

Where Quantum and Fluids entangle

Quantum Computational Fluid Dynamics
Extended Gate-Level QCFD Set
Project Number# 101080085

Call:	HORIZON-CL4-2021-DIGITAL-EMERGING-02
Topic:	HORIZON-CL4-2021-DIGITAL-EMERGING-02-10
Type of action:	HORIZON Research and Innovation Actions
Granting authority:	European Commission-EU
Project starting date: fixed date:	1 November 2022
Project end date:	30 April 2027
Project duration:	54 months
Project Coordinator:	University of Hamburg (UHH)
Deliverable number:	D6.2
WP contributing to the deliverable:	WP6 – Gate-level simulations
Deliverable Type:	Data
Dissemination level:	Public
Due Submission date:	30/10/2027
Prepared By:	FZJ, TUC, PlanQC
Internal Reviewers:	UHH, TUHH, FZJ, ENG, TUC, PlanQC, TUM
Final Approval:	Prof. Dr. Dieter Jaksch

Revision History

Version	Date	Who	Changes
R0	30.10.2026	FM, FC, JJ, AS	

Executive Summary

Deliverable D6.2 extends the gate-level set of work package 6. Focus on circuit level simulations to go through all the steps and emulate the behaviour of a quantum computer for simple problems. It asks which quantum algorithms can advance the differential equations of computational fluid dynamics (CFD) with circuits that devices of today and of the near future can run, and how far those circuits can be reduced. Every algorithm is judged by numbers read off compiled circuits: qubits, two-qubit gates, depth and circuit executions. Where possible, the test problems are those of the project benchmark set D2.1 [1]. The repository doi.org/10.25592/uhhfdm.22085 contains the necessary data.

Main results

- **A variational time step ran on hardware.** Based on testing via circuit-based simulations, the Lubasch variational time stepping was run on IBM hardware: the viscous Burgers equation was advanced five steps on IBM quantum processors with a final mean squared error of 2.1 %; optimised adders need about half the two-qubit gates, and error mitigation keeps the norm within 1 % of the ideal value (Section 1).
- **The number of measurements problem of nonlinear terms is partly solved.** A diagonal block encoding keeps the success probability of the nonlinear step near 15–30 % on up to 10^7 grid points, where the state-based product falls below 10^{-4}, at the price of a classical tensor-train copy of the field (Section 2).
- **Block-encoding circuits reduced in depth by 4–15×** on IBM heavy-hex and square-lattice layouts by coherent permutation (Section 3).
- **Resource comparison favours the variational algorithm on today's devices.** On a three-qubit Burgers problem the amplitude-encoded variational circuit has a two-qubit depth of 30–40 against 2.4–3.4×10^6 for a block-encoding with quantum singular value transform (QSVT); the price is many circuit executions and gradients that shrink with qubit number, most strongly for the nonlinear term (Section 4).
- **Two applications work in simulation.** The Grad–Shafranov equation reaches an infidelity of about 10^{-3} on four qubits (Section 5), and the fourth-order Cahn–Hilliard equation with its cubic term stays below an RMSE of 0.05 over 30 time steps on two qubits with 10,000 shots (Section 6).
- **New algorithms were tested at gate level:** learned circuits that are up to 10^3 times more accurate than Trotter steps of equal size (Section 7), time marching with unit normalisation whose circuit shell alone needs 252 CX gates per step, QNPU building blocks whose gates grow linearly with the number of qubits (Section 8), and the reduction in depth of multi-control Toffli circuits, commonly used in variational algorithms, from logarithmic to constant through the use of measurement and feedback (Section 9).

Almost all results are noiseless, small in size and one-dimensional. The main open questions are the trainability of variational circuits at larger qubit numbers, shots and noise.

Contents

1 Platforms and algorithms 7
1.1 Equations and benchmark cases 7
1.2 Algorithm families 8
1.3 Optimising the circuits on a superconducting processor 8
2 Measurement-efficient nonlinear circuits 9
2.1 Diagonal block encoding of the prefactor 10
2.2 Benchmarks on turbulent flows 10
3 Advances in block encoding 11
3.1 Coherent permutation and MCX compression 11
3.2 Hardware resources 12
4 Numerical comparison of algorithms in terms of circuit size 13
4.1 Linear benchmark: HHL, VQLS and physics-informed circuits 13
4.2 Lubasch-type variational time stepping: trainability and resources 14
4.3 Physics-informed quantum machine learning 15
4.4 Block encoding and QSVT 15
4.5 Which algorithm for devices of today 16
5 Variational quantum solution of the Grad–Shafranov equation 17
5.1 Picard and Weak cost functions 17
5.2 Convergence and the hybrid strategy 17
5.3 Resources of the boundary circuits 18
6 Cahn–Hilliard equation 19
6.1 Results on two qubits 19
6.2 Gate requirements for larger grids 20
6.3 2D implementation 20
7 Operator-learning circuits 22
7.1 Propagators and CFD operators 23
8 Variational generalised linear phenomena 25
8.1 Building blocks for band matrices 25
8.2 Test problems 26
9 Optimizing circuit depth for VQA 25
9.1 Depth optimization for CNOT ladders 25
9.2 Minimum Toffoli depth for the multi-controlled Toffoli gate via teleportation 26
10 Summary 279
11 References 30

