Quantum Science and Technology

OPEN ACCESS

RECEIVED
23 May 2025

REVISED
18 December 2025

ACCEPTED FOR PUBLICATION
16 January 2026

PUBLISHED
2 February 2026

PAPER

Quantum finite element algorithm for solving Euler–Bernoulli and heat transfer PDEs with Dirichlet, Neumann, and Robin boundary conditions

Dingjie Lu*, Zhao Wang, Jun Liu, Yangfan Li, Wei-Bin Ewe and Zhuangjian Liu*

Institute of High Performance Computing IHPC, Agency for Science, Technology and Research (A*STAR), Singapore 138632, Singapore

* Authors to whom any correspondence should be addressed.

E-mail: ludj@a-star.edu.sg and liuzj@a-star.edu.sg

Keywords: variational quantum algorithm, partial differential equations, finite element method

Abstract

This paper introduces a quantum-enhanced finite element method (FEM) designed for noisy intermediate-scale quantum (NISQ) devices, leveraging variational quantum algorithms (VQAs) to solve engineering partial differential equations. We demonstrate the framework by solving the Euler–Bernoulli beam and the NAFEMS T4 heat transfer problems, which involve Dirichlet, Neumann, and Robin boundary conditions. A key innovation is a 'set-to-zero' strategy that incorporates boundary conditions through a correction matrix, K_{bc}, allowing for flexible imposition at any node without domain decomposition. The global stiffness matrix is decomposed into a constant number of Pauli terms, $O(1)$, using the method by Sato *et al* while boundary terms are handled with a sublinearly scaling partial Pauli measurement technique. The algorithm achieves logarithmic qubit scaling ($n = \lceil \log_2 N \rceil$ qubits for N degrees of freedom(DOF)) and employs shallow, hardware-efficient circuits with empirically trainable depth for small-scale systems. Validation on the Qiskit statevector simulator shows high accuracy. For the Euler–Bernoulli beam problem with 4 to 64 DOF, the algorithm achieves relative errors of 0.5%–1.5% and fidelities of 0.998–0.999. For the NAFEMS T4 heat transfer benchmark, a 5.4% relative error is observed. The VQA converges robustly within 77–350 iterations, though barren plateaus are a known challenge for scaling to larger systems. This work establishes a scalable framework for quantum FEM, offering a significant memory advantage over classical methods and advancing the potential for quantum-enhanced engineering simulations.

1. Introduction

Quantum computing offers unprecedented potential for solving large-scale partial differential equations (PDEs) that govern engineering phenomena, from structural mechanics to heat transfer [1–8]. The exponential growth of quantum state space enables encoding solutions from $N = 2^n$ grid points using only n qubits, promising dramatic computational advantages over classical methods [9, 10]. Among quantum approaches suitable for near-term devices, variational quantum algorithms (VQAs) have emerged as particularly promising due to their resilience to noise and shallow circuit requirements [11, 12].

Recent advances in quantum algorithms for PDEs can be broadly categorized into three approaches. Variational methods, pioneered by Lubasch *et al* [13] with the quantum nonlinear processing unit framework, have been successfully extended to multidimensional nonlinear PDEs [14] and computational fluid dynamics with structure-resolving schemes [15]. Hamiltonian simulation methods have achieved notable success, including hardware implementations of wave equations [16], scalable circuits for hyperbolic PDEs [17], and the innovative Schrödingerization technique that transforms general linear PDEs into quantum-solvable forms [18]. Time-marching algorithms have addressed advection-diffusion problems

through Hamiltonian embedding [19] and block encoding with correction terms [20]. These approaches demonstrate quantum computing's potential for diverse PDE classes.

Despite this progress, most quantum PDE algorithms employ finite difference methods (FDMs) for spatial discretization, leveraging local stencils for efficient quantum circuit design [16–22]. Both FDM and finite element method (FEM) offer distinct advantages for solving PDEs. FDM provides robust and efficient solutions for mixed boundary conditions and variable materials, with recent quantum implementations successfully demonstrating these capabilities [23, 24]. FEM excels particularly in handling complex geometries and non-uniform mesh refinement, allowing for higher accuracy in specific regions through adaptive meshing. In classical FEM, boundary conditions are imposed through direct modification of the global stiffness matrix K, adjustment of the load vector $\mathbf{f}$, or a combination thereof. Our quantum FEM framework complements existing quantum FDM approaches by providing systematic incorporation of higher-order basis functions, and flexibility in mesh topology for complex engineering geometries. Both methods have their place in the quantum PDE solver toolkit, with the choice depending on specific problem requirements. Implementing FEM on quantum computers presents unique challenges absent in FDM approaches. First, the global stiffness matrix K in FEM is assembled from element contributions, resulting in non-local coupling that is challenging to represent using simple shift operators (also known as adder operators in some quantum literature [21]) as in FDM [25]. Second, imposing boundary conditions requires modifying the global matrix structure while preserving quantum encoding efficiency. Third, the load vector $\mathbf{f}$ containing boundary condition information must be efficiently encoded in quantum amplitudes [26, 27]. These challenges have likely contributed to the absence of quantum FEM algorithms in the literature, despite FEM's advantages for engineering applications.

This paper presents a quantum finite element algorithm for solving the Euler–Bernoulli beam and heat transfer PDEs with Dirichlet, Neumann, and Robin boundary conditions. We address the fundamental challenges of quantum FEM implementation through two key innovations. First, we develop an efficient Pauli decomposition of the global stiffness matrix into $O(1)$ terms independent of system size [22, 25], enabling efficient quantum evaluation of $\langle\phi|K|\phi\rangle$. Second, we introduce the 'set-to-zero' strategy for boundary conditions that modifies the stiffness matrix through a correction term K_{bc}. Neumann and Robin conditions are supported via quantum oracles, enabling mixed boundary conditions without domain decomposition.

Our approach differs fundamentally from finite difference boundary treatments [21] that modify local stencils. Instead, we derive K_{bc} from the global stiffness matrix structure, ensuring that the modified system preserves the variational properties essential to FEM. This enables handling of boundary conditions at any nodes in the discretization domain. The quantum implementation employs logarithmic qubit scaling and shallow circuits designed for noisy intermediate-scale quantum (NISQ) compatibility [2, 28]; however, scalability and trainability for global cost functions remain open research challenges [12].

We validate our quantum FEM algorithm against industry-standard NAFEMS benchmarks [29], demonstrating accurate solutions for both the Euler–Bernoulli beam with point loads and 2D heat conduction with mixed boundary conditions. Following established benchmarking practices for quantum algorithms [30], we evaluate our method using both quantum-specific metrics and classical validation standards. The remainder of this paper is organized as follows: section 2 presents the theoretical framework for reformulating PDEs into quantum-compatible form within the FEM framework. Sections 3 and 4 details the quantum circuit implementation for the Euler–Bernoulli beam problem and the NAFEMS T4 heat transfer problem. Section 5 provides comprehensive results including convergence analysis, accuracy validation, and resource scaling. Finally, section 6 summarizes our findings and outlines future directions for quantum-enhanced engineering simulations.

2. Formulation of the minimization problem

The application of VQAs to solve partial differential equations (PDEs) requires recasting the governing equations into a minimization problem suitable for quantum implementation. This section details the derivation of a quantum-compatible objective functional, starting from a general linear elliptic PDE, through its variational form and finite element discretization, with explicit handling of Dirichlet, Neumann, and Robin boundary conditions.

2.1. General PDE and variational formulation

Let $\Omega \subset \mathbb{R}^d$ be a bounded domain with boundary $\partial\Omega$ partitioned into disjoint subsets Γ_{D}, Γ_{N}, and Γ_{R} for Dirichlet, Neumann, and Robin conditions, respectively. We consider a linear, self-adjoint, elliptic

differential operator of order $2m$ [31] (p 117):

$$Lu = \sum_{|\alpha|\leqslant m}\sum_{|\beta|\leqslant m}(-1)^{|\alpha|}D^{\alpha}\left(a_{\alpha\beta}(x)D^{\beta}u\right), \tag{1}$$

where α,β are multi-indices and the coefficients $a_{\alpha\beta}(x)$ ensure ellipticity [32](p 488). For the Euler–Bernoulli equation ($m=2$), Lu models beam bending; for the heat Equation ($m=1$), it models steady-state heat transfer. The strong form of the PDE is $Lu=r$ in Ω, supplemented by the boundary conditions

$$u = g_{\mathrm{D}} \quad \text{on } \Gamma_{\mathrm{D}} \quad \text{(Dirichlet)}, \tag{2}$$

$$B_{\mathrm{N}}[u] = g_{\mathrm{N}} \quad \text{on } \Gamma_{\mathrm{N}} \quad \text{(Neumann)}, \tag{3}$$

$$B_{\mathrm{R}}[u] = \alpha u + B_{\mathrm{N}}[u] = g_{\mathrm{R}} \quad \text{on } \Gamma_{\mathrm{R}} \quad \text{(Robin)}, \tag{4}$$

where $r, g_{\mathrm{D}}, g_{\mathrm{N}}, g_{\mathrm{R}}$ are prescribed functions and $B_{\mathrm{N}}[u]$ represents normal derivative terms.

