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 based on the round-off error for the one-dimensional second-order ordinal differential equations. This method uses a formula to save several computations on fine grids. The application of our method to a complex-valued Helmholtz equation in space shows that the highest attainable accuracy can be accurately predicted, while the CPU time required is much less.

Keywords: Finite Element Method (FEM), differential equation, round-off error, highest attainable accuracy, estimation.

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 model, such as the simplification of the domain and governing equations and the approximation of the initial and boundary conditions; next, truncation errors due to the discretization of

*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 computational domain and 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 error resulting from the artificially controlled tolerance of iterative solvers. We focus on the error led by the truncation and round-off if not stated otherwise.

One tacitly assumes that the two types of errors are well-balanced. That is, 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 are adopted. However, with the popularity of high-order approximations, the round-off error is likely to play a role with only a small number of degrees of freedom (DoFs). 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 [4, 5, 6, 7]. 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 round-off on the error for the solution and its first and second derivative, and propose a practical approach for obtaining the highest attainable accuracy determined by the round-off error ($E_{\min}$). The scope is restricted to one-dimensional second-order model problems, for which the existence of the second derivative of the solution is guaranteed in the weak sense. Moreover, we consider both the standard finite element method (FEM) and mixed FEM[8].

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 error and the approach to predict $E_{\min}$ are discussed in Section 3. The constants used in the approach are determined in Section 4, followed by an algorithm for realizing the approach and its application in Section 5. 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, the diffusion equation is found when $r(x) = 0$, and the Helmholtz equation 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. Here, 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 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 the coordinate 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:

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

with

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

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

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

Imposing the Dirichlet boundary conditions in the weak sense[9], the weak form reads:

Weak form 2

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

$$\begin{aligned}&(\eta_x,\, du_x) + (\eta,\, ru) - (\eta,\, du_x n)_{\Gamma_D} + (\eta_x,\, un)_{\Gamma_D} - (\eta,\, \rho un)_{\Gamma_D} \\ &= (\eta,\, f) + (\eta,\, hn)_{\Gamma_N} + (\eta_x,\, gn)_{\Gamma_D} - (\eta,\, \rho gn)_{\Gamma_D} \qquad \forall \eta \in H^1(I),\end{aligned} \tag{4}$$

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

Note that, the terms in the right-hand sides of Eqs. (3)–(4) consist of information of Neumann boundary conditions which vanish if they are not prescribed.

We approximate u by a linear combination of a finite number of basis functions:

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

Here, m is the number of DoFs (N_h), φ_i are C^0-continuous Lagrange basis functions supported by Gauss-Lobatto points and u_i are the values of u_h at the DoFs. The resulting system of equations reads

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

where A is the stiffness matrix, F the right-hand side and U the discretized u.

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 $v \in H^1_N(I)$ 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 v and u by a linear combination of a finite number of basis functions:

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

$$u \approx u_h = \sum_{j=1}^{p} u_{cj} \psi_j \text{ in cell } c, \tag{8b}$$

where n is the number of DoFs for v_h, φ_i are of the same type of basis functions used in Eq. (5), and v_i are the values of v_h at the DoFs; p is the number of DoFs of u_h in each cell, ψ_j are discontinuous Lagrange basis functions, and $u_{c,j}$ are the values of u_h at the DoFs. Finally, 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.

2.3. Numerical implementation

All results are computed in IEEE-754 double precision [10] using the deal.II finite element code [11].

2.3.1. Solution technique

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. To compute the occurring integrals, sufficiently accurate Gaussian quadrature formulas are used. Furthermore, to solve the system of equations, the UMFPACK solver [12] is used with priority. Last but not least, 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

For the numerical results var_h of the variable var, 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 approximation var_{exc} is available, or [13]

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

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 Q or $\widetilde{Q}$ 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[14].

