

# Boundary treatment for variational quantum simulations of partial differential equations on quantum computers

Paul Over <sup>a</sup>,<sup>\*</sup> Sergio Bengoechea <sup>a</sup>, Thomas Rung <sup>a</sup>, Francesco Clerici <sup>b</sup>,  
Leonardo Scandurra <sup>b</sup>, Eugene de Villiers <sup>c</sup>, Dieter Jaksch <sup>d,e</sup>

<sup>a</sup> Institute for Fluid Dynamics and Ship Theory, Hamburg University of Technology, Am Schwarzenberg-Campus 4, 21073, Hamburg, Germany

<sup>b</sup> ENGYS Srl, Via del Follatoio, 12, 34148 Trieste TS, Italy

<sup>c</sup> ENGYS Ltd., London, SW18 3SX, United Kingdom

<sup>d</sup> Institute for Quantum Physics, University of Hamburg, Luruper Chaussee 149, 22761, Hamburg, Germany

<sup>e</sup> Clarendon Laboratory, University of Oxford, Parks Road, Oxford OX1 3PU, United Kingdom

## ARTICLE INFO

Dataset link: <https://doi.org/10.25592/uhhfdm.14123>

### Keywords:

Computational fluid dynamics  
Variational quantum algorithms  
Quantum computing  
Boundary conditions

## ABSTRACT

The paper presents a variational quantum algorithm to solve initial-boundary value problems described by second-order partial differential equations. The approach uses hybrid classical/quantum framework that is well suited for quantum computers of the current noisy intermediate-scale quantum era. The partial differential equation is initially translated into an optimal control problem with a modular control-to-state operator (ansatz). The objective function and its derivatives required by the optimizer can efficiently be evaluated on a quantum computer by measuring an ancilla qubit, while the optimization procedure employs classical hardware. The focal aspect of the study is the treatment of boundary conditions, which is tailored to the properties of the quantum hardware using a correction technique. For this purpose, the boundary conditions and the discretized terms of the partial differential equation are decomposed into a sequence of unitary operations and subsequently compiled into quantum gates. The accuracy and gate complexity of the approach are assessed for second-order partial differential equations by classically emulating the quantum hardware. The examples include steady and unsteady diffusive transport equations for a scalar property in combination with various Dirichlet, Neumann, or Robin conditions. The results of this flexible approach display a robust behavior and a strong predictive accuracy in combination with a remarkable *polylog* complexity scaling in the number of qubits of the involved quantum circuits. Remaining challenges refer to adaptive ansatz strategies that speed up the optimization procedure.

## 1. Introduction

Partial Differential Equations (PDE) are ubiquitous for modeling problems in science and engineering. Areas of application include structural and fluid mechanics, electrodynamics, thermodynamics, and quantum physics. Today, sophisticated and dedicated numerical methods are well-established and widely available in all of those areas. However, the increasing demands to resolve wider ranges of spatial and temporal scales with great precision make these simulations expensive and energy-consuming. Furthermore, chip sizes for classical Central Processing Units (CPUs) are expected to converge over the next decade [1] due to the limits of reducing the transistor's size further, putting an end to Moore's Law [2]. In regard to increasing computing power, Quantum Computers (QCs) promise to address some future hardware challenges. Two categories for QC-supported PDE solutions

have been proposed. They are either based on the direct encoding of the PDE solution in a large quantum circuit or on a Variational Quantum Algorithm (VQA), which evaluates shallow circuits in combination with classical optimization methods [3].

For the first category, Quantum Linear Solvers (QLS) are applied to solve the algebraic equation systems derived from the discretization of linear PDEs. In a pioneering work [4], Harrow, Hassidim, and Lloyd presented the HHL quantum algorithm to compute the solution of a Linear System of Equations (LSE). The HHL algorithm promises an exponential speedup, provided that the state preparation and information readout can efficiently be performed on the QC, which is by no means self-evident. The algorithm works best for non-stiff LSE but scales worse than classical methods for stiff problems. In Ref. [5], a modification of the HHL algorithm is applied to solve a LSE originating

<sup>\*</sup> Corresponding author.

E-mail addresses: [paul.over@tuhh.de](mailto:paul.over@tuhh.de) (P. Over), [sergio.bengoechea@tuhh.de](mailto:sergio.bengoechea@tuhh.de) (S. Bengoechea).

from a Finite-Element Method while the temporal evolution of a linear PDE is approximated in Ref. [6]. Computational Fluid Dynamic (CFD) applications are reported in Refs. [7–15]. Steijl and Barakos [7] used a vortex-in-cell CFD method and employed a quantum Fourier solver for the Poisson equation. The works of Brearley and Laizet [8] and Over et al. [9] treat linear transport phenomena described by the advection and the advection-diffusion equation via a Hamiltonian simulation. In Ref. [10] and [11] the PDE system is spatially discretized into a system of coupled Ordinary Differential Equations (ODEs) and approximated via a Quantum Amplitude Estimation Algorithm (QAEA) proposed by Kacewicz [16]. In Ref. [12] and [13], the PDE solution is spatially approximated with truncated Fourier or Chebyshev series, in which the unknown coefficients are determined by a QLS. In the study of Cao et al. [14], the HHL is used to solve the Poisson equation within a predictor–corrector method to solve the incompressible Navier–Stokes equation. The approaches mentioned above mostly require a large number of qubits devoted to error correction, and the number of operations associated with the HHL exceeds the capabilities of current QC hardware. Hence, these methods are more suitable for future fault-tolerant quantum machines and less suitable for QCs of the current Noise Intermediate-Scale Quantum (NISQ) generation [3].

The VQA is an alternative category to solve PDE problems and is dedicated to state-of-the-art NISQ devices [17]. The noise-tolerant procedure is based on the efficient evaluation of a cost function by the QC, while the classical hardware optimizes the parameters of the ansatz circuit that encodes the solution on the QC. As a consequence, the PDE to solve must be reformulated as an optimization problem. The main advantage of this approach is the use of a small number of gate operations, making it more robust to decoherence effects of the quantum registers that compose the circuits.

VQAs are used in many applications such as the identification of ground states [17], or solving LSE [18]. Recent applications to fluid dynamics and related PDEs are, for example, reported in Refs. [19–25]. Burgers’ advection-diffusion equation is studied in Ref. [19] where the discretized sparse matrix problem is decomposed in a finite series of unitary operators and solved by the variational quantum solver proposed in Ref. [20]. In Ref. [21], quantum neural networks are trained with a hybrid VQA to solve systems of nonlinear differential equations that model a supersonic nozzle flow. The work of Sato et al. [22] is of particular relevance for this study and explores the ability of VQAs to solve the Poisson equation by minimizing the potential energy of an elliptic PDE. To this end, the dynamics of the Poisson equation can be linearly decomposed in a series of parameterized shallow quantum circuits. The idea of energy minimization from Sato et al. [22] is extended to time-dependent problems in Leong et al. [23,24].

The present effort aims to combine the quantum framework described in Lubasch et al. [26] with the work of Sato et al. [22]. We focus on a flexible treatment of engineering boundary conditions on QCs and deploy a Hadamard test-based structure to gain quantum advantage [27]. The related single-qubit measurements reduce expensive sampling operations to evaluate the results encoded in a quantum state and thus the associated loss of efficiency [6,15,21]. Accurate, flexible, and robust treatment of diverse boundary conditions is crucial for engineering applications. Despite the importance, a rigorous algorithmic implementation in the context of QC is not trivial, as demonstrated by the variety of specific approaches proposed in the literature. The works of Cao et al. [14] and Childs et al. [12] are restricted to Dirichlet conditions, while a related study of Childs et al. [13] mirrors the solution into the symmetric and antisymmetric sub-spaces for Neumann and Dirichlet cases, respectively. The mixed boundary conditions for the Burgers’ equation suggested in Ref. [11] can only be enforced on classical hardware. The algorithm suggested in Ref. [28] imposes Dirichlet or Neumann conditions on the discrete wave equation prior to the derivation of the Hamiltonian operator, i.e., the solutions in the Hilbert space (trial functions) need to satisfy the boundary conditions. Suau et al. [3] present a modification of this approach,

where only Dirichlet boundaries are included. For VQA applications, the boundary conditions in Ref. [21] are included either as part of the objective function or as an additional constraint on the ansatz function. The Dirichlet and Neumann boundary operators employed by Sato et al. [22] do not form unitary operators. This either requires implementation on classical hardware due to the necessary non-unitary corrections or measuring the whole quantum register, which in turn reduces the efficiency.

The paper suggests combining a classical CFD ghost-point technique with a unitary decomposition of the boundary operators by modifying the route proposed in Ref. [22]. The decomposed boundary operators are subsequently compiled into quantum gates and finally included in the cost function using a deferred correction strategy. The proposed approach permits the full-quantum implementation of arbitrary Dirichlet, Neumann, and Robin conditions without additional constraints on the trial or ansatz functions.

The remainder of the paper is structured as follows: Section 2 introduces the mathematical model, and the boundary treatment is described in Section 3. Subsequently Sections 4 and 5 outline the computational model. The latter also addresses the QC implementation and the quantum circuits. Section 6 discusses means to reduce the gate complexity of the utilized quantum circuits. Section 7 describes the employed optimization procedure in brief and Section 8 is devoted to numerical results. The applications are concerned with steady and transient heat conduction. Particular emphasis is given to temperature distributions obtained with various types of boundary conditions, which are compared to results of classical Finite-Differences (FD). Final conclusions and future directions are outlined in Section 9.

The presented quantum framework is emulated with IBM’s QISKIT environment [29]. Within the publication, vectors and tensors are defined with reference to Cartesian coordinates. Note that binary representations follow the *little-endian* convention.

## 2. Mathematical model

The considered PDE is first introduced and discretized in time. Subsequently, the PDE is cast into an optimization problem, and the objective function is prepared for being evaluated on QC hardware before the spatial discretization is outlined.

### 2.1. Governing equation

The presented approach is restricted to unsteady, spatially 1D problems. An extension to two- or three spatial dimensions is straightforward [30]. For brevity, we use a normalized spatial coordinate  $x \in [0, 1]$  and a time domain  $t \in [0, T]$ . The problems considered herein are described by an unsteady reaction/diffusion PDE for an unknown field or state variable  $y(x, t)$ , viz.

$$\frac{\partial y}{\partial t} - \nu \frac{\partial^2 y}{\partial x^2} - \zeta y p = f \quad \text{in } \Omega_T := (0, 1) \times (0, T]. \quad (1)$$

In Eq. (1), the notation  $\partial^2 y / \partial x^2$  describes the Laplace operator acting on  $y$ . Moreover, the right-hand side term  $f(x)$  denotes a time-independent source,  $p(x, t)$  is a time-dependent potential that is independent of  $y$ ,  $\zeta$  is a simple (real) coefficient, and  $\nu$  refers to the inherently positive (real) diffusivity. In contrast to previous works published in Refs. [14,22–24,31], the proposed framework to solve Eq. (1) applies to arbitrary combinations of Dirichlet and Neumann conditions. The details of the boundary treatment will be outlined in Section 3.

## 2.2. Temporal discretization

The time horizon  $T$  is discretized into  $N_t$  equidistant time instants  $t^l$ , where  $l \in [0, N_t]$  and  $\Delta t$  marks a constant time step. In line with traditional finite approximation strategies, time derivatives are approximated by implicit backward FD, e.g., an implicit first-order Euler scheme or a second-order three-time-level scheme

$$\begin{aligned} \frac{\partial y(x, t^l)}{\partial t} &= \frac{y(x, t^l) - y(x, t^{l-1})}{\Delta t} + \mathcal{O}(\Delta t), \\ \frac{\partial y(x, t^l)}{\partial t} &= \frac{3y(x, t^l) - 4y(x, t^{l-1}) + y(x, t^{l-2})}{2\Delta t} + \mathcal{O}(\Delta t^2). \end{aligned} \quad (2)$$

The examples included herein use the implicit Euler scheme. This results in the following modifications of the potential term  $p \rightarrow p - 1/(\zeta \Delta t)$  and the source term  $f \rightarrow f + y(x, t^{l-1})/\Delta t$ , which yields a spatial PDE at time  $t^l$  in residual form

$$R(y(x, t^l)) = - \left( v \frac{\partial^2 y}{\partial x^2} + \zeta y p + f \right) = 0 \quad \text{in } \Omega := (0, 1). \quad (3)$$

## 2.3. Optimal control problem

A weak form of Eq. (3) follows from a weighted residual formulation and reads

$$\begin{aligned} \int_{\Omega} z(x, t^l) R(y(x, t^l)) dx = 0 \quad \text{or} \\ - \int_{\Omega} z \left( v \frac{\partial^2 y}{\partial x^2} + \zeta y p \right) dx - \int_{\Omega} z f dx = 0, \end{aligned} \quad (4)$$

$\underbrace{\hspace{10em}}_{a(z,y)} \quad \underbrace{\hspace{10em}}_{F(z)}$

where  $z(x, t^l)$  is a weighting function. For  $z = y$ , the solution to Eq. (4) is equivalent to the solution of the variational problem [32,33] characterized by minimizing an objective function  $J$ ,

$$\min_y J(y) \quad \text{with} \quad J(y) := a(y, y) - 2F(y). \quad (5)$$

Mind that the equivalence between the solutions restricts the function space of  $y$  to an elliptic  $a(z, y)$  [32] and that the computed solution is, of course, only an approximation to  $y$ .

The limited ability to represent arbitrary states  $y$  on QC hardware, yields the *partition of unity* constraint given by the  $L^2$  scalar product as [34,35]

$$(y^*, y)_{L^2} = \int_{\Omega} y^*(x, t^l) y(x, t^l) dx = 1, \quad (6)$$

where the asterisk indicates the complex conjugate. Turning our attention to the variational framework in Eq. (5), the function  $y$  is made variable by the control  $v' \in \mathbb{R}^{c+1}$ ,  $v' := (\lambda'_0, \lambda'_c)^T$  ( $c \in \mathbb{N}$ ), using the time-dependent ansatz

$$y(x, t^l, v') = \lambda'_0 u(x, t^l, \lambda'_c) \quad \text{with} \quad (u^*, u)_{L^2} = 1 \quad (7)$$

to enforce the normalization restriction (6). Suppressing the temporal index  $(\dots)^l$  and substituting the ansatz of Eq. (7) into the objective function, one obtains the (time-dependent) minimization problem

$$\begin{aligned} \min_{\mathbf{v}} J(\mathbf{v}) \quad \text{with} \quad J(\mathbf{v}) := & a(y(\mathbf{v}), y(\mathbf{v})) - 2F(y(\mathbf{v})), \\ = & -v \lambda_0^2 \int_{\Omega} u(x, \lambda_c) \frac{\partial^2}{\partial x^2} u(x, \lambda_c) dx \\ & - \zeta \lambda_0^2 \int_{\Omega} u(x, \lambda_c) p(x) u(x, \lambda_c) dx \\ & - 2\lambda_0 \int_{\Omega} u(x, \lambda_c) f(x) dx. \end{aligned} \quad (8)$$

For each time step, this minimization requires the derivatives of the objective function with respect to the control. The Gâteaux derivative with respect to the control  $\mathbf{v}$  provides a first-order optimality condition [34]

$$\frac{\partial J}{\partial \mathbf{v}} \delta \mathbf{v} = \frac{\partial J}{\partial y} \frac{\partial y}{\partial \mathbf{v}} \delta \mathbf{v} = \begin{pmatrix} \nabla_{\lambda_0} J \\ \nabla_{\lambda_c} J \end{pmatrix} \cdot \begin{pmatrix} \delta \lambda_0 \\ \delta \lambda_c \end{pmatrix} = 0 \quad \forall \delta \mathbf{v}. \quad (9)$$

The partial derivatives of the objective function with respect to the control are not discussed in the continuous form here. Instead, a QC-compatible discretized form is developed in Section 7.