To derive the weak form, we multiply the PDE by a test function $v \in V = \{v \in H^m(\Omega) : v = 0 \text{ on } \Gamma_{\mathrm{D}}\}$, integrate by parts, and substitute the Neumann and Robin conditions. This yields the variational problem: find $u \in U = \{w \in H^m(\Omega) : w = g_{\mathrm{D}} \text{on } \Gamma_{\mathrm{D}}\}$ such that

$$a(u,v) = \ell(v) \quad \forall v \in V, \tag{5}$$

where $a(u,v)$ is a bilinear form representing the system's internal energy and $\ell(v)$ is a linear functional for the work of external forces, defined as:

$$a(u,v) = \int_{\Omega}\sum_{|\alpha|\leqslant m}\sum_{|\beta|\leqslant m} a_{\alpha\beta}(x)\left(D^{\beta}u\right)\left(D^{\alpha}v\right)\mathrm{d}\Omega + \int_{\Gamma_{\mathrm{R}}}\alpha uv\,\mathrm{d}\Gamma, \tag{6}$$

$$\ell(v) = \int_{\Omega} rv\,\mathrm{d}\Omega + \int_{\Gamma_{\mathrm{N}}} g_{\mathrm{N}}v\,\mathrm{d}\Gamma + \int_{\Gamma_{\mathrm{R}}} g_{\mathrm{R}}v\,\mathrm{d}\Gamma. \tag{7}$$

The Robin condition contributes to both the bilinear form (stiffness) and the linear functional (load).

In the classical setting, this variational problem is equivalent to minimizing the quadratic energy functional

$$\Pi(u) = \tfrac{1}{2}a(u,u) - \ell(u) \tag{8}$$

in the space V, where $a(\cdot,\cdot)$ is symmetric, continuous, and coercive [33, 34]. This guarantees a unique global minimizer $u \in V$, corresponding to the weak solution of the PDE. After finite-element discretization, the solution $u_h = \sum_{i=1}^{N} v_i\phi_i$ yields a strictly convex quadratic program in the nodal vector $\mathbf{v} \in \mathbb{R}^N$, and the effective stiffness matrix K_{eff} is symmetric positive definite (SPD), ensuring well-posedness.

In the quantum formulation, rather than optimizing the components v_i directly, we represent the solution through the amplitudes of a parametrized quantum state $|\phi(\boldsymbol{\theta})\rangle = U(\boldsymbol{\theta})|0\rangle$. The corresponding variational loss $\Pi(\boldsymbol{\theta})$ inherits the classical energy structure but becomes non-convex in the parameters $\boldsymbol{\theta}$. Thus, while the SPD property of K_{eff} ensures a well-defined physics-informed objective, classical convexity guarantees do not directly carry over to the variational parameter space. In practice, for the problem sizes considered here ($N \leqslant 64$), the resulting optimization landscape proved tractable, and we consistently observed convergence to physically meaningful solutions. This behavior aligns with other VQA-based studies for linear systems and PDEs, and suggests that further advances in problem-inspired ansätze and scalable optimization strategies will continue to enhance robustness as system size increases [12, 35, 36].

2.2. Finite element discretization and boundary conditions

The FEM approximation discretizes the solution as $u_h = \sum_{i=1}^{N} v_i\phi_i$, where $\{\phi_i\}_{i=1}^{N}$ are basis functions and $\mathbf{v} = [v_1,\ldots,v_N]^{\mathrm{T}}$ are the nodal coefficients. Applying the Galerkin method transforms the weak form into a linear system $K\mathbf{v} = \mathbf{f}$. The entries of the stiffness matrix K and load vector $\mathbf{f}$ are:

$$K_{ij} = a(\phi_i,\phi_j) = \int_{\Omega}\sum_{|\alpha|,|\beta|\leqslant m} a_{\alpha\beta}\left(D^{\beta}\phi_i\right)\left(D^{\alpha}\phi_j\right)\mathrm{d}\Omega + \int_{\Gamma_{\mathrm{R}}}\alpha\phi_i\phi_j\,\mathrm{d}\Gamma, \tag{9}$$

$$f_j = \ell(\phi_j) = \int_{\Omega} r\phi_j\,\mathrm{d}\Omega + \int_{\Gamma_{\mathrm{N}}} g_{\mathrm{N}}\phi_j\,\mathrm{d}\Gamma + \int_{\Gamma_{\mathrm{R}}} g_{\mathrm{R}}\phi_j\,\mathrm{d}\Gamma. \tag{10}$$

Boundary conditions are systematically incorporated. Neumann conditions contribute directly to the load vector $\mathbf{f}$ through the integral over Γ_N. Robin conditions add a matrix K_{Robin} (from the Γ_R integral in equation (9)) to the stiffness matrix and contribute to $\mathbf{f}$ (from the Γ_R integral in equation (10)). Dirichlet conditions are enforced using our 'set-to-zero' strategy, which adds a correction matrix K_{bc} to K and adjusts the load vector to a final form $\mathbf{f}_1$ (see appendix B for details).

The final discrete energy functional to be minimized is:

$$\Pi_{\text{d}}(\mathbf{v}) = \frac{1}{2}\mathbf{v}^{\text{T}}(K + K_{\text{bc}} + K_{\text{Robin}})\mathbf{v} - \mathbf{v}^{\text{T}}\mathbf{f}_1, \tag{11}$$

where the effective stiffness matrix $K_{\text{eff}} = K + K_{\text{bc}} + K_{\text{Robin}}$ is SPD.

2.3. Quantum formulation and optimization

To solve the minimization problem on a quantum computer, we map the discrete vectors to quantum states using amplitude encoding, which requires $n = \lceil \log_2 N \rceil$ qubits for N degrees of freedom (DOF). The solution vector is parameterized as $\mathbf{v} = c\boldsymbol{\phi}$, where $\|\boldsymbol{\phi}\| = 1$, and encoded into a quantum state prepared by a parameterized ansatz, $|\phi(\boldsymbol{\theta})\rangle = U(\boldsymbol{\theta})|0\rangle^{\otimes n}$. The final load vector $\mathbf{f}_1$, which incorporates all boundary data, is normalized and encoded as the state $|f_1\rangle$ [26, 27].

The effective stiffness matrix $K_{\text{eff}} = K + K_{\text{bc}} + K_{\text{Robin}}$ is decomposed into a sum of Pauli operators for measurement on quantum hardware. The VQA minimizes the energy functional of equation (11). We use a hybrid quantum–classical approach based on the Rayleigh–Ritz method, where the scaling factor c can be optimized analytically for a fixed $\boldsymbol{\theta}$:

$$c^*(\boldsymbol{\theta}) = \frac{\langle \phi(\boldsymbol{\theta}) | f_1 \rangle}{\langle \phi(\boldsymbol{\theta}) | K_{\text{eff}} | \phi(\boldsymbol{\theta}) \rangle}.$$

Substituting this back into the energy functional yields the final objective function to be minimized with respect to the ansatz parameters $\boldsymbol{\theta}$:

$$\Pi_{\text{d}}(\boldsymbol{\theta}) = -\frac{1}{2}\frac{|\langle \phi(\boldsymbol{\theta}) | f_1 \rangle|^2}{\langle \phi(\boldsymbol{\theta}) | K_{\text{eff}} | \phi(\boldsymbol{\theta}) \rangle}. \tag{12}$$

The overlap $\langle \phi(\boldsymbol{\theta}) | f_1 \rangle$ is typically measured using the Hadamard test. The denominator $\langle \phi(\boldsymbol{\theta}) | K_{\text{eff}} | \phi(\boldsymbol{\theta}) \rangle$ is evaluated using the partial Pauli measurement (PPM) technique described in section 3.1. As $K_{\text{eff}} = K + K_{\text{bc}} + K_{\text{Robin}}$, the bulk stiffness matrix K is represented with $\mathcal{O}(1)$ Pauli terms following the construction of Sato *et al* [25], while the boundary contributions K_{bc} and K_{Robin} introduce at most $\mathcal{O}(\sqrt{N})$ Pauli terms. These terms are grouped and measured efficiently under the PPM framework. The optimization proceeds in a hybrid loop: expectation values are estimated on quantum hardware, and a classical optimizer (e.g. BFGS) updates the parameters $\boldsymbol{\theta}$. The process is iterated until convergence, yielding the solution $\mathbf{v} = c^*(\boldsymbol{\theta}^*)|\phi(\boldsymbol{\theta}^*)\rangle$.