3. Approach to finding the optimal number of DoFs

3.1. Theoretical evolution of the error

The conceptual sketch of E_h against N_h in the log-log axes can be found in Fig. 1. 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 the above two phases, E_h is controlled by the truncation error E_{T}. In the latter phase, E_h can be represented by

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

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

When N_h is even larger, E_h increases with N_h since it is controlled by the round-off error E_{R}. In this phase, the errors is denoted by the orange circles; the slope for the line approximating E_h tends to be fixed[15, 16], and hence, E_h can be represented by

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

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.

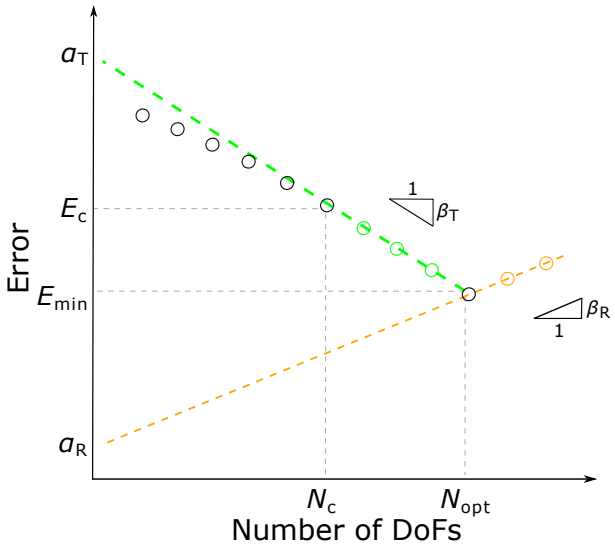


Fig. 1. Conceptual sketch of the error against the number of DoFs.

3.2. Implementation process

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

$$\alpha_T = E_c / {N_c}^{-\beta_T}. \tag{13}$$

After this point, both the development of E_T and E_R are known. Obviously, N_{opt} occurs when $E_T + E_R$ is the smallest. By solving

$$\frac{d(E_T + E_R)}{dN} = 0, \tag{14}$$

we can predict

$$N_{opt} = \left(\frac{\alpha_T \beta_T}{\alpha_R \beta_R} \right)^{\frac{1}{\beta_T + \beta_R}}, \tag{15a}$$

and hence, the highest attainable accuracy

$$E_{min} = \alpha_T {N_{opt}}^{-\beta_T} + \alpha_R {N_{opt}}^{\beta_R}. \tag{15b}$$

4. Determination of the error constants in Fig. 1

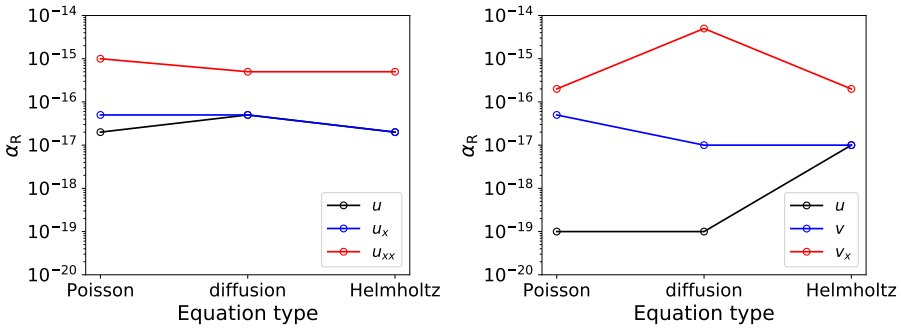
To determine the constants α_R and β_R in Fig. 1, we investigate three benchmark equations using various element degrees, followed by a wide range of second-order differential equations using one particular element degree. Furthermore, we analyse the influence of the solution strategy and boundary condition focusing on one element degree.

4.1. Benchmark equations

The benchmark equations are shown in Table 1, for which the element degree ranges from 1 to 5. The values of α_{R} are shown in Fig. 2, which are as expected when using the double precision[17]. The values of β_{R} are 2 using the standard FEM and 1 using the mixed FEM, which will be taken as constants if not stated otherwise. Note that, α_{R} and β_{R} for one particular variable are basically the same for all the element degrees.

