

Variational Quantum Algorithm for Solving Second-Order Linear Differential Equations

Ryo Sugaya *, Tomohiro Sogabe, Tomoya Kemmochi and Shao-Liang Zhang

Department of Applied Physics, Graduate School of Engineering, Nagoya University, Nagoya 464-8601, Japan; sogabe@na.nuap.nagoya-u.ac.jp (T.S.); kemmochi@na.nuap.nagoya-u.ac.jp (T.K.); zhang@na.nuap.nagoya-u.ac.jp (S.-L.Z.)

* Corresponding author: r-sugaya@na.nuap.nagoya-u.ac.jp

Received date: 17 February 2025; Accepted date: 17 April 2025; Published online: 22 May 2025

Abstract: Variational quantum algorithms (VQAs) have attracted attention as quantum algorithms employing on noisy intermediate-scale quantum devices. VQAs for solving the Poisson equation were recently proposed, as this method transforms the linear equation with a symmetric positive definite matrix obtained by discretizing the Poisson equation into a minimization problem. In this study, we propose a VQA for second-order linear differential equations including the Poisson equation on the basis of the referenced study, where the coefficient matrix obtained through discretization is a non-symmetric matrix. Furthermore, in our VQAs, we succeeded in decomposing the coefficient matrices arising from the periodic and the Dirichlet boundary condition into linear combinations of measurement operators and unitary matrices.

Keywords: quantum computing; variational quantum algorithm; linear equations

1. Introduction

Solving partial differential equations (PDEs) is key to describing various issues in condensed matter physics, fluid simulation, electromagnetic field analysis, and other fields as mathematical models [1]. Therefore, identifying efficient algorithms for solving PDEs is crucial. Thus, numerous studies have explored the numerical analysis of PDEs.

Quantum algorithms (QAs), which are computational algorithms employing quantum computers, have attracted attention owing to their ability to solve specific issues more efficiently compared with classical computers. Notably, QAs have been demonstrated to outperform classical computers in solving certain problems. Hence, QAs have been deployed to solve problems in various fields such as machine learning [2,3], integer factorization [4], and combinatorial optimization [5]. Further, the Harrow–Hassidim–Lloyd (HHL) algorithm [6], a QA for solving linear equations, can solve linear equations obtained via PDEs discretization. However, the HHL algorithm requires a sufficient number of qubits and error correction techniques on quantum computers, which may require a long development time. To resolve this shortcoming, quantum-classical hybrid algorithms, employing noisy intermediate-scale quantum (NISQ) devices [7] with intermediate numbers of qubits, combined with classical computers, have gained attention. Among these quantum-classical hybrid algorithms, the variational quantum algorithms (VQAs) can operate on NISQ devices.

VQAs [8] are algorithms for minimizing objective functions containing approximate solution vectors. These approximate solution vectors are generated by quantum gates (ansatz gates) depending on parameters. The parameter updates are performed using iterative algorithms to minimize the objective function represented by approximate solution vectors. Thus, VQA is highly versatile and has been applied to various tasks such as solving eigenvalue problems using the variational quantum eigenvalue solver [9,10], quantum machine learning [2], and quantum approximate optimization algorithms [5].

To obtain the value of the objective function using VQA, the computation is performed by measuring the expectations. Furthermore, two conventional methods for calculating expectations on a quantum computer are projective measurements and the Hadamard test [11]. To compute expectations using projective measurement and the Hadamard test, the matrices involved must be measurement operators and unitary, respectively. Therefore, to implement VQAs, decomposing a matrix into a linear combination of unitary matrices and measurement operators represents a crucial challenge.

Many researchers have proposed methods for solving linear equations using VQAs [12–18]. Notably, Sato et al. proposed VQAs to solve linear equations obtained by the discretization of the Poisson equation [18]. To solve $Ax = \mathbf{b}$ for a symmetric positive definite matrix A , the minimizing problem for $\mathbf{x}^\dagger Ax/2 - \mathbf{x}^\dagger \mathbf{f}$ was solved using VQAs in the previous study, which was aimed at solving the Poisson equation, where \mathbf{x}^\dagger denotes the conjugate transpose of \mathbf{x} . Furthermore, the number of terms involving expectations in the objective function value was accelerated to $O(1)$ regardless of the matrix size by decomposing the coefficient matrix into a linear combination of unitary matrices and measurement operators. However, A must be a symmetric positive definite to ensure its formulation as a convex quadratic programming problem. Additionally, as A in the Poisson equation exhibits a specific structure, extending the methods employed in previous studies to other PDEs may be challenging. For example, for differential equations containing advection terms in fluid simulations, A becomes non-symmetric, thereby complicating the use of existing VQAs to solve the problem.

Thus, we propose VQAs for solving general second-order linear differential equations containing constant coefficients, e.g., the Poisson equation. Generally, second-order linear differential equations containing constant coefficients include advection terms, thus yielding non-symmetric coefficient matrices when discretized. For example, A obtained from the discretization of the Poisson equation with the Dirichlet boundary condition represents a special tridiagonal Toeplitz matrix that is symmetric and positive definite. Conversely the A obtained by discretizing a second-order linear differential equation containing constant coefficients and the Dirichlet boundary condition is a general tridiagonal Toeplitz matrix. To handle the non-symmetric A , we considered the normal equation $A^\dagger Ax = A^\dagger \mathbf{b}$ for the original linear equation ($Ax = \mathbf{b}$) because $A^\dagger A$ is always a symmetric positive definite when A is a nonsingular, making it reducible to a convex quadratic programming problem¹. Furthermore, solving the normal equation is equivalent to solving the least squares problem for minimizing the norm $\|Ax - \mathbf{b}\|$. Similar to previous study [18], as matrices $A^\dagger A$ and A in the normal equation are not unitary matrices, they must be decomposed into linear combinations of unitary matrices and measurement operators. Thus, for A in second-order linear differential equations containing constant coefficients with the periodic boundary condition, we decomposed them into a linear combination of unitary matrices and measurement operators. Furthermore, by adding a correction term to A with the periodic boundary condition, we attempted the decomposition of the A for the case of the Dirichlet boundary condition. However, as the matrices employed for this correction do not possess unitarity and Hermiticity, we proposed a method for converting these correction matrices into projection operators or unitary matrices, ensuring their applicability even under the Dirichlet boundary condition.

The remainder of this paper is structured as follows: Section 2 presents an overview of the extant studies on VQAs for solving the Poisson equation. Next, Section 3 reports the construction the objective function for the constant-coefficient second-order linear differential equations on the basis of VQAs. Thereafter, we demonstrated the decomposition method for the matrices included in the objective function under the periodic boundary condition. Next, we present the decomposition method for the correction matrices for cases involving the imposition of the Dirichlet boundary condition is imposed on section. Section 4 presents numerical examples using the proposed method implemented by Qiskit in this section. Finally, Section 5 concludes the paper and discussing the future directions.

2. Variational Quantum Algorithm for the Poisson Equation

Here, we describe VQA for solving the Poisson equation based on Sato et al.’s study [18]. For the domain $\Omega = (L, R)^d$, ($L, R \in \mathbb{R}, L < R$), the d -dimensional Poisson equation with a given function $f : \Omega \rightarrow \mathbb{R}$ is given by the following equation:

$$-\Delta u(\mathbf{x}) = f(\mathbf{x}) \quad \mathbf{x} \in \Omega. \quad (1)$$

To simply explain the previous study, we consider here the Dirichlet boundary condition:

$$u(\mathbf{x}) = 0 \quad \mathbf{x} \in \partial\Omega.$$

For simplicity, we restrict the study to the case where $d = 1$. Thus, by discretizing Eq. (1) using the finite difference method with the interval-partition width $h = (R - L)/(N + 1)$, we derive the following linear equation:

$$A\mathbf{u} = \mathbf{f}, \quad (2)$$

¹ A weakness of this approach is that $A^\dagger A$ may become highly ill-conditioned. Conversely, $A^\dagger A$ will be well conditioned if A is close to unitary; in this case A^\dagger acts as a good preconditioner for A . Furthermore, preconditioning techniques have been developed for variational quantum linear solvers, see Ref. [19] for example.

where the matrix $A \in \mathbb{R}^{N \times N}$ is a symmetric positive-definite matrix, and $\mathbf{u}, \mathbf{f} \in \mathbb{R}^N$. Therefore, solving the differential equation Eq. (1) reduces to solving Eq. (2). Furthermore, as matrix A is positive definite and symmetric, solving the linear equation Eq. (2) is equivalent to determining the minimization solution \mathbf{v}_{\min} of the following objective function $E(\mathbf{v})$:

$$E(\mathbf{v}) := \frac{1}{2} \mathbf{v}^\dagger A \mathbf{v} - \mathbf{v}^\dagger \mathbf{f},$$

where \mathbf{v} and \mathbf{f} are assumed to be real vectors, and thus $\mathbf{v}^\dagger \mathbf{f} = \mathbf{f}^\dagger \mathbf{v}$.