List of figures and tables

Figures

Figure 1. Structure of the report 7
Figure 2. Variational Burgers dynamics on IBM Marrakesh and circuit optimisation 9
Figure 3. Hybrid quantum–classical tensor-network scheme for the nonlinear time step 10
Figure 4. Measurement-efficient nonlinear circuits for the Taylor–Green vortex 11
Figure 5. Block-encoding circuit and index-mapping oracles 12
Figure 6. Two-qubit depth of three index-mapping circuits before and after optimisation 13
Figure 7. Scaling of the circuit sizes with the number of grid points 14
Figure 8. Lubasch-type variational circuits: gradient variance and hardware resources 15
Figure 9. Physics-informed quantum machine learning: gradient variance and hardware resources 16
Figure 10. Heat equation by block encoding and QSVT 16
Figure 11. Weak, Picard and hybrid optimisation 18
Figure 12. Two-qubit gates and depth of the neutralising circuits 19
Figure 13. Hadamard-test circuits for the Cahn–Hilliard cost function 19
Figure 14. Cahn–Hilliard results on two qubits 20
Figure 15. Two-qubit gates and depth of the nonlinear Cahn–Hilliard circuit 20
Figure 16. Evolution of the 2D Cahn-Hilliard equation 21
Figure 17. Evolution of the 2D Cahn-Hilliard equation with shots 21
Figure 18. Final measurement of the the 2D Cahn-Hilliard equation with shots 22
Figure 19. Operator-learning pipeline 24
Figure 20. Compression of propagators 24
Figure 21. Second-difference operator learned with four-qubit gates 24
Figure 22. Gate counts of the ansatz and of the QNPU building blocks 25
Figure 23. Error against the classical solution for four unsteady problems 26
Figure 24. Step by step transition from a unitary to a non-unitary circuit 27
Figure 25. Comparison between a unitary and a non-unitary circuit 28
Figure 26. Comparison among MCT decompositions 28
Figure 27. Teleportation-based decomposition of an MCT gate 29

Tables

Table 1. Test problems of this report and the corresponding D2.1 benchmark cases 7
Table 2. Resources on the heat-equation benchmark, 8 grid points, 8 time steps [25] 13
Table 3. Hardware resources of the block-encoding/QSVT pathway [30] 16
Table 4. Algorithms for a three-qubit Burgers problem (analysis by A. Setty, FZJ) 17
Table 5. RMSE of one Cahn–Hilliard time step against finite [35] 20
Table 6. Learned circuits: resources and accuracy [36] 23
Table 7. Compiled CX gates of the circuit shell of one 2D time step (computed for this report) 25
Table 8. Result and limit of each section 29

List of abbreviations

Acronym / Short Name	Meaning
EU	European Union
EC	European Commission
QCFD	Quantum Computational Fluid Dynamics
UHH	University of Hamburg
TUHH	Technical University of Hamburg
FZJ	Jülich Research Centre
TUM	Technical University of Munich
ENG	ENGYS SRL
TUC	Technical University of Crete
Planqc	Planqc GMBH
CDE	Communication, Dissemination and Exploitation
KPI	Key Performance Indicator
WP	Work Package
PO	Project Officer
SME	Small and Medium Enterprise
WPL	Work Package Leader
BE	Block encoding
CDS / UDS / LUDS / QUICK / FB	Central / upwind / linear upwind / quadratic upstream / flux-blended convection schemes
CFD	Computational fluid dynamics
DE / PDE	(Partial) differential equation
GS	Grad–Shafranov (equation)
HHL	Harrow–Hassidim–Lloyd linear-system algorithm
KITE	Name used in the project for the time-marching algorithm of Section 8
LCU	Linear combination of unitaries
MCX	Multi-controlled X gate
MSE / RMSE	(Root) mean squared error
NISQ	Noisy intermediate-scale quantum
PINN / QPINN / PIQML	Physics-informed neural network / quantum PINN / physics-informed quantum machine learning
QNPU	Quantum nonlinear processing unit
QSVT	Quantum singular value transformation
SGEO	Sequential grid-based explicit optimisation
TCI / EST / TTS	Tensor cross interpolation / efficient state tomography / tensor-train simulation
TDJ / TGV	Temporally developing jet / Taylor–Green vortex
TT	Tensor train
VQA	Variational quantum algorithm
VQLS	Variational quantum linear solver
ZNE	Zero-noise extrapolation

1 Platforms and algorithms

All amplitude encoding algorithms of this report solve a discretised equation in the same way: the field on $N = 2^n$ grid points is stored in the amplitudes of an n-qubit state, a circuit applies the discretised operator, the state is advanced in time or iteration, and a result is read out. Amplitude storage is the source of the hoped-for advantage; it also makes applying a non-unitary operator (possible only with a probability of success) and reading out the field (requiring a large number of measurements) the hard parts. The algorithms differ in how they apply the operator, whether the circuit is fixed or trained, and how they treat the nonlinear term. Figure 1 shows the flow between the different simulation studies that we have conducted.