3. Implementation on Euler–Bernoulli beam problem

We apply our quantum FEM to the Euler–Bernoulli beam problem, a fourth-order PDE ($m = 2$) given by $EI\frac{\mathrm{d}^4 w}{\mathrm{d}x^4} = q$, where w is deflection, E is Young's modulus, I is the moment of inertia, and q is a distributed load. The problem is discretized and mapped to the quantum objective functional in equation (12). The solution vector $\mathbf{v}$ and load vector $\mathbf{f}_1$ are encoded using $n = \lceil \log_2 N \rceil$ qubits, with the solution state $|\phi(\boldsymbol{\theta})\rangle$ prepared by the Qiskit RealAmplitudes [2, 37] ansatz as shown in figure 3.

3.1. Stiffness matrix and boundary conditions

The element stiffness matrix K_e for a beam element with normalized parameters ($E = I = l_e = 1$) is a 4×4 matrix decomposed into six Pauli terms (see appendix A). Following Sato *et al* [25], the global stiffness matrix K is assembled from three components, $K = K_a + K_b - K_c$, constructed using shift operators, which maintains an $O(1)$ Pauli decomposition independent of system size.

Boundary conditions are handled by the framework in section 2.2. For the beam problems in section 5, we focus on Dirichlet conditions (e.g. cantilever, fixed-fixed), enforced via the 'set-to-zero' strategy which generates a correction matrix K_{bc}. For a cantilever beam fixed at its first node, K_{bc} modifies the first two rows and columns of the global stiffness matrix to enforce zero displacement and rotation (see appendix B for an example). The expectation of the effective stiffness matrix $\langle \phi | (K + K_{\text{bc}}) | \phi \rangle$ is computed by measuring the $O(1)$ terms for K and additional terms for K_{bc} using our PPM method [38]. The number of PPM terms scales with the number of constrained DOF, which is a small constant for typical beam problems, ensuring sublinear overhead.

4. Implementation on 2D heat transfer problem

We extend our method to the NAFEMS T4 benchmark [29], a 2D steady-state heat transfer problem ($m = 1$) governed by $-\nabla \cdot (k\nabla T) = 0$, where k is thermal conductivity and T is temperature. The domain is discretized with linear quadrilateral elements on a 5×3 mesh, resulting in 24 DOF, which requires $n = \lceil \log_2 24 \rceil = 5$ qubits for amplitude encoding. The VQA framework, including the RealAmplitudes ansatz, follows that of the beam problem.

4.1. Stiffness matrix and boundary conditions

The element stiffness matrix K_e for a linear quadrilateral element is a 4×4 matrix decomposed into four Pauli terms (see appendix A). To assemble the global stiffness matrix K for a 2D mesh, the elements are grouped into four tessellations. Using shift operators in both x- and y-directions, the full matrix K is constructed as a sum of four matrices, maintaining an $O(1)$ Pauli decomposition as in the 1D case [25].

This benchmark involves mixed boundary conditions: Dirichlet ($T = 100\,^\circ$C), Neumann (zero heat flux), and Robin (convective cooling) as shown in figure 2. These are incorporated as described in section 2.2. The Dirichlet condition is enforced with K_{bc} via the 'set-to-zero' strategy. The Neumann condition modifies the load vector $\mathbf{f}_1$. The Robin condition adds a matrix K_{Robin} and also contributes to $\mathbf{f}_1$. The boundary correction matrices $K_{bc} + K_{Robin}$ are handled using PPM. The number of terms scales as $O(\sqrt{N})$ with the number of boundary nodes, which is efficient for 2D problems.

5. Results and discussion

This section presents numerical results for the Euler–Bernoulli beam problem and the NAFEMS T4 heat transfer benchmark, validating the accuracy, convergence, and computational efficiency of our quantum-enhanced FEM using the Qiskit statevector simulator. The VQA outlined in section 2 was applied to both problems, with implementations detailed in sections 3 and 4. We evaluate solution accuracy through relative errors, root mean square error (RMSE), normalized RMSE, and fidelity against analytical or benchmark solutions, assess convergence via mesh refinement and BFGS optimization iterations, and demonstrate logarithmic qubit scaling ($n = \lceil \log_2 N \rceil$). For the problems studied ($N \leqslant 64$), we observe that the RealAmplitudes ansatz with 3–5 repetitions (circuit depth $d_a = 7$ to 11) provides sufficient expressibility while remaining trainable, supporting shallow, hardware-efficient circuits designed for NISQ compatibility. The ansatz $U(\boldsymbol{\theta})$ uses Qiskit's RealAmplitudes circuit [39], consisting of alternating single-qubit RY rotations and linear-entangling CNOT layers. All simulations reported herein use noiseless statevector sampling to validate the correctness of the quantum FEM framework; performance under shot noise and on real quantum hardware, including noise-robust optimization strategies, is the subject of our ongoing research. BFGS optimization was run for up to 500 iterations with a convergence tolerance of 10^{-4}. For all the cases no gradient was feed to the optimizer and $\boldsymbol{\theta}$ were initialized randomly at start. While barren plateaus are a known challenge in VQAs with global cost functions [12], our empirical results for $n \leqslant 6$ show robust convergence. However, these results are heuristic rather than guaranteed at scale. Future work will investigate local-cost formulations and problem-tailored ansätze to address potential scalability limitations.

5.1. Euler–Bernoulli beam results

The Euler–Bernoulli beam problem, a fourth-order PDE, was solved with unitless parameters: length $L = 10.0$, Young's modulus $E = 1.0$, and second moment of inertia $I = 1.0$, ensuring physical consistency across four boundary conditions: periodic (PBC), cantilever, simply-supported (SSB), and fixed-fixed (FFB), as shown in figure 1 (SCHEMATIC). A mesh convergence study was conducted with qubit counts from $n = 2$ (4 DOFs) to $n = 6$ (64 DOFs), evaluating loss convergence, deflection profiles, and rotation profiles against analytical solutions. The stiffness matrix K was decomposed into $O(1)$ Pauli terms [25], and boundary corrections K_{bc} were handled via our PPM method [38]. The number of terms in K_{bc} depends on the affected DOFs.

Figure 1 presents results in matrix format, with rows for PBC, SSB, FFB, and cantilever, and columns for schematic, loss convergence (LOSS), deflection (DEFLECTION), and rotation (ROTATION). Figure 1 (LOSS) shows the temporal evolution of the objective function, converging toward target loss values that differ markedly by boundary condition: $PBC \approx 0.003$, $FFB \approx 0.002$, $SSB \approx 0.01$, and $cantilever \approx 0.15$. These target values are the exact minimum of the classical energy functional after the respective BCs are applied. The raw stiffness matrix K (before any boundary correction) is identical for all four cases because the mesh, geometry, and material properties are the same. The difference in performance manifest the physical effects of the boundary constraints. The PBC case converges fastest (≈77 iterations),

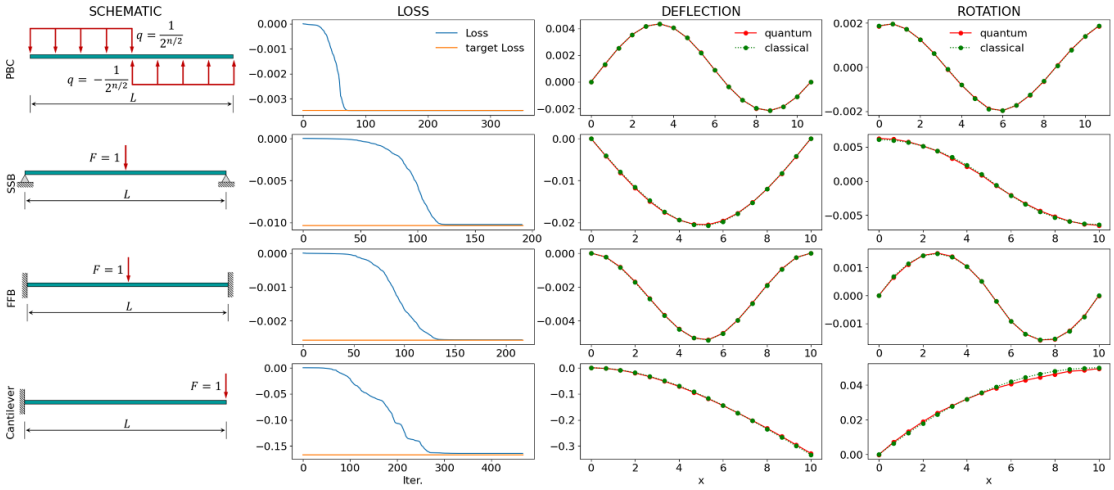

Figure 1. Schematics, loss convergence history, deflection (unit: 1), and rotation (unit: rad) for four representative Euler–Bernoulli beam problems with Dirichlet, Neumann, and Robin boundary conditions. First column (SCHEMATIC): Illustrates the four boundary configurations: PBC (pinned–pinned), SSB (simply supported–built-in), FFB (free–free), and Cantilever. Second column (LOSS): Temporal evolution of the quantum objective function during optimization iterations, converging toward the target loss (yellow solid line). The discrepancy between converged loss and target loss is quantitatively analyzed in figure 4 (LEFT). Third column (DEFLECTION) and fourth column (ROTATION): Simulated beam profiles for deflection and rotation. The red solid line represents results from the quantum finite element algorithm (QFEM), while the green dotted line corresponds to classical FEM solutions. Quantitative error metrics (RMSE and fidelity) are reported in figure 4 (MID, RIGHT).

consistent with findings by Pellow *et al* [40] for symmetric problems. The cantilever case requires up to 350 iterations due to its asymmetric fixed-free boundary, while SSB and FFB converge in ≈150 iterations, reflecting intermediate complexity. Figure A2 (appendix D) confirms that even coarse meshes ($n = 4$, 16 DOFs) achieve high accuracy, with minimal improvement beyond $n = 5$ (32 DOFs). Accuracy is quantified using relative error $100 \times (1 - |O_t - O_p|/|O_t|)$, where O_t and O_p are target and predicted objective values [35, 41–43].