In the following sections of this paper, we will consider $N = 2^n$ with respect to the number of qubits $n \in \mathbb{Z}^+$.

To minimize $E_v(\mathbf{v})$, the vector \mathbf{v} is assumed to be represented as $\mathbf{v} = r |\psi(\boldsymbol{\theta})\rangle$ and determine \mathbf{v}_{\min} from this representation. Here, $r \in \mathbb{R}$ is the norm of \mathbf{v} , and $|\psi(\boldsymbol{\theta})\rangle$ is a quantum state vector depending on the parameter $\boldsymbol{\theta}$. In this study, $|\psi(\boldsymbol{\theta})\rangle$ generated by unitary matrices $U(\boldsymbol{\theta})$ is conventionally referred to as an ansatz gate, which is constructed by stacking the layers of R_y gates and CNOT gates. Notably, the ansatz parameters $\boldsymbol{\theta}$ is given by the following

$$\boldsymbol{\theta} = [\theta_1^1, \dots, \theta_1^{2m}, \theta_2^1, \dots, \theta_2^{2m}, \dots, \theta_n^1, \dots, \theta_n^{2m}]^\top,$$

where the dimension of the parameter vector $\boldsymbol{\theta}$ is linear with respect to the number of R_y gate m and number of qubits n .

On the basis of the foregoing, using r and $\boldsymbol{\theta}$, the objective function $E_v(\mathbf{v})$ is redefined as follows:

$$E(\mathbf{v}) = \mathcal{F}(r, \boldsymbol{\theta}) = \frac{1}{2} r^2 \langle \psi(\boldsymbol{\theta}) | A | \psi(\boldsymbol{\theta}) \rangle - r \langle \psi(\boldsymbol{\theta}) | f \rangle. \quad (3)$$

By minimizing $\mathcal{F}(r, \boldsymbol{\theta})$ with respect to r , we can reduce the problem to a minimization problem with respect to $\boldsymbol{\theta}$.

$$E(\boldsymbol{\theta}) = -\frac{1}{2} \frac{|\langle \psi(\boldsymbol{\theta}) | f \rangle|^2}{\langle \psi(\boldsymbol{\theta}) | A | \psi(\boldsymbol{\theta}) \rangle}.$$

Furthermore, for any quantum state vector $|\phi\rangle$ or $|\psi\rangle$, we define

$$|\phi, \psi\rangle := \frac{1}{\sqrt{2}} (|0\rangle |\phi\rangle + |1\rangle |\psi\rangle). \quad (4)$$

Then, the following can be obtained using Eq. (4):

$$E(\boldsymbol{\theta}) = -\frac{1}{2} \frac{[\langle \psi(\boldsymbol{\theta}), f | X \otimes I^{\otimes n} | \psi(\boldsymbol{\theta}), f \rangle]^2}{\langle \psi(\boldsymbol{\theta}) | A | \psi(\boldsymbol{\theta}) \rangle}. \quad (5)$$

Furthermore, the quantum state $|\phi, \psi\rangle$ shown in Eq. (4) is constructed by the quantum circuit in Figure 1. Here, the unitary matrices U_ϕ and U_ψ satisfy $|\phi\rangle = U_\phi |0\rangle^{\otimes n}$ and $|\psi\rangle = U_\psi |0\rangle^{\otimes n}$, respectively. The construction of multiple controlled- U_ϕ and controlled- U_ψ requires $O(n^2)$ quantum gates and a circuit depth of $O(n)$ [20].

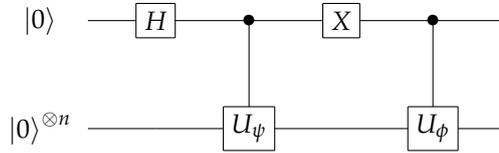


Figure 1. Quantum circuit diagram for generating $|\phi, \psi\rangle$.

The coefficient matrix $A^{(d)}$ in the d -dimensional Poisson equation is expressed using matrix A as follows:

$$A^{(d)} = A \otimes I^{\otimes n(d-1)} + I^{\otimes n} \otimes A \otimes I^{\otimes n(d-2)} + \dots + I^{\otimes n(d-1)} \otimes A. \quad (6)$$

Following Eq. (6), assuming a problem can be formulated in one dimension, it can also be formulated in higher dimensions. Therefore, in this paper, we will discuss the one-dimensional VQA in this paper.

3. Variational Quantum Algorithms for Second-Order Linear Differential Equations

3.1. Formulation

We consider the following d -dimensional second-order linear PDEs:

$$-a\nabla \cdot (\nabla u(\mathbf{x})) + \mathbf{b} \cdot \nabla u(\mathbf{x}) + cu(\mathbf{x}) = f(\mathbf{x}), \quad \mathbf{x} \in \Omega, \quad (7)$$

where $a, c \in \mathbb{R}$, $\mathbf{b} \in \mathbb{R}^N$, $\Omega = (L, R)^d$ and $f : \Omega \rightarrow \mathbb{R}$. When $a \neq 0$ and $c = 0$, Eq. (7) represents the steady advection-diffusion equation and when $a \neq 0$, $c = 0$ and $\mathbf{b} = \mathbf{0}$, Eq. (7) represents the d -dimension Poisson equation. We discretize this equation (7) using the central difference method. In one dimension, let the mesh size be $h = (R - L)/(N + 1)$. Then, we obtain the following difference equation:

$$-a \frac{u_{i+1} - 2u_i + u_{i-1}}{h^2} + b \frac{u_{i+1} - u_{i-1}}{2h} + cu_i = f_i, \quad 1 \leq i \leq N.$$

This difference equation is represented by the following linear equation:

$$M\mathbf{u} = \mathbf{f}, \quad (8)$$

where $M \in \mathbb{R}^{N \times N}$ and $\mathbf{u}, \mathbf{f} \in \mathbb{R}^N$. The components of the coefficient matrix M in Eq. (8) depend on the boundary conditions of PDEs. Additionally, let M_p and M_d denote M with the periodic boundary condition and Dirichlet boundary condition, $u_0 = u_N$ and $u_0 = u_N = 0$ respectively. Here, the matrices M_p and M_d are given as follows:

$$M_p := \begin{bmatrix} \alpha & \gamma & 0 & \dots & 0 & 0 & \beta \\ \beta & \alpha & \gamma & \dots & 0 & 0 & 0 \\ 0 & \beta & \alpha & \dots & 0 & 0 & 0 \\ \vdots & & & \ddots & & & \vdots \\ 0 & 0 & 0 & \dots & \alpha & \gamma & 0 \\ 0 & 0 & 0 & \dots & \beta & \alpha & \gamma \\ \gamma & 0 & 0 & \dots & 0 & \beta & \alpha \end{bmatrix},$$

$$M_d := \begin{bmatrix} \alpha & \gamma & 0 & \dots & 0 & 0 & 0 \\ \beta & \alpha & \gamma & \dots & 0 & 0 & 0 \\ 0 & \beta & \alpha & \dots & 0 & 0 & 0 \\ \vdots & & & \ddots & & & \vdots \\ 0 & 0 & 0 & \dots & \alpha & \gamma & 0 \\ 0 & 0 & 0 & \dots & \beta & \alpha & \gamma \\ 0 & 0 & 0 & \dots & 0 & \beta & \alpha \end{bmatrix},$$

where $\Omega = (0, 1)$, a mesh size $h = 1/(N + 1)$, $\alpha = 2a/h^2 + c$, $\beta = -b/(2h) - a/h^2$, and $\gamma = b/(2h) - a/h^2$. Let us solve the linear equation in Eq. (8). Assuming the matrix M is symmetric positive definite, solving Eq. (8) will be equivalent to minimizing an objective function

$$\mathcal{F}(\mathbf{v}) := \frac{1}{2} \mathbf{v}^\dagger M \mathbf{v} - \mathbf{v}^\dagger \mathbf{f}.$$