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
$\|f(x)\|_2$	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 $u(x)$	$e^{-(x-1/2)^2}$	$\sin(2\pi x)$	$ae^{(1+i)x}+(1-a)e^{-ix}$, $a=1/((1-i)e^{1+2i}+1)$
$\|u(x)\|_2$	0.92	0.71	1.26



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

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

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

Table 2 An example for the evolution of the order of convergence.

(a) The standard FEM

	Refinement level		
	2	3	4
u	3.97	3.99	4.00
u_x	4.02	4.00	4.00
u_{xx}	3.87	3.98	4.00

(b) The mixed FEM

	Refinement level		
	2	3	4
u	3.96	3.99	4.00
v	4.02	4.00	4.00
v_x	3.86	3.98	4.00

4.2. Wide range of second-order differential equations

First, to cover a wide range of $\|u\|_2$, together with $\|f\|_2$, for the Poisson equation, we investigate the cases shown in Table 3, for which the distribution of $\|u\|_2$ and $\|f\|_2$ can be found in Fig. 3 for Cases 1–4. Second, we investigate various $d(x)$ shown in Table 4 for the diffusion equations with $u = e^{-(x-1/2)^2}$. Last, we consider $r(x)$ from the first five cases of $d(x)$ 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 $\|u\|_2$ and $\|f\|_2$.

Case	$f(x,c)$	$u(x,c)$	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	$-e^{-c(x-1/2)^2}\cdot\big(4c^2(x-1/2)^2-2c\big)$	$e^{-c(x-1/2)^2}$	
5	0	$c^{-1}x$	

For the three types of equations, α_{R} for different unknowns are shown in Fig. 4 – Fig. 6. Note that, the variable in the x axis is $\|u\|_2$ for u and v, and $\|v\|_2$ for v_x in Fig. 4(b).

For the Poisson equations, α_{R} is linearly proportional to the variable in the x axis, for which the ratios are 2e-17, 5e-17 and 5e-16 for u, u_x and u_{xx} respectively using the standard FEM, and 1e-18, 1e-16 and 5e-16 for u, v and v_x respectively using the mixed FEM. To make α_{R} independent of the unknowns, we propose the scaling schemes shown in Table 5, which allow us to recover the above constants irrespective of the magnitude of the unknowns. Note that, two schemes are needed for the mixed FEM: M_1 for u and v_x, and M_2 for v_x.

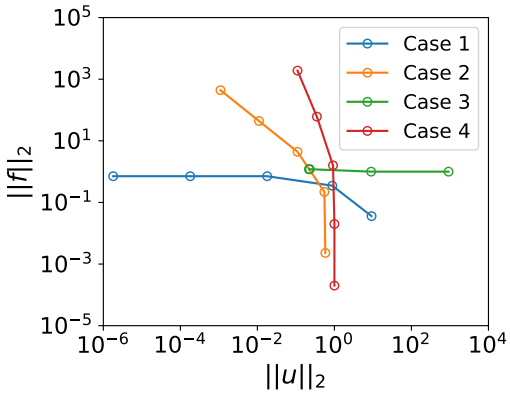
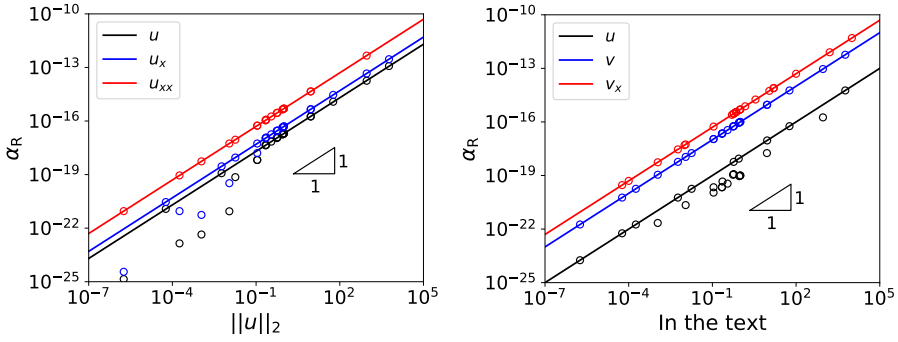


Fig. 3. Distribution of $\|u\|_2$ and $\|f\|_2$ for the Poisson equations in Table 3.