Figure 4 (LEFT) shows accuracy exceeding 98.5%, with PBC reaching 99.9% (0.1% error) and averaging 99.3% across cases. Figure A1 (appendix D) reports relative errors of 0.5%–1.5%. RMSE values, computed as $\sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}$, are below 1.3×10^{-4} for PBC, SSB, and FFB, and 1.0–2.4×10^{-3} for cantilever (figure 4 (MID)). Normalized RMSE, defined as $\text{RMSE}_n = \text{RMSE}/(v_{\max} - v_{\min}) \times 100$, is below 1.0% for deflection and 1.0%–2.0% for rotation (figure 4 (RIGHT)). Fidelity, $\mathcal{F} = |\langle\psi_\theta|\psi_\phi\rangle|^2$, ranges from 0.998 to 0.999 (figure 4 (RIGHT)), aligning with high-accuracy quantum PDE solvers [25, 44]. The cantilever beam exhibits the largest target loss, the slowest convergence, and the highest final RMSE because its free-end condition imposes the most drastic modification of the original K. This asymmetric correction creates a more complex optimization landscape for the VQA. Consequently, the optimizer encounters more chanllege and is more susceptible to barren-plateaus-like phenomena [1, 43], leading to amplified errors in both deflection and rotation–most pronounced at the unsupported tip. These metrics confirm robust performance across diverse boundary conditions, with the cantilever case highlighting the critical role of boundary-induced landscape complexity in quantum FEM solvers.

5.2. NAFEMS T4 heat transfer results

The NAFEMS T4 benchmark, a 2D steady-state heat transfer problem with mixed Dirichlet, Neumann, and Robin boundary conditions, was solved using the implementation in section 4. A 5 × 3 mesh (24 DOFs, 5 qubits) was used for validation. The stiffness matrix K was decomposed into $O(1)$ Pauli terms via four-tessellations [25], and boundary terms $K_{bc} + K_{Robin}$ were handled with PPM. The predicted temperature at point E (0.6, 0.2) is 19.3 °C, compared to the NAFEMS target of 18.3 °C, yielding a 5.4% relative error. Abaqus reports 18.91 °C (3.3% error), confirming competitive performance on a coarse grid.

Figure 2 shows, from left to right: the problem schematic with boundary conditions, the quantum FEM temperature field, the absolute error field compared to Abaqus, and the relative error field (%) compared to Abaqus. The absolute error remains below 0.6 °C throughout the domain and is physically consistent, with slightly higher values near the Neumann boundaries (AD), where, the absence of a temperature constraint allows the solution to vary freely, and small differences lead to larger local deviations than on Dirichlet or Robin boundaries. The relative error reaches up to 6.7% near the top Robin

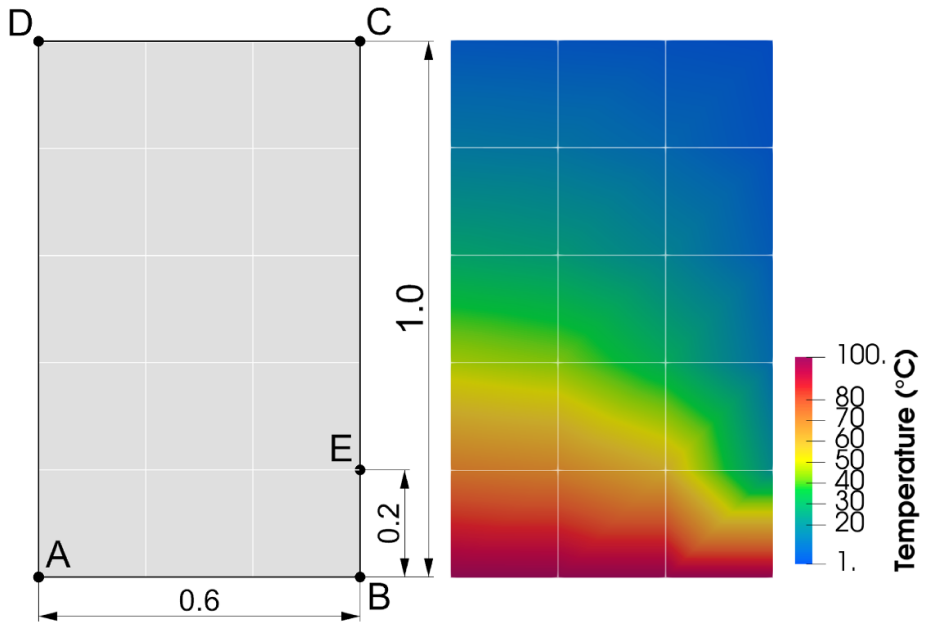

Figure 2. NAFEMS T4 benchmark: Two-dimensional heat transfer with mixed boundary conditions. (a) Problem geometry showing a rectangular plate (0.6 m × 1.0 m) with labeled edges and point E at coordinates (0.6, 0.2) from the bottom-left corner. (b) quantum FEM temperature field illustrating the heat transfer across the plate with Dirichlet (constant temperature) boundary condition on edge AB, Neumann (zero heat flux) condition on edge DA, and Robin (convection) boundary conditions on edges BC and CD. This combination of boundary conditions demonstrates the method's capability to handle complex, real-world heat transfer scenarios; (c) absolute error (°C) compared to Abaqus; (d) relative error (%) compared to Abaqus.

boundary (CD), where temperature approaches 0 °C. This elevated percentage is not due to poor solution quality but results from division by near-zero reference values in the relative error formula, amplifying small absolute discrepancies. Point E, located on the Robin boundary BC shows a low relative error of 5.4%, which underrepresents the peak interior error. The logarithmic qubit scaling (5 qubits for 24 DOFs) and circuit depth $O(7+n)$ to $O(11+n)$ for $n=5$ further highlight efficiency over classical FEM's linear memory scaling.

5.3. Quantum resource requirements

The quantum-enhanced FEM achieves efficiency through scalable Pauli decompositions and logarithmic qubit scaling. The core stiffness matrix K is decomposed into $O(1)$ Pauli terms for both 1D and 2D problems, leveraging Sato *et al*'s approach [25] (appendix A). Boundary terms $K_{\text{bc}} + K_{\text{Robin}}$ are decomposed via PPM [38] into a number of terms influenced by the affected DOFs and their row, column location (p, q) in the stiffness matrix, with terms (i, j) sharing the same circuit if the bitwise XOR are the same, $i \text{XOR } j = p \text{XOR } q$, reducing the effective number of unique circuits (appendix C). For standard boundary conditions on $\partial\Omega$, the number of terms is sublinear in N: small constants in 1D (e.g. the two edge points influenced by boundary DOFs), $O(\sqrt{N})$ in 2D (e.g. perimeter nodes in NAFEMS T4), and $O(N^{2/3})$ in 3D (e.g. surface nodes). The 'set-to-zero' strategy enables flexible boundary condition imposition, though non-standard internal constraints could increase terms to $O(N)$, which is rare. Qubit requirements scale as $n = \lceil \log_2 N \rceil$, offering a memory advantage over classical FEM's $O(N)$ scaling [10].