However, as M is not always a symmetric positive-definite matrix, VQA for the Poisson equation cannot be applied to the optimization problem. Hence, we solve the following normal equation:

$$M^\dagger M \mathbf{u} = M^\dagger \mathbf{f} \quad (9)$$

instead of Eq. (8). Assuming M is nonsingular, the coefficient matrix $M^\dagger M$ will be a symmetric positive-definite matrix. Thus, we define the objective function $E(v)$, as follows:

$$\mathcal{F}_{normal}(v) := \frac{1}{2} v^\dagger M^\dagger M v - v^\dagger M^\dagger f \quad (10)$$

to solve Eq. (9). Further, we express Eq. (10) as a function that depends on θ , similar to the previous section. Thus, we obtain Eq. (11)

$$r_{opt} = \frac{\langle \psi(\theta) | M^\dagger | f \rangle}{\langle \psi(\theta) | M^\dagger M | \psi(\theta) \rangle}. \quad (11)$$

Additionally, we obtain the following objective function $\mathcal{E}(\theta)$ by substituting Eq. (11) for Eq. (3)

$$\mathcal{E}(\theta) = -\frac{1}{2} \frac{[\langle \psi(\theta) | M^\dagger | f \rangle]^2}{\langle \psi(\theta) | M^\dagger M | \psi(\theta) \rangle}. \quad (12)$$

Thus, Algorithm 1 expresses how to solve second-order linear PDEs using a VQA. Here, the formulation of the gradient of the objective function $\nabla \mathcal{E}(\theta)$ is discussed in the Appendix A.

Algorithm 1 VQAs for second-order linear PDEs

Require: Objective function \mathcal{E} and gradient $\nabla \mathcal{E}$;

an initial ansatz parameter θ_0 ; a tolerant parameter of gradient norm ϵ

Ensure: Optimized ansatz parameter θ_{opt} and an optimized quantum state $|\psi(\theta_{opt})\rangle$

- 1: **while** $\|\nabla \mathcal{E}(\theta)\| \geq \epsilon$ **do**
 - 2: Encode $|\psi(\theta)\rangle$ on a quantum computer
 - 3: Evaluate $\mathcal{E}(\theta)$ and $\nabla \mathcal{E}(\theta)$ on a quantum computer
 - 4: Update θ using a optimization method employing $\mathcal{E}(\theta)$ and $\nabla \mathcal{E}(\theta)$ on a classical computer.
 - 5: **end while**
-

3.2. Evaluation of the Objective Function

As M is not unitary, we cannot directly calculate $\langle \psi(\theta) | M_p^\dagger M_p | \psi(\theta) \rangle$ and $\langle \psi(\theta) | M_p^\dagger | f \rangle$ in Eq. (3) on a quantum computer. Therefore, we first decompose $M_p^\dagger M_p$ and M_p^\dagger into unitary matrices. M_p is decomposed into

$$M_p = \alpha I^{\otimes n} + \beta P + \gamma P^\dagger, \quad (13)$$

where P is a shift operator defined by

$$P := \sum_{i=0}^{N-1} |(i+1) \bmod N\rangle \langle i|$$

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

Thus, matrix $M^\dagger M$ is represented by

$$\begin{aligned} M_p^\dagger M_p &= (\alpha I^{\otimes n} + \beta P^\dagger + \gamma P)(\alpha I^{\otimes n} + \beta P + \gamma P^\dagger) \\ &= (\alpha^2 + \beta^2 + \gamma^2) I^{\otimes n} + \alpha(\beta + \gamma)(P^\dagger + P) \\ &\quad + \beta\gamma(P P + P^\dagger P^\dagger). \end{aligned} \quad (14)$$

Employing Eq. (13). Let us denote

$$\begin{aligned} |\psi_+(\boldsymbol{\theta})\rangle &:= P |\psi(\boldsymbol{\theta})\rangle, \\ |\psi_-(\boldsymbol{\theta})\rangle &:= P^\dagger |\psi(\boldsymbol{\theta})\rangle \end{aligned}$$

with respect to the quantum state $|\psi(\boldsymbol{\theta})\rangle$. Further, we introduce $|\psi_+(\boldsymbol{\theta})\rangle$ and $|\psi_-(\boldsymbol{\theta})\rangle$ newly defined, into the following expectation terms of observable $M^\dagger M$:

$$\begin{aligned} &\langle \psi(\boldsymbol{\theta}) | M_p^\dagger M_p | \psi(\boldsymbol{\theta}) \rangle \\ &= (\alpha^2 + \beta^2 + \gamma^2) \\ &\quad + \alpha(\beta + \gamma) \langle \psi(\boldsymbol{\theta}) | (P + P^\dagger) | \psi(\boldsymbol{\theta}) \rangle \\ &\quad + \beta\gamma \langle \psi(\boldsymbol{\theta}) | (PP + P^\dagger P^\dagger) | \psi(\boldsymbol{\theta}) \rangle \end{aligned}$$

respectively. The first term $\langle \psi(\boldsymbol{\theta}) | (P + P^\dagger) | \psi(\boldsymbol{\theta}) \rangle$ can yield

$$\begin{aligned} &\langle \psi(\boldsymbol{\theta}) | (P + P^\dagger) | \psi(\boldsymbol{\theta}) \rangle \\ &= \langle \psi(\boldsymbol{\theta}) | P | \psi(\boldsymbol{\theta}) \rangle + \langle \psi(\boldsymbol{\theta}) | P^\dagger | \psi(\boldsymbol{\theta}) \rangle \\ &= \langle \psi(\boldsymbol{\theta}) | \psi_+(\boldsymbol{\theta}) \rangle + \langle \psi_+(\boldsymbol{\theta}) | \psi(\boldsymbol{\theta}) \rangle \\ &= 2 \langle \psi_+(\boldsymbol{\theta}), \psi(\boldsymbol{\theta}) | X \otimes I^{\otimes n} | \psi_+(\boldsymbol{\theta}), \psi(\boldsymbol{\theta}) \rangle. \end{aligned}$$

The second term $\langle \psi(\boldsymbol{\theta}) | (PP + P^\dagger P^\dagger) | \psi(\boldsymbol{\theta}) \rangle$ in Eq. (13) can be evaluated as

$$\begin{aligned} &\langle \psi(\boldsymbol{\theta}) | (PP + P^\dagger P^\dagger) | \psi(\boldsymbol{\theta}) \rangle \\ &= \langle \psi(\boldsymbol{\theta}) | PP | \psi(\boldsymbol{\theta}) \rangle + \langle \psi(\boldsymbol{\theta}) | P^\dagger P^\dagger | \psi(\boldsymbol{\theta}) \rangle \\ &= \langle \psi_-(\boldsymbol{\theta}) | \psi_+(\boldsymbol{\theta}) \rangle + \langle \psi_+(\boldsymbol{\theta}) | \psi_-(\boldsymbol{\theta}) \rangle \\ &= 2 \langle \psi_+(\boldsymbol{\theta}), \psi_-(\boldsymbol{\theta}) | X \otimes I^{\otimes n} | \psi_+(\boldsymbol{\theta}), \psi_-(\boldsymbol{\theta}) \rangle. \end{aligned}$$