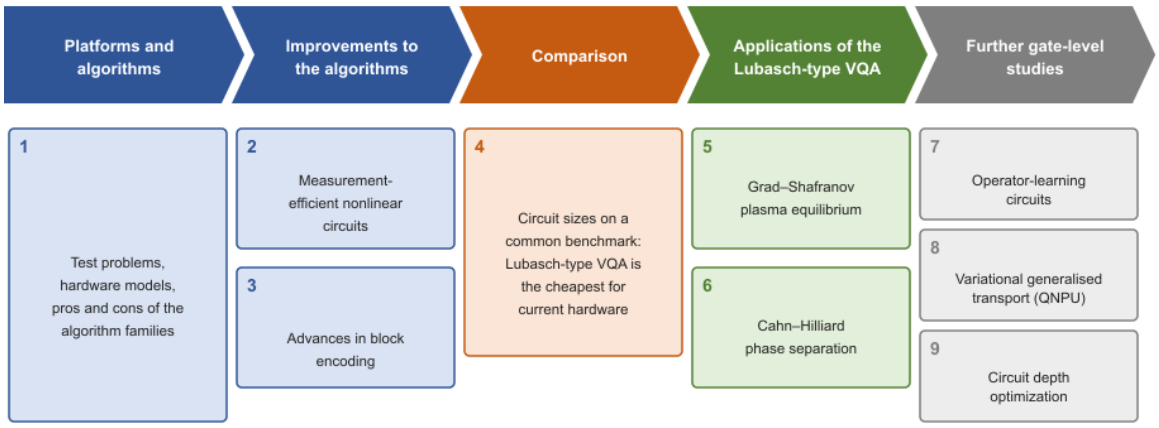


FIGURE 1. STRUCTURE OF THE REPORT

Foundations, improvements of the algorithms, comparison, applications of the Lubasch-type algorithm and novel algorithms.

1.1 Equations and benchmark cases

The test problems are aligned with the core benchmark set D2.1 of the project [1], which provides classical finite-difference and finite-volume reference data, and the accuracy measures used here are those defined there: absolute and L_2 errors, fidelity and trace distance. D2.1 stresses that a direct performance comparison between quantum and classical solvers is not meaningful at this stage; the cases serve to validate and develop the algorithms. Note that even for classical simulation of the quantum circuits, large problem sizes are not possible due to the exponential scaling of the Hilbert space. Table 1 maps the problems of this report to the D2.1 cases.

TABLE 1. TEST PROBLEMS OF THIS REPORT AND THE CORRESPONDING D2.1 BENCHMARK CASES

Equation or model (key characteristics)	Setting in this report	D2.1 case	Sect.
Heat equation, variable diffusivity (linear, parabolic)	1D, Gaussian diffusivity, Dirichlet 0/1, 16 and 64 interior points, 40 steps	3.1.1 Transient heat conduction with non-uniform diffusivity	8
Taylor–Green vortex (TGV) benchmark	3D Taylor–Green vortex (Re = 10–500)	3.2.1 Taylor–Green-vortex-like advection–diffusion	2
Heat equation, constant diffusivity	1D, Dirichlet, 8 points, 8 backward-Euler steps (benchmark)	3.2.2 Unsteady heat equation including boundary conditions	4
Burgers / Navier–Stokes equations (quadratic)	1D Burgers on classical and quantum hardware; 2D temporally developing jet (Re 10–1000)	3.2.7 Rayleigh–Bénard as next target	1, 2, 4, 8
Grad–Shafranov (nonlinear, elliptic)	2D rectangle, four qubits, three boundary types	–	5
Cahn–Hilliard (cubic, fourth order)	1D periodic, two qubits, 30 steps	–	6
Operators of CFD solvers	Second difference, NACA 0012 panel matrix, sparse matrices	–	3, 7

1.2 Algorithm families

Quantum linear-system solvers (HHL, QSVT) [2, 3] invert the discretised operator coherently; they have provable complexity bounds but need block encodings, state preparation and fault tolerance, and nonlinear equations only through Carleman linearisation [4, 5]. Coherent time marching (Section 8) applies a block-encoded time step repeatedly. On the other hand, variational algorithms train a shallow circuit: VQLS for the linear system [6], physics-informed circuits on the PDE residual [7], and the time stepping of Lubasch et al. [8, 9], on which the project concentrates. In the latter, the field at time t is $\lambda\, U(\theta)|0\rangle$ with a classical norm λ, and the parameters of the next step minimise the cost function

$$C(\lambda, \theta) = \| \lambda\, U(\theta)|0\rangle - \lambda_0 \sum_k \quad c_k\, O_k\, U(\theta_0)|0\rangle \|^2, \tag{1}$$

where each operator O_k of the discretised update (identity, shifts, nonlinear products) contributes overlaps measured with one-ancilla Hadamard tests. A nonlinear term of degree p needs $p-1$ further copies of the register, so the degree of the nonlinearity sets the qubit count, and there is no truncation as in Carleman linearisation. The variational families move the cost from circuit depth to number of circuit executions; this exchange is attractive while depth is limited by coherence, and it is the reason for the working choice of the project (Section 4).