The circuit depth for the core stiffness matrix evaluation is $O(d_a + n)$, where $d_a \in [7, 11]$ corresponds to 3–5 repetitions of the RealAmplitudes ansatz, and $O(n)$ accounts for shift operators and PPM unitaries (appendices A and C). In our proof-of-concept demonstrations (up to $n=6$), this fixed-depth ansatz provides sufficient expressibility while remaining trainable. However, this is an empirical observation and does not guarantee that a fixed repetition count suffices for larger FEM discretizations. For bigger problems, more repetitions or expressive ansätze may be needed, potentially increasing depth. The time complexity for a single circuit evaluation is $O(t_a + n^2)$ for the core stiffness matrix, where the n^2 term arises from the shift operator's circuit depth [45], and $O(t_a + t_f)$ for the force term, where t_a, t_f are the ansatz and amplitude encoding time for the force term and implemented in a single circuit [26]. Boundary term evaluation via PPM contributes an additional time complexity of $O(B(t_a + t_f))$. The total time complexity is:

$$O\left(n_i n_q \left(t_a + n^2\right) + n_i \left(t_a + t_f\right) + n_i B \left(t_a + t_f\right)\right),$$

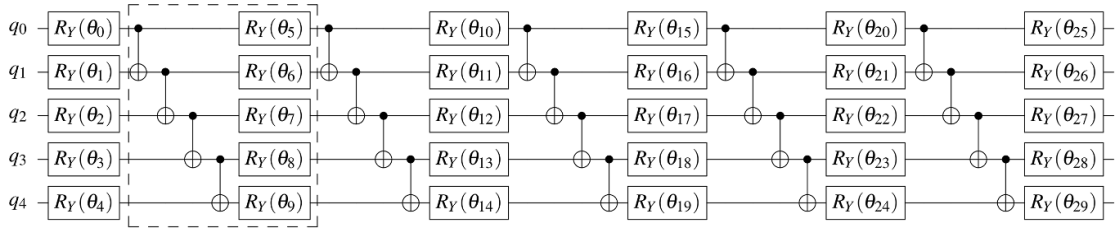

Figure 3. Circuit diagram showcs the ansatz with linear entanglement employed in this work. The ansatz consists of five layers acting on an five-qubit register, q_i, initialized to the computational basis state $|0\rangle^{\otimes 5}$ where $q_i = |0\rangle$. Each layer comprises CNOT gates arranged in a linear entanglement structure followed by single-qubit rotation gates R_Y, applied to each qubit, as shown in the dashed box.

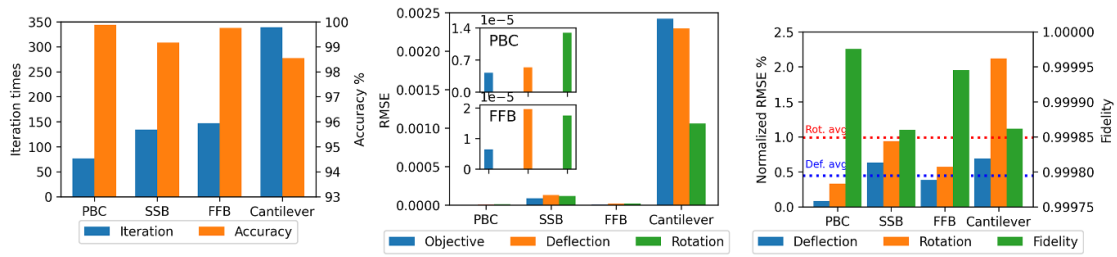

Figure 4. LEFT: The bar charts depict the relationship between iteration times (primary axis) based on figure 1 and accuracy level (secondary axis) upon convergence. The accuracy level is quantified by the variation between the target loss and the corresponding predicted result at convergence, scaled by a factor of 100. MID: Analysis of RMSE for objective, deflection, and rotation under four boundary conditions compared to the classical solutions. The top left section zooms in on PBC and FFB due to limitations in visualization. RIGHT: Accuracy indicators: (1) Quantitative analysis of normalized RMSE (primary axis) scaled by a factor of 100 for deflection and rotation across four boundary conditions compared with the classical solutions. Average Values are indicated by dashed lines. (2) Fidelity analysis (secondary axis) across four boundary conditions with respect to the classical solutions.

where $n_i \leqslant 500$ is the number of BFGS iterations, n_q is the number of circuits for the stiffness matrix evaluation, and B is the number of boundary terms. The force term requires only one circuit ($n_q' = 1$), as it is determined classically before quantum implementation. The n^2 term in the stiffness matrix evaluation and the B-dependent boundary term partially offset the logarithmic qubit scaling advantage in time complexity, but for small n (e.g. $n = 5$–6) and sublinear B, this remains manageable for NISQ devices. Unlike HHL, our VQA leverages NISQ devices with shallow circuits ($d_a = 7$–11), achieving errors of 0.5%–5.4% within 500 iterations. Compared to other VQAs [25, 46], our method's constant Pauli terms and sublinear boundary scaling enables practical application for engineering problems.

6. Conclusions

This paper introduces a quantum-enhanced FEM that leverages VQAs to address PDEs in structural mechanics and heat transfer, showcasing their potential for engineering applications. Using the Qiskit statevector simulator, we solved the Euler–Bernoulli beam problem (fourth-order PDE) and the NAFEMS T4 heat transfer benchmark (second-order PDE), achieving robust results across mesh convergence studies (4 to 64 DOFs). Our approach efficiently handles diverse boundary conditions (periodic, cantilever, simply-supported, fixed-fixed for 1D; Dirichlet, Neumann, Robin for 2D) through a 'set-to-zero' strategy and PPM [38], decomposes the stiffness matrix K into $O(1)$ Pauli terms using Sato *et al*'s method [25], and handle the sublinear scaling boundary terms $K_{bc} + K_{Robin}$: small constants in 1D, $O(\sqrt{N})$ in 2D, $O(N^{2/3})$ in 3D with efficient circuit reuse (appendix C). Logarithmic qubit scaling ($n = \lceil \log_2 N \rceil$) provides a memory advantage over classical FEM's $O(N)$ scaling, while shallow circuits ($d_a = 7 - -11$) and a time complexity of $O(n_i n_q (d_a + n^2) + n_i (d_a + t_f) + n_i B(d_a + t_f))$ ensure NISQ feasibility, despite the n^2 shift operator term tempering the logarithmic advantage. The beam problem achieved 0.5%–1.5% relative errors and fidelities of 0.998–0.999, converging in 77–350 BFGS iterations, while NAFEMS T4 yielded a 5.4% error (19.3 °C vs. 18.3 °C), competitive with Abaqus (3.3% error). While VQAs face barren plateaus, which can cause vanishing gradients and hinder optimization, our work does not address this issue but would benefit from further discussion on ansatz expressibility and trainability. Future efforts will focus on mitigating barren plateaus, enhancing accuracy for complex boundary conditions, and extending to 3D and multi-physics problems, leveraging quantum hardware

advances to potentially surpass classical FEM and establishing a foundation for quantum-enhanced FEM in computational science and engineering.

Data availability statement

No new data were created or analysed in this study.

Acknowledgments

This work was supported by the National Research Foundation, Singapore and A*STAR under its Quantum Engineering Programme (NRF2021-QEP2-02-P04), Hybrid Quantum-Classical Computing HQCC 1.0 (SC23/247534CI) and A*STAR C230917003.

Author contributions

Dingjie Lu 0000-0002-7501-0161
Conceptualization (lead), Data curation (lead), Formal analysis (lead), Funding acquisition (lead), Investigation (lead), Methodology (lead), Project administration (lead), Resources (lead), Software (lead), Supervision (equal), Validation (lead), Visualization (lead), Writing – original draft (lead), Writing – review & editing (lead)

Zhao Wang
Data curation (equal), Formal analysis (equal), Visualization (equal), Writing – original draft (equal)

Jun Liu 0000-0001-9896-1843
Conceptualization (equal), Methodology (equal), Software (equal)

Yangfan Li 0000-0002-7155-8277
Formal analysis (equal), Methodology (equal), Software (equal)

Wei-Bin Ewe 0000-0002-4600-0634
Conceptualization (equal), Methodology (equal), Software (equal), Writing – review & editing (equal)

Zhuangjian Liu 0000-0002-3412-2116
Funding acquisition (equal), Project administration (equal), Supervision (equal)

Appendix A. Element and global stiffness matrix details

A.1. Euler–Bernoulli beam

The element stiffness matrix K_e for the Euler–Bernoulli beam is derived from the weak form of the governing equation. With normalized parameters $E = 1$, $I = 1$, and $l_e = 1$, the matrix is:

$$K_e = \begin{bmatrix} \frac{12EI}{l_e^3} & \frac{6EI}{l_e^2} & \frac{-12EI}{l_e^3} & \frac{6EI}{l_e^2} \\ \frac{6EI}{l_e^2} & \frac{4EI}{l_e} & \frac{-6EI}{l_e^2} & \frac{2EI}{l_e} \\ \frac{-12EI}{l_e^3} & \frac{-6EI}{l_e^2} & \frac{12EI}{l_e^3} & \frac{-6EI}{l_e^2} \\ \frac{6EI}{l_e^2} & \frac{2EI}{l_e} & \frac{-6EI}{l_e^2} & \frac{4EI}{l_e} \end{bmatrix} = \begin{bmatrix} 12 & 6 & -12 & 6 \\ 6 & 4 & -6 & 2 \\ -12 & -6 & 12 & -6 \\ 6 & 2 & -6 & 4 \end{bmatrix}. \tag{A1}$$