From the foregoing, the denominator in Eq. (3) can be decomposed to

$$\begin{aligned} &\langle \psi(\boldsymbol{\theta}) | M_p^\dagger M_p | \psi(\boldsymbol{\theta}) \rangle \\ &= (\alpha^2 + \beta^2 + \gamma^2) I^{\otimes n} \\ &\quad + 2\alpha(\beta + \gamma) \langle \psi_+(\boldsymbol{\theta}), \psi(\boldsymbol{\theta}) | X \otimes I^{\otimes n} | \psi_+(\boldsymbol{\theta}), \psi(\boldsymbol{\theta}) \rangle \\ &\quad + 2\beta\gamma \langle \psi_+(\boldsymbol{\theta}), \psi_-(\boldsymbol{\theta}) | X \otimes I^{\otimes n} | \psi_+(\boldsymbol{\theta}), \psi_-(\boldsymbol{\theta}) \rangle, \end{aligned}$$

and we can calculate Eq. (14) on a quantum computer, where each term in the equation is represented as the expectation of the Pauli matrices. Thereafter, we evaluate the numerator in Eq. (3). The numerator can be decomposed into

$$\begin{aligned} \langle \psi(\boldsymbol{\theta}) | M_p^\dagger | f \rangle &= \alpha \langle \psi(\boldsymbol{\theta}) | f \rangle + \beta \langle \psi(\boldsymbol{\theta}) | P^\dagger | f \rangle + \gamma \langle \psi(\boldsymbol{\theta}) | P | f \rangle \\ &= \alpha \langle \psi(\boldsymbol{\theta}) | f \rangle + \beta \langle \psi_+(\boldsymbol{\theta}) | f \rangle + \gamma \langle \psi_-(\boldsymbol{\theta}) | f \rangle \\ &= \alpha \langle \psi(\boldsymbol{\theta}), f | X \otimes I^{\otimes n} | \psi(\boldsymbol{\theta}), f \rangle \\ &\quad + \beta \langle \psi_+(\boldsymbol{\theta}), f | X \otimes I^{\otimes n} | \psi_+(\boldsymbol{\theta}), f \rangle \\ &\quad + \gamma \langle \psi_-(\boldsymbol{\theta}), f | X \otimes I^{\otimes n} | \psi_-(\boldsymbol{\theta}), f \rangle. \end{aligned}$$

Note that if $|\psi(\boldsymbol{\theta})\rangle, |\psi_+(\boldsymbol{\theta})\rangle, |\psi_-(\boldsymbol{\theta})\rangle$ and $|f\rangle$ are real-valued vector, then the inner products $\langle \psi(\boldsymbol{\theta}) | f \rangle, \langle \psi_+(\boldsymbol{\theta}) | f \rangle, \langle \psi_-(\boldsymbol{\theta}) | f \rangle$ are commutative.

3.3. Dirichlet Boundary Condition

The proposed method can be applied to the Dirichlet boundary condition. Notably, M_d can also be decomposed into

$$M_d = \alpha I^{\otimes n} + \beta P + \gamma P^\dagger - M_{BC}, \quad (15)$$

where the matrix M_{BC} is given as follows:

$$M_{BC} = \begin{bmatrix} 0 & 0 & 0 & \cdots & 0 & 0 & \beta \\ 0 & 0 & 0 & \cdots & 0 & 0 & 0 \\ 0 & 0 & 0 & \cdots & 0 & 0 & 0 \\ & & & \vdots & & & \\ 0 & 0 & 0 & \cdots & 0 & 0 & 0 \\ 0 & 0 & 0 & \cdots & 0 & 0 & 0 \\ \gamma & 0 & 0 & \cdots & 0 & 0 & 0 \end{bmatrix} = \beta |0\rangle \langle N-1| + \gamma |N-1\rangle \langle 0|.$$

Thus, $M_d^\dagger M_d$ can be decomposed into

$$\begin{aligned} M_d^\dagger M_d &= (\alpha I^{\otimes n} + \beta P^\dagger + \gamma P - M_{BC}^\dagger)(\alpha I^{\otimes n} + \beta P + \gamma P^\dagger - M_{BC}) \\ &= M_p^\dagger M_p - M_p^\dagger M_{BC} - M_{BC}^\dagger M_p + M_{BC}^\dagger M_{BC}. \end{aligned} \quad (16)$$

To calculate the expectation of Eq. (16) on a quantum computer, we decompose the matrix M_{BC} into symmetric and non-symmetric matrices, V and W , respectively.

$$M_{BC} = \frac{\beta + \gamma}{2} V + \frac{\beta - \gamma}{2} W. \quad (17)$$

Here, V and W are given as follows:

$$V = \begin{bmatrix} 0 & 0 & 0 & \cdots & 0 & 0 & 1 \\ 0 & 0 & 0 & \cdots & 0 & 0 & 0 \\ 0 & 0 & 0 & \cdots & 0 & 0 & 0 \\ & & & \vdots & & & \\ 0 & 0 & 0 & \cdots & 0 & 0 & 0 \\ 0 & 0 & 0 & \cdots & 0 & 0 & 0 \\ 1 & 0 & 0 & \cdots & 0 & 0 & 0 \end{bmatrix} = |0\rangle \langle N-1| + |N-1\rangle \langle 0|,$$

$$W = \begin{bmatrix} 0 & 0 & 0 & \cdots & 0 & 0 & 1 \\ 0 & 0 & 0 & \cdots & 0 & 0 & 0 \\ 0 & 0 & 0 & \cdots & 0 & 0 & 0 \\ & & & \vdots & & & \\ 0 & 0 & 0 & \cdots & 0 & 0 & 0 \\ 0 & 0 & 0 & \cdots & 0 & 0 & 0 \\ -1 & 0 & 0 & \cdots & 0 & 0 & 0 \end{bmatrix} = |N-1\rangle \langle 0| - |0\rangle \langle N-1|.$$

To compute the numerator and denominator of $\mathcal{E}(\theta)$ (Eq. (12)) on a quantum computer, we introduce two decomposition methods for V and W .

3.3.1. Computing the Numerator of the Objective Function

Employing Eq. (16), the numerator of Eq. (12) is decomposed into the following terms:

$$\begin{aligned} \langle \psi(\theta) | M_d^\dagger M_d | \psi(\theta) \rangle &= \langle \psi(\theta) | M_p^\dagger M_p | \psi(\theta) \rangle \\ &\quad - \langle \psi(\theta) | M_p^\dagger M_{BC} + M_{BC}^\dagger M_p | \psi(\theta) \rangle + \langle \psi(\theta) | M_{BC}^\dagger M_{BC} | \psi(\theta) \rangle. \end{aligned} \quad (18)$$

The first term in Eq. (18) can be computed using the proposed method with the periodic boundary condition. Therefore, the second term in Eq. (18) is decomposed into

$$\begin{aligned}
& \langle \psi(\boldsymbol{\theta}) | M_p^\dagger M_{BC} + M_{BC}^\dagger M_p | \psi(\boldsymbol{\theta}) \rangle \\
&= \frac{\beta + \gamma}{2} \langle \psi(\boldsymbol{\theta}) | M_p^\dagger V + V^\dagger M_p | \psi(\boldsymbol{\theta}) \rangle + \frac{\beta - \gamma}{2} \langle \psi(\boldsymbol{\theta}) | M_p^\dagger W + W^\dagger M_p | \psi(\boldsymbol{\theta}) \rangle \\
&= \alpha(\beta + \gamma) \langle \psi(\boldsymbol{\theta}) | V | \psi(\boldsymbol{\theta}) \rangle \\
&\quad + \beta \frac{\beta + \gamma}{2} \langle \psi(\boldsymbol{\theta}) | P^\dagger V + V^\dagger P | \psi(\boldsymbol{\theta}) \rangle + \beta \frac{\beta - \gamma}{2} \langle \psi(\boldsymbol{\theta}) | P^\dagger W + W^\dagger P | \psi(\boldsymbol{\theta}) \rangle \\
&\quad + \gamma \frac{\beta + \gamma}{2} \langle \psi(\boldsymbol{\theta}) | PV + V^\dagger P^\dagger | \psi(\boldsymbol{\theta}) \rangle + \gamma \frac{\beta - \gamma}{2} \langle \psi(\boldsymbol{\theta}) | PW + W^\dagger P^\dagger | \psi(\boldsymbol{\theta}) \rangle.
\end{aligned} \tag{19}$$

We perform the decomposition using Eq. (17). Afterward, we extract the following outer products with a computational basis from the matrix included in each term in Eq. (19):

$$\begin{aligned}
P^\dagger V + V^\dagger P &= 2 |N-1\rangle \langle N-1| + |N-2\rangle \langle 0| + |0\rangle \langle N-2|, \\
PV + V^\dagger P^\dagger &= 2 |0\rangle \langle 0| + |1\rangle \langle N-1| + |N-1\rangle \langle 1|, \\
P^\dagger W + W^\dagger P &= |N-2\rangle \langle 0| + |0\rangle \langle N-2| - 2 |N-1\rangle \langle N-1|, \\
PW + W^\dagger P^\dagger &= |1\rangle \langle N-1| + |N-1\rangle \langle 1| - 2 |0\rangle \langle 0|.
\end{aligned}$$

To compute the expectations in a quantum computer, these outer products must be denoted as measurement operators. Further, by performing matrix decomposition using measurement operators, we localize the quantum circuits and reduce the number of measurements involved in expectation-value measurement.