## 2.4. Spatial discretization

The unit interval is discretized by  $N_p + 2$  equidistantly spaced points  $x_k$ , with  $k \in [0, N_p + 1]$ . As alluded to earlier, the time  $t \in [0, T]$  is discretized by  $N_t + 1$  equidistant time instants  $t^l$ , which leads to the structured grid depicted in Fig. 1. As indicated by the Figure, we distinguish between interior points (closed circles) and ghost points (open circles), which serve to introduce boundary conditions. Similar to classical finite approximation schemes, variables located along the spatial boundaries can either be part of the unknowns or separated from the unknowns. In the latter case, the boundary conditions are represented by the influence of prescribed ghost point variables on the interior point approximations. In the case of Neumann or Robin conditions, this usually involves the additional discretization of the spatial derivative.

The procedure is as follows: Boundary nodes are assigned to ghost nodes (○) and the spatially discretized objective function (8) is evaluated for the interior nodes (●), with the contributions of the ghost nodes arising from discretized boundary conditions. Thus, the discrete ansatz function vector  $(u(x_0, t^l), u(x_1, t^l), \dots, u(x_{N_p+1}, t^l)) = \mathbf{u}^l$  is represented using the orthogonal basis  $e_k$ , viz.

$$\mathbf{u}^l = u'_0 e_0 + \underbrace{\sum_{k=1}^{N_p} u'_k e_k + u'_{N_p+1} e_{N_p+1}}_{\text{register}}, \quad (10)$$

with the interior to be stored in a register of length  $\log_2(N_p)$  as shown in Section 4. The derivatives of the Laplace operator required to evaluate the objective function (8) are approximated by

$$\frac{\partial^2 u^l_k}{\partial x^2} = \frac{u^l_{k+1} - 2u^l_k + u^l_{k-1}}{\Delta x^2} + \mathcal{O}(\Delta x^2), \quad (11)$$

which is second-order accurate for an equidistant grid.

## 3. Discrete objective function and boundary treatment

The objective function (8) consists of contributions from the Laplace operator  $j_L$ , the potential  $j_p$ , and the source term  $j_S$ . The integration

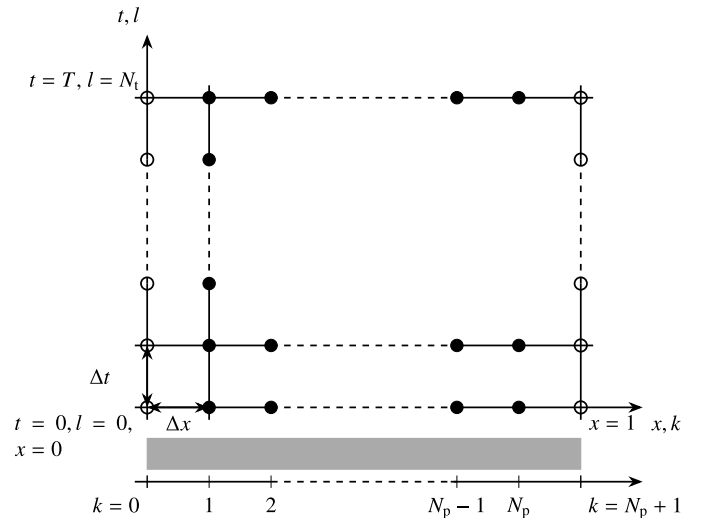


Fig. 1. Illustration of the spatial-temporal discretization on an equidistant grid with boundary/ghost nodes indicated by open symbols (○) and interior nodes by closed symbols (●).

over the interior domain in Eq. (8) is fractioned into sub-integrals around the centrally spaced interior nodes, which are approximated by a second-order accurate midpoint rule, i.e.,

$$J^I = \int_{\Omega} (j_L + j_P + j_S)^I dx \approx \sum_{k=1}^{N_p} (j_{L_k} + j_{P_k} + j_{S_k})^I \Delta x. \quad (12)$$

Subsequently, the discretized boundary conditions are introduced. These are formulated as corrections to established periodic boundary conditions, which yield augmentations of the discrete objective function (12). The approach is compatible for an evaluation on QC hardware and is discussed below using the left boundary ( $k = 0$ ) as an example. The procedure for the right boundary is analogous and is therefore not discussed in detail. The time index is again deliberately suppressed for the sake of clarity.

The potential and source term contributions  $j_{P_k}$ ,  $j_{S_k}$  to Eq. (12) are strictly local and therefore do not interact with the ghost point (boundary) values. The contribution of the discrete Laplace operator, however, is non-local (but homogeneous) and interacts with the boundary. At the first interior point ( $k = 1$ ), the contribution reads

$$j_{L_1} = -\left(\frac{v\lambda_0^2}{\Delta x^2}\right) u_1 (u_0 - 2u_1 + u_2), \quad (13)$$

where  $u_0$  is an unknown that must be closed by the boundary condition. As already mentioned, the strategy refers to corrections of periodic boundaries, for which the left ghost point receives the value from the right interior point and the right ghost point gets the value from the left interior point, i.e.,  $u_0 = u_{N_p}$  and  $u_{N_p+1} = u_1$ . The periodic contribution of the left boundary to  $J$  reads

$$j_{L_1}^P = -\left(\frac{v\lambda_0^2}{\Delta x^2}\right) u_1 (u_{N_p} - 2u_1 + u_2). \quad (14)$$

On the other hand, a Dirichlet condition  $D = u_0$  yields

$$j_{L_1}^D = -\left(\frac{v\lambda_0^2}{\Delta x^2}\right) u_1 (D - 2u_1 + u_2). \quad (15)$$

Aiming to reconstruct the Dirichlet condition (15) from the periodic approach (14), one arrives at

$$\begin{aligned} j_{L_1}^D &= j_{L_1}^P + u_1 \left(\frac{v\lambda_0^2}{\Delta x^2}\right) (u_{N_p} - D) \\ &= \underbrace{j_{L_1}^P + u_1 u_{N_p} \left(\frac{v\lambda_0^2}{\Delta x^2}\right)}_{\text{new } J_1\text{-contrib. } j_{DN_1}} \underbrace{-u_1 D \left(\frac{v\lambda_0^2}{\Delta x^2}\right)}_{\text{source-term modif. } f_1}. \end{aligned} \quad (16)$$

In the case of Neumann conditions, the prescribed gradient  $N = \partial u / \partial x|_0$  first needs to be approximated. The exemplary use of simple first-order FD, viz.  $N \approx (u_1 - u_0) / (\Delta x)$  to substitute  $u_0 = u_1 - N \Delta x$  in Eq. (13) yields

$$\begin{aligned} j_{L_1}^N &= -\left(\frac{v\lambda_0^2}{\Delta x^2}\right) u_1 (u_1 - N \Delta x - 2u_1 + u_2) \\ &= \underbrace{j_{L_1}^P + u_1 u_{N_p} \left(\frac{v\lambda_0^2}{\Delta x^2}\right)}_{\text{new } J_1\text{-contrib. } j_{DN_1}} \underbrace{-u_1 u_1 \left(\frac{v\lambda_0^2}{\Delta x^2}\right)}_{\text{new } J_1\text{-contrib. } j_{N_1}} \underbrace{+u_1 N \left(\frac{v\lambda_0^2}{\Delta x}\right)}_{\text{source-term modif. } f_1}. \end{aligned} \quad (17)$$

Note that the contribution  $j_{DN_1}$ , which neutralizes the periodic term, occurs for any conditions. The technique can be combined with higher-order FD to approximate derivative expressions in Neumann or Robin conditions. To demonstrate this procedure, a central second-order FD approximates the left boundary at  $x_0$ , viz.  $N \approx (u_1 - u_{-1}) / (2\Delta x)$ . The terms  $u_{-1}$  and  $u_0$  are recursively eliminated by combining the discrete Laplace operator of Eq. (13) for  $x_0$  and  $x_1$  and the second-order derivative approximation at  $x_0$ . In the case of second-order Neumann, the modifications result in the additional contribution of  $0.5u_1 f_0 v \lambda_0^2$  to

the source term  $f_1$ . Here, the  $f_0$  indicates the extrapolation of the right-hand side  $f$  in Eq. (1) to  $x_0$ . Analogously, the source term on the right boundary  $f_{N_p}$  is modified by  $0.5u_{N_p} f_{N_p+1} v \lambda_0^2$ .

At the right boundary, the corresponding correction terms read

$$\begin{aligned} j_{L_{N_p}}^D &= j_{L_{N_p}}^P + u_{N_p} \left(\frac{v\lambda_0^2}{\Delta x^2}\right) (u_1 - D) \\ &= \underbrace{j_{L_{N_p}}^P + u_1 u_{N_p} \left(\frac{v\lambda_0^2}{\Delta x^2}\right)}_{\text{new } J_{N_p}\text{-contrib. } j_{DN_{N_p}}} \underbrace{-u_{N_p} D \left(\frac{v\lambda_0^2}{\Delta x^2}\right)}_{\text{source-term modif. } f_{N_p}} \quad \text{and} \quad (18) \\ j_{L_{N_p}}^N &= \underbrace{j_{L_{N_p}}^P + u_1 u_{N_p} \left(\frac{v\lambda_0^2}{\Delta x^2}\right)}_{\text{new } J_{N_p}\text{-contrib. } j_{DN_{N_p}}} \underbrace{-u_{N_p} u_{N_p} \left(\frac{v\lambda_0^2}{\Delta x^2}\right)}_{\text{new } J_{N_p}\text{-contrib. } j_{N_{N_p}}} \\ &\quad \underbrace{-u_{N_p} N \left(\frac{v\lambda_0^2}{\Delta x}\right)}_{\text{source-term modif. } f_{N_p}}. \end{aligned} \quad (19)$$

Again, the contribution  $j_{DN_{N_p}}$ , which neutralizes the periodic entries, occurs for both Neumann and Dirichlet conditions, and it is identical for both boundary locations ( $j_{DN_{N_p}} = j_{DN_1}$ ). The actual correction implementation is given in Eq. (20). It employs three terms: (a) the periodic baseline term, (b) corrections based upon combinations of unknown interior points ( $j_{DN_k}$ ,  $j_{N_k}$ ), and (c) modifications of the source terms  $j_{S_k}$  arising from known boundary condition values ( $D$ ,  $N$ ), viz.

$$\begin{aligned} J(\lambda_0, \lambda_c) &= \sum_{k=1}^{N_p} \underbrace{-\left(\frac{v\lambda_0^2}{\Delta x^2}\right) u_k(\lambda_c) (u_{k+1}(\lambda_c) - 2u_k(\lambda_c) + u_{k-1}(\lambda_c)) \Delta x}_{j_{L_k} : \text{periodic \& interior term}} \\ &\quad + \underbrace{\left(\frac{v\lambda_0^2}{\Delta x^2}\right) (u_1(\lambda_c) u_{N_p}(\lambda_c) + u_1(\lambda_c) u_{N_p}(\lambda_c)) \Delta x}_{j_{DN_1} + j_{DN_{N_p}} : \text{neutralizing left/right term (Dirichlet \& Neumann)}} \\ &\quad - \underbrace{\left(\frac{v\lambda_0^2}{\Delta x^2}\right) u_1^2(\lambda_c) \Delta x}_{j_{N_1} : \text{left Neumann cond.}} - \underbrace{\left(\frac{v\lambda_0^2}{\Delta x^2}\right) u_{N_p}^2(\lambda_c) \Delta x}_{j_{N_{N_p}} : \text{right Neumann cond.}} \\ &\quad + \sum_{k=1}^{N_p} \underbrace{-\zeta \lambda_0^2 u_k(\lambda_c) p_k u_k(\lambda_c) \Delta x}_{j_{P_k} : \text{potential term}} + \sum_{k=1}^{N_p} \underbrace{-2\lambda_0 u_k(\lambda_c) \tilde{f}_k \Delta x}_{j_{S_k} : \text{source term}}. \end{aligned} \quad (20)$$

This approach is applied in the sense of the desired boundary conditions for all time steps and can be adapted to combinations of Dirichlet, Neumann, Robin or periodic conditions. Equation (20) utilizes a modified source term  $\tilde{f}_k$  which inherits the additional sources given in Eqs. (16)–(19). For the example of a Dirichlet conditions on the left boundary, cf. Eq. (16), the source term modifies to  $j_{S_1} = -2\lambda_0 u_1 \tilde{f}_1$ , with  $\tilde{f}_1 = f_1 + 0.5Dv\lambda_0/\Delta x^2$ .

The following two sections outline the framework for evaluating Eq. (20) on a QC. Section 4 describes the VQA framework, which shares features with Ref. [26]. Subsequently, the quantum circuits used to calculate the different contributions ( $j_{L_k}$ ,  $j_{S_k}$ ,  $j_{P_k}$ ) to Eq. (20) on a QC are described in Section 5. The gate complexity of these circuits is optimized in Section 6 before they will be used to approximate the derivative of the objective function with respect to the parameters in Section 7, which in turn is required to advance the optimization.

Before we continue with the QC implementation of the different objective function contributions, we remark that the Neumann terms  $j_{N_1}$  and  $j_{N_{N_p}}$  could also be cast into potential term contributions derived

from the reciprocal of the diffusive time scale, i.e.,  $\bar{p}_1 = p_1 + \nu/(\zeta \Delta x^2)$  and  $\bar{p}_{N_p} = p_{N_p} + \nu/(\zeta \Delta x^2)$ .

#### 4. Quantum framework

As outlined above, a quantum register of  $n = \log_2(N_p)$  qubits is required to evaluate the objective function on a QC, cf. Eq. (10). Each qubit possesses two distinguishable computational states  $|0\rangle$  and  $|1\rangle$ , which form an orthonormal basis of a complex Hilbert space and are frequently identified with basis vectors  $|0\rangle = (1, 0)^T$  and  $|1\rangle = (0, 1)^T$  [36]. The  $n$  qubit register spans a Hilbert space whose basis is formed by tensor products of the computational basis states of individual qubits, e.g.,  $e_{i=0} = |0\rangle^{\otimes n} = |0\rangle \otimes |0\rangle \otimes \dots \otimes |0\rangle$ , where  $\otimes n$  indicates an  $n$ -fold tensor product. Thus, the  $i$ th element of the orthonormal basis can be expressed via the binary representation  $\text{bin}(i)$  as  $|\text{bin}(i)\rangle = e_i$ , following the *little-endian* convention where the least significant qubit corresponds to the right-most register position ( $q_1$ ) and the most significant qubit corresponds to the left-most register position ( $q_n$ ), i.e.,  $|q_n q_{n-1} \dots q_1\rangle$ . Furthermore, the least significant qubit ( $q_1$ ) is drawn as the uppermost qubit in the network figures and the lowermost qubit ( $q_n$ ) indicates the most significant. This encoding enables the representation of the discrete vector  $u$  by the amplitudes of the quantum register, viz.  $u = \sum_{k=1}^{N_p} u_k e_{k-1} = \sum_{k=1}^{N_p} u_k |\text{bin}(k-1)\rangle = |u\rangle$ , where the offset of the indices results from the technicality that the basis vectors  $e_k$  start at  $k=0$  but the interior unknowns  $u_k$  begin with  $k=1$ . Mind that, though complex  $u_k$  are generally allowed, we assume real solutions of Eq. (1), hence  $u_k^* = u_k$ , and that the time index is mostly suppressed in this Section to improve the readability.

##### 4.1. Variational quantum algorithm

The VQA methodology aims to minimize an objective function  $J^l$  given in Eq. (20), by optimizing the control parameters  $v^l = (\lambda_0^l, \lambda_c^l)^T$  for every time step  $l$ . The hybrid framework utilizes both classical and quantum hardware, as displayed in Fig. 2. The function  $J^l$  is evaluated on the quantum device, as illustrated by the shaded box in

Fig. 2, whereas the optimization process runs on the classical hardware that controls the process and iteratively adjusts the control  $v$ . For a particular set of ansatz parameters  $\lambda_c$ , the objective function is evaluated from dedicated, individual so-called Quantum-Nonlinear-Processing-Units (QNPU) for each group of terms in Eq. (20), i.e., for  $j_{L_k}, j_{DN}, j_N, j_{P_k}, j_{S_k}$ , cf. Section 5. An update  $\Delta v$  is determined by the optimizer as outlined in Section 7.1 before a new iteration starts.