This matrix can be decomposed into a sum of six Pauli terms:

$$K_e = 8\mathbb{I} \otimes \mathbb{I} + 4\mathbb{I} \otimes Z - 5X \otimes \mathbb{I} - 7X \otimes Z - 6Y \otimes Y + 6Z \otimes X. \tag{A2}$$

The global stiffness matrix K is assembled using a shift operator P, which cyclically permutes the computational basis states: $P = \sum_i |(i+1) \mod 2^n\rangle\langle i|$. The global matrix is constructed as $K = K_a + K_b - K_c$, where $K_a = \mathbb{I}^{\otimes n} \otimes K_e$, $K_b = P^{-1}K_aP$, and $K_c = P^{-1}(I_0^{\otimes n} \otimes K_e)P$. This structure, detailed in [25], ensures that K can be represented by a constant number of Pauli terms, regardless of the system size N. The expectation of K_c is evaluated as $\langle\phi|K_c|\phi\rangle = \langle P\phi|(I_0^{\otimes n} \otimes K_e)|P\phi\rangle$, where $I_0 = |0\rangle\langle 0|$, by first applying

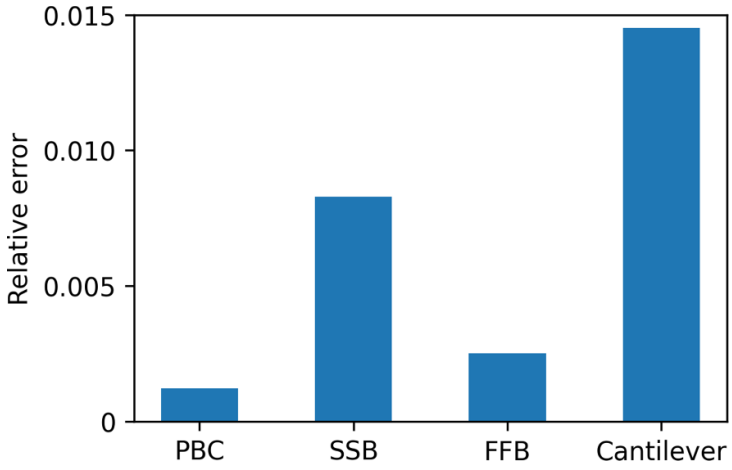

Figure A1. Relative error (%) between target and predicted objective values for periodic (PBC), simply-supported (SSB), fixed-fixed (FFB), and cantilever boundary conditions, computed as $100 \times (1 - |\mathcal{O}_t - \mathcal{O}_p|/|\mathcal{O}_t|)$.

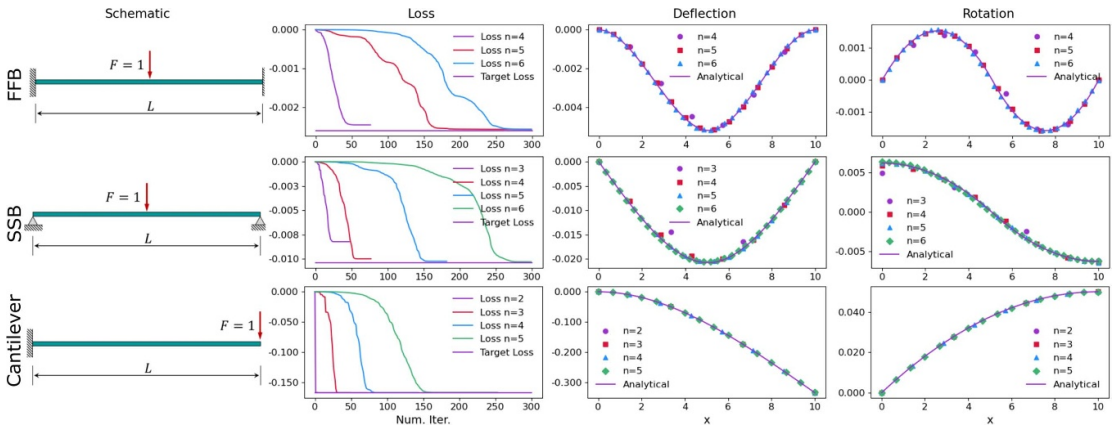

Figure A2. Mesh convergence study in matrix format. Rows represent boundary conditions: fixed-fixed beam (FFB, top), simply-supported beam (SSB, middle), and cantilever beam (bottom). Columns show schematic diagram (SCHEMATIC), loss convergence history (LOSS), deflection profile (DEFLECTION), and rotation profile (ROTATION) for different numbers of qubits n. Analytical solutions are included in deflection and rotation plots for comparison.

the shift circuit for P and then measuring the Pauli terms of K_e on the relevant subspace. The matrix for a 6-DOF system are:

$$K = \begin{bmatrix} 12 & 6 & -12 & 6 & & & & & & & & \\ 6 & 4 & -6 & 2 & & & & & & & & \\ -12 & -6 & 24 & 0 & -12 & 6 & & & & & & \\ 6 & 2 & 0 & 8 & -6 & 2 & & & & & & \\ & & -12 & -6 & & & \ddots & & & & & \\ & & 6 & 2 & & & & & & & & \\ & & & & \ddots & & & & & & & \\ & & & & & & & -12 & 6 & & & \\ & & & & & & 6 & 2 & 0 & 8 & & \\ & & & & & & -12 & -6 & 24 & 0 & -12 & 6 \\ & & & & & & & & -12 & -6 & 12 & -6 \\ & & & & & & & & 6 & 2 & -6 & 4 \end{bmatrix}. \quad \text{(A3)}$$

$$K_{\mathrm{a}} = \begin{bmatrix} 12 & 6 & -12 & 6 & & & & & & & & \\ 6 & 4 & -6 & 2 & & & & & & & & \\ -12 & -6 & 12 & -6 & & & & & & & & \\ 6 & 2 & -6 & 4 & & & & & & & & \\ & & & & \ddots & & & & & & & \\ & & & & & 12 & 6 & -12 & 6 & & & \\ & & & & & 6 & 4 & -6 & 2 & & & \\ & & & & & -12 & -6 & 12 & -6 & & & \\ & & & & & 6 & 2 & -6 & 4 & & & \end{bmatrix}, \tag{A4}$$

$$K_{\mathrm{b}} = \begin{bmatrix} 12 & -6 & & & & & & & & & -12 & -6 \\ -6 & 4 & & & & & & & & & 6 & 2 \\ & & 12 & 6 & -12 & 6 & & & & & & \\ & & 6 & 4 & -6 & 2 & & & & & & \\ & & -12 & -6 & 12 & -6 & & & & & & \\ & & 6 & 2 & -6 & 4 & & & & & & \\ & & & & & & \ddots & & & & & \\ & & & & & & & 12 & 6 & -12 & 6 & \\ & & & & & & & 6 & 4 & -6 & 2 & \\ & & & & & & & -12 & -6 & 12 & -6 & \\ & & & & & & & 6 & 2 & -6 & 4 & \\ -12 & 6 & & & & & & & & & 12 & 6 \\ -6 & 2 & & & & & & & & & 6 & 4 \end{bmatrix}, \tag{A5}$$

$$K_{\mathrm{c}} = \begin{bmatrix} 12 & -6 & 0 & \cdots & 0 & -12 & -6 \\ -6 & 4 & 0 & \cdots & 0 & 6 & 2 \\ 0 & 0 & 0 & \cdots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & 0 & 0 & 0 \\ -12 & 6 & 0 & \cdots & 0 & 12 & 6 \\ -6 & 2 & 0 & \cdots & 0 & 6 & 4 \end{bmatrix}. \tag{A6}$$

A.2. 2D heat transfer

For the 2D steady-state heat conduction problem on a unit square quadrilateral element, the element stiffness matrix K_{e} is:

$$K_{\mathrm{e}} = \frac{k}{6}\begin{bmatrix} 4 & -1 & -2 & -1 \\ -1 & 4 & -1 & -2 \\ -2 & -1 & 4 & -1 \\ -1 & -2 & -1 & 4 \end{bmatrix} = \frac{k}{6}\left(4I\otimes I - I\otimes X - 2X\otimes I - X\otimes X\right). \tag{A7}$$

To assemble the global stiffness matrix K for a 2D grid, the mesh is divided into four non-overlapping tessellations $(\mathcal{T}_0, \mathcal{T}_1, \mathcal{T}_2, \mathcal{T}_3)$. The global matrix is the sum of contributions from each tessellation, $K = \sum_{i=0}^{3} K_{\mathcal{T}_i}$. Using shift operators P_x and P_y for the x- and y-directions, these contributions are related by $K_{\mathcal{T}_1} = P_x^{-1} K_{\mathcal{T}_0} P_x$, $K_{\mathcal{T}_2} = P_y^{-1} K_{\mathcal{T}_0} P_y$, etc where $K_{\mathcal{T}_0}$ contains the element matrices for the first tessellation. This method, also from Sato *et al* [25], again results in an $O(1)$ Pauli decomposition for the global stiffness matrix K.