Here, we define the measurement operators $|GHZ\rangle$, $|GHZ_+\rangle$, and $|GHZ_-\rangle$, as

$$|GHZ\rangle := \frac{|0\rangle + |N-1\rangle}{\sqrt{2}} = U_{GHZ} |0\rangle, \tag{20}$$

$$|GHZ_+\rangle := \frac{|1\rangle + |N-1\rangle}{\sqrt{2}} = U_{GHZ_+} |0\rangle, \tag{21}$$

$$|GHZ_-\rangle := \frac{|0\rangle + |N-2\rangle}{\sqrt{2}} = U_{GHZ_-} |0\rangle. \tag{22}$$

Here these unitary matrices, $U_{GHZ}, U_{GHZ_+}, U_{GHZ_-} \in \mathbb{R}^{N \times N}$ are equivalent to the quantum circuits shown in Figures 2, 3 and 4, respectively: these quantum circuits are constructed using $\mathcal{O}(n)$ fundamental quantum gates.

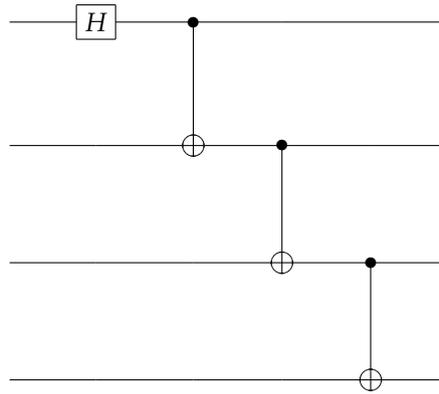


Figure 2. Quantum circuit diagram for U_{GHZ}

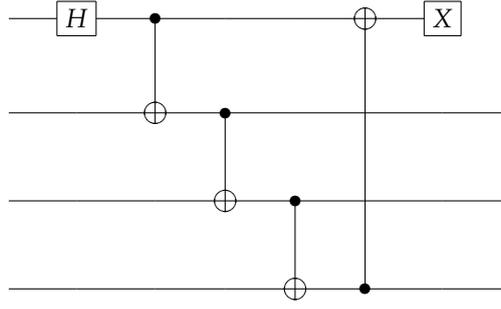


Figure 3. Quantum circuit diagram for U_{GHZ+}

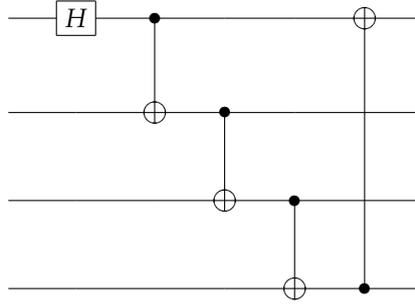


Figure 4. Quantum circuit diagram for U_{GHZ-}

Furthermore, these outer products can be introduced as projection operators using Eqs. (20) and (21) and (22).

$$\begin{aligned}
 |N-1\rangle\langle 0| + |0\rangle\langle N-1| &= 2|GHZ\rangle\langle GHZ| - |0\rangle\langle 0| - |N-1\rangle\langle N-1|, \\
 |N-1\rangle\langle 1| + |1\rangle\langle N-1| &= 2|GHZ_+\rangle\langle GHZ_+| - |1\rangle\langle 1| - |N-1\rangle\langle N-1|, \\
 |N-2\rangle\langle 0| + |0\rangle\langle N-2| &= 2|GHZ_-\rangle\langle GHZ_-\rangle - |0\rangle\langle 0| - |N-2\rangle\langle N-2|.
 \end{aligned}$$

From the above, Eq. (19) is given as follows, and the expectation can be obtained from the projection measurement for each term.

$$\begin{aligned}
 &\langle \psi(\theta) | M_p^\dagger M_{BC} + M_{BC}^\dagger M_p | \psi(\theta) \rangle \\
 &= 2\alpha(\beta + \gamma) |\langle \psi(\theta) | U_{GHZ} | 0 \rangle|^2 + 2\beta\gamma |\langle \psi(\theta) | U_{GHZ_+} | 0 \rangle|^2 \\
 &\quad + 2\beta\gamma |\langle \psi(\theta) | U_{GHZ_-} | 0 \rangle|^2 - \{\alpha(\beta + \gamma) + \gamma(\beta - 2\gamma)\} |\langle \psi(\theta) | 0 \rangle|^2 \\
 &\quad - \{\alpha(\beta + \gamma) + \beta(\gamma - 2\beta)\} |\langle \psi(\theta) | N-1 \rangle|^2 - \beta\gamma |\langle \psi(\theta) | 1 \rangle|^2 - \beta\gamma |\langle \psi(\theta) | N-2 \rangle|^2.
 \end{aligned}$$

The third term of Eq. (18) is given as follows:

$$\langle \psi(\theta) | M_{BC}^\dagger M_{BC} | \psi(\theta) \rangle = \gamma^2 |\langle \psi(\theta) | 0 \rangle|^2 + \beta^2 |\langle \psi(\theta) | N-1 \rangle|^2.$$

From the above, Eq. (18) is decomposed into the following terms:

$$\begin{aligned}
 \langle \psi(\theta) | M_d^\dagger M_d | \psi(\theta) \rangle &= (\alpha^2 + \beta^2 + \gamma^2) + \alpha(\beta + \gamma) \langle \psi(\theta) | (P + P^\dagger) | \psi(\theta) \rangle \\
 &\quad + \beta\gamma \langle \psi(\theta) | (PP + P^\dagger P^\dagger) | \psi(\theta) \rangle - 2\alpha(\beta + \gamma) |\langle \psi(\theta) | U_{GHZ} | 0 \rangle|^2 \\
 &\quad - 2\beta\gamma |\langle \psi(\theta) | U_{GHZ_-} | 0 \rangle|^2 - 2\beta\gamma |\langle \psi(\theta) | U_{GHZ_+} | 0 \rangle|^2 \\
 &\quad + \{\alpha(\beta + \gamma) + \gamma(\beta - \gamma)\} |\langle \psi(\theta) | 0 \rangle|^2 + \beta\gamma |\langle \psi(\theta) | 1 \rangle|^2 \\
 &\quad + \beta\gamma |\langle \psi(\theta) | N-2 \rangle|^2 + \{\alpha(\beta + \gamma) + \beta(\gamma - \beta)\} |\langle \psi(\theta) | N-1 \rangle|^2.
 \end{aligned}$$

3.3.2. Computing the Denominator of the Objective Function

Employing Eq. (15), we decomposed the denominator of Eq. (12) as follows:

$$\langle \psi(\boldsymbol{\theta}) | M_{BC}^\dagger | f \rangle = \frac{\beta + \gamma}{2} \langle \psi(\boldsymbol{\theta}) | V | f \rangle + \frac{\beta - \gamma}{2} \langle \psi(\boldsymbol{\theta}) | W | f \rangle. \quad (23)$$