Figure 3(a) describes two equivalent options of the general layout of circuits on the quantum device. Grossly speaking, both options consist of a measuring (aka. ancilla) qubit in the upper “wire” and calculation qubits in the lower “wires”. In general, the qubits are initialized to the base state  $|0\rangle$ . The calculation qubits are then introduced to an ansatz circuit called  $U$ , highlighted in gray in Fig. 3(a). The ansatz  $U$  can be denoted by a  $N_p \times N_p$  matrix composed of an embedded combination of two-qubit units, i.e., fundamental gates  $\tilde{U}(4 \times 4)$ , as illustrated in Fig. 3(b) for six qubits. The particular combination employed in this research refers to a scalable depth bricklayer topology and shares some similarities with machine learning topologies. The benefit of the bricklayer topology over alternative topologies has been communicated in Ref. [37].

The employed fundamental gate is displayed in dark gray in Figs. 3(b), 3(c). It has been suggested by Vatan and Williams [38] and is frequently labeled  $SO4$ . This gate operates in real and imaginary space, which is rather unusual for engineering applications that aim for real-valued results [22,23]. Its application promises a richer ansatz function space that, in turn, supports the use of shallower depth  $d$  in  $U$ . As displayed in Fig. 3(c), the fundamental gate consists of  $R_y, R_z$  and CNOT gates, where  $R_y$  and  $R_z$  are parameterized versions of classical Pauli  $\sigma_y$  and  $\sigma_z$  matrices [36], cf. Appendix A.1, and the  $(\dots)^\dagger$  symbol indicates the transposed of the complex conjugate.

Turning back to Fig. 3(a), the result of the ansatz circuit  $U(\lambda_c)$  is the parameterized state  $|u(\lambda_c)\rangle$ . Subsequently, a particular transformation matrix  $T_l$  or  $\tilde{T}_l$  is introduced in case of boundary contributions  $j_{DN}$  and  $j_N$  before the data enters a boundary-specific QNPU. The transformation essentially modifies the order of qubits in the register  $|u\rangle$  where necessary. As will be shown in Section 6, two implementations are conceivable, cf. Fig. 3(a), by either using highly complex multi-controlled gates or by employing additional carry qubits.

The prepared state  $(T_l|u\rangle)$  or  $(\tilde{T}_l|u\rangle)$  inputs to the main QNPU gate, indicated in light gray in Fig. 3(a), through the input port (IP) and outputs the specific objective function contributions for a group of terms ( $j_{L_k}, j_{DN}, j_N, j_{P_k}, j_{S_k}$ ) through an output port (OP). Finally, the register is reversed to its initial order after applying the boundary-specific QNPU. To this end, the Hadamard test method inherently reverses the gates in the upper variant of Fig. 3(a). In contrast, the implementation example at the bottom of Fig. 3(a) requires the explicit application of  $\tilde{T}_l^\dagger$ . Note that for periodic conditions, where neither  $j_{DN}$  nor  $j_N$  contributions occur, no transformation is needed and  $T_l, \tilde{T}_l, \tilde{T}_l^\dagger$  simply degenerates to the identity matrix  $I$  in line with the upper part of Fig. 3(a). The use of the Hadamard test [39] is motivated by previous studies [27], which indicated quantum advantage with this approach. Here, rather than assessing the individual contributions at each grid point  $x_k$  [39], the integral over all interior points is directly compiled, for example  $\sum_k j_{L_k}$ , and measured statistically on the upper “wire” by the expectation value denoted by  $\langle \sigma_z \rangle$  [36]. This is indicated by the upper “wire” of Fig. 3(a) that connects to the calculation “wires” through a control port (CP).

Before we discuss the individual QNPUs in Section 5, the measurement strategy implemented here should be described for the sake of completeness. We rely on the density (or autocorrelation) matrix  $\rho_{\text{ancilla}}$  of the ancilla qubit to approximate the expectation  $\langle \sigma_z \rangle$  at the measurement gauge in Fig. 3(a). To eliminate stochastic influences, evaluating QC-calculated results is usually based on evaluating sufficiently large samples. However, if the QC is only emulated, as in the current study, the procedure can be simplified using the trace-out method [36].

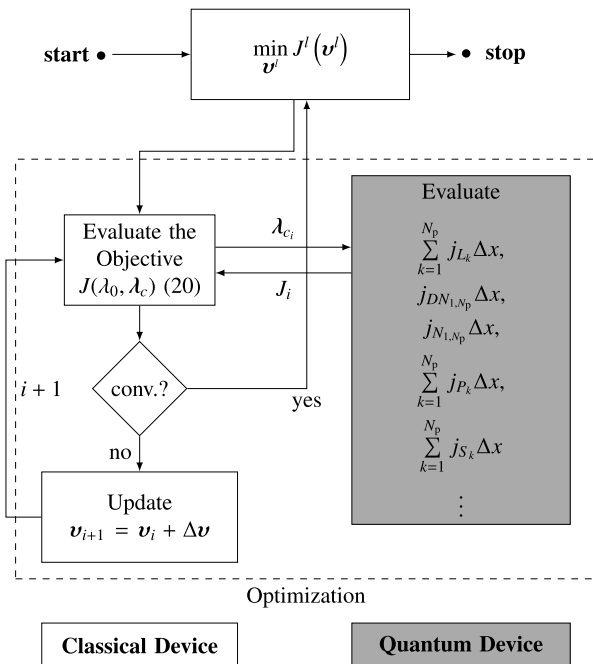
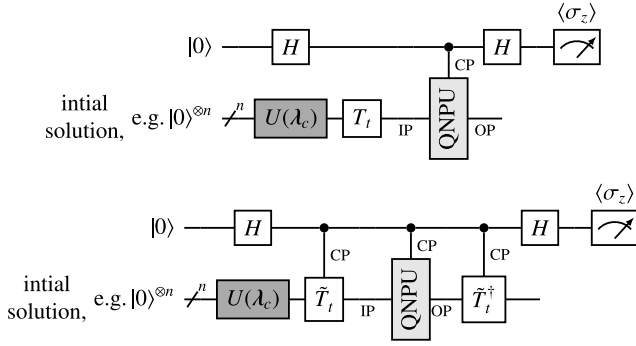
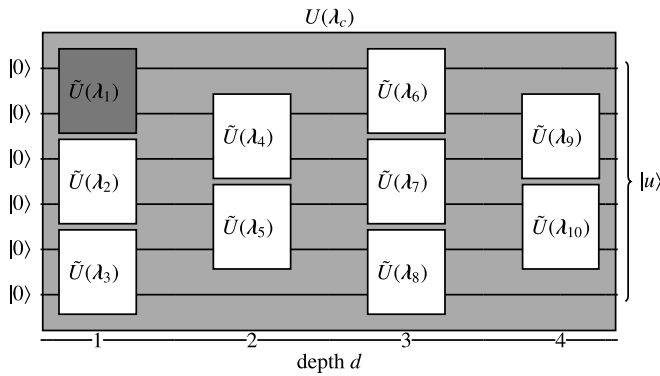


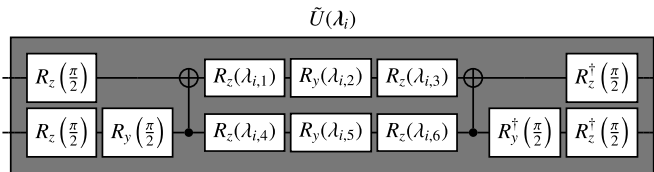
Fig. 2. Sketch of the hybrid VQA optimization (time dependencies in the optimization step are suppressed for the sake of clarity).



(a) Sketch of two VQ circuits composed of the ansatz  $U(\lambda_c)$ , the  $T_t$  or  $\tilde{T}_t(\tilde{T}_t^\dagger)$  transformation, a specific QNPU, and a measurement qubit (ancilla qubit) for indirect measurements.  $H$  indicates a Hadamard gate, CP denotes a control port, IP marks the input port & OP the output port, and  $\langle \sigma_z \rangle$  provides the expectation value of the measurement operation, cf. [26].



(b) Example of an ansatz circuit  $U(\lambda_c)$  composed from  $a = 10$  fundamental gates  $\tilde{U}(\lambda_i)$  featuring a bricklayer topology, parameterized by  $\lambda_c = (\lambda_1, \lambda_2, \dots, \lambda_a)^T$  parameters to encode a test function  $|u\rangle$  on  $n = 6$  qubits, cf. [26].



(c) Two qubit SO4 ansatz  $\tilde{U}(\lambda_i)$  to create an optimal variational state [38] based on  $\lambda_i = (\lambda_{i,1}, \lambda_{i,2}, \dots, \lambda_{i,6})^T$  individual parameters. In the first layer ( $d = 1$ ) the ansatz has only three parameters in  $\lambda_i$ , the rotation gates corresponding to  $\lambda_{i,4}$ ,  $\lambda_{i,5}$ , and  $\lambda_{i,6}$  are identity gates.

Fig. 3. Illustration of the employed nested VQA circuits.

First, the whole system state  $|\psi\rangle$  (ancilla and calculation qubits) is computed by a state-vector simulation. As we are only interested in the information contained in the ancilla portion of the state  $|\psi\rangle$ , the remainder is traced out, and the density operator  $\rho_{\text{ancilla}} = \text{Tr}_{\text{QNPU}}\{|\psi\rangle\langle\psi|\}$  of the ancilla remains. The state-vector simulation thus yields the same result as the measurement of the ancilla performed on real hardware, without noise and sampling errors [36].

### 5. Quantum-Nonlinear-Processing-Units

For the introduction of our library of Quantum-Nonlinear-Processing-Units (QNPU), we use a complex number notation. All QNPU are highlighted in light gray in the respective figures, which links them to the VQ circuits depicted in Fig. 3(a). They are adjustable

to an arbitrary equidistant one-dimensional discretization and have been designed to keep the gate sequence shallow. Quantum gates are often associated with matrices or matrix operations. Therefore, the interested reader is referred to Appendix A.2 for a brief introduction to converting gate sequences into matrices.

#### 5.1. Periodic Laplace operator

The Laplacian is given by the FD stencil in Eq. (11) and partly resembles the classical bit shift operator [36,40] due to shifted products, e.g.,  $u_k^* u_{k+1}^l, u_k^* u_{k-1}^l$ . In contrast to other approaches [22,41] the QNPU circuit employed herein is based on the adder circuit with a polynomial scaling of the gate complexity, cf. Section 6. Using  $n = 2$  qubits, a half-adder circuit can be used, which consists of one CNOT gate and an additional Toffoli gate. For  $n > 2$ , additional  $(n - 2)$  carry qubits (auxiliary qubits) are required, which results in a full-adder circuit, cf. [26]. Assuming a periodic state, it can be shown [26] that the general structure of the discrete Laplace operator is represented by

$$\sum_{k=1}^{N_p} u_k |\text{bin}(k-1)\rangle \rightarrow \sum_{k=1}^{N_p} u_{k+1} |\text{bin}(k-1)\rangle \quad \text{with} \quad 1 = \sum_{k=1}^{N_p} u_k^* u_k. \quad (21)$$

Mind that an adder QNPU to compute a periodic Laplace operator has previously been reported by Lubasch et al. [26]. Figure 4 depicts an example of the employed Laplace QNPU circuit for  $n = 6$  qubits and  $n - 2 = 4$  carry bits. The circuit involves  $n - 2 = 4$  CNOT gates and  $2n - 2 = 10$  Toffoli gates.

#### 5.2. External contributions (potential and source terms)

The computation of external contributions follows from two subsequent steps. First, the representation of the discrete linear source  $\tilde{f}_k^l$  and potential terms  $p_k^l$  are assembled. This enables computing their contributions to the objective function  $(j_S, j_P)$  in a second step.

For simple, analytically described terms, e.g.,  $\exp(x)$  or  $\sin(x)$ , Lubasch et al. [26] introduced corresponding gate sequences that allow such functions to be represented on a QC. Other suggestions intensively deploy Matrix-Product-State (MPS) techniques, [42–44]. In our case, however, we first solve a prior optimization problem to represent arbitrary external contributions by quantum gates, taking the ansatz  $U(\lambda)$  into account, viz.

$$\bar{\lambda}_c = \min_{\lambda_c} (1 - \lambda_0 \langle u(\lambda_c) | g \rangle) \quad \text{with} \quad \frac{1}{\lambda_0^2} = \sum_{k=1}^{N_p} g_k^* g_k. \quad (22)$$

In Eq. (22)  $g = \tilde{f}$  or  $g = p$  for the source and the potential term, respectively. The benefit of this approach is that it re-uses existing implementations and also keeps the depth of the resulting circuits

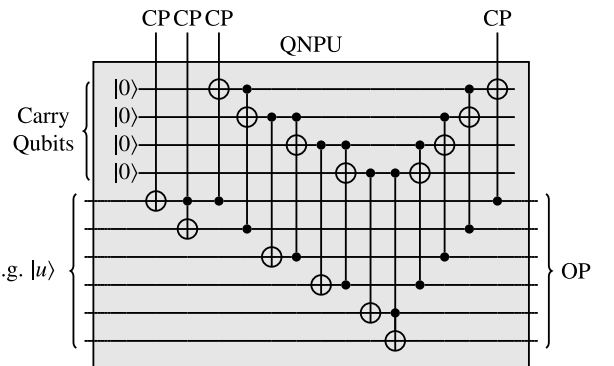
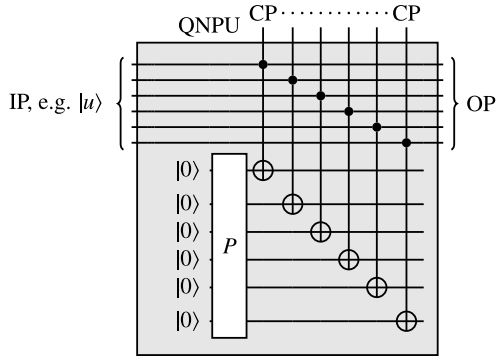
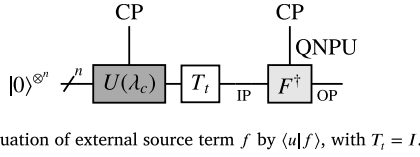


Fig. 4. Exemplary quantum circuit of the discrete Laplacian in the Hadamard test configuration. The example refers to  $n = 6$  qubits and  $n - 2 = 4$  carry bits. It requires  $n - 2 = 4$  CNOT and  $2n - 2 = 10$  Toffoli gates. For  $n = 2$  qubits, this circuit is exactly implementing a half-adder, cf. [26].



(a) Exemplary quantum circuit for a discrete potential function  $p = (p_1, p_2, p_3, \dots, p_{N_p})^T$  represented by the gate “P”.



(b) Evaluation of external source term  $f$  by  $\langle u|f\rangle$ , with  $T_t = I$ .

Fig. 5. External contributions on the Hadamard test. The displayed example refers to  $N_p = 64$ ,  $n = 6$  qubits.

shallow. The optimal parameters  $\bar{\lambda}_c$  determined in this way result in the required gates  $U(\bar{\lambda}_c) = P$  and  $U(\bar{\lambda}_c) = F$  to be used inside the QNPU that compute the potential and the source term contributions to  $J$  in Eq. (20).

The contribution of the potential term  $p(x_k, t^l)$  to  $J$  follows from  $\sum_{k=1}^{N_p} [\lambda_0^2 u_k^* p_k u_k]^l$ . The corresponding QNPU circuit is assembled by Toffoli gates according to Fig. 5(a). Since the product  $u_k^* p_k u_k$  is non-unitary, additional  $n$  ancilla qubits occur in the circuit [45]. Similarly, the representation of  $\langle \tilde{f} | = U(\bar{\lambda}_c) | 0 \rangle$  is implemented into the source term QNPU (light gray) given in Fig. 5(b). Note that thanks to the versatility of the  $SO4$  ansatz, the minimum depth  $d = 1$  is usually sufficient to reconstruct  $\tilde{f}$  and  $p$ . Instead of the uncontrolled ansatz  $U(\lambda)$  displayed in Fig. 3(a), the linearity of the source term requires that  $U(\lambda)$  is also controlled.

### 5.3. Boundary conditions

With the intention of developing a quantum equivalent of the ghost point approach discussed in Section 3, we first elaborate on a matrix-based representation to facilitate the understanding of the quantum implementation. Subsequently, the novel QNPU for the boundary treatment in variational quantum simulations are introduced.