For a 3×3 element mesh (4×4 nodal grid, 16 DOFs) the matrix are:

$$K_{\mathcal{T}_0} = \begin{bmatrix}
4 & -1 & 0 & 0 & -1 & -2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
-1 & 4 & 0 & 0 & -2 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 4 & -1 & 0 & 0 & -1 & -2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & -1 & 4 & 0 & 0 & -2 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
-1 & -2 & 0 & 0 & 4 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
-2 & -1 & 0 & 0 & -1 & 4 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & -1 & -2 & 0 & 0 & 4 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & -2 & -1 & 0 & 0 & -1 & 4 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 4 & -1 & 0 & 0 & -1 & -2 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 4 & 0 & 0 & -2 & -1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 4 & -1 & 0 & 0 & -1 & -2 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 4 & 0 & 0 & -2 & -1 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & -2 & 0 & 0 & 4 & -1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -2 & -1 & 0 & 0 & -1 & 4 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & -2 & 0 & 0 & 4 & -1 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -2 & -1 & 0 & 0 & -1 & 4
\end{bmatrix}$$

$$P_x = \begin{bmatrix}
0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0
\end{bmatrix},$$

$$P_y = \begin{bmatrix}
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0
\end{bmatrix}.$$

$$K = \begin{bmatrix} 16 & -2 & 0 & -2 & -2 & -2 & 0 & -2 & 0 & 0 & 0 & 0 & -2 & -2 & 0 & -2 \\ -2 & 16 & -2 & 0 & -2 & -2 & -2 & 0 & 0 & 0 & 0 & 0 & -2 & -2 & -2 & 0 \\ 0 & -2 & 16 & -2 & 0 & -2 & -2 & -2 & 0 & 0 & 0 & 0 & 0 & -2 & -2 & -2 \\ -2 & 0 & -2 & 16 & -2 & 0 & -2 & -2 & 0 & 0 & 0 & 0 & -2 & 0 & -2 & -2 \\ -2 & -2 & 0 & -2 & 16 & -2 & 0 & -2 & -2 & -2 & 0 & -2 & 0 & 0 & 0 & 0 \\ -2 & -2 & -2 & 0 & -2 & 16 & -2 & 0 & -2 & -2 & -2 & 0 & 0 & 0 & 0 & 0 \\ 0 & -2 & -2 & -2 & 0 & -2 & 16 & -2 & 0 & -2 & -2 & -2 & 0 & 0 & 0 & 0 \\ -2 & 0 & -2 & -2 & -2 & 0 & -2 & 16 & -2 & 0 & -2 & -2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -2 & -2 & 0 & -2 & 16 & -2 & 0 & -2 & -2 & -2 & 0 & -2 \\ 0 & 0 & 0 & 0 & -2 & -2 & -2 & 0 & -2 & 16 & -2 & 0 & -2 & -2 & -2 & 0 \\ 0 & 0 & 0 & 0 & 0 & -2 & -2 & -2 & 0 & -2 & 16 & -2 & 0 & -2 & -2 & -2 \\ 0 & 0 & 0 & 0 & -2 & 0 & -2 & -2 & -2 & 0 & -2 & 16 & -2 & 0 & -2 & -2 \\ -2 & -2 & 0 & -2 & 0 & 0 & 0 & 0 & -2 & -2 & 0 & -2 & 16 & -2 & 0 & -2 \\ -2 & -2 & -2 & 0 & 0 & 0 & 0 & 0 & -2 & -2 & -2 & 0 & -2 & 16 & -2 & 0 \\ 0 & -2 & -2 & -2 & 0 & 0 & 0 & 0 & 0 & -2 & -2 & -2 & 0 & -2 & 16 & -2 \\ -2 & 0 & -2 & -2 & 0 & 0 & 0 & 0 & -2 & 0 & -2 & -2 & -2 & 0 & -2 & 16 \end{bmatrix}.$$

Appendix B. The 'set-to-zero' strategy for Dirichlet conditions

The 'set-to-zero' strategy enforces Dirichlet boundary conditions by modifying the stiffness matrix K and load vector $\mathbf{f}$. Let $S = \{i_1, i_2, \ldots, i_k\}$ be the set of indices corresponding to degrees of freedom (DOFs) where Dirichlet conditions $v_i = c_i$ are prescribed. For each $i \in S$, the off-diagonal elements in row and column i of K are set to zero, the diagonal element K_{ii} is preserved, and the load vector is adjusted to enforce the conditions. The modified system is $(K + K_{\mathrm{bc}})\mathbf{v} = \mathbf{f}_1$, where the correction matrix K_{bc} contains the negated off-diagonal elements, and the new load vector $\mathbf{f}_1$ is:

$$f_{1,i} = \begin{cases} K_{ii} c_i & \text{if } i \in S, \\ f_i - \sum_{j \in S} K_{ij} c_j & \text{if } i \notin S. \end{cases} \tag{B1}$$

Consider a 6-DOF Euler–Bernoulli beam with Dirichlet conditions $v_1 = c_1$ and $\theta_1 = c_2$ at the first node ($S = \{1, 2\}$). The original system is $K_0 \mathbf{v} = \mathbf{f}_0$. Applying the strategy, the off-diagonal elements in the first two rows and columns of K_0 are moved into K_{bc} (with a negative sign) and subsequently added back, effectively zeroing them in the main matrix while preserving symmetry. The modified matrix $K + K_{\mathrm{bc}}$ has zeroed off-diagonal elements for the constrained DOFs. For the cantilever case where $c_1 = c_2 = 0$ and a unit load is applied at the free end ($q_5 = 1$), the resulting correction matrix K_{bc} is:

$$K_{\mathrm{bc}} = \begin{bmatrix} 0 & -6 & 12 & -6 & 0 & 0 \\ -6 & 0 & 6 & -2 & 0 & 0 \\ 12 & 6 & 0 & 0 & 0 & 0 \\ -6 & -2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}, \tag{B2}$$

and the final load vector is $\mathbf{f}_1 = [0, 0, 0, 0, 1, 0]^{\mathrm{T}}$.

Appendix C. Evaluation of boundary term expectation $\langle \phi | K_{\mathrm{bc}} | \phi \rangle$

The expectation value $\langle \phi | K_{\mathrm{bc}} | \phi \rangle$ is evaluated using the PPM method [38]. PPM was originally introduced in our prior work as a general framework for accelerating quantum expectation estimation through binary arithmetic preprocessing and circuit reuse. In the present manuscript, we do not re-derive the PPM formalism; instead, we adopt it as a measurement primitive and focus on its concrete integration into quantum finite element formulations, particularly for boundary stiffness operators.

Specifically, the boundary stiffness matrix K_{bc} is decomposed into a sum of sparse matrices, each containing non-zero off-diagonal entries only at a single pair of indices (p, q) corresponding to boundary DOFs. Each such matrix K_{pq} is associated with a unitary transformation T_{pq}, constructed from CNOT gates, that rotates K_{pq} into a simple Pauli observable. The circuit depth of T_{pq} scales as $\mathcal{O}(n)$, where $n = \lceil \log_2 N \rceil$ is the number of qubits.

The number of such boundary-localized terms scales with the number of boundary DOFs, denoted B, which is typically sublinear in the total system size: $\mathcal{O}(1)$ in one dimension and $\mathcal{O}(\sqrt{N})$ in two

dimensions. Moreover, as established in [38], circuit reuse is possible for pairs (p, q) sharing the same bitwise XOR pattern $(p \oplus q)$, significantly reducing the number of distinct circuits required. This reuse property allows $\langle\phi|K_{\mathrm{bc}}|\phi\rangle$ to be estimated with overall measurement complexity scaling as $\mathcal{O}(\log N)$, preserving the logarithmic advantage of the quantum FEM formulation.

Appendix D. Supplementary numerical results

Figure A1 shows the relative error between the final objective value achieved by the VQA and the target value for the four beam boundary conditions. The error is consistently low, ranging from 0.5% to 1.5%, demonstrating the high accuracy of the method.