The given matrix factorization is limited to cases where V and W could be decomposed into measurement operators, represented as the expected values of V and W , respectively. Alternatively, we consider the decomposition method for calculation using a quantum computer. Furthermore, matrices V and W can be decomposed as follows:

$$V = \frac{V_{BC} + V_{BC}^\dagger}{2},$$

$$W = \frac{-W_{BC} + W_{BC}^\dagger}{2},$$

where the unitary matrices V_{BC} and W_{BC} are defined as

$$V_{BC} := \begin{bmatrix} 0 & 0 & 0 & \cdots & 0 & 0 & \cdots & 0 & 0 & 1 \\ 0 & 0 & 0 & \cdots & 0 & 0 & \cdots & 0 & 1 & 0 \\ 0 & 0 & 0 & \cdots & 0 & 0 & \cdots & 1 & 0 & 0 \\ & & & & & & & \vdots & & \\ 0 & 0 & 0 & \cdots & 0 & 1 & \cdots & 0 & 0 & 0 \\ 0 & 0 & 0 & \cdots & -1 & 0 & \cdots & 0 & 0 & 0 \\ & & & & & & & \vdots & & \\ 0 & 0 & -1 & \cdots & 0 & 0 & \cdots & 0 & 0 & 0 \\ 0 & -1 & 0 & \cdots & 0 & 0 & \cdots & 0 & 0 & 0 \\ 1 & 0 & 0 & \cdots & 0 & 0 & \cdots & 0 & 0 & 0 \end{bmatrix},$$

$$W_{BC} := \begin{bmatrix} 0 & 0 & 0 & \cdots & 0 & 0 & \cdots & 0 & 0 & 1 \\ 0 & 0 & 0 & \cdots & 0 & 0 & \cdots & 0 & 1 & 0 \\ 0 & 0 & 0 & \cdots & 0 & 0 & \cdots & 1 & 0 & 0 \\ & & & & & & & \vdots & & \\ 0 & 0 & 0 & \cdots & 0 & 1 & \cdots & 0 & 0 & 0 \\ 0 & 0 & 0 & \cdots & 1 & 0 & \cdots & 0 & 0 & 0 \\ & & & & & & & \vdots & & \\ 0 & 0 & 1 & \cdots & 0 & 0 & \cdots & 0 & 0 & 0 \\ 0 & 1 & 0 & \cdots & 0 & 0 & \cdots & 0 & 0 & 0 \\ -1 & 0 & 0 & \cdots & 0 & 0 & \cdots & 0 & 0 & 0 \end{bmatrix}.$$

From the above, the expectation (23) is decomposed into the linear combination of the expectation of V_{BC} and W_{BC} .

$$\begin{aligned} & \langle \psi(\boldsymbol{\theta}) | M_{BC}^\dagger | f \rangle \\ &= \frac{\beta + \gamma}{2} \left(\langle \psi(\boldsymbol{\theta}) | V_{BC} | f \rangle + \langle \psi(\boldsymbol{\theta}) | V_{BC}^\dagger | f \rangle \right) + \frac{\gamma - \beta}{2} \left(\langle \psi(\boldsymbol{\theta}) | W_{BC} | f \rangle - \langle \psi(\boldsymbol{\theta}) | W_{BC}^\dagger | f \rangle \right) \\ &= \frac{\beta + \gamma}{4} \left(\langle \psi(\boldsymbol{\theta}), V_{BC} f | X \otimes I^{\otimes n} | \psi(\boldsymbol{\theta}), V_{BC} f \rangle + \langle \psi(\boldsymbol{\theta}), V_{BC} f | X \otimes I^{\otimes n} | \psi(\boldsymbol{\theta}), V_{BC}^\dagger f \rangle \right) \\ & \quad + \frac{\gamma - \beta}{4} \left(\langle \psi(\boldsymbol{\theta}), W_{BC} f | X \otimes I^{\otimes n} | \psi(\boldsymbol{\theta}), W_{BC} f \rangle - \langle \psi(\boldsymbol{\theta}), W_{BC}^\dagger f | X \otimes I^{\otimes n} | \psi(\boldsymbol{\theta}), W_{BC}^\dagger f \rangle \right). \end{aligned}$$

Furthermore, the quantum circuits corresponding to V_{BC} and W_{BC} are shown in Figures 5 and 6, respectively. The quantum circuits of V_{BC} and W_{BC} comprise $n - 1$ control bits and one target multi-Toffoli gate. Since V_{BC} and W_{BC} include multi-controlled Toffoli gates, they require a circuit depth of $O(n)$ and are composed of $O(n)$ elementary quantum gates, with an ancilla qubit [21]. Furthermore, considering the circuit cost of Eq. (4), one can see that a total

of $O(n^2)$ elementary quantum gates is required. Therefore, from the perspective of gate implementation costs [22], decomposing into projection operators is effective for obtaining the expected values of matrices M_{BC}^\dagger and $M_{BC}^\dagger M_{BC}$.

Classical shadows are a useful technique for efficiently estimating expectation values [23]. In particular, by using random Clifford circuits, it is possible to compute expectation values with respect to arbitrary matrices. However, since the upper bound of the variance for random Clifford measurements scales as $O(\text{tr}(M^2))$, the variance depends on the values of α , β , and γ . That is, as these values increase, the variance also increases. In contrast, in our method, the error in the expectation value does not depend on the magnitude of the individual matrix components (α , β , and γ). Therefore, when $\text{tr}(M)$ is large, our method can be considered advantageous

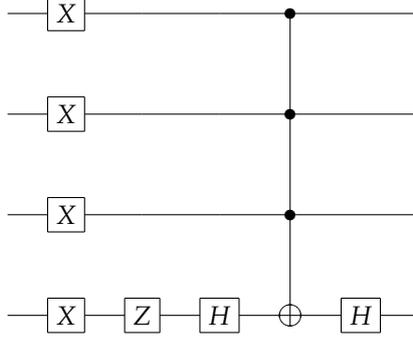


Figure 5. Quantum circuit diagram of V_{BC} .

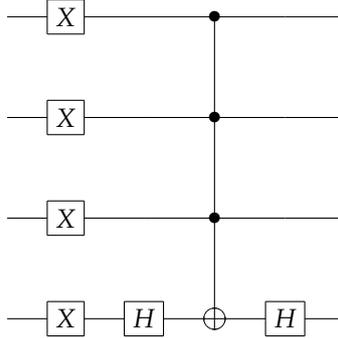


Figure 6. Quantum circuit diagram of W_{BC} .

Additionally, expectation value estimation using the Extended Bell Measurement (XBM) is also a useful approach [24]. XBM is a technique for estimating expectation values of banded matrices and can compute expectation values involving tridiagonal matrices using $O(n)$ measurement operators. The measurement operators used in XBM consist of at most $n - 1$ CNOT gates and one Hadamard gate. XBM is applicable to general problems including Eq. (7). However, simultaneous measurements using XBM yield not only the matrix elements required for computation but also additional and irrelevant elements, which may lead to a higher number of shots required to compute the expectation values. In contrast, our method, particularly with the periodic boundary condition, is tailored for second-order linear differential equations with constant coefficients, and therefore avoids computing irrelevant elements. We will investigate the relationship between the number of unnecessary elements and the required number of shots elsewhere in the future.

4. Numerical Experiments

In this section, we present numerical-analysis experiments using the proposed method. In this experiment, a quantum computing simulation is performed using a classical computer. Further, Qiskit (version 0.38.0) is used as the quantum computation library [25], and Qiskit provided a simulator backend. In this experiment, we exclusively employed the Qiskit StatevectorSimulator. As StatevectorSimulator performed matrix-vector operations internally, it does not account for the sampling errors that may arise in the calculation of the expectation value.

The initial values of the variational parameters were randomly selected from the $0 \leq \theta_i^t \leq 4\pi$ range. Additionally, the BFGS method, a type of gradient-descent method, was employed as the parameter-update method.

Further, we constructed a specific ansatz gate layer (Figure 7) and the number of the ansatz-gate layer m was 3.

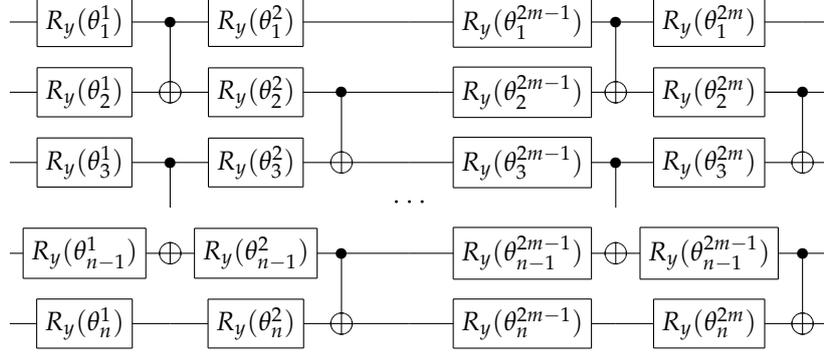


Figure 7. Quantum circuit diagram for the ansatz gate $U(\theta)$.