The implementation of boundary conditions is based on the correction of the periodic contributions in the derivative matrix. To outline the transfer of Eq. (20) into quantum-inspired or true quantum computations, we restrict ourselves to Dirichlet conditions imposed for a simple Laplace equation  $-\nu \Delta y = 0$  in the two-qubit case. Next, let us recover a slightly reduced version of Eq. (20) for this problem in a matrix-based approach, assuming  $\mathbf{u} \in \mathbb{C}^{N_p}$  and  $A \in \mathbb{R}^{N_p \times N_p}$ . For the sake of clarity, we also define  $(\nu \lambda_0^2 / 4x) = 1$ , viz.

$$\mathbf{u}^\dagger \cdot A \cdot \mathbf{u} = \begin{pmatrix} u_1^* & u_2^* & u_3^* & u_4^* \end{pmatrix} \cdot \begin{pmatrix} 2 & -1 & 0 & -1 \\ -1 & 2 & -1 & 0 \\ 0 & -1 & 2 & -1 \\ -1 & 0 & -1 & 2 \end{pmatrix} \cdot \begin{pmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \end{pmatrix} \quad (23)$$

$$= 2u_1^*u_1 - u_1^*u_2 - u_2^*u_1 + 2u_2^*u_2 - u_2^*u_3 - u_3^*u_2 + 2u_3^*u_3 - u_3^*u_4 - u_4^*u_3 + 2u_4^*u_4 - \underbrace{u_1^*u_4 - u_4^*u_1}_{\text{periodic boundary}}$$

$$T_t = \begin{pmatrix} 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix}$$

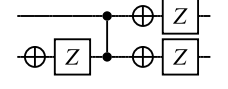
(a) Matrix representation of  $T_t$ .

$$B = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 \\ 0 & 0 & 1 & 0 \end{pmatrix}$$

(b) Matrix representation of  $B$ .



(c) Gate representation of  $T_t$  [22].



(d) Gate representation of  $B$ .

Fig. 6. Matrices and their corresponding circuits for two-qubits.

The sum on the right-hand side of Eq. (23) distinguishes between regular interior contributions and periodic boundary contributions. In line with Eq. (20), the periodic contributions of  $-u_1^*u_4 - u_4^*u_1$  need to be eliminated by another term in the cost function, viz.  $J = \sum_{k=1}^4 j_{L_k} + j_{DN_1} + j_{DN_4}$ . This is achieved by

$$J_{DN} = j_{DN_1} + j_{DN_4} = \underbrace{u_1^*u_4 + u_4^*u_1}_{\text{boundary}} = \mathbf{u}^\dagger \cdot D \cdot \mathbf{u}$$

$$= \begin{pmatrix} u_1^* & u_2^* & u_3^* & u_4^* \end{pmatrix} \cdot \begin{pmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{pmatrix} \cdot \begin{pmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \end{pmatrix} \quad (24)$$

The matrix  $D \in \mathbb{R}^{N_p \times N_p}$  needs to be unitary for its representation by quantum gates. This distinguishes the quantum-inspired approaches from true QC methods. A QC-compatible representation of  $D$  follows from a modification with the basis of its null space removing the zero eigenvalues and intuitively matching the boundary contributions of Eq. (24). Accordingly, the unitary matrix  $C \in \mathbb{R}^{N_p \times N_p}$  is proposed as

$$J_{DN} = \mathbf{u}^\dagger \cdot C \cdot \mathbf{u} = \begin{pmatrix} u_1^* & u_2^* & u_3^* & u_4^* \end{pmatrix} \cdot \begin{pmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & -1 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{pmatrix} \cdot \begin{pmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \end{pmatrix} \quad (25)$$

$$= \underbrace{u_1^*u_4 - u_2^*u_3 + u_3^*u_2 + u_4^*u_1}_{\text{if } \mathbf{u} \in \mathbb{R}^n} = \underbrace{2u_1u_4}_{\text{boundary}}$$

The unitary matrix  $C$  only inheres nonzero entries on the anti-diagonal. The upper and lower corners must have the same sign to neutralize the periodic contributions of matrix  $A$ . The other terms need to change sign with respect to the main diagonal for these contributions to be canceled when the expectation value  $\langle \sigma_z \rangle$  is measured. Note that for the summation of the interior anti-diagonal elements, one obtains zero only if  $\mathbf{u} \in \mathbb{R}^n$ .

The remaining task is to design the circuits that implement the transformation described by the matrix  $C$ . To this end, the heuristic approach requires another decomposition, i.e.,  $C = T_t^{-1} \cdot B \cdot T_t$ , where  $T_t$  relates to a permutation operator and  $B$  to a boundary QNPU, which both directly translate into quantum gates, cf. Fig. 6.

In general, deriving the gate sequence corresponding to a given matrix is neither trivial nor unique. The application of heuristic approaches leads to the gate-level implementation for two qubits outlined in Figs. 6(c) and 6(d). In contrast to other methods [22,23], the proposed boundary corrections are represented as unitary circuits and, therefore, as quantum gates. Further insight into the conversion between matrix and circuit representation is given in Appendix A.2. For the sake of completeness, the four qubit version of the circuits in Figs. 6(c) and 6(d) with the corresponding unitary matrix  $C$  are also included in Appendix A.3.

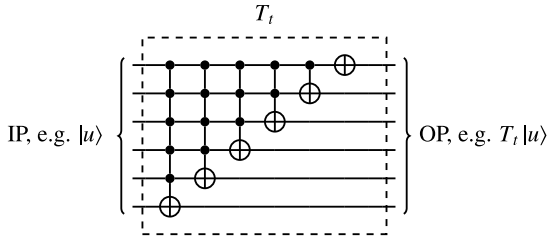


Fig. 7. Six qubit example of the permutation operator  $T_i$  represented by a sequence of multi-controlled Toffoli gates, cf. [22].

The extension to an  $n$  qubit framework implies the multi-qubit implementation of the transformation  $T_i$  with a sequence of  $n - 2$  multi-controlled NOT gates, cf. Fig. 7. Similarly, the boundary corrections  $j_{DN}$  and  $j_N$  consist of  $2n - 1$  CNOTs and controlled-Z gates for an arbitrary number of qubits. Furthermore, one  $n$  multi-qubit controlled Z gate for both circuits and one  $n$  multi-controlled NOT gate for  $j_{DN}$  complete the circuits, as illustrated in Fig. 8 for  $n = 6$  qubits. Both circuits shown in Figs. 8(a) and 8(b) use operations that only manipulate the dynamics at the boundaries. In the interior domain, the effect of the operator neutralizes for the expectation value  $\langle \sigma_z \rangle$ .

The resulting circuits for  $T_i$ ,  $j_{DN}$ , and  $j_N$  in Figs. 7 and 8 inhere sequences of multi-controlled gates, which substantially deteriorate the gate complexity. In this regard, the most efficient implementation depends on the available QC hardware and its support for multi-controlled gates. Alternatively, additional carry qubits can be used to obtain more shallow circuits. This technique is illustrated in more detail in Section 6.

### 6. Gate complexity

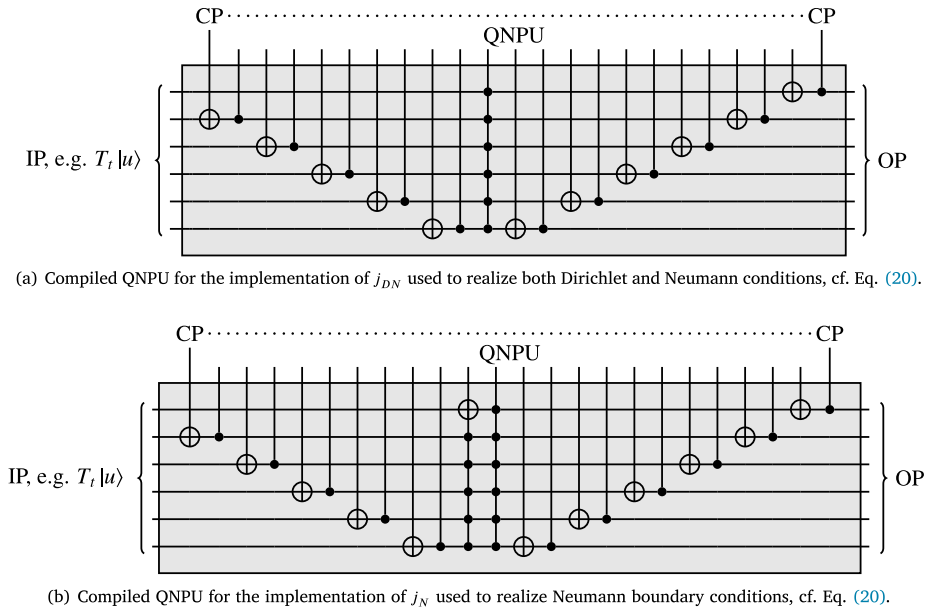
To scrutinize the scalability of the algorithm, the number of single- ( $R_x, R_y, U, \dots$ ) and two-qubit (CNOT,  $\dots$ ) gates per qubit are outlined in this Section. For this purpose, multi-qubit gates are decomposed using known identities, which are available, for example, in QISKIT [29]. The scaling of the number of gates involved in the  $SO4$  ansatz with respect to the number of qubits  $n$  and the depth  $d$  is reported in Fig. 9(a). It is observed that the number of gates scales linearly with  $n$  and  $d$ , i.e.,  $\mathcal{O}(nd)$ . The graphs displayed in Fig. 9(b) illustrate the polynomial relation between the number of gates and the number of qubits for the

adder ( $j_L$ ), the source ( $j_S$ ), the potential ( $j_P$ ) and the boundary circuits — the latter using either multi-controlled gates ( $j_{DN}, j_N$ ) described in Section 5.3 or the corresponding modifications described below ( $\tilde{j}_{DN}, \tilde{j}_N$ ). Most QNPUs in Fig. 9(b) depict a favorable complexity. The polylog complexity of the adder circuit ( $\mathcal{O}(\log(n)^r) : r \in \mathbb{R}$ ) is due to the very local reach of the FD stencil for approximating the second derivative. In this case, the resulting circuit conserves the local property. An analysis of the source and potential QNPUs also reveals a linear scaling of  $\mathcal{O}(n)$ , which underlines the importance of an efficient state initialization for arbitrary generic functions [15,21,46–48], cf. Section 5.2. The similarity to the scaling behavior examined for the  $SO4$  ansatz is obvious and indicates the strong influence of the prior optimization of  $U(\lambda_c)$  to determine  $\tilde{f}$  or  $p$ , cf. Eq. (22). The related overhead in the number of gates with respect to the ansatz is due to the control operation of the ansatz in both QNPUs ( $j_S, j_P$ ) and the additional CNOT gates following the  $P$  gate for the potential QNPU, see Fig. 5(a). Note that for the values given in Fig. 9, the depth of the ansatz for  $j_P$  and  $j_S$  is assigned to  $d = 1$ .

The boundary QNPUs ( $j_{DN}, j_N$ ), however, scale polynomially, on the maximum order of  $\mathcal{O}(n^5)$ . This was also observed by other researchers when using multi-controlled quantum gates, for example, [22–24]. The reason for that is the non-locality of the implementation and the multi-controlled feature, which yields interaction with more than just the nearest neighbor qubits. In practical applications, e.g., an implementation on NISQ hardware, there is a demand for circuit efficiency. To obtain more shallow circuits, additional carry qubits are used. They implement intermediate information associated with AND operations by a staggered series of Toffoli gates. The technique is illustrated for a multi-controlled G gate (MCG) in Fig. 10.

The example consists of  $n = 6$  qubits and requires  $n - 2 = 4$  carry qubits to transform the deep multi-controlled gate on the left of Fig. 10 into  $n + 2$  Toffoli- and one controlled G gate on the right of Fig. 10. Mind that after applying G, all carry bits need to be reversed to their initial state, i.e.,  $|0\rangle$ .

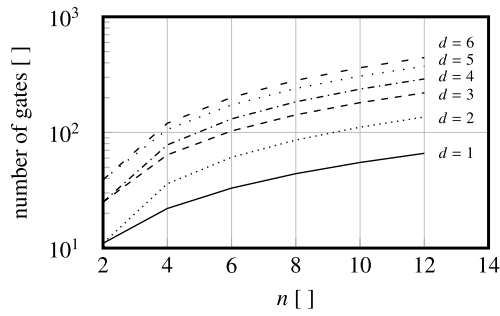
Applying this concept to the transformation circuit  $T_i$  outlined in Fig. 7 allows to derive a shallower alternative  $\tilde{T}_i$  resulting in a permutation operator, cf. Fig. 11(a). This Figure also provides the corresponding circuit for the inverted transformation  $\tilde{T}_i^\dagger$  in Fig. 11(b). The latter is required for measuring the expectation value  $\langle \sigma_z \rangle = \langle 0|U^\dagger \tilde{T}_i^\dagger [\text{QNPU}] T_i U|0\rangle$  as illustrated by the Hadamard-test based structure in Fig. 3(a). Both  $\tilde{T}_i$  and  $\tilde{T}_i^\dagger$  are controlled by an ancilla qubit,



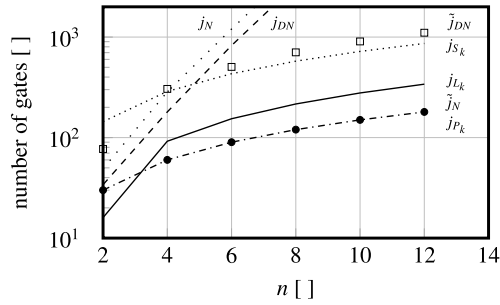
(a) Compiled QNPU for the implementation of  $j_{DN}$  used to realize both Dirichlet and Neumann conditions, cf. Eq. (20).

(b) Compiled QNPU for the implementation of  $j_N$  used to realize Neumann boundary conditions, cf. Eq. (20).

Fig. 8. Illustration of the novel boundary correction QNPUs for a  $n = 6$  qubits setup.



(a) Complexity for the  $SO4$  ansatz.



(b) Complexity for all QNPU.

Fig. 9. Gate complexity of the employed QNPU and ansatz gate compilations.

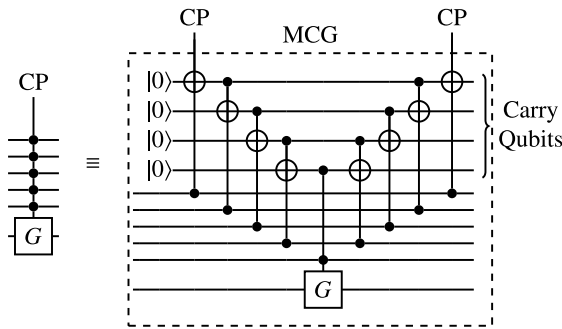


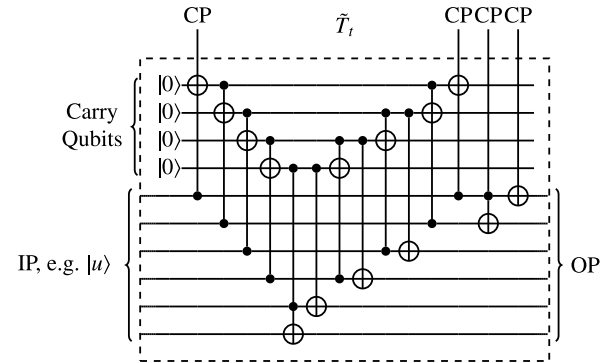
Fig. 10. Equivalent networks using a deep qubit-saving implementation (left) and a shallow circuit depth approach (right) to realize a multi-controlled gate  $G$ .

which follows the same strategy as depicted for the QNPU assembly in Fig. 3(a).