Appendix E. Mesh convergence study

A mesh convergence study was performed to validate the method's accuracy and stability across different discretization levels. Figure A2 shows results for the fixed-fixed, simply-supported, and cantilever beams with varying numbers of qubits ($n = 2$ to $n = 6$). For all cases, the deflection and rotation profiles show excellent agreement with analytical solutions, even for coarse meshes. For instance, the simply-supported case shows that $n = 4$ (16 DOFs) already provides high accuracy, with minimal improvement for finer meshes. The cantilever beam results also show convergence at moderate mesh sizes, with negligible difference between $n = 4$ and $n = 5$. While larger systems require more optimization iterations, the study confirms that the method is robust and maintains high accuracy across different mesh densities and boundary conditions.

References

[1] McClean J R, Romero J, Babbush R and Aspuru-Guzik A 2016 The theory of variational hybrid quantum-classical algorithms *New J. Phys.* **18** 023023
[2] Bravo-Prieto C, LaRose R, Cerezo M, Subasi Y, Cincio L and Coles P J 2023 Variational quantum linear solver *Quantum* **7** 1188
[3] Schillo N and Sturm A 2025 Variational quantum algorithms for differential equations on a noisy quantum computer *IEEE Trans. Quantum Eng.* **6** 1
[4] Renardy M and Rogers R C 2006 *An Introduction to Partial Differential Equations* vol 13 (Springer) (https://doi.org/10.1007/b97427)
[5] Nouy A 2010 A priori model reduction through proper generalized decomposition for solving time-dependent partial differential equations *Comput. Methods Appl. Mech. Eng.* **199** 1603
[6] Avalos G and Geredeli P G 2019 Stability analysis of coupled structural acoustics PDE models under thermal effects and with no additional dissipation *Math. Nachr.* **292** 939
[7] Rabczuk T, Ren H and Zhuang X 2019 A nonlocal operator method for partial differential equations with application to electromagnetic waveguide problem *Comput. Mater. Contin.* **59** 31
[8] Bharadwaj S S and Sreenivasan K R 2023 Hybrid quantum algorithms for flow problems *Proc. Natl Acad. Sci.* **120** e2311014120
[9] Shor P W 1994 Algorithms for quantum computation: discrete logarithms and factoring *Proc. 35th Annual Symp. on Foundations of Computer Science* (IEEE) pp 124–34
[10] Nielsen M A and Chuang I L 2010 *Quantum Computation and Quantum Information* 10th edn (Cambridge University Press) (https://doi.org/10.1017/CBO9780511976667)
[11] Preskill J 2018 Quantum computing in the NISQ era and beyond *Quantum* **2** 79
[12] Cerezo M, Sone A, Volkoff T, Cincio L and Coles P J 2021 Cost function dependent barren plateaus in shallow parametrized quantum circuits *Nat. Commun.* **12** 1791
[13] Lubasch M, Joo J, Moinier P, Kiffner M and Jaksch D 2020 Variational quantum algorithms for nonlinear problems *Phys. Rev. A* **101** 010301
[14] Sarma A, Watts T W, Moosa M, Liu Y and McMahon P L 2024 Quantum variational solving of nonlinear and multidimensional partial differential equations *Phys. Rev. A* **109** 062616
[15] Jaksch D, Givi P, Daley A J and Rung T 2023 Variational quantum algorithms for computational fluid dynamics *AIAA J.* **61** 1885
[16] Wright L, Keever C M, First J T, Johnston R, Tillay J, Chaney S, Rosenkranz M and Lubasch M 2024 Noisy intermediate-scale quantum simulation of the one-dimensional wave equation *Phys. Rev. Res.* **6** 043169
[17] Sato Y, Kondo R, Hamamura I, Onodera T and Yamamoto N 2024 Hamiltonian simulation for hyperbolic partial differential equations by scalable quantum circuits *Phys. Rev. Res.* **6** 033246
[18] Hu J, Jin S, Liu N and Zhang L 2024 Quantum circuits for partial differential equations via Schrödingerisation *Quantum* **8** 1563
[19] Brearley P and Laizet S 2024 Quantum algorithm for solving the advection equation using Hamiltonian simulation *Phys. Rev. A* **110** 012430
[20] Over P, Bengoechea S, Brearley P, Laizet S and Rung T 2025 Quantum algorithm for the advection-diffusion equation by direct block encoding of the time-marching operator *Phys. Rev. A* **112** L010401
[21] Over P, Bengoechea S, Rung T, Clerici F, Scandurra L, de Villiers E and Jaksch D 2025 Boundary treatment for variational quantum simulations of partial differential equations on quantum computers *Comput. Fluids* **288** 106508
[22] Bengoechea S, Over P, Jaksch D and Rung T 2025 Toward variational quantum algorithms for generalized linear and nonlinear transport phenomena *AIAA J.* 1
[23] Guseynov N, Huang X and Liu N 2025 Gate construction of block-encoding for Hamiltonians needed for simulating partial differential equations *Phys. Rev. Res.* **7** 033100
[24] Guseynov N, Huang X and Liu N 2025 Quantum framework for simulating linear PDES with robin boundary conditions (arXiv:2506.20478)

[25] Sato Y, Kondo R, Koide S, Takamatsu H and Imoto N 2021 Variational quantum algorithm based on the minimum potential energy for solving the Poisson equation *Phys. Rev. A* **104** 052409
[26] Zoufal C, Lucchi A and Woerner S 2019 Quantum generative adversarial networks for learning and loading random distributions *npj Quantum Inf.* **5** 103
[27] Grimsley H R, Economou S E, Barnes E and Mayhall N J 2019 An adaptive variational algorithm for exact molecular simulations on a quantum computer *Nat. Commun.* **10** 3007
[28] Zhang J, Feng F and Zhang Q-J 2024 Quantum computing method for solving electromagnetic problems based on the finite element method *IEEE Trans. Microw. Theory Tech.* **72** 948
[29] NAFEMS 1990 *The Standard NAFEMS Benchmarks* Publication P18
[30] Lubinski T, Johri S, Varosy P, Coleman J, Zhao L, Necaise J, Baldwin C H, Mayer K and Proctor T 2023 Application-oriented performance benchmarks for quantum computing *IEEE Trans. Quantum Eng.* **4** 1
[31] Bathe K 1982 *Finite Element Procedures in Engineering Analysis* (Prentice-Hall)
[32] Courant R and Hilbert D 2013 *Methoden der Mathematischen Physik* (Springer)
[33] Ciarlet P 2002 *The Finite Element Method for Elliptic Problems* (*Classics in Applied Mathematics*) (Society for Industrial and Applied Mathematics) (https://doi.org/10.1137/1.9780898719208)
[34] Brenner S and Scott R 2007 *The Mathematical Theory of Finite Element Methods* (*Texts in Applied Mathematics*) (Springer) (https://doi.org/10.1007/978-0-387-75934-0)
[35] McClean J R, Boixo S, Smelyanskiy V N, Babbush R and Neven H 2018 Barren plateaus in quantum neural network training landscapes *Nat. Commun.* **9** 4812
[36] Cerezo M *et al* 2021 Variational quantum algorithms *Nat. Rev. Phys.* **3** 625
[37] Bravo-Prieto C, Lumbreras-Zarapico J, Tagliacozzo L and Latorre J I 2020 Scaling of variational quantum circuit depth for condensed matter systems *Quantum* **4** 272
[38] Lu D, Li Y, Koh D E, Wang Z, Liu J and Liu Z Towards enhancing quantum expectation estimation of matrices through partial Pauli decomposition techniques and post-processing (arXiv:2401.17640)
[39] Javadi-Abhari A *et al* 2024 *Quantum Computing With QIskit* (arXiv:2405.08810 [quant-ph])
[40] Pellow-Jarman A, Sinayskiy I, Pillay A and Petruccione F 2021 A comparison of various classical optimizers for a variational quantum linear solver *Quantum Inf. Process.* **20** 202
[41] Uvarov A, Biamonte J D and Yudin D 2020 Variational quantum eigensolver for frustrated quantum systems *Phys. Rev. B* **102** 075104
[42] Sato Y, Watanabe H C, Raymond R, Kondo R, Wada K, Endo K, Sugawara M and Yamamoto N 2023 Variational quantum algorithm for generalized eigenvalue problems and its application to the finite-element method *Phys. Rev. A* **108** 022429
[43] Kyriienko O, Paine A E and Elfving V E 2021 Solving nonlinear differential equations with differentiable quantum circuits *Phys. Rev. A* **103** 052416
[44] Leong F Y, Koh D E, Ewe W-B and Kong J F 2023 Variational quantum simulation of partial differential equations: applications in colloidal transport *Int. J. Numer. Methods Heat Fluid Flow* **33** 3669
[45] Maslov D 2016 Advantages of using relative-phase toffoli gates with an application to multiple control toffoli optimization *Phys. Rev. A* **93** 022311
[46] Liu H-L, Wu Y-S, Wan L-C, Pan S-J, Qin S-J, Gao F and Wen Q-Y 2021 Variational quantum algorithm for the Poisson equation *Phys. Rev. A* **104** 022418