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

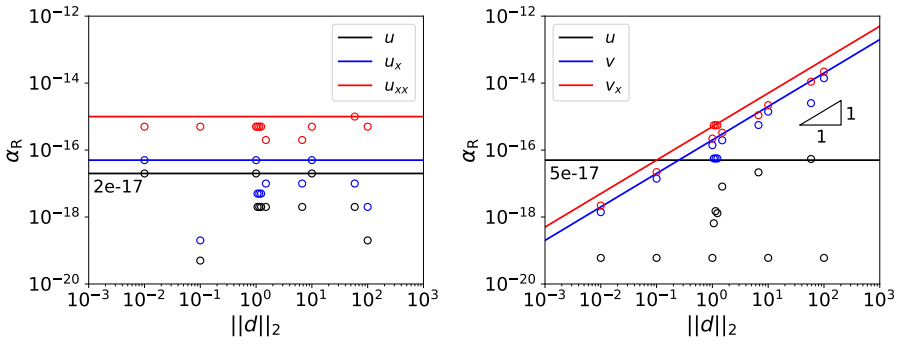
#	$d(x)$	$\|d(x)\|_2$	#	$d(x)$	$\|d(x)\|_2$
1	0.01	0.01	7	$1+\sin(10x)$	1.14
2	0.1	0.1	8	$1+\sin(100x)$	1.06
3	1	1	9	$1+x$	1.5
4	10	10	10	$1+10x$	6.7
5	100	100	11	$1+100x$	58.6
6	$1+\sin(x)$	1.23			



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

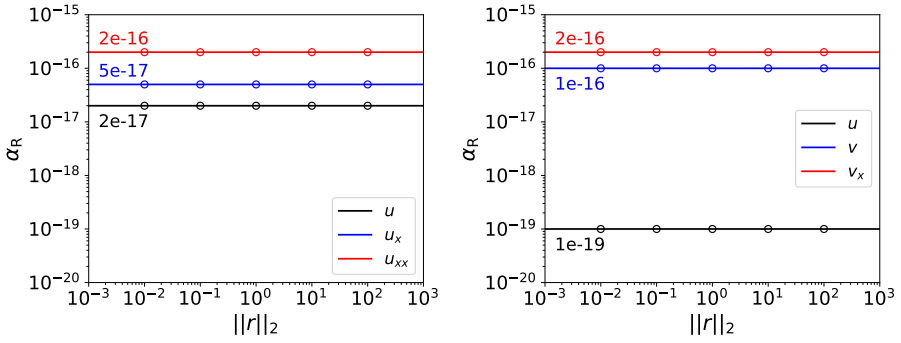
Fig. 4. α_{R} for the Poisson equations in Table 3.

For the diffusion equations, α_{R} is linearly proportional to $\|d(x)\|_2$ for v and v_x using the mixed FEM, for which the ratios are 2e-16 and 5e-16, respectively, while it is relatively independent of $d(x)$ in other scenarios. For the Helmholtz equations, α_{R} is independent of $r(x)$.



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

Fig. 5. α_{R} for the diffusion equations with $u = e^{-(x-1/2)^2}$ and various $\|d\|_2$ in Table 4.



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

Fig. 6. α_{R} for the Helmholtz equations with $u = e^{-(x-1/2)^2}$, $\|d\|_2 = 1$ and $r(x)$ taken from the first five cases of $d(x)$ in Table 4.

Table 5 System of equations using various scaling schemes.

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

In summary, by using the scaling schemes in Table 5, we obtain α_{R} in Table 6 that are independent of the unknowns. Note that, these numbers are also valid for the benchmark equations.

Table 6 Variable-independent α_{R} for the second-order differential equations.