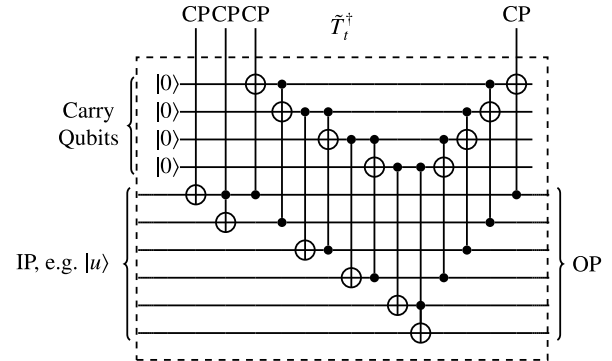
Introducing the above-mentioned shallow circuit alternatives to the  $j_{DN}$  circuit is straightforward and allows replacing the multi-controlled  $Z$  gate by the usage of  $n - 2$  carry qubits. As already outlined at the end of Section 3, the QNPU related to the Neumann condition ( $j_N$ ) shown in Fig. 8(b) can be cast in terms of the potential QNPU, depicted in Fig. 5(a). A consistent application of shallow circuit (MCG) strategies yields the complexity graphs for  $\tilde{j}_N, \tilde{j}_{DN}$  marked by symbols in Fig. 9. Advantageous differences to the initial implementation  $j_N, j_{DN}$ , given by lines in Fig. 9, are clearly visible but, of course, come at the expense of additional  $(n-2)$  carry qubits. The graph displays an efficient scaling below  $\mathcal{O}(n^2)$ . In terms of grid-point behavior, the complexity is of *polylog* scaling. We again emphasize that the chosen reduction in complexity, i.e., the number of carry qubits used, depends on the capabilities of the QC hardware.

## 7. Optimization strategy

A common denominator of all VQAs is the efficiency of the optimization process. As stated in Ref. [49], this is non-trivial since the



(a) Forward transformation  $\tilde{T}_t$ .



(b) Backward (adjoint) transformation  $\tilde{T}_t^\dagger$ .

Fig. 11. Alternative implementations of the forward ( $T_t, \tilde{T}_t$ ) and backward ( $T_t^\dagger, \tilde{T}_t^\dagger$ ) transformation introduced in Fig. 7.

problem covered by VQA is theoretically NP-hard, meaning that the optimization is often characterized as non-convex and features multiple local minima and/or vanishing derivatives [23,45,50]. To update the controls ( $\lambda_0, \lambda_c$ ) along the route outlined in Fig. 2, we apply the sequence of a zero- and a first-order method. The combination aims at a larger sample size to address non-convex functions and higher accuracy in the vicinity of the minima and might be replaced by a more sophisticated strategy. The optimization starts with the zero-order approach and employs an evolutionary *Particle-Swarm Optimization* (PSO) algorithm [51] using  $p$  particles. In the case of non-periodic BC, the PSO step typically involves  $\mathcal{O}(1000)$  iterations, where around  $p \approx 100n$  particles are employed. The best candidate initializes the control  $v$  for the subsequent first-order classical *Gradient Descent* (GD) method. The required derivative of the objective function with respect to the controls  $\nabla_v J(v_i)$  is obtained from the QC and follows the expressions given below, cf. Eq. (28) and Eq. (31). Convergence is assessed from the absolute change of the derivative

$$\text{ch}(\nabla J_i^l) = \|\nabla J_i^l - \nabla J_{i-1}^l\|_{l_2} \leq 10^{-m}, \quad (26)$$

where  $m \in [3, 7]$  depends on the number of employed qubits, cf. Section 8. The solution is advanced in time as described in Fig. 1 and in the Algorithm 1. For time steps  $l > 1$ , the optimal parameters of the previous step  $v^{l-1}$  serve as an initial guess for the PSO to reduce the efforts, as proposed by [22,23].

The whole framework is implemented by an in-house toolbox, with QISKIT v.0.42.1 [29] as a quantum backend.

**Algorithm 1** VQ Algorithm

---

**Require:**  $N_t, \tilde{f}, \text{maxit}_{\text{PSO}}, \text{tol}_{\text{PSO}}, \text{maxit}_{\text{GD}}, \text{tol}_{\text{GD}}$

$l \leftarrow 1$

**while**  $l \leq N_t$  **do** ▷ Time Evolution

**if**  $l = 1$  **then**

$\tilde{f}^1 \leftarrow \frac{y(x,0,D_0)}{\Delta t} + \tilde{f}^1$  ▷ Initial setup of the external contribution

$v^1 \leftarrow (\lambda_0^1, \lambda_c^1)^\top$  ▷ Random initialization of  $\lambda_0^1 \in [0, 1]$  and  $\lambda_c^1 \in [0, 4\pi]^c$  for all  $p$  particles

**else**

$\tilde{f}^l \leftarrow \frac{y(x,l,D_{l-1})}{\Delta t} + \tilde{f}^l$  ▷ Adjust the contributions

$v^l \leftarrow v^{l-1}$  ▷ Propagate previous final control  $v^{l-1}$

**end if**

$v^l \leftarrow \text{PSO}(v^l)$  ▷ Iterate until  $\text{maxit}_{\text{PSO}}$  or  $\text{tol}_{\text{PSO}} > \text{ch}(\nabla J_l^l)$   
(cf. Fig. 2)

$v^l \leftarrow \text{GD}(v^l)$  ▷ Iterate until  $\text{maxit}_{\text{GD}}$  or  $\text{tol}_{\text{GD}} > \text{ch}(\nabla J_l^l)$   
(cf. Fig. 2)

$l \leftarrow l + 1$

**end while**

---

**7.1. Derivative computation**

The determination of the descent direction is supported by the QC. To this end, the quantum circuits derived in Sections 4 and 5 are reused, as outlined below. The gradient of the objective function with respect to the scalar parameter  $\lambda_0$  follows from Eq. (20) and simply reads

$$\begin{aligned} \nabla_{\lambda_0} J(\lambda_0, \lambda_c) &= \sum_{k=1}^{N_p} - \left( \frac{2v\lambda_0}{\Delta x^2} \right) u_k(\lambda_c) \underbrace{\left( u_{k+1}(\lambda_c) - 2u_k(\lambda_c) + u_{k-1}(\lambda_c) \right)}_{\nabla_{\lambda_0} j_{L_k}} \Delta x \\ &\quad + \underbrace{\left( \frac{2v\lambda_0}{\Delta x^2} \right) \left( u_1(\lambda_c) u_{N_p}(\lambda_c) + u_1(\lambda_c) u_{N_p}(\lambda_c) \right)}_{\nabla_{\lambda_0} (j_{DN_1} + j_{DN_{N_p}})} \Delta x \\ &\quad - \underbrace{\left( \frac{2v\lambda_0}{\Delta x^2} \right) \left( u_1(\lambda_c)^2 + u_{N_p}(\lambda_c)^2 \right)}_{\nabla_{\lambda_0} (j_{N_1} + j_{N_{N_p}})} \Delta x \\ &\quad + \sum_{k=1}^{N_p} \underbrace{-2\zeta \lambda_0 u_k(\lambda_c) p_k u_k(\lambda_c)}_{\nabla_{\lambda_0} j_{P_k}} \Delta x + \sum_{k=1}^{N_p} \underbrace{-2u_k(\lambda_c) \tilde{f}_k}_{\nabla_{\lambda_0} j_{S_k}} \Delta x. \end{aligned} \quad (27)$$

Again, the temporal index has been omitted for better clarity. Comparing Eq. (20) with Eq. (27) reveals

$$\begin{aligned} \nabla_{\lambda_0} J &= \frac{1}{\lambda_0} \left[ \sum_{k=1}^{N_p} (2j_{L_k} + 2j_{P_k} + j_{S_k}) \Delta x \right] \\ &\quad + \frac{2}{\lambda_0} \left[ \left( j_{DN_1} + j_{DN_{N_p}} + j_{N_1} + j_{N_{N_p}} \right) \Delta x \right], \end{aligned} \quad (28)$$

which can be evaluated on the QC. The derivative of  $J$  with respect to  $\lambda_c$  can also be computed on the QC without deriving a new gate sequence. The derivative of the objective function with respect to the control vector  $\lambda_c$  follows from

$$\begin{aligned} \nabla_{\lambda_c} J(\lambda_0, \lambda_c) &= \sum_{k=1}^{N_p} \nabla_{\lambda_c} (j_{L_k} + j_{P_k} + j_{S_k}) \Delta x \\ &\quad + (\nabla_{\lambda_c} (j_{DN_1} + j_{DN_{N_p}}) + \nabla_{\lambda_c} (j_{N_1} + j_{N_{N_p}})) \Delta x. \end{aligned} \quad (29)$$

For the quadratic terms  $j_L$ ,  $j_P$ ,  $j_{DN}$ , and  $j_N$ , an exact representation of the derivative is given by the *Parameter Shift Rule*, cf. [52–54]. The derivative follows using only two additional evaluations of existing quantum circuits, and the nonlinearity is implicitly considered.

The linear source term  $\nabla_{\lambda_c} j_{S_k}$  is approximated with second-order accurate central differences that operate on the complete (linear) objective function contribution [55], viz.

$$\begin{aligned} \nabla_{\lambda_c} j_{S_k} &= -2\lambda_0 (\nabla_{\lambda_c} u_k) \tilde{f}_k \\ &= \frac{j_{S_k}(\lambda_0, \lambda_c + h e_c) - j_{S_k}(\lambda_0, \lambda_c - h e_c)}{2h} + \mathcal{O}(h^2). \end{aligned} \quad (30)$$

Mind that Eq. (30) has to be executed for all entries of the parameter vector  $\lambda_c$ , and the step size is typically assigned to a constant value  $h$ , depending on the number of control parameters. Applying these techniques yields an expression for the derivative of the objective function  $J$  with respect to the control vector  $\lambda_c$  that utilizes the existing circuits, viz.

$$\begin{aligned} \nabla_{\lambda_c} J(\lambda_0, \lambda_c) &= \sum_{k=1}^{N_p} \frac{1}{2} \left( j_{L_k}(\lambda_0, \lambda_c + \frac{\pi}{2} e_c) - j_{L_k}(\lambda_0, \lambda_c - \frac{\pi}{2} e_c) \right) \Delta x \\ &\quad + \frac{1}{2} \left( j_{DN}(\lambda_0, \lambda_c + \frac{\pi}{2} e_c) - j_{DN}(\lambda_0, \lambda_c - \frac{\pi}{2} e_c) \right) \Delta x \\ &\quad + \frac{1}{2} \left( j_N(\lambda_0, \lambda_c + \frac{\pi}{2} e_c) - j_N(\lambda_0, \lambda_c - \frac{\pi}{2} e_c) \right) \Delta x \\ &\quad + \sum_{k=1}^{N_p} \left( \frac{1}{2} \left( j_{P_k}(\lambda_0, \lambda_c + \frac{\pi}{2} e_c) - j_{P_k}(\lambda_0, \lambda_c - \frac{\pi}{2} e_c) \right) \Delta x \right. \\ &\quad \left. + \sum_{k=1}^{N_p} \left( \nabla_{\lambda_c} j_{S_k} \right) \Delta x \right). \end{aligned} \quad (31)$$

**8. Numerical results**

This Section describes the application of the VQA to two prototypical CFD problems, i.e., the steady heat conduction (Section 8.1) and the transient heat conduction (Section 8.2). Results are discussed for a range of Dirichlet, Neumann, and Robin boundary-condition combinations. The validation data is obtained from FD-solutions of the PDE problem. It employs the same discretization as the VQA to approximate the Laplace operator and the temporal derivative, cf. Eqs. (2) and (11), respectively.

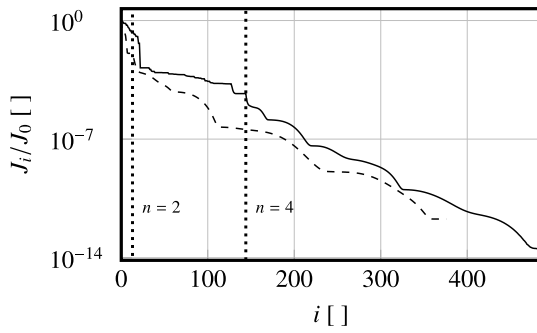
The predictive deviation of the VQA-solutions  $y^{\text{VQ}}$  from FD-solutions  $y^{\text{FD}}$  is assessed using either the  $l_2$ -norm  $\epsilon_{l_2}$  or the frequently employed trace distance  $\epsilon_{\text{tr}}$ , cf. [20,22,41], viz.

$$\epsilon_{l_2} = \sqrt{\sum_k (y_k^{\text{FD}} - y_k^{\text{VQ}})^2}, \quad \epsilon_{\text{tr}} = \sqrt{\left( 1 - \left| \sum_k \frac{y_k^{\text{FD}}}{\|y^{\text{FD}}\|_{l_2}} \frac{y_k^{\text{VQ}}}{\|y^{\text{VQ}}\|_{l_2}} \right|^2 \right)}. \quad (32)$$

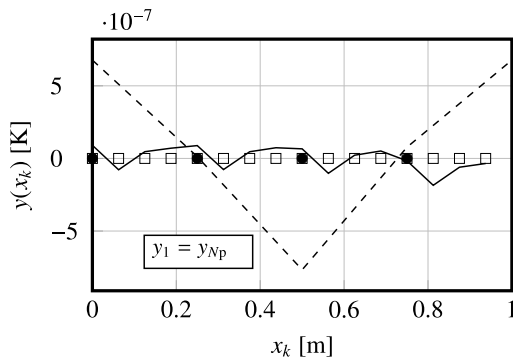
Time-averaged measures are employed to assess the transient heat conduction and indicated by  $\bar{\epsilon}_{l_2}$ ,  $\bar{\epsilon}_{\text{tr}}$ . The number of control variables  $c$  of  $\lambda_c$  is also stated in the results tables. A full dataset of the performed experiments is available via [56].

**8.1. Steady heat conduction (Poisson equation)**

The first case refers to the steady heat conduction, where the time derivative in Eq. (1) vanishes, and  $y$  [K] marks the temperature. The physical potential  $p(x)$  [1/s] vanishes, and the diffusivity is assigned to a unit value  $v = 1$  [m<sup>2</sup>/s]. In the case of pure Neumann boundary conditions, the source reads  $f(x) = 1$  for  $x \leq 0.5$  and  $f(x) = -1$  for  $x > 0.5$ . In all other cases, the unit source responds to  $f(x) \equiv 1$  [K/s]. Mind that singular solutions, e.g., caused by pure Neumann conditions, are fixed by defining a central zero crossing. The application of the VQA framework to this Poisson problem is presented for the following seven scenarios: in-/homogeneous Dirichlet, in-/homogeneous Neumann, mixed Neumann/Dirichlet conditions, Robin conditions, and periodic conditions. First and second-order derivative approximations are applied to Neumann, mixed Neumann/Dirichlet, and Robin boundaries. The displayed results are obtained for a coarse grid, featuring  $N_p = 4$  interior points ( $n = 2$  qubits; dashed and dash-dotted lines),



(a) Evolution of the normalized objective function over the optimization count  $i$ . Dotted vertical lines indicate the transition from global (PSO) to local (GD) optimization.



(b) Comparisons of VQA- and FD-results, ( $n = 2$ , closed circles;  $n = 4$ , open squares).

Fig. 12. Verification results of the VQA application for a periodic steady heat conduction problem using  $N_p = 4$  ( $n = 2$ ,  $d = 1$ ; dashed lines) and  $N_p = 16$  ( $n = 4$ ,  $d = 1$ ; solid lines) interior points.

and a finer grid, featuring  $N_p = 16$  interior points ( $n = 4$  qubits; dotted and solid lines). The FD-results are depicted by closed ( $N_p = 4$ ) and open ( $N_p = 16$ ) symbols.

The periodic case serves as a verification reference and it is displayed in Fig. 12. VQA-results were obtained from shallow circuits  $U(\lambda_c)$  using a depth of  $d = 1$ , cf. Fig. 3(b). Figure 12(a) shows the convergence of the objective function with the dotted vertical lines indicating the transition from global (PSO) to local (GD) optimization (cf. Alg. 1). The VQA-results are compared with FD-solutions in Fig. 12(b) and indicate an excellent agreement with the expected solution  $y_k \equiv 0$ . The VQA-result improves as the grid is refined. The latter is also indicated by a reduction of  $\epsilon_{l_2}$  from an already satisfactory value of  $10^{-6}$  to  $10^{-7}$ , cf. Tab. 1. The reduction goes hand in hand with lower amplitude but higher frequency wiggles displayed in Fig. 12(b), which are reflected by an augmented trace distance  $\epsilon_{tr}$  and should be viewed with caution due to the vanishing mean value for this case.