Platforms. Circuits are built and transpiled with Qiskit [10] for IBM heavy-hex processors (Heron r3 "Boston", Marrakesh, Kingston; native CZ, √X, Rz, X), the square-lattice Nighthawk r1 "Miami", all-to-all trapped ions (rxx, rz, r) and logical gate sets for fault-tolerant estimates. Apart from the hardware study below, all results are noiseless.

1.3 Optimising the circuits on a superconducting processor

An FZJ study [11] ran the variational time stepping on both classical emulators and IBM Marrakesh for the 1D Burgers equation $\partial_t u = \nu \partial_x^2 u - u \partial_x u$ with periodic boundaries, using 10,000 shots per operator evaluation. In the viscous case ($\nu = 0.01$, $Re \approx 100$, five steps of $\Delta t = 0.06$) the convection steepens the profile into a sharp gradient; in the inviscid case, the exact solution is linear. The ansatz is a nearest-neighbour chain of R_y–CNOT blocks with 2–5 parameters termed as hardware efficient ansatz. The nonlinear block is the deepest part because the Hadamard qubit controls every operation; it was reduced by dropping controls on gates that leave the all-zero state unchanged, by relative-phase Toffoli gates [12] (three instead of six CNOTs) and by a modified controlled-R_y that saves one two-qubit gate. For an N-qubit field register the adder that implements the shifts falls from $13N - 14$ to $7N - 11$ two-qubit gates, i.e. from 64 to 31 for $N = 6$ and from 142 to 73 for $N = 12$ (all-to-all connectivity).

All circuits use XY4 dynamical decoupling as a resource for error mitigation. For the overlaps the study introduces Hadamard verification: under global depolarising noise of strength η every overlap is attenuated by $1 - \eta$, which is estimated by omitting the final Hadamard gate and comparing the probability of the all-zero outcome with that of the ancilla alone,

$$\langle O \rangle_{ideal} = \frac{\langle O \rangle_{noisy}}{1-\eta}. \tag{2}$$

It needs two circuits of the original depth, where digital zero-noise extrapolation needs three of increasing depth, and it outperformed both quadratic and exponential extrapolation (Figure 2).

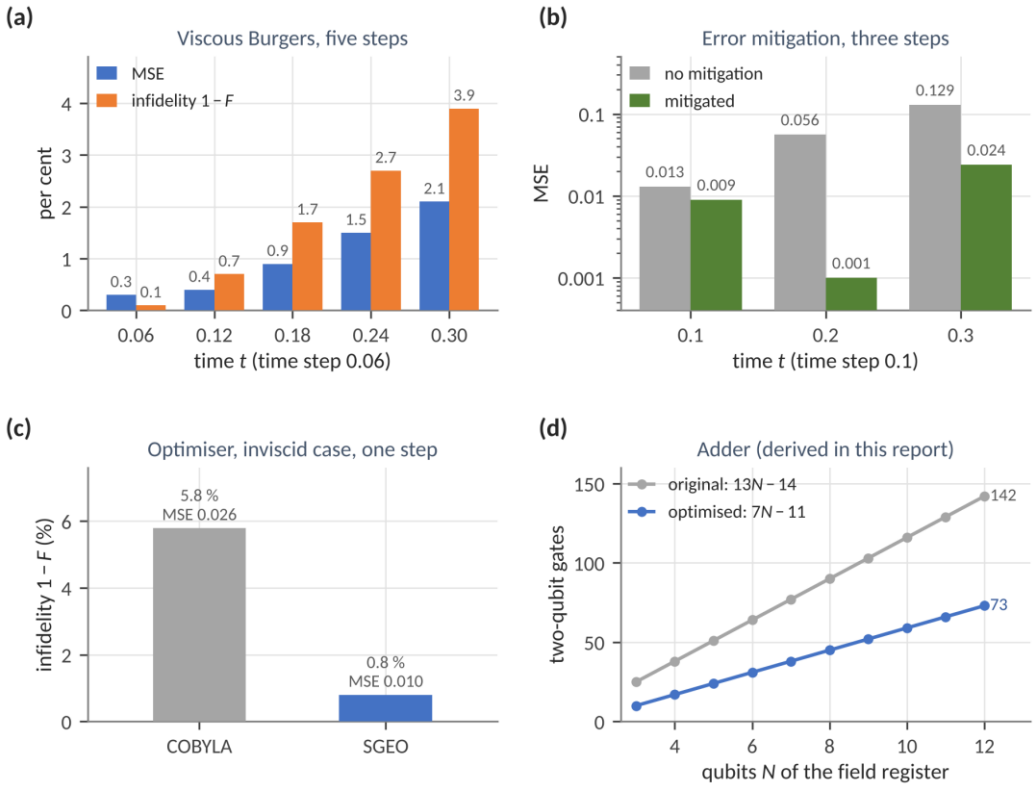


FIGURE 2. VARIATIONAL BURGERS DYNAMICS ON IBM MARRAKESH AND CIRCUIT OPTIMISATION

(a) Viscous case, five steps: MSE and infidelity against the error-free evolution [11]. (b) MSE over three steps of 0.1 with and without mitigation. (c) Inviscid case, one cold-start step: COBYLA versus SGEO. (d) Two-qubit gates of the adder before and after optimisation.

