the UMFPACK solver is used to solve the system of equations for the left-hand side being M. For tol_{prm} being 10^{-16} and 10^{-10}, the errors are shown in Fig. 4(b), in comparison with that obtained from solving the monolithic Eq. (9) directly. It shows that the monolithic solution approach yields by far the most accurate solution and derivative values. Remarkably,



(a) Solution

(b) Solution

(c) First derivative

(d) First derivative

(e) Second derivative

(f) Second derivative

Fig. 4. Comparison of the errors using the CG solver and the UMFPACK solver.

the round-off error for v_x increases fastest using the Schur complement approach even though tol_{prm} is sufficiently small, i.e. $tol_{prm} = 10^{-16}$. When tol_{prm} is less strict, i.e. $tol_{prm} = 10^{-10}$, the iteration error dominates the discretization error before the round-off error.

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 system of equations with the left-hand side being $B^\top M^{-1} B$. For the same tolerances, the errors are basically the same with that shown in Fig. 4(b).

In summary, when tol_{prm} is strict enough, for the standard FEM, the CG solver gives the same accuracy for u and u_x as the UMFPACK solver , while the UMFPACK solver produces higher accuracy for u_{xx}; for the mixed FEM, the accuracy for all the three variables is the highest when using the monolithic approach.

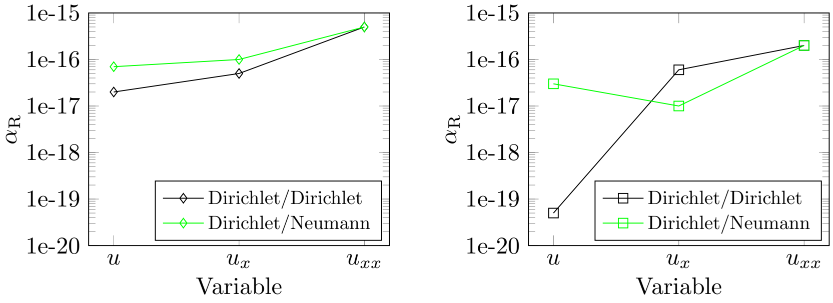
When tol_{prm} is less strict, the applications of the CG solver on both the standard and mixed FEM methods introduce iteration errors.

4.3.2. Boundary conditions

In this section, the influence of boundary condition types on the round-off error are investigated. 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. 5(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.

The mixed FEM. Using the mixed FEM, the offsets α_{R} for the two types of boundary conditions are depicted in Fig. 5(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. 5. 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.

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

Based on the strategy given in section 3 and error constants validated in section 4, we introduce a novel a posteriori algorithm for determining $E_{\min}$. The default and custom settings of the algorithm are given in Table 6, followed by several coefficients.

Table 6 Settings of the algorithm.

Item	Default	Custom
Problem	-	• the differential equation to be solved • its associated boundary conditions
Grid	• initial number of vertices: 2 • refined globally afterwards	-
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}

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:

$$REF_{\min} = \begin{cases} 9 - p & \text{for p} < 6, \\ 4 & \text{otherwise.} \end{cases} \tag{17}$$

  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 relaxation coefficient c_r for seeking the theoretical order of convergence, with the following default values:

$$c_r = \begin{cases} 0.9 & \text{for p} < 4, \\ 0.7 & \text{for } 4 \leqslant \text{p} < 10, \\ 0.5 & \text{otherwise.} \end{cases} \tag{18}$$

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.

5.1. *Application*

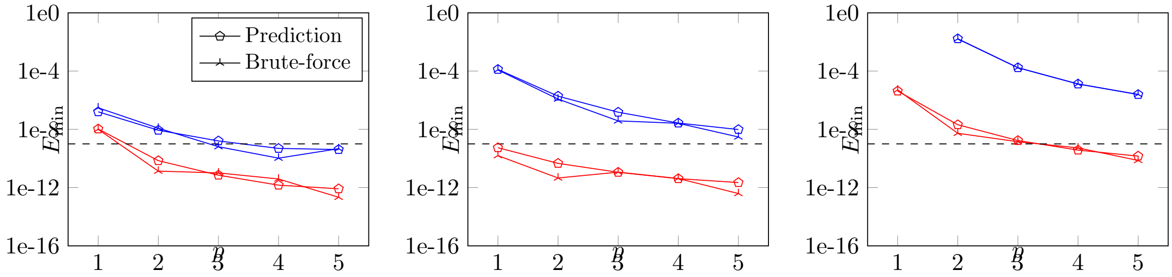
In what follows, we validate the strategy discussed in Section 3 by using the following Helmholtz problem:

$$\left((0.01 + x)(1.01 - x)u_x\right)_x - (0.01i)u(x) = 1.0, \qquad x \in I = (0, 1), \tag{19}$$

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}.

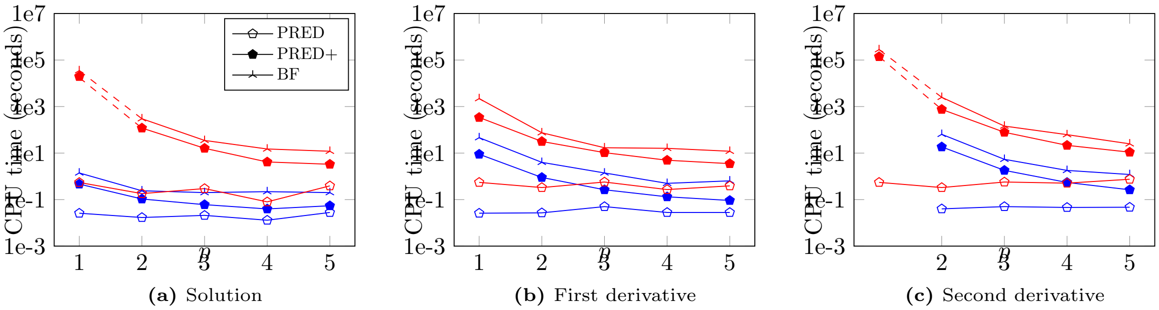
Using the prediction approach and the brute-force approach, $E_{\min}$ are compared in Fig. 6. As can be seen, $E_{\min}$ can be predicted correctly.



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

Fig. 6. Comparison of $E_{\min}$ for Eq. (19) 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. 7. 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. 8 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. 7. Comparison of the CPU time to obtain $E_{\min}$ for Eq. (19) using the algorithm and the brute-force refinement. The blue color denotes the standard FEM, and the red color denotes the mixed FEM.

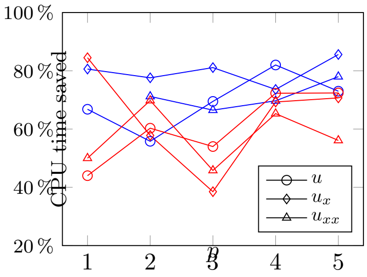


Fig. 8. Percentage of CPU time saved using the algorithm. The blue color denotes the standard FEM, and the red color denotes the mixed FEM.

Furthermore, the dashed line indicating the desired error tolerance in Fig. 6 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.

6. 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 yields

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

By applying Gauss's theorem for the first term of the left-hand side of Eq. (A.1), we obtain

$$(\eta_x,\, du_x) + (\eta,\, ru) = (\eta,\, f) + \left(\eta,\, du_x n\right)_{\Gamma_N}, \tag{A.2}$$

which gives that shown in Eq. (3).

Appendix A.2. The mixed FEM

First, Eq. (6a) is multiplied by a test function of v, i.e. $w \in H^1_{N0}(I)$, and integrated over I, which yields

$$(d^{-1}v + u_x, w) = 0, \tag{A.3a}$$

and then, it becomes

$$(w,\, d^{-1}v) - (w_x,\, u) = -(w,\, gn)_{\Gamma_D}, \tag{A.3b}$$

by applying Gauss's theorem.

Next, Eq. (6b) is multiplied by a test function of u, i.e. $q \in L^2(I)$, and integrated over I, yielding

$$-(q,\, v_x) + (q,\, ru) = (q,\, f). \tag{A.4}$$

Eq. (A.3b) and Eq. (A.4) result in those shown in Eq. (7).

References