As indicated in Fig. 13, the VQA predictions perfectly agree with the FD reference for all Dirichlet, Neumann, Robin, or mixed boundary conditions and grid resolutions. Furthermore, the first-order accurate approximation of Neumann conditions is obviously improved by using second-order techniques in both the VQA and the FD-results, cf. for example Fig. 13(c). Note that, the increase in the order of accuracy does not imply higher computational costs, since the quantum circuits described in Section 5 can be employed without additional modifications and only the source term  $\tilde{f}$  is adjusted. It is noteworthy to remark that using second-order Neumann boundary conditions does not induce a substantial change in the Robin boundary condition experiment, which is confirmed by the FD results. The required depth  $d$  increases for non-periodic conditions on the finer grid. To this end, a depth  $d = 3$  is

sufficient to mimic all investigated boundary conditions in conjunction with the finer grid.

Table 1 quantifies the deviation of the VQA-results from the FD-results for all cases depicted in Fig. 13. The Table displays a deviation of  $\epsilon_{l_2} \approx 10^{-6}$ – $10^{-7}$  for two qubits ( $N_p = 4$ ), disregarding the imposed boundary conditions. The good agreement of  $\epsilon_{l_2}$  for the periodic case, where no specific boundary condition circuits are required, and the non-periodic cases reveal that no substantial sensitivity of the attainable accuracy is observed when applying the current strategy for a quantum-based implementation of boundary conditions. For the finer discretization associated with  $n = 4$  qubits, Tab. 1 shows higher values of  $\epsilon_{l_2} \approx 10^{-3}$  for the non-periodic boundary conditions and thereby a less satisfactory agreement with the periodic arrangement. The latter is attributed to difficulties of finding the global optimum in non-convex optimization and highlights a bottleneck of the VQA in combination with versatile ansatz functions  $U(\lambda_c)$ , cf. Fig. 3. Moreover, it also outlines that sufficient accuracy of the ancilla measurements is crucial for detecting the (global) minimum.

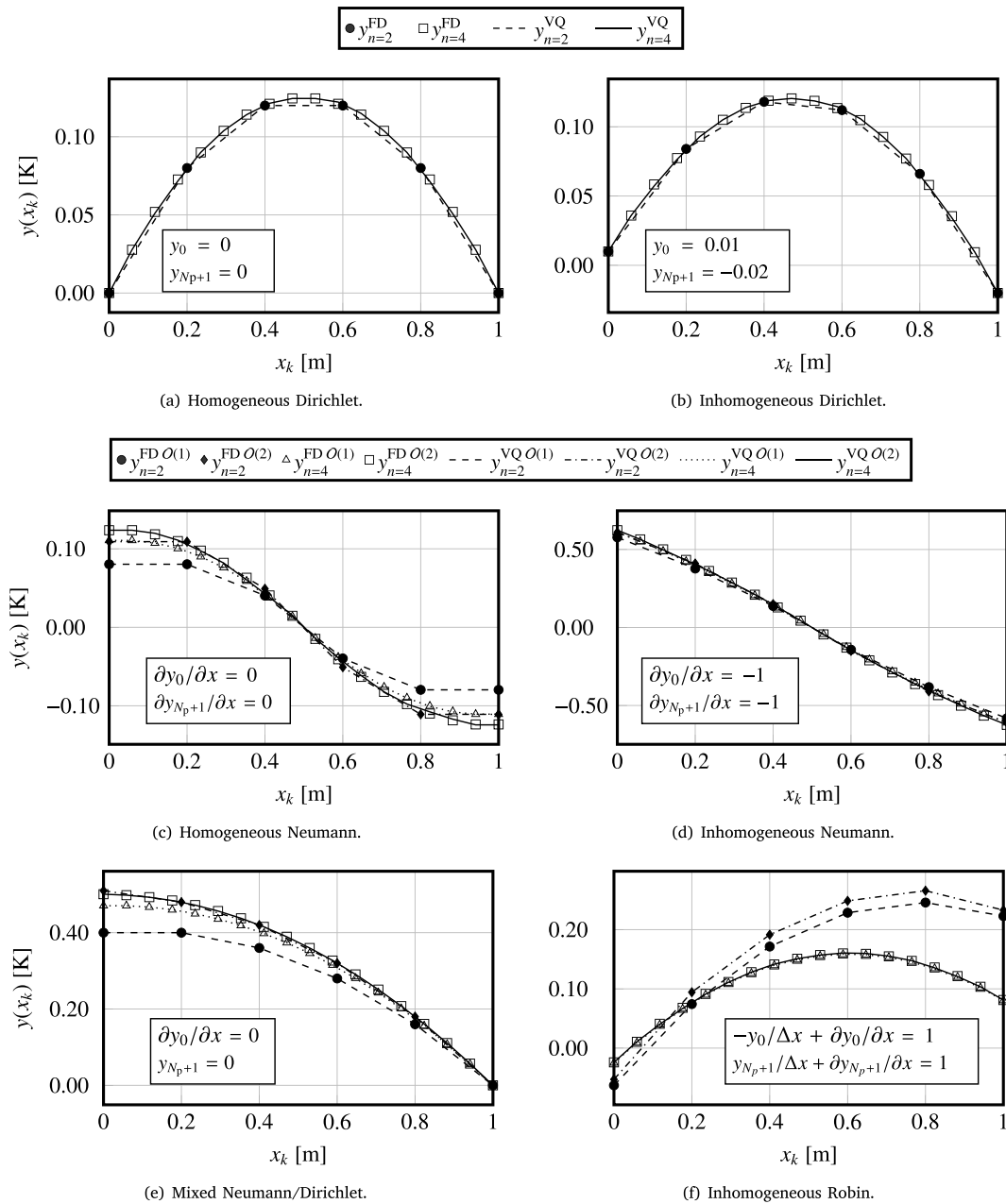
The trace distance  $\epsilon_{tr}$  is given in the third column of Tab. 1 and can be interpreted as point-wise correlation between the quantum- and the FD-result. The trace distance is related to the convergence to the global minimum, while the  $\epsilon_{l_2}$  measures the expressibility of the ansatz. Since the two measures address different aspects, considering just one of them seems insufficient. Moreover, the trace distance sometimes displays numerical zero values, cf. Tab. 1, which does not necessarily guarantee a perfect agreement and also advocates assessing both measures. In addition to the error measures used to assess the VQA-results, the origin of the observed differences in predictive accuracy is also of importance. Interestingly, the results reported by Sato et al. [22] – for example – show a much better trace distance compared to the present results for periodic boundaries, although the predictive agreement of the current results appears to be slightly better. A key difference between the current and similar recent efforts, e.g., in Ref. [22], refers to the employed  $SO4$  ansatz illustrated in Fig. 3(c), which seems to feature an improved ansatz expressibility but is also harder to train. Furthermore, the number of control variables  $c$  given in Tab. 1 roughly scales as  $\mathcal{O}(n d)$ , which suggests room for future improvements in the design of the quantum ansatz circuits towards more problem-specific strategies [57].

## 8.2. Transient heat conduction (Heat equation)

The second case deals with the transient heat conduction where the time derivative in Eq. (3) is treated as discussed in Section 2. The contributions of the potential term  $p(x, t)$  [1/s] as well as the physical source term  $f(x)$  [K/s] in Eq. (1) are neglected. The diffusivity is again set to a unit value of  $\nu = 1$  [m<sup>2</sup>/s]. Four different boundary condition settings are investigated using a constant time step of  $\Delta t \nu / \Delta x^2 = 1 / (2^{|n-3|} (N_p - n))$ . The discretizations are the same two equidistant grids used in Section 8.1, i.e., a coarse ( $N_p = 4$ ) and a fine grid ( $N_p = 16$ ). All cases studied should reach a steady state, and due to the different convergence speeds to the steady state, different time step numbers are shown for different boundary and initial conditions. The convergence of the optimizer is controlled by Eq. (26) using a threshold level of  $10^{-7}$  for two qubits and  $10^{-4}$  for four qubits, respectively.

The predicted temperature distributions are depicted in Fig. 14, where arrows indicate the temporal evolution from the initial to the final time step, and FD-results are again illustrated by closed (coarse grid) and open (fine grid) symbols. Mind that a larger scatter of depth  $d$ , cf. Fig. 3(b), is seen for the employed ansatz  $U(\lambda_c)$  which is motivated by the complexity of the initial solution as explained below.

Figure 14(a) shows the results obtained for the homogeneous Dirichlet problem over the first 30 time instants. Since the results for the two grids are very similar, we mostly confine the discussion to coarse grid results for the sake of brevity. The initial solution was assigned

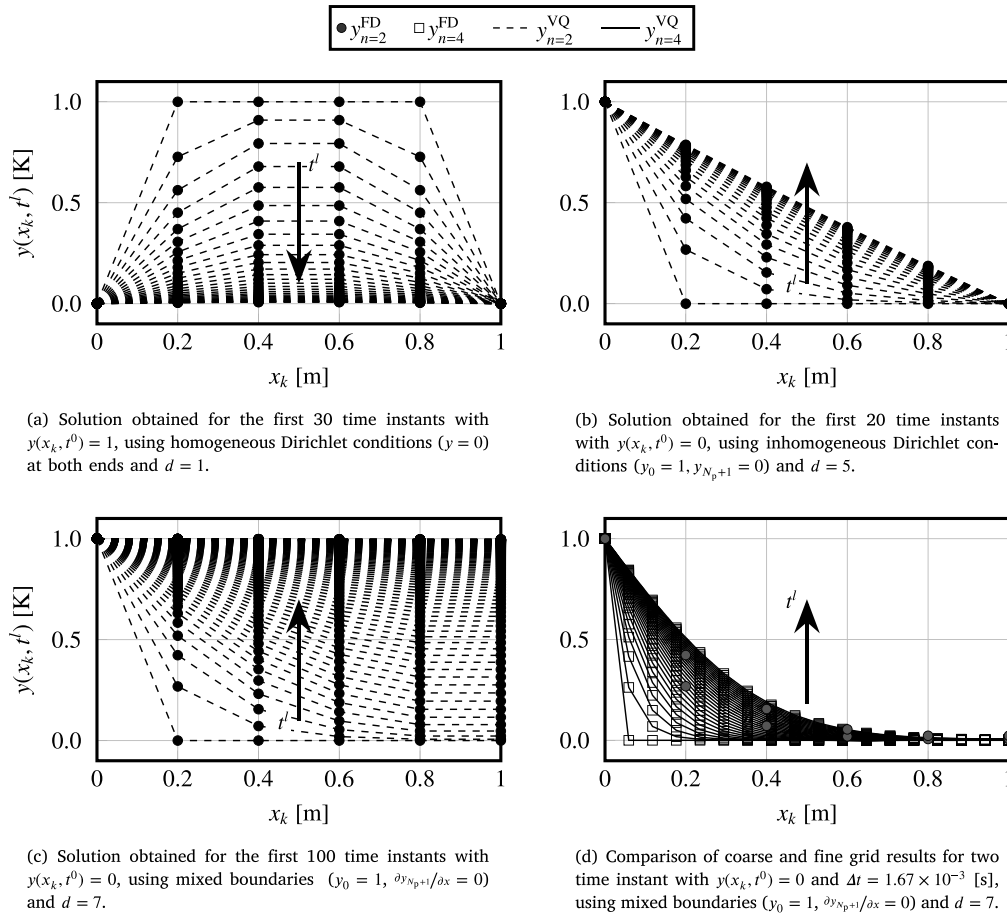


**Fig. 13.** Comparisons of VQA-result (lines) and FD-results (symbols) for the steady heat conduction in combination with different boundary conditions on a grid with  $N_p = 4$  ( $n = 2$ ; dashed or dash-dotted lines & closed symbols) and  $N_p = 16$  ( $n = 4$ ; dotted or solid lines & open symbols) interior points. The depth of the ansatz circuit  $U(\lambda_c)$  is  $d = 1$  for  $n = 2$  qubits and  $d = 3$  for  $n = 4$  qubits, cf. Fig. 3(b).

**Table 1**

Deviations of VQA and FD-results in the  $l_2$ -norm ( $\epsilon_{l_2}$ ) supplemented by their trace distance ( $\epsilon_{tr}$ ) for results depicted in Fig. 13. The degrees of freedom (DOF) are indicated by the number of control variables  $c$ .

Boundary setting	$l_2$ -norm $\epsilon_{l_2}$		Trace distance $\epsilon_{tr}$		DOF $c$		
	Interior points $N_p$	4	16	4	16	4	16
Periodic		$1.23 \times 10^{-6}$	$3.26 \times 10^{-7}$	$9.70 \times 10^{-1}$	$1.0 \times 10^0$	3	6
Homog. Dirichlet		$1.5 \times 10^{-7}$	$2.54 \times 10^{-3}$	0.00	$6.76 \times 10^{-3}$	3	24
Homog. Neumann $\mathcal{O}(1)$		$1.37 \times 10^{-8}$	$1.82 \times 10^{-3}$	$6.32 \times 10^{-8}$	$5.05 \times 10^{-3}$	3	24
Homog. Neumann $\mathcal{O}(2)$		$5.03 \times 10^{-9}$	$5.53 \times 10^{-3}$	$1.49 \times 10^{-8}$	$1.36 \times 10^{-2}$	3	24
Inhomog. Dirichlet		$8.49 \times 10^{-8}$	$2.83 \times 10^{-3}$	$4.31 \times 10^{-7}$	$7.87 \times 10^{-3}$	3	24
Inhomog. Neumann $\mathcal{O}(1)$		$1.68 \times 10^{-6}$	$4.22 \times 10^{-3}$	$1.65 \times 10^{-6}$	$2.55 \times 10^{-3}$	3	24
Inhomog. Neumann $\mathcal{O}(2)$		$3.52 \times 10^{-8}$	$5.28 \times 10^{-3}$	$2.58 \times 10^{-8}$	$2.78 \times 10^{-3}$	3	24
Mixed Neumann/Dirichlet $\mathcal{O}(1)$		$6.01 \times 10^{-8}$	$8.38 \times 10^{-4}$	0.00	$5.66 \times 10^{-4}$	3	24
Mixed Neumann/Dirichlet $\mathcal{O}(2)$		$1.26 \times 10^{-8}$	$4.44 \times 10^{-4}$	0.00	$2.87 \times 10^{-4}$	3	24
Inhomog. Robin $\mathcal{O}(1)$		$3.32 \times 10^{-9}$	$1.86 \times 10^{-3}$	0.00	$3.7 \times 10^{-3}$	3	24
Inhomog. Robin $\mathcal{O}(2)$		$2.99 \times 10^{-8}$	$2.21 \times 10^{-3}$	$5.16 \times 10^{-8}$	$4.34 \times 10^{-3}$	3	24



**Fig. 14.** Comparisons of VQA-result (lines) and FD-results (symbols) for the transient heat conduction in combination with four different boundary conditions on grids with  $N_p = 4$  ( $n = 2$ ; dashed lines) and  $N_p = 16$  ( $n = 4$ ; solid lines) interior points.

to  $y(x_k, t^0) = 1$  in the interior, which yields negative heat fluxes at the boundaries in combination with the imposed homogeneous Dirichlet boundaries. The Figure indicates an excellent agreement between the VQA-results and the FD-results. This also applies to the results obtained for the inhomogeneous Dirichlet problem depicted in Fig. 14(b), where the initial solution refers to  $y(x_k, t^0) = 0$  in the interior domain, and a constant heat flux from the left to the right is predicted at the twentieth time step. An interesting detail refers to the increased depth  $d$  of the ansatz circuit  $U(\lambda_c)$ , cf. Fig. 3(b). The increase is explained by the different initial conditions and the related steep solution in the initial phase that requires a deeper ansatz circuit, even when using the fairly rich SO4 approach for  $\tilde{U}(\lambda_i)$ . On the other hand, deeper circuits are afflicted with more parameters to be optimized, and the optimization effort could be reduced for lowered depth circuits as the solution advances to the simpler steady state.