The error grows with the number of steps: the mean squared error (MSE) rises from 0.3 % after the first to 2.1 % after the fifth step and the fidelity falls from 99.9 % to 96.1 %. The norm is the most sensitive quantity. With mitigation it stays within 1 % of the ideal value (MSE 0.009, 0.001, 0.024 over three steps), but without it it deviates by more than 80 % and the field collapses; the dominant failure mechanism is the recursive amplification of normalisation errors, not the local gate noise. The optimiser matters significantly: from a cold start SGEO reaches 99.2 % fidelity where COBYLA reaches 94.2 %, and with warm starts SGEO converges in about ten iterations per time step. The deepest circuits contain up to 60 layers of CZ gates. The study uses few qubits, one dimension and one device; the fields were reconstructed from the parameters by noiseless circuit simulation.

2 Measurement-efficient nonlinear circuits

We now discuss the central variational circuit estimation problem and how the circuits can be modified to compensate for it [15]. A block-encoded operator acts on an amplitude-encoded state only when its auxiliary registers (encoding nonlinearity) are found in the all-zero state. The probability α_{succ} of this event enters the sampling cost directly: estimating a cost function to accuracy ε needs

$$m \propto \frac{1}{\varepsilon^2 \, \alpha_{succ}} \tag{3}$$

number of measurements. For the linear operators of a PDE, α_{succ} is fixed by the operator and does not change under refinement [13]. For the nonlinear term of the Lubasch-type algorithm it does: the product $u_k \partial_{x_k} u_i$ is formed from two amplitude-encoded copies of the state, the resulting block encoding has the subnormalisation $\|u\|_2$, which grows with the grid, and for generic fields α_{succ} falls as $1/N$ [14]. Because the nonlinear term is what makes CFD

hard, this is the central scaling problem of the working algorithm of the project. Siegl et al. [15] quantify it for turbulent flow and show how this scaling impediment can be removed.

2.1 Diagonal block encoding of the prefactor

Multiplying a state by a field is the action of a diagonal matrix. Block-encoded directly, $diag(u)$ has a natural normalisation $max|u|$ which is set by the physics (the peak velocity) and is independent of N. The present scheme reconstructs the prefactor u_k as a tensor train (TT) from the parametrised circuit of the current step, contracts every TT tensor with a delta tensor to obtain a diagonal TT operator, and compiles it into a unitary $\hat{U}_\phi$ by quantum tensor programming [13, 16], whose depth grows at most as χ^2, with the bond dimension χ. Only the prefactor is reconstructed while the differentiated field stays in the quantum state. Linear and nonlinear parts are combined by an LCU circuit [17] with mixing angle ω (Figure 3), and a Hadamard test with an adapted final rotation returns the cost and fidelity of the trained state, so convergence can be monitored without reference data.

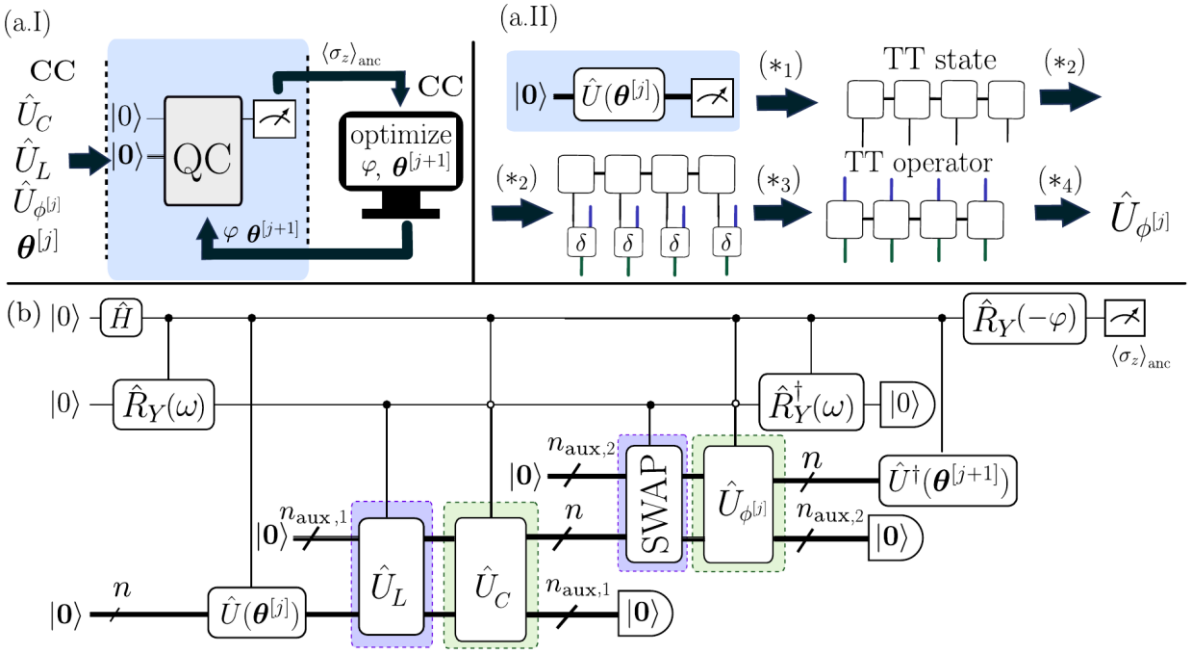


FIGURE 3. HYBRID QUANTUM–CLASSICAL TENSOR-NETWORK SCHEME FOR THE NONLINEAR TIME STEP

(a.I) Variational loop: the quantum circuit applies the compiled operators, the classical computer updates $\boldsymbol{\theta}$ and $\boldsymbol{\lambda}$. (a.II) The field is reconstructed as a TT state, turned into a diagonal TT operator and compiled into $\hat{U}_\phi$. (b) Circuit of one step: linear (violet) and nonlinear (green) blocks combined with angle ω; auxiliary registers are post-selected on $|0\rangle$. Adapted from [15].