(a) The standard FEM

u	2e-17
u_x	5e-17
u_{xx}	1e-15

(b) The mixed FEM

u	5e-17
v	2e-16 $\times \|\|d(x)\|\|_2$
v_x	5e-16

4.3. Sensitivity analysis

We focus on the benchmark Poisson equation.

4.3.1. Boundary conditions

We consider two aspects for the influence of boundary conditions: the type of boundary conditions and the weak imposition of the Dirichlet boundary condition using the standard FEM.

For the first part, the Dirichlet boundary condition at the left boundary ($x = 0$) is kept while that 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. For the standard FEM with P_2 elements, using the Dirichlet/Neumann boundary condition, the offset α_{R} is slightly larger than that using the Dirichlet/Dirichlet boundary condition for u and u_x, by a factor of 3.5 and 2 respectively; α_{R} is equal to that using the Dirichlet/Dirichlet boundary condition for u_{xx}. For the mixed FEM with P_4/P_3^{disc} elements, using the Dirichlet/Neumann boundary condition, α_{R} increases to 3e-17 for u, is 5 times smaller for v, and does not change for v_x.

For the second part, the weak imposition of the Dirichlet boundary condition produces the same error when the penalty parameter is large enough, e.g. 10^6.

In summary, the boundary condition basically makes no difference to the values of α_{R} we proposed in Table 6.

4.3.2. Solution strategy

The alternative solution method to the UMFPACK solver is the iterative Conjugate Gradient (CG) method[18], which can be applied when the left-hand side is symmetric and positive definite. We focus on the tolerance of the CG solver, denoted by tol_{prm}: the iteration stops when the norm of the residual is smaller than it.

For the standard FEM, the CG method can be applied directly. However, for the mixed FEM, since the left-hand side of Eq. (9) is indefinite, this method is used after segregating Eq. (9) based on the Schur

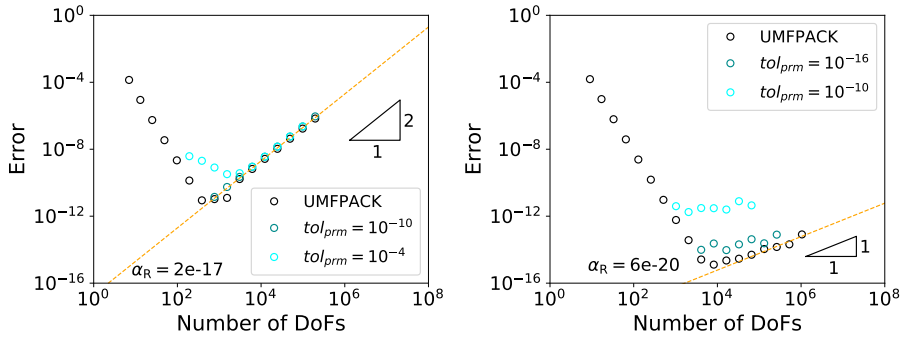
complement, which results in

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

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

In the solution process of Eq. (16), the CG solver can be used for the left-hand side being either $B^\top M^{-1} B$ (Schur complement) or M. In particular, we investigate the former while the UMFPACK solver is used for the left-hand side being M.

Focusing on u with the approximation order of 3, the evolution of the error for various tol_{prm} using the CG solver is illustrated in Fig. 7, in comparison with that using the UMFPACK solver. It shows that the CG solver introduces iteration errors when tol_{prm} is less strict. Specially, for the mixed FEM, the accuracy using the CG solver can not be as high as that using the UMFPACK solver. This proves the correctness of choosing the UMFPACK solver to obtain higher accuracy.



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

Fig. 7. Comparison of the errors using the CG solver and the UMFPACK solver for u.

5. Algorithm and its application

Based on the approach given in section 3, and scaling schemes and error constants provided in section 4, we introduce a practical algorithm and apply it to a complex-valued Helmholtz equation.

5.1. Algorithm

In this algorithm, we define the following coefficients and use them in the steps given below.