Results obtained for a mixed Dirichlet/Neumann problem are outlined in Figs. 14(c) and 14(d). The initial temperature distribution is assigned to  $y(x_k, t^0) = 0$  in the interior as well as to the Neumann condition on the right side. The final solution should agree with the left side Dirichlet value  $y(0, t^l) = 1$  in the whole domain. For the coarse grid,  $N_t = 100$  time instants are displayed in Fig. 14(c). They indicate a remarkable visual agreement between the VQA-results and the FD-results over the entire simulation period. In line with the FD-results, the VQA-results unveil that the temperature on the right increases for the adiabatic condition until the global heat exchange finally vanishes. Fig. 14(d) compares the evolution of the solution on the coarse and the fine grid for the initial two time instants. Results indicate that the predictive accuracy of the VQA is similar on both grids.

The time evolution of the  $\epsilon_{l_2}$  measure is displayed, using a *log-log* representation in Fig. 15(a) and a *semi-log* representation in Fig. 15(b).

For all cases, the measures remain at a low level. For the homogeneous Dirichlet case with  $d = 1$ , displayed in Fig. 14(a), the error increases significantly after approximately 10 time steps but remains small, i.e., under  $10^{-5}$ . In contrast, the ansatz with  $d = 7$  that was used to compute the inhomogeneous steady state solution depicted in Fig. 14(c) outperforms the trainability in the homogeneous Dirichlet setting ( $d = 1$ ). Despite a general temporal error transport, the implicit numerical scheme remains stable and does not indicate any amplification of the error, which is in the range of the error of the steady-state solution, cf. Section 8.1. As indicated by Fig. 15(a), the maximum values agree with the results of the steady heat conduction case provided by Tab. 1. The corresponding Tab. 2 lists the time-averaged values of the  $l_2$ -norm  $\bar{\epsilon}_{l_2}$  and the trace distance measure  $\bar{\epsilon}_{tr}$ . Since the changes of the solution over time are small, maximum values of  $\bar{\epsilon}_{l_2}$  and  $\bar{\epsilon}_{tr}$  are bounded, and the averaged values remain within the range of the steady state results.

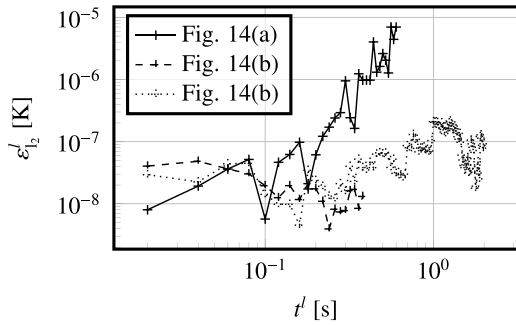
## 9. Conclusion

The study presents a flexible and general boundary condition treatment for the numerical solution of PDEs on a QC. The strategy is based on the VQA approach by Lubasch et al. [26]. To this end, initial-boundary value problems are reformulated as optimal control problems where all terms, including the boundary conditions, are cast into case-independent QNPU. The boundary treatment combines ghost points with boundary cost function contributions that complement the objective function of the control problem. The method relaxes the requirements for the ansatz functions  $U(\lambda_c)$ , which must only be valid for the interior points that remain decoupled from the boundary points

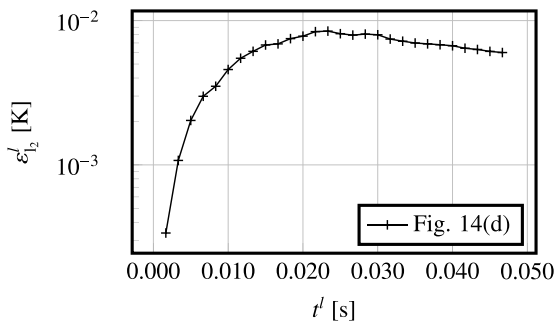
**Table 2**

Time-averaged deviation measures  $\bar{\epsilon}_l$  and  $\bar{\epsilon}_{tr}$  for the results depicted in Fig. 14. The degrees of freedom (DOF) are indicated by the number of control variables  $c$ .

Boundary setting	$l_2$ -norm $\bar{\epsilon}_l$		Trace distance $\bar{\epsilon}_{tr}$		DOF $c$		
	Interior points $N_p$	4	16	4	16	4	16
Homog. Dirichlet		$6.70 \times 10^{-6}$	-	$4.79 \times 10^{-5}$	-	3	-
Inhomog. Dirichlet		$1.84 \times 10^{-8}$	-	$2.77 \times 10^{-7}$	-	15	-
Mixed Dirichlet/Neumann		$8.08 \times 10^{-8}$	$6.10 \times 10^{-3}$	$2.12 \times 10^{-8}$	$4.07 \times 10^{-3}$	21	60



(a) Evolution for coarse-grid results displayed in Fig. 14(a)-14(c).



(b) Evolution of fine-grid results displayed in Fig. 14(d).

**Fig. 15.** Temporal evolution of the deviation measure  $\epsilon_l^t$  for the results displayed in Fig. 14.

and avoids the use of penalty terms. Verification studies are performed for steady and unsteady heat conduction problems, covering a variety of boundary conditions.

The obtained VQA-results reveal an excellent agreement with classical FD-results. The increase of the circuit depth could attenuate the optimization efficiency, while multi-controlled gates limit the implementation on QC hardware [18,22,45]. Analogously to the Laplacian circuits, improvements in this regard are achieved by replacing the multi-controlled gates with additional carry qubits. The remarkable flexibility and accuracy of the proposed approach are thereby complemented by a favorable scaling of the quantum circuits with the number of qubits.

As for all VQA approaches, the choice of the ansatz is crucial since it defines the underlying function space for the PDE solution. At the same time, the approach can introduce a non-convex property into the optimization. The hardware-efficient bricklayer topology of the employed ansatz shows favorable expressibility in terms of small deviations from FD solutions in noise-less QC simulations. The richer expressibility of the  $SO4$  ansatz in combination with shallow depths  $d$  is advantageous, for example, to express general source terms. However, the strategy also unveils unfavorable with regard to the case-dependent increase of control parameters  $\lambda_c$  and depth  $d$ , and the related poorer trainability. The combination of challenging trainability and the non-convex nature of the optimization can imply serious convergence issues in tracking the global optimum for larger numbers of qubits.

Future research will be devoted to encoding first derivatives as a QNPU operator to represent convective fluxes [58]. In view of extending the present algorithm to 2D and 3D spatial problems, the non-convex optimization problem will be addressed by selecting more appropriate (trainable, expressible, shallow) dynamic ansatz strategies. For instance, strategies for avoiding barren plateaus in the cost function [59] include utilizing classical shadows [60] and convolutional neural networks [61]. For the time evolution, the use of adaptive ansatz depths and circuit re-compilation techniques [62] seem promising ways to save computational cost and increase the range of applications towards industrial QCFD in the current NISQ era.

### CRedit authorship contribution statement

**Paul Over:** Conceptualization, Writing – original draft, Writing – review & editing, Data curation, Formal analysis, Methodology, Software, Investigation, Validation, Visualization. **Sergio Bengoechea:** Conceptualization, Conceptualization, Writing – original draft, Writing – review & editing, Data curation, Formal analysis, Methodology, Software, Investigation, Validation, Visualization. **Thomas Rung:** Funding acquisition, Writing – original draft, Writing – review & editing, Conceptualization, Methodology, Project administration, Resources, Supervision. **Francesco Clerici:** Writing – review & editing. **Leonardo Scandurra:** Writing – review & editing. **Eugene de Villiers:** Supervision, Project administration, Resources, Funding acquisition. **Dieter Jaksch:** Project administration, Resources, Supervision, Writing – review & editing, Funding acquisition.

### Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

### Acknowledgments

This publication and the current work have received funding from the European Union’s Horizon Europe research and innovation program (HORIZON-CL4-2021-DIGITAL-EMERGING-02-10) under grant agreement No. 101080085 (QCFD). The authors thank N.-L. van Hülst, G. S. Reese, E. L. Fesefeldt, D. Schmeckpeper, P. M. Müller, and M. Kiffner for fruitful discussions.

### Appendix

In this Appendix, the fundamental quantum gates and the gate sequence of the quantum half-adder are summarized together with their corresponding matrix representation. In addition, the matrix gate conversion for the four qubit version of the circuits in Fig. 6 is also given.

### A.1. Fundamentals

The Dirac notation defines  $\psi = \psi^i e_i = |\psi\rangle$  (called “ket psi”) and  $\psi^\dagger = \psi_j e^j = \langle\psi|$  (called “bra psi”). Quantum circuits are accordingly assembled by inner (dot/scalar product) and outer products (tensor/dyadic product) using the computational basis of a qubit, viz.  $|0\rangle = (1, 0)^\top$  and  $|1\rangle = (0, 1)^\top$ . The inner product contracts two identical coordinates, e.g., in a scalar  $z = \phi^\dagger \psi = \langle\phi|\psi\rangle = \sum_k \phi^k \psi_k$ . The outer product instead conserves the sum of the ranks (orders) of the multiplied factors, e.g., in the matrix  $\phi \otimes \psi^\dagger = |\phi\rangle \otimes \langle\psi| = |\phi\rangle \langle\psi| \rightarrow \sum_k \sum_l \phi^k \psi_l e_k \otimes e^l$ . The application of these basic rules allows to derive various operations, e.g., the fundamental set of one-qubit operations

$$\sigma_0 = I = |0\rangle\langle 0| + |1\rangle\langle 1| = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \tag{A.1}$$

$$\sigma_x = X = |0\rangle\langle 1| + |1\rangle\langle 0| = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}, \tag{A.2}$$

$$\sigma_y = Y = i(|1\rangle\langle 0| - |0\rangle\langle 1|) = \begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix}, \tag{A.3}$$

$$\sigma_z = Z = |0\rangle\langle 0| - |1\rangle\langle 1| = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}, \tag{A.4}$$

$$R_y(\theta) = \exp\left(i\frac{-\theta}{2}\sigma_y\right) = \begin{bmatrix} \cos\left(\frac{\theta}{2}\right) & -\sin\left(\frac{\theta}{2}\right) \\ \sin\left(\frac{\theta}{2}\right) & \cos\left(\frac{\theta}{2}\right) \end{bmatrix}, \tag{A.5}$$

$$R_z(\theta) = \exp\left(i\frac{-\theta}{2}\sigma_z\right) = \begin{bmatrix} e^{-i\theta/2} & 0 \\ 0 & e^{i\theta/2} \end{bmatrix} \tag{A.6}$$

and the two-qubit operations (A.7)

$$\text{CNOT} = I \otimes |0\rangle\langle 0| + \sigma_x \otimes |1\rangle\langle 1| = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}, \tag{A.8}$$

$$\text{CZ} = I \otimes |0\rangle\langle 0| + \sigma_z \otimes |1\rangle\langle 1| = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & -1 \end{bmatrix}. \tag{A.9}$$

Mind that the shape of the CNOT matrix results from the *little-endian* convention.

### A.2. Matrix gate conversion

We present two approaches to derive the corresponding matrix representation of the two-qubit quantum half-adder in Fig. A.16. The first technique, referred to as the “formal way”, details each intermediate state  $|\phi_r\rangle; r \in \{0, 1, 2\}$  of the circuit while the second scans all the possible outputs to construct the matrix representation of the network by inspection. The latter is restricted to real-valued states and, we referred to as the “fast way”. The intermediate break-points, indicated by red dashed lines, serve as orientation in the following computations.

#### The formal way

$$|\phi_0\rangle = |\psi_2\rangle \otimes |\psi_1\rangle = |\psi_2\psi_1\rangle \tag{A.10}$$

$$\begin{aligned} |\phi_1\rangle &= \text{CNOT}|\psi_2\psi_1\rangle = (I \otimes |0\rangle\langle 0| + \sigma_x \otimes |1\rangle\langle 1|)|\psi_2\psi_1\rangle \\ &= (I \otimes |0\rangle\langle 0|)|\psi_2\psi_1\rangle + (\sigma_x \otimes |1\rangle\langle 1|)|\psi_2\psi_1\rangle \\ &= I|\psi_2\rangle \otimes |0\rangle\langle 0| + \sigma_x|\psi_2\rangle \otimes |1\rangle\langle 1| \end{aligned} \tag{A.11}$$

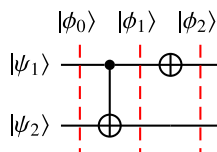


Fig. A.16. Quantum half-adder circuit.

$$\begin{aligned} |\phi_2\rangle &= I|\psi_2\rangle \otimes (\sigma_x|0\rangle\langle 0| + |1\rangle\langle 1|)|\psi_1\rangle \\ &= (I \otimes \sigma_x|0\rangle\langle 0|)|\psi_2\psi_1\rangle + (\sigma_x \otimes |1\rangle\langle 1|)|\psi_2\psi_1\rangle \\ &= \underbrace{(I \otimes \sigma_x|0\rangle\langle 0| + \sigma_x \otimes |1\rangle\langle 1|)}_A |\psi_2\psi_1\rangle \end{aligned} \tag{A.12}$$

$$\begin{aligned} A &= \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \otimes \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} + \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix} + \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \otimes \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix} \\ &= \begin{bmatrix} 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \end{aligned} \tag{A.13}$$

$$|\phi_2\rangle = A|\psi_2\psi_1\rangle = A|\phi_0\rangle. \tag{A.14}$$

**The fast way** — see from Eq. (A.15) to Eq. (A.18) below.

$$\begin{aligned} |\phi_0\rangle &= |\psi_2\psi_1\rangle = |00\rangle = (1, 0, 0, 0)^\top \\ |\phi_1\rangle &= |00\rangle \end{aligned} \tag{A.15}$$

$$|\phi_2\rangle = |01\rangle \quad \underline{|00\rangle \rightarrow |01\rangle},$$

$$\begin{aligned} |\phi_0\rangle &= |\psi_2\psi_1\rangle = |01\rangle = (0, 1, 0, 0)^\top \\ |\phi_1\rangle &= |11\rangle \end{aligned} \tag{A.16}$$

$$|\phi_2\rangle = |10\rangle \quad \underline{|01\rangle \rightarrow |10\rangle},$$

$$\begin{aligned} |\phi_0\rangle &= |\psi_2\psi_1\rangle = |10\rangle = (0, 0, 1, 0)^\top \\ |\phi_1\rangle &= |10\rangle \end{aligned} \tag{A.17}$$

$$|\phi_2\rangle = |11\rangle \quad \underline{|10\rangle \rightarrow |11\rangle},$$

$$\begin{aligned} |\phi_0\rangle &= |\psi_2\psi_1\rangle = |11\rangle = (0, 0, 0, 1)^\top \\ |\phi_1\rangle &= |01\rangle \end{aligned} \tag{A.18}$$

$$|\phi_2\rangle = |00\rangle \quad \underline{|11\rangle \rightarrow |00\rangle}.$$

The underlined statements deliver the information to reconstruct the rows of matrix A. By inspection, it is straightforward to extract the matrix representation in Eq. (A.12).

### A.3. Boundary circuit: Matrix and gate representation for four-qubits

The boundary circuits for  $n = 4$ , and the corresponding transformation matrix C are depicted in Fig. A.17. In the case of more grid points (or qubits), the different negative signed terms in the interior of C do not fall only on the antidiagonal, cf. Fig. A.17(d). Consequently, at least two matrix configurations work, and thus, the design of circuits in Figs. A.17(a) and A.17(b) is not unique with respect to the number of qubits. However, there are three important properties to be respected. First, the matrix entries are required to be symmetric with respect to the antidiagonal and the main diagonal. Second, the interior entries need to have a different sign with respect to their main diagonal-mirrored counterparts in order to cancel when taking the expectation value  $\langle\sigma_z\rangle$ . Last and most importantly, the corners of the antidiagonal have to be of the same sign to match the periodic entries of the derivative matrix.

### Data availability

The data is available via <https://doi.org/10.25592/uhhfdm.14123> [56].