To compare the numerical solutions obtained with the proposed method, the solutions obtained using LU decomposition for the linear equation (Eq. (8)) were utilized.

The following PDEs was considered for the numerical calculations:

$$-\frac{d^2u}{dx^2} + 0.2\frac{du}{dx} + 0.1u = c_f, \quad (24)$$

where the boundary condition in Eq. (24) was the Dirichlet boundary condition. Furthermore, the right-hand vector is defined, as follows:

$$|f\rangle = 2^{n/2}c_f(H^{\otimes n}|0\rangle^{\otimes n}),$$

where the matrix H is the Hadamard gate and the coefficient $c_f = 0.1$.

Figure 8 shows that the solution obtained using the proposed method was plotted, exhibiting significant agreement with the analytical solution. Furthermore, Table 1 shows that regardless of the increase in the number of qubits, ϵ_c and ϵ_q exhibited nearly identical values, indicating that the solution obtained using the proposed method performed at a comparable level with that obtained via classical computation. Therefore, we concluded that the proposed method functioned appropriately as a method for solving second-order linear differential equations with constant coefficients.

Table 1. Relative errors, ϵ_c and ϵ_q for $n = 3, 4$ and 5 qubit cases.

the number of qubits	the relative error ϵ_c	the relative error ϵ_q
3 qubits	7.8455×10^{-4}	7.8478×10^{-4}
4 qubits	2.2014×10^{-4}	2.2027×10^{-4}
5 qubits	5.8441×10^{-5}	4.4997×10^{-4}

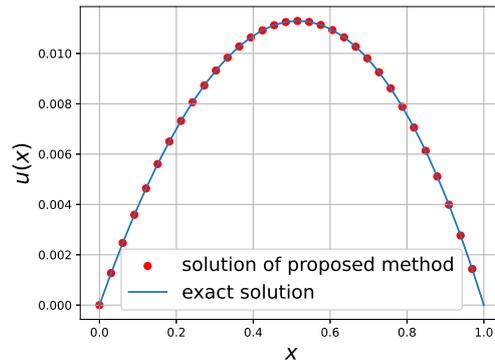


Figure 8. Plot of the analytical solution of Eq. (24) for 5 qubits, and the solution obtained using the proposed method.

5. Conclusion

In this study, we proposed a VQA for solving constant-coefficient second-order linear differential equations with the periodic and the Dirichlet boundary condition, and our finding contributed the following two points to the extant literature:

1. We provided $\mathcal{E}(\boldsymbol{\theta})$ for the normal equations corresponding to the discretized linear equations and demonstrated a matrix decomposition method for reducing them to a computationally feasible form on a quantum computer.
2. For the case involving the imposition of the Dirichlet boundary condition, we proposed a method that focused on the difference between the matrices obtained with the periodic and Dirichlet boundary condition, following by decomposing this difference into a linear combination of unitary matrices and measurement operators.

To solve the linear equation, $M\mathbf{u} = \mathbf{f}$, the condition number is given by

$$\kappa(M) = \|M\| \cdot \|M^{-1}\|.$$

When the condition number κ was large, the numerical solution tended to be unstable. Particularly, as κ of the normal equations $M^\dagger M\mathbf{u} = M^\dagger \mathbf{f}$ ($\kappa(M^\dagger M)$) was approximately the square of $\kappa(M)$, directly solving the normal equations presented a numerically unstable problem. In classical computation, preconditioning is conventionally performed to improve κ . Therefore, QAs that improve κ of the coefficient matrices and reduce the computational cost of the objective function. However, studying the preconditioning matrices for efficient quantum computation is a challenging task, requiring entirely different approaches from the existing preconditioning methods [19]. In VQAs, It is important to avoid barren plateaus when computing the objective function is an important issue. Barren plateaus refer to the phenomenon where the gradient of the objective function vanishes exponentially with respect to the number of qubits. This issue arises due to the execution of deep quantum circuits or measurements involving global quantum operations [26]. Although a complete solution to the barren plateaus problem has not yet been discovered, strategies that mitigate its effects are preferable. For example, designing ansatz gates specialized for specific problems has been proposed as a potential approach [27].

Author Contributions

The first author R.S. executed the project and wrote the manuscript. The second author T.S.: review, editing, supervision; the third author T.K.: review, editing. The 4th author S.-L.Z.: review, editing. All the authors discussed the project and All authors have read and agreed to the published version of the manuscript.

Funding

This work was supported by JSPS KAKENHI Grant Numbers JP20K20397 and JP25H00449.

Conflicts of Interest

The authors declare no conflicts of interest.

Data Availability Statement

The data supporting the findings of this study are available at Nagoya university. Interested researchers may contact the corresponding author for access.

Acknowledgments

We appreciate the valuable comments from the reviewers.

Appendix A

To update the angle parameters $\boldsymbol{\theta}$, we perform the optimization using gradient-descent methods such as the BFGS algorithm [28]. In general VQAs, gradient values are computed by the parameter-shift rule [29].

The gradient of the quantum state vector with respect to $\boldsymbol{\theta}$ ($|\psi(\boldsymbol{\theta})\rangle$) is given as follows:

$$\begin{aligned} \frac{\partial}{\partial \boldsymbol{\theta}} |\psi(\boldsymbol{\theta})\rangle &= \frac{\partial}{\partial \boldsymbol{\theta}} U(\boldsymbol{\theta}) |0\rangle^{\otimes n} \\ &= \left[\frac{\partial U(\boldsymbol{\theta})}{\partial \theta_1^1}, \frac{\partial U(\boldsymbol{\theta})}{\partial \theta_1^2}, \dots, \frac{\partial U(\boldsymbol{\theta})}{\partial \theta_i^{t-1}}, \frac{\partial U(\boldsymbol{\theta})}{\partial \theta_i^t}, \frac{\partial U(\boldsymbol{\theta})}{\partial \theta_i^{t-1}}, \dots, \frac{\partial U(\boldsymbol{\theta})}{\partial \theta_n^m} \right]^\top |0\rangle^{\otimes n} \end{aligned}$$

where θ_i^t represents the variational parameter for the i -th qubit of the ansatz gate in the t -th layer.

As shown in the quantum circuit in Figure 7, the quantum gate dependent on θ is only the R_y gate. Therefore, the derivative of the R_y gate corresponding to a certain angle θ_i^t of the ansatz gate is given by the following:

$$\begin{aligned}\frac{\partial R_y(\theta_i^t)}{\partial \theta_i^t} &= \frac{\partial}{\partial \theta_i^t} e^{-i\theta_i^t Y} \\ &= -\frac{iY}{2} e^{-i\theta_i^t Y} \\ &= \frac{1}{2} R_y(\theta_i^t + \pi),\end{aligned}$$

and we only consider the change due to the differentiation of the R_y gate for the quantum gate $U(\theta)$. Therefore, the derivative with respect to θ_i^t for the entire ansatz gate $U(\theta)$ is given as follows:

$$\frac{\partial U(\theta)}{\partial \theta_i^t} = \frac{1}{2} U(\theta_1^1, \theta_2^1, \dots, \theta_{i-1}^t, \theta_i^t + \pi, \theta_{i+1}^t, \dots, \theta_{2m+1}^n).$$

Additionally, the derivative of a quantum vector $|\psi(\theta)\rangle$ is given as follows:

$$\begin{aligned}\frac{1}{2} |\psi^{i,t}(\theta)\rangle &:= \frac{\partial}{\partial \theta_i^t} |\psi(\theta)\rangle \\ &= \frac{1}{2} U(\theta_1^1, \theta_2^1, \dots, \theta_{i-1}^t, \theta_i^t + \pi, \theta_{i+1}^t, \dots, \theta_{2m+1}^n) |0\rangle^n.\end{aligned}\tag{A1}$$

Next, the gradient of $\mathcal{E}(\theta)$ with respect to θ_i^t is given as follows:

$$\begin{aligned}\frac{\partial E(\theta)}{\partial \theta_i^t} &= -\frac{\langle \psi(\theta) | M^\dagger |f\rangle \frac{\partial}{\partial \theta_i^t} \langle \psi(\theta) | M^\dagger |f\rangle}{\langle \psi(\theta) | M^\dagger M | \psi(\theta) \rangle} \\ &\quad + \frac{1}{2} \frac{[\langle \psi(\theta) | M^\dagger |f\rangle]^2 \frac{\partial}{\partial \theta_i^t} \langle \psi(\theta) | M^\dagger M | \psi(\theta) \rangle}{\langle \psi(\theta) | M^\dagger M | \psi(\theta) \rangle^2},\end{aligned}$$

and the derivatives of the expectation values in this equation are given by

$$\begin{aligned}\frac{\partial}{\partial \theta_i^t} \langle \psi(\theta) | M^\dagger M | \psi(\theta) \rangle, \\ \frac{\partial}{\partial \theta_i^t} \langle \psi(\theta) | M^\dagger |f\rangle,\end{aligned}\tag{A2}$$

which can be expressed by Eq. (A1).