The TT copy can be obtained in three ways: tensor cross interpolation (TCI) [18] or efficient state tomography (EST), both needing $O(\chi^2)$ expectation values, or a classical TT simulation of the already trained circuit (TTS), which is measurement-free and costs $O(\chi^3)$, the same order as the compilation. In the tests EST and TTS reach the optimal (SVD) bond dimension, TCI more than twice that value at large Reynolds numbers, because single amplitudes scale as $N^{-1/2}$ and shot noise dominates.

2.2 Benchmarks on turbulent flows

The scheme was tested on DNS data of the momentum equation for a 2D temporally developing jet (TDJ, Re = 10–1000, up to 2^{20} points) and a 3D Taylor–Green vortex (TGV, Re = 10–500, up to 2^{24} points) (Figure 4).

- **Success probability.** The state-based product loses about two orders of magnitude over the TDJ range (to about 10^{-4}) and more than four over the TGV range (to about 10^{-6}). With the diagonal block encoding α_{succ} stays flat at 15–30 % in the turbulent regime, so the shot count no longer grows with (powers of) N; on the largest TGV

grid, this is about six circuit repetitions per accepted one instead of more than 10^6. Only laminar TGV cases give values near 10^{-2}.

- **Bond dimension.** The prefactor needs a bond dimension about 2× (TGV) to 3× (TDJ) smaller than a fully classical TT evaluation of the advective terms, because a field is more compressible than its derivative or the nonlinear product; the finite accuracy of the circuit hardly changes this.
- **Time cost.** With $T_H = T_{QC} + T_{CL}$, $T_{QC} \sim d/(\varepsilon^2 \alpha_{succ})$ and $T_{CL} \sim \chi^3$, the classical overhead is up to three orders of magnitude below the classical TT Hadamard product $T_{TN} \sim \chi_{TN}^4$ at the highest Reynolds numbers, and the ratio grows with Re.

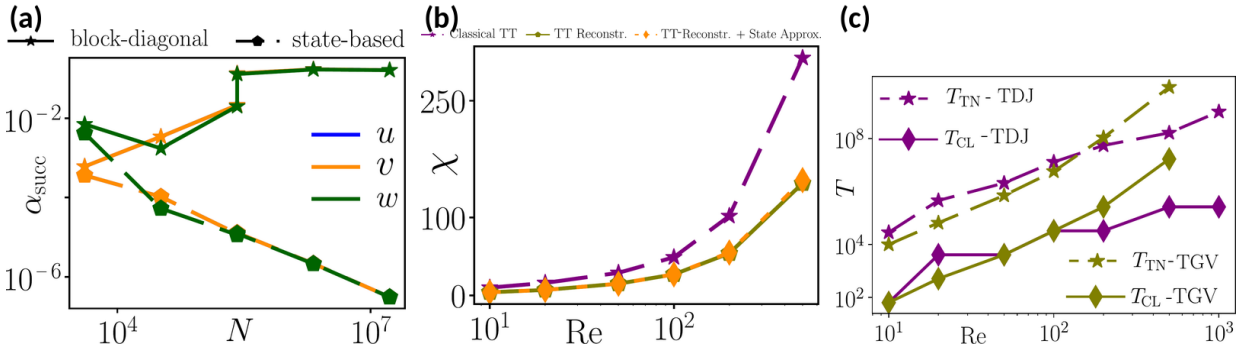


FIGURE 4. MEASUREMENT-EFFICIENT NONLINEAR CIRCUITS FOR THE TAYLOR–GREEN VORTEX

(a) Success probability versus grid points, diagonal block encoding (solid) and state-based product (dashed); (b) bond dimension versus Reynolds number for the classical TT and the reconstructed prefactor; (c) classical cost of the fully classical TT solver (T_{TN}) and of the hybrid scheme (T_{CL}).

Summary. The shot count of the nonlinear step using this methodology no longer depends on the grid. The problem is solved only in part: measurement-based reconstruction needs 10^4–10^6 shots per expectation value for percent-level accuracy, TTS moves the work to a classical cost that grows with the bond dimension and hence with Re, and all results are noiseless emulations of one Euler step. Outlook includes coupled problems such as Rayleigh–Bénard convection (D2.1 case 3.2.7 [1, 19]), precompiled diagonal operators for immersed boundaries, and cheaper reconstruction with classical shadows.