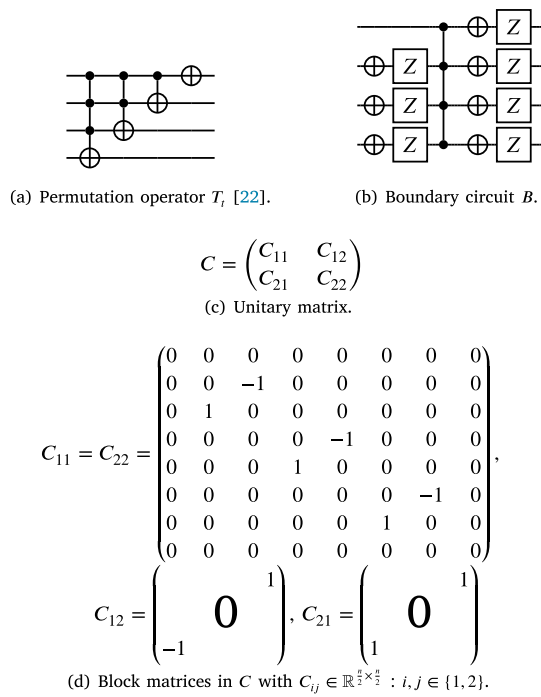


Fig. A.17. Boundary circuits and matrix  $C$  for four qubits.

References

[1] Burg D, Ausubel JH. Moore's Law revisited through Intel chip density. PLOS ONE 2021;16(8). [Online]. Available: <https://doi.org/10.1371/journal.pone.0256245>.

[2] Khan HN, Hounshell DA, Fuchs ERH. Science and research policy at the end of Moore's Law. Nat Electron 2018;1(1):14–21. [Online]. Available: <https://doi.org/10.1038/s41928-017-0005-9>.

[3] Suau A, Staffelbach G, Calandra H. Practical quantum computing: Solving the wave equation using a quantum approach. ACM Trans Quantum Comput 2021;2(1). [Online]. Available: <https://doi.org/10.1145/3430030>.

[4] Harrow AW, Hassidim A, Lloyd S. Quantum algorithm for linear systems of equations. Phys Rev Lett 2009;103(15). [Online]. Available: <https://doi.org/10.1103/PhysRevLett.103.150502>.

[5] Montanaro A, Pallister S. Quantum algorithms and the finite element method. Phys Rev A 2016;93(3). [Online]. Available: <https://doi.org/10.1103/PhysRevA.93.032324>.

[6] Berry DW. High-order quantum algorithm for solving linear differential equations. J Phys A 2014;47(10). [Online]. Available: <https://doi.org/10.1088/1751-8113/47/10/105301>.

[7] Steijl R, Barakos GN. Parallel evaluation of quantum algorithms for computational fluid dynamics. Comput & Fluids 2018;173. [Online]. Available: <https://doi.org/10.1016/j.compfluid.2018.03.080>.

[8] Brearley P, Laizet S. Quantum algorithm for solving the advection equation using Hamiltonian simulation. Phys Rev A 2024;110:012430. [Online]. Available: <https://doi.org/10.1103/PhysRevA.110.012430>.

[9] Over P, Bengoechea S, Brearley P, Laizet S, Rung T. Quantum algorithm for the advection-diffusion equation with optimal success probability. 2024, arXiv (pre-print). [Online]. Available: <https://doi.org/10.48550/arXiv.2410.07909>.

[10] Gaitan F. Finding flows of a Navier–Stokes fluid through quantum computing. npj Quantum Inf 2020;6(1). [Online]. Available: <https://doi.org/10.1038/s41534-020-00291-0>.

[11] Oz F, Vuppala RKSS, Kara K, Gaitan F. Solving Burgers' equation with quantum computing. Quantum Inf Process 2021;21(1). [Online]. Available: <https://doi.org/10.1007/s11128-021-03391-8>.

[12] Childs AM, Liu J-P. Quantum spectral methods for differential equations. Comm Math Phys 2020;375(2):1427–57. [Online]. Available: <https://doi.org/10.1007/s00220-020-03699-z>.

[13] Childs AM, Liu J-P, Ostrander A. High-precision quantum algorithms for partial differential equations. Quantum 2020;5. [Online]. Available: <https://doi.org/10.22331/q-2021-11-10-574>.

[14] Cao Y, Papageorgiou A, Petras I, Traub J, Kais S. Quantum algorithm and circuit design solving the Poisson equation. New J Phys 2013;15(1). [Online]. Available: <https://doi.org/10.1088/1367-2630/15/1/013021>.

[15] Chen Z-Y, Xue C, Chen S-M, Lu B-H, Wu Y-C, Ding J-C, Huang S-H, Guo G-P. Quantum finite volume method for computational fluid dynamics with classical input and output. 2021, arXiv (pre-print). [Online]. Available: <https://doi.org/10.48550/arXiv.2102.03557>.

[16] Kacewicz B. Almost optimal solution of initial-value problems by randomized and quantum algorithms. J Complexity 2006;22(5):676–90. [Online]. Available: <https://doi.org/10.1016/j.jco.2006.03.001>.

[17] Peruzzo A, McClean J, Shadbolt P, Yung M-H, Zhou X-Q, Love PJ, Aspuru-Guzik A, O'Brien JL. A variational eigenvalue solver on a photonic quantum processor. Nature Commun 2014;5(1). [Online]. Available: <https://doi.org/10.1038/ncomms5213>.

[18] Cerezo M, Arrasmith A, Babbush R, Benjamin SC, Endo S, Fujii K, McClean JR, Mitarai K, Yuan X, Cincio L, Coles PJ. Variational quantum algorithms. Nat Rev Phys 2021;3(9). [Online]. Available: <https://doi.org/10.1038/s42254-021-00348-9>.

[19] Demirdjian R, Gunlycke D, Reynolds CA, Doyle JD, Tafur S. Variational quantum solutions to the advection–diffusion equation for applications in fluid dynamics. Quantum Inf Process 2022;21(9). [Online]. Available: <https://doi.org/10.1007/s11128-022-03667-7>.

[20] Bravo-Prieto C, LaRose R, Cerezo M, Subasi Y, Cincio L, Coles PJ. Variational quantum linear solver. Quantum 2023;7. [Online]. Available: <https://doi.org/10.22331/q-2023-11-22-1188>.

[21] Kyriienko O, Paine AE, Elfving VE. Solving nonlinear differential equations with differentiable quantum circuits. Phys Rev A 2021;103(5). [Online]. Available: <https://doi.org/10.1103/PhysRevA.103.052416>.

[22] Sato Y, Kondo R, Koide S, Takamatsu H, Imoto N. Variational quantum algorithm based on the minimum potential energy for solving the Poisson equation. Phys Rev A 2021;104(5). [Online]. Available: <https://doi.org/10.1103/PhysRevA.104.052409>.

[23] Leong FY, Ewe W-B, Koh DE. Variational quantum evolution equation solver. Sci Rep 2022;12(1). [Online]. Available: <https://doi.org/10.1038/s41598-022-14906-3>.

[24] Leong FY, Koh DE, Ewe W-B, Kong JF. Variational quantum simulation of partial differential equations: applications in colloidal transport. Internat J Numer Methods Heat Fluid Flow 2023;33(11). [Online]. Available: <https://doi.org/10.1108/hff-05-2023-0265>.

[25] Jaksch D, Givi P, Daley AJ, Rung T. Variational quantum algorithms for computational fluid dynamics. AIAA J 2023;61(5):1885–94. [Online]. Available: <https://doi.org/10.2514/1.J062426>.

[26] Lubasch M, Joo J, Moinier P, Kiffner M, Jaksch D. Variational quantum algorithms for nonlinear problems. Phys Rev A 2020;101(1). [Online]. Available: <https://doi.org/10.1103/PhysRevA.101.010301>.

[27] Guseynov NM, Zhukov AA, Pogosov WV, Lebedev AV. Depth analysis of variational quantum algorithms for the heat equation. Phys Rev A 2023;107(5). [Online]. Available: <https://doi.org/10.1103/PhysRevA.107.052422>.

[28] Costa PCS, Jordan S, Ostrander A. Quantum algorithm for simulating the wave equation. Phys Rev A 2019;99(1). [Online]. Available: <https://doi.org/10.1103/PhysRevA.99.012323>.

[29] Javadi-Abhari A, Treinish M, Krsulich K, Wood CJ, Lishman J, Gacon J, Martiel S, Nathon PD, Bishop LS, Cross AW, Johnson BR, Gambetta JM. Quantum computing with Qiskit. 2024, arXiv (pre-print). [Online]. Available: <https://doi.org/10.48550/arXiv.2405.08810>.

[30] Lynch RE, Rice JR, Thomas DH. Direct solution of partial difference equations by tensor product methods. Numer Math 1964;6(1):185–99. [Online]. Available: <https://doi.org/10.1007/bf01386067>.

[31] Arrazola JM, Kalajdzievski T, Weedbrook C, Lloyd S. Quantum algorithm for nonhomogeneous linear partial differential equations. Phys Rev A 2019;100(3). [Online]. Available: <https://doi.org/10.1103/PhysRevA.100.032306>.

[32] Grossmann C, Roos H-G, Stynes M. Numerical treatment of partial differential equations. Springer Berlin Heidelberg; 2007. [Online]. Available: <https://doi.org/10.1007/978-3-540-71584-9>.

[33] Glowinski R. Variational methods for the numerical solution of nonlinear elliptic problems. Philadelphia, PA: Society for Industrial and Applied Mathematics; 2015. [Online]. Available: <https://doi.org/10.1137/1.9781611973785>.

[34] Troutman JL. Variational calculus and optimal control. New York: Springer; 1996. [Online]. Available: <https://doi.org/10.1007/978-1-4612-0737-5>.

[35] Werner D. Funktionalanalysis. Springer; 2005. [Online]. Available: <https://doi.org/10.1007/978-3-662-55407-4>.

[36] Nielsen MA, Chuang IL. Quantum computation and quantum information. Cambridge University Pr.; 2010. [Online]. Available: <https://doi.org/10.1017/CBO9780511976667>.

[37] Nakaji K, Yamamoto N. Expressibility of the alternating layered ansatz for quantum computation. Quantum 2021;5. [Online]. Available: <https://doi.org/10.22331/q-2021-04-19-434>.

[38] Vatan F, Williams C. Optimal quantum circuits for general two-qubit gates. Phys Rev A 2004;69(3). [Online]. Available: <https://doi.org/10.1103/PhysRevA.69.032315>.

[39] Endo S, Cai Z, Benjamin SC, Yuan X. Hybrid quantum-classical algorithms and quantum error mitigation. J Phys Soc Japan 2021;90(3). [Online]. Available: <https://doi.org/10.7566/jpsj.90.032001>.

- [40] Vedral V, Barenco A, Ekert A. Quantum networks for elementary arithmetic operations. *Phys Rev A* 1996;54(1). [Online]. Available: <https://doi.org/10.1103/PhysRevA.54.147>.
- [41] Liu H-L, Wu Y-S, Wan L-C, Pan S-J, Qin S-J, Gao F, Wen Q-Y. Variational quantum algorithm for the Poisson equation. *Phys Rev A* 2021;104. [Online]. Available: <https://doi.org/10.1103/PhysRevA.104.022418>.
- [42] Oseledets IV. Constructive representation of functions in low-rank tensor formats. *Constr Approx* 2012;37(1):1–18. [Online]. Available: <https://doi.org/10.1007/s00365-012-9175-x>.
- [43] Gonzalez-Conde J, Watts TW, Rodriguez-Grasa P, Sanz M. Efficient quantum amplitude encoding of polynomial functions. *Quantum* 2024;8:1297. [Online]. Available: <https://doi.org/10.22331/q-2024-03-21-1297>.
- [44] Oseledets I, Tyrtysnikov E. TT-cross approximation for multidimensional arrays. *Linear Algebra Appl* 2010;432(1):70–88. [Online]. Available: <https://doi.org/10.1016/j.laa.2009.07.024>.
- [45] Sarma A, Watts TW, Moosa M, Liu Y, McMahon PL. Quantum variational solving of nonlinear and multidimensional partial differential equations. *Phys Rev A* 2024;109:062616. [Online]. Available: <https://doi.org/10.1103/PhysRevA.109.062616>.
- [46] Creevey FM, Hill CD, Hollenberg LCL. GASP: a genetic algorithm for state preparation on quantum computers. *Sci Rep* 2023;13(1). [Online]. Available: <https://doi.org/10.1038/s41598-023-37767-w>.
- [47] Melnikov AA, Termanova AA, Dolgov SV, Neukart F, Perelshtein MR. Quantum state preparation using tensor networks. *Quantum Sci Technol* 2023;8(3). [Online]. Available: <https://doi.org/10.1088/2058-9565/acd9e7>.
- [48] Araujo IF, Park DK, Petruccione F, da Silva AJ. A divide-and-conquer algorithm for quantum state preparation. *Sci Rep* 2021;11(1). [Online]. Available: <https://doi.org/10.1038/s41598-021-85474-1>.
- [49] Bittel L, Kliesch M. Training variational quantum algorithms is NP-hard. *Phys Rev Lett* 2021;127(12). [Online]. Available: <https://doi.org/10.1103/PhysRevLett.127.120502>.
- [50] Wiersema R, Killoran N. Optimizing quantum circuits with Riemannian gradient flow. *Phys Rev A* 2023;107. [Online]. Available: <https://doi.org/10.1103/PhysRevA.107.062421>.
- [51] Kennedy J, Eberhart R. Particle swarm optimization. In: Proceedings of ICNN'95-international conference on neural networks. Vol. 4, IEEE; 1995, p. 1942–8. [Online]. Available: <https://doi.org/10.1109/ICNN.1995.488968>.
- [52] Schuld M, Bergholm V, Gogolin C, Izaac J, Killoran N. Evaluating analytic gradients on quantum hardware. *Phys Rev A* 2019;99(3). [Online]. Available: <https://doi.org/10.1103/PhysRevA.99.032331>.
- [53] Banchi L, Crooks GE. Measuring analytic gradients of general quantum evolution with the stochastic parameter shift rule. *Quantum* 2021;5. [Online]. Available: <https://doi.org/10.22331/q-2021-01-25-386>.
- [54] Crooks GE. Gradients of parameterized quantum gates using the parameter-shift rule and gate decomposition. 2019, arXiv (pre-print). [Online]. Available: <https://doi.org/10.48550/arXiv.1905.13311>.
- [55] Nocedal J, Wright SJ. Numerical optimization. 2nd ed.. New York, NY: Springer; 2006. [Online]. Available: <https://doi.org/10.1007/978-0-387-40065-5>.
- [56] Over P, Bengoechea S, Rung T, Clerici F, Scandurra L, De Villiers E, Jaksch D. Boundary treatment for variational quantum simulations of partial differential equations on quantum computers [data set]. 2024, QCFD Boundary Conditions Data: Version v1. [Online]. Available: <https://doi.org/10.25592/uhhfdm.14123>.
- [57] Holmes Z, Sharma K, Cerezo M, Coles PJ. Connecting ansatz expressibility to gradient magnitudes and barren plateaus. *PRX Quantum* 2022;3. [Online]. Available: <https://doi.org/10.1103/PRXQuantum.3.010313>.
- [58] Bengoechea S, Over P, Jaksch D, Rung T. Towards Variational Quantum Algorithms for generalized linear and nonlinear transport phenomena. 2024, arXiv (pre-print). [Online]. Available: <https://doi.org/10.48550/arXiv.2411.14931>.
- [59] Ragone M, Bakalov BN, Sauvage F, Kemper AF, Ortiz Marrero C, Larocca M, Cerezo M. A Lie algebraic theory of barren plateaus for deep parameterized quantum circuits. *Nature Commun* 2024;15(1). [Online]. Available: <https://doi.org/10.1038/s41467-024-49909-3>.
- [60] Sack SH, Medina RA, Michailidis AA, Kueng R, Serbyn M. Avoiding barren plateaus using classical shadows. *PRX Quantum* 2022;3. [Online]. Available: <https://doi.org/10.1103/PRXQuantum.3.020365>.
- [61] Pesah A, Cerezo M, Wang S, Volkoff T, Sornborger AT, Coles PJ. Absence of barren plateaus in quantum convolutional neural networks. *Phys Rev X* 2021;11. [Online]. Available: <https://doi.org/10.1103/PhysRevX.11.041011>.
- [62] Jaderberg B, Agarwal A, Leonhardt K, Kiffner M, Jaksch D. Minimum hardware requirements for hybrid quantum—classical DMFT. *Quantum Sci Technol* 2020;5(3). [Online]. Available: <https://doi.org/10.1088/2058-9565/ab972b>.