Employing Eq. (A1), the derivative with respect to θ_i^t in Eq. (A2), is given by

$$\begin{aligned}\frac{\partial}{\partial \theta_i^t} \langle \psi(\theta) | M^\dagger M | \psi(\theta) \rangle &= \frac{1}{2} \left(\langle \psi^{i,t}(\theta) | M^\dagger M | \psi(\theta) \rangle + \langle \psi(\theta) | M^\dagger M | \psi^{i,t}(\theta) \rangle \right) \\ &= \langle \psi^{i,t}(\theta), \psi(\theta) | X \otimes M^\dagger M | \psi^{i,t}(\theta), \psi(\theta) \rangle,\end{aligned}$$

and

$$\frac{\partial}{\partial \theta_i^t} \langle \psi(\theta) | M^\dagger |f\rangle = \frac{1}{2} \langle \psi^{i,t}(\theta) | M^\dagger |f\rangle.$$

Note that $M^\dagger M$ is a symmetric matrix.

References

1. T. Sogabe. *Krylov Subspace Methods for Linear Systems : Principles of Algorithms*. No. 60 in Springer series in computational mathematics. Springer, Singapore, 2022.
2. J. Biamonte, P. Wittek, N. Pancotti, P. Rebentrost, N. Wiebe, and S. Lloyd. Quantum machine learning. *Nature*, Vol. 549, No. 7671, pp. 195–202, 2017.
3. H.-Y. Huang, K. Bharti, and P. Rebentrost. Near-term quantum algorithms for linear systems of equations with regression loss functions. *New Journal of Physics*, Vol. 23, No. 11, 113021, 2021.
4. P.W. Shor. Algorithms for quantum computation: discrete logarithms and factoring. In *Proceedings 35th Annual Symposium on Foundations of Computer Science*, pp. 124–134, 1994.
5. L. Zhou, S.-T. Wang, S. Choi, H. Pichler, and M. D. Lukin. Quantum approximate optimization algorithm: performance, mechanism, and implementation on near-term devices. *Physical Review X*, Vol. 10, No. 2, 021067, 2020.
6. A.W. Harrow, A. Hassidim, and S. Lloyd. Quantum algorithm for linear systems of equations. *Physical Review Letters*, Vol. 103, No. 15, 150502, 2009.
7. J. Preskill. Quantum computing in the NISQ era and beyond. *Quantum*, Vol. 2, 79, 2018.
8. M. Cerezo, A. Arrasmith, R. Babbush, S. C. Benjamin, S. Endo, K. Fujii, J. R. McClean, K. Mitarai, X. Yuan, L. Cincio, and P. J. Coles. Variational quantum algorithms. *Nature Reviews Physics*, Vol. 3, No. 9, pp. 625–644, 2021.
9. A. T. Amos, C. Laughlin, and G. R. Moody. A generalized eigenvalue equation for the hydrogen atom. *Chemical Physics Letters*, Vol. 3, No. 6, pp. 411–413, 1969.
10. A. Peruzzo, J. McClean, P. Shadbolt, M.-H. Yung, X.-Q. Zhou, P. J. Love, A. Aspuru-Guzik, and J. L. O’Brien. A variational eigenvalue solver on a photonic quantum processor. *Nature Communications*, Vol. 5, No. 1, 4213, 2014.
11. Michael A. Nielsen and Issac L. Chuang. *Quantum Computation and Quantum Information*. Cambridge University Press, 2010.
12. M. Ali and M. Kabel. Performance study of variational quantum algorithms for solving the poisson equation on a quantum computer. *Phys. Rev. Appl.*, Vol. 20, 014054, 2023.
13. F. Y. Leong, W.-B. Ewe, and D. E. Koh. Variational quantum evolution equation solver. *Scientific Reports*, Vol. 12, No. 1, 10817, 2022.
14. F. Y. Leong, D. E. Koh, J. F. Kong, S. T. Goh, J. Y. Khoo, W.-B. Ewe, H. Li, J. Thompson, and D. Poletti. Solving fractional differential equations on a quantum computer: A variational approach. *AVS Quantum Science*, Vol. 6, No. 3, 033802, 2024.
15. H.-L. Liu, Y.-S. Wu, L.-C. Wan, S.-J. Pan, S.-J. Qin, F. Gao, and Q.-Y. Wen. Variational quantum algorithm for the Poisson equation. *Physical Review A*, Vol. 104, No. 2, 022418, 2021.
16. M. Lubasch, J. Joo, P. Moinier, M. Kiffner, and D. Jaksch. Variational quantum algorithms for nonlinear problems. *Physical Review A*, Vol. 101, No. 1, 010301, 2020.
17. P. Over, S. Bengoechea, T. Rung, F. Clerici, L. Scandurra, E. de Villiers, and D. Jaksch. Boundary treatment for variational quantum simulations of partial differential equations on quantum computers. *Computers & Fluids*, Vol. 288, 106508, 2025.
18. Y. Sato, R. Kondo, S. Koide, H. Takamatsu, and N. Imoto. Variational quantum algorithm based on the minimum potential energy for solving the Poisson equation. *Physical Review A*, Vol. 104, No. 5, 052409, 2021.
19. A. Hosaka, K. Yanagisawa, S. Koshikawa, I. Kudo, X. Alifu, and T. Yoshida. Preconditioning for a variational quantum linear solver, 2023. arXiv:2312.15657 [quant-ph].
20. A. J. da Silva and D. K. Park. Linear-depth quantum circuits for multiqubit controlled gates. *Phys. Rev. A*, Vol. 106, 042602, 2022.
21. Y. He, M.-X. Luo, E. Zhang, H.-K. Wang, and X.-F. Wang. Decompositions of n-qubit toffoli gates with linear circuit complexity. *International Journal of Theoretical Physics*, Vol. 56, No. 7, pp. 2350–2361, 2017.
22. T. Satoh, S. Oomura, M. Sugawara, and N. Yamamoto. Pulse-engineered controlled-V gate and its applications on superconducting quantum device. *IEEE Transactions on Quantum Engineering*, Vol. 3, pp. 1–10, 2022.
23. H.-Y. Huang, R. Kueng, and J. Preskill. Predicting many properties of a quantum system from very few measurements. *Nature Physics*, Vol. 16, No. 10, pp. 1050–1057, 2020.
24. R. Kondo, Y. Sato, S. Koide, S. Kajita, and H. Takamatsu. Computationally efficient quantum expectation with extended bell measurements. *Quantum*, Vol. 6, 688, 2022.

25. G. Aleksandrowicz, T. Alexander, P. Barkoutsos, L. Bello, Y. Ben-Haim, et al., Qiskit: An open-source framework for quantum computing, 2019. Available at: <https://zenodo.org/record/2562111>
26. M. Cerezo, A. Sone, T. Volkoff, L. Cincio, and P. J. Coles. Cost function dependent barren plateaus in shallow parametrized quantum circuits. *Nature Communications*, Vol. 12, No. 1, 1791, 2021.
27. Z. Holmes, K. Sharma, M. Cerezo, and P. J. Coles. Connecting ansatz expressibility to gradient magnitudes and barren plateaus. *PRX Quantum*, Vol. 3, 010313, 2022.
28. D. F. Shanno. Conditioning of quasi-newton methods for function minimization. *Mathematics of Computation*, Vol. 24, No. 111, pp. 647–656, 1970.
29. K. Mitarai, M. Negoro, M. Kitagawa, and K. Fujii. Quantum circuit learning. *Phys. Rev. A*, Vol. 98, 032309, 2018.