– a minimal number of h-refinements as a precondition, denoted by $R_{\min}$, with the following default values:

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

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

– a stopping criterion c_s for seeking the scaling factor $\|var\|_2$ in Table 5, which is 0.001 by default.

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

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

Step-1. '*INPUT*'. In this step, the custom input shown in Table 7 has to be provided.

Table 7 Custom input of the algorithm.

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

Step-2. '*NORMALIZATION*'. This step is used to find the scaling factors to normalize the variable of interest since the scaling factors do not exist for most practical problems. The specific procedure can be found in Algorithm 1.

Algorithm 1: NORMALIZATION

```
while N_h < N_max do
    if |(||var_h||_2 − ||var_2h||_2) / ||var_h||_2| < c_s then
        ||var||_2 ← ||var_h||_2;
        break;
    else
        h ← h/2;
        calculate ||var_h||_2;
    end
end
```

Step-3. '*PREDICTION*'. This step illustrates how to find $E_{\min}$ based on the order of convergence. The procedure for carrying out this step can be found in Algorithm 2.

Algorithm 2: PREDICTION

while $\widetilde{E_h} > E_{\mathrm{R}}$ **and** $N_h < N_{\max}$ **do**
  $\widetilde{Q} \leftarrow \log_2\left(\widetilde{E_{2h}}/\widetilde{E_h}\right)$;
  if $\widetilde{Q} \geqslant \beta_{\mathrm{T}} \times c_r$ **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 $\widetilde{E_h}$ using Eq. (10b) with proper scaling schemes;
  end
end

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

5.2. Application

The problem reads[19]:

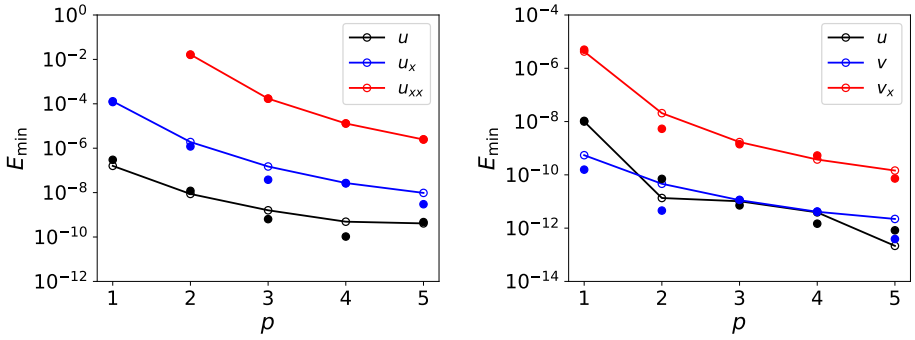
$$((0.01+x)(1.01-x)u_x)_x - (0.01i)u(x) = 1.0, \qquad x \in I = (0,1), \tag{19}$$

with boundary conditions imposed as follows: $u(0) = 0$ and $u_x(1) = 0$.

For the element degree p being $\{1, 2, \ldots, 5\}$ using both the standard and mixed FEMs, $E_{\min}$ predicted by the algorithm are given in Fig. 8. They fits that using the brute-force approach well. However, the CPU time is saved much using the algorithm, see Fig. 9 for the CPU time used and Fig. 10 for the percentage of CPU time saved using the algorithm. Basically, the CPU time can be saved more than 60% and 40% for the standard FEM and the mixed FEM, respectively.