3 Advances in block encoding

A block encoding (BE) represents a generally non-unitary matrix A as the upper-left block of a larger unitary,

$$U_A = (A/\alpha \ * * *), \qquad \|A/\alpha\| \leq 1. \tag{4}$$

It is the input of QSVT [3], quantum linear-system solvers [2] and Hamiltonian simulation [17], and the operators of Sections 2, 7 and 8 are compiled into block encodings. Its cost on a device is set by the gates of U_A and by the subnormalisation α, which lowers the success probability of everything built on top; a cheaper block encoding therefore improves all these methods. Dense matrices can be block-encoded with state preparation and LCU [20]; the operators of PDE solvers are sparse, and on regular grids highly structured. For them the construction of [21, 22] uses a state-preparation oracle (PREP/UNPREP), which stores every distinct value once, and an index-mapping oracle that places each value on its diagonal and rows (Figure 5a). The work of Setty [23] optimises the index mapping, whose multi-controlled X (MCX) gates dominate the cost and must be routed through the connectivity of the hardware.

3.1 Coherent permutation and MCX compression

The index mapping is built from three operations, each conditioned on the data register: shift (move an element to its diagonal), delete (remove it from rows where it does not occur) and insert (place it only in selected rows). Three transformations reduce the circuit depth:

- **Composition theorem.** A product of 2^n MCX gates with the same target whose control patterns agree on a fixed set F and enumerate all patterns on the remaining n controls equals one MCX gate controlled on F only. The controls are orthogonal projectors, so the gates commute and sum to the identity on the free qubits.
- **Coherent permutation.** Where the control patterns lack this structure, a permutation of amplitudes restores it. It is built from amplitude swaps along shortest Hamming paths; the assignment of states is the solution of a linear sum assignment problem [24], and F is chosen so that the compressed gate acts on neighbouring qubits. Zero-padding states of the data register are free slack.
- **Insert by delete-everywhere.** Inserting an element into a few rows would need up to 2^n deletions; deleting it everywhere (one gate, by the theorem) and undoing the deletion in the target rows costs two MCX gates.

The permutation is applied before each compressed gate and undone afterwards (architecture **a**), or the ordering is tracked through the circuit and inverted once at the end (architecture **b**). The subnormalisation of the construction is the sum of the distinct entry magnitudes; whether $\alpha = \|A\|$ can be reached is open.

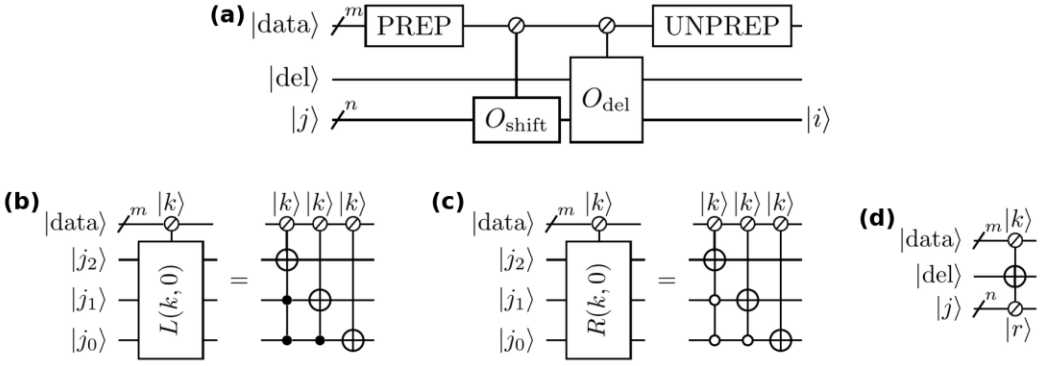


FIGURE 5. BLOCK-ENCODING CIRCUIT AND INDEX-MAPPING ORACLES
(a) Circuit structure: PREP, shift and delete oracles conditioned on the data register, UNPREP. (b, c) Left- and right-shift oracles for three matrix qubits as MCX gates. (d) Delete oracle.

3.2 Hardware resources

Three index-mapping circuits on three matrix qubits were transpiled to the heavy-hex layout of IBM Heron r3 and the square lattice of IBM Nighthawk r1 (Figure 6). Coherent permutation and compressed oracles reduce the two-qubit depth by 4.8× and 4.1× (shift, 173→36 and 144→35), 15.0× and 11.0× (delete in four rows, 704→47 and 515→47) and 6.8× and 7.6× (insert, 2,234→330 and 2,069→272). After optimization, the two layouts give almost the same depth, so the construction also removes most of the routing overhead. The pipeline was demonstrated on a complex tridiagonal matrix (six data values on three data qubits) and a structured real 32×32 matrix with 14 distinct values. The gain relies on repeated entries, which finite-difference operators on regular grids provide; the study covers transpilation of single operations only.