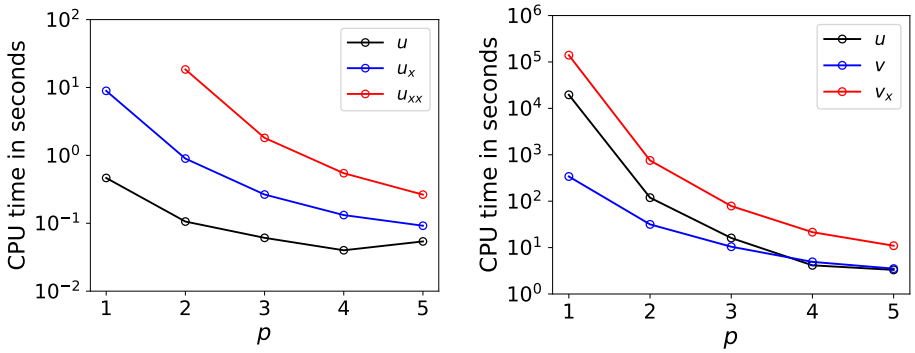
6. Conclusions

A novel approach is presented to predict the highest attainable accuracy for one-dimensional second-order ordinary differential equations using the finite element methods. In contrast to the brute-force approach, which uses monolithic h-refinements, this approach uses only a few coarse grid refinements. This approach takes advantage of the properpty of the order of convergence and the bound for the round-off error, and it is viable for the solution and its first and second derivative, using both the standard FEM and the mixed



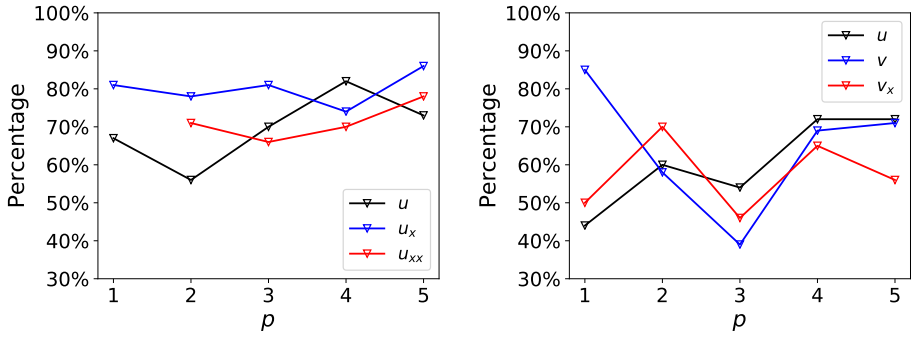
(a) The standard FEM **(b)** The mixed FEM

Fig. 8. $E_{\min}$ for Eq. (19) using the algorithm. The filled circle denotes results using the brute-force refinement.



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

Fig. 9. CPU time used to obtain $E_{\min}$ for Eq. (19) using the algorithm.



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

Fig. 10. Percentage of CPU time saved to obtain $E_{\min}$ for Eq. (19) using the algorithm.

FEM with 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 two-dimensional second-order problems, where the influence of the linear system solver, local mesh refinement and boundary conditions might be significantly different from one-dimensional 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

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

By applying Gauss's theorem, we obtain

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

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

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

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

$$(w,\, d^{-1}v) - (w_x,\, u) = -(w,\, gn)_{\Gamma_D}. \tag{A.4b}$$

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

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

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

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] Fuyun Ling and J Proakis. Numerical accuracy and stability: Two problems of adaptive estimation algorithms caused by round-off error. In *Acoustics, Speech, and Signal Processing, IEEE International Conference on ICASSP'84.*, volume 9, pages 571–574. IEEE, 1984.

[5] Shan-Cong Mou, Yu-Xuan Luan, Wen-Tao Ji, Jian-Fei Zhang, and Wen-Quan Tao. An example for the effect of round-off errors on numerical heat transfer. *Numerical Heat Transfer, Part B: Fundamentals*, 72(1):21–32, 2017.

[6] Mark Ainsworth and J Tinsley Oden. A procedure for a posteriori error estimation for hp finite element methods. *Computer Methods in Applied Mechanics and Engineering*, 101(1-3):73–96, 1992.

[7] 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.

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

[9] 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.

[10] 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.

[11] 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.

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

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

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

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

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

[17] Estimate of the condition number. https://www.alglib.net/matrixops/rcond.php. Accessed: 2020-3-15.

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

[19] Alexander S Chernetsky, Henk M Schuttelaars, and Stefan A Talke. The effect of tidal asymmetry and temporal settling lag on sediment trapping in tidal estuaries. *Ocean Dynamics*, 60(5):1219–1241, 2010.