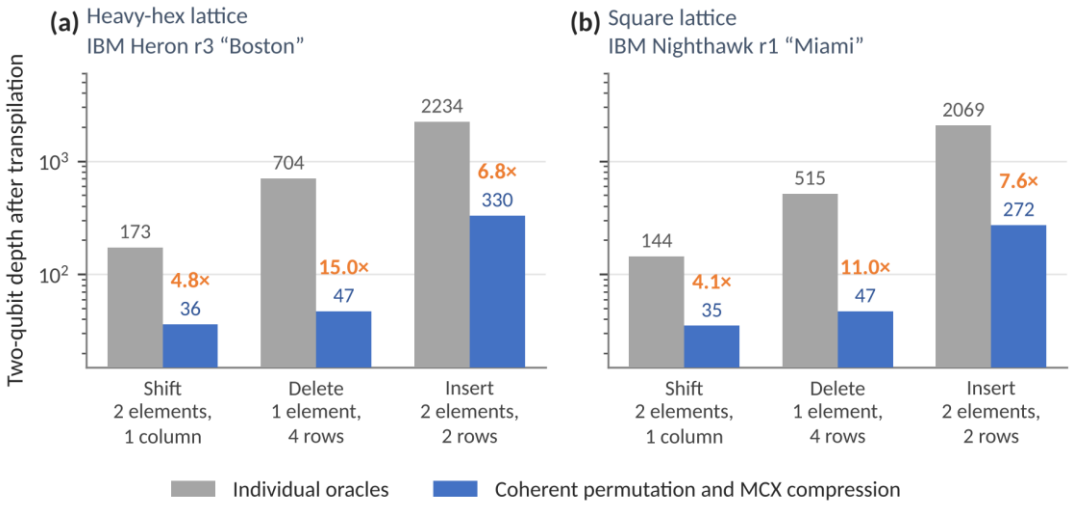


FIGURE 6. TWO-QUBIT DEPTH OF THREE INDEX-MAPPING CIRCUITS BEFORE AND AFTER OPTIMISATION
Heavy-hex (IBM Heron r3) and square-lattice (IBM Nighthawk r1) layouts, three matrix qubits. Grey: individual oracles; blue: coherent permutation with compressed oracles; orange: reduction factor.

4 Numerical comparison of algorithms in terms of circuit size

A solver is described by the circuits it runs, and each circuit by four numbers: qubits, depth, two-qubit gates and the number of executions (including measurement shots). The cost of a solver is roughly the number of executions times the size of the circuit that runs. Linear solvers (HHL, QSVT) run one deep circuit per time step, divided by a success probability; coherent time marching keeps the state in the register for the whole evolution (Section 8); variational algorithms run many shallow circuits, with cost evaluations per iteration, circuits per evaluation (one per operator term) and shots per circuit. This section compares the algorithms in these units on two problems: a linear heat-equation benchmark and a three-qubit Burgers problem.

4.1 Linear benchmark: HHL, VQLS and physics-informed circuits

Method	Error vs exact sim.	Qubits	Circuit depth	2Q gates	Circuit evaluations
Classical direct solve	4.0e−2 (floor)	–	–	–	–
HHL, 7 clock qubits [2]	4.6e−2	11	1,604	940	8
VQLS, local cost [6]	4.0e−2	3 + 1	35 (ansatz 8)	23 (ansatz 8)	11,406
QPINN [7]	6.9e−3 (best of 6)	4	17–29	24	12,552,904
Classical PINN [26]	2.1e−2	–	–	–	–

TABLE 2. RESOURCES ON THE HEAT-EQUATION BENCHMARK, 8 GRID POINTS, 8 TIME STEPS [25] VQLS depth and gates: the executed Hadamard test including the preparation of the right-hand side (bare ansatz in brackets). Circuit evaluations are noiseless evaluations, not number of shots. Medians over five random starts: 2.3e−2 (VQLS, four qubits), 1.5e−1 (QPINN), 1.6e−2 (classical PINN).

The project benchmark [25] solves $u_t = u_{xx}$ on $(0,1)$ with Dirichlet ends and a two-mode initial condition by backward Euler on $N = 8$ points (8 steps, condition number 2.77), noiselessly and with transpiled circuits (gate set u, cx). HHL uses 7 auxiliary ("clock") qubits for phase estimation with an idealised dense e^{iAt} and exact state preparation, so its depth is a lower bound; the Variational Quantum Linear Solver (VQLS) uses a four-layer real-amplitude ansatz with a local cost function; the QPINN is a function-encoded data re-uploading circuit; a 337-

parameter classical PINN is used as a control. All methods reach errors in the 1% range given by the discretization floor.

Quantum linear algebra cannot go below this floor, while the VQLS reproduces the linear system to 4.5×10^{-7} and still ends at 4.0×10^{-2}. Only the function-encoded circuit, which does not discretise, goes below it, but its median over random starts performs worse than that of the small classical network. The computational cost columns reverse their ranking: HHL runs 8 circuits of depth 1,604, while the QPINN has 1.3×10^{7} circuits of depth 17–29, because second derivatives by parameter shift need $O(m^2)$ circuits per collocation point.

Looking at scaling after circuit simplification (Figure 7), the dense-unitary HHL grows as $N^{1.95}$ in depth, the executed VQLS circuit as $N^{0.94}$ because it contains the preparation of the right-hand side, and the Pauli decomposition of the Laplacian needs N^2 Hadamard tests per cost evaluation where a five-term LCU needs 25. The gradient of a global VQLS cost falls by a factor of 0.22 -- 0.41 per added qubit at every depth; but a local cost is trainable only when the circuit is shallow [27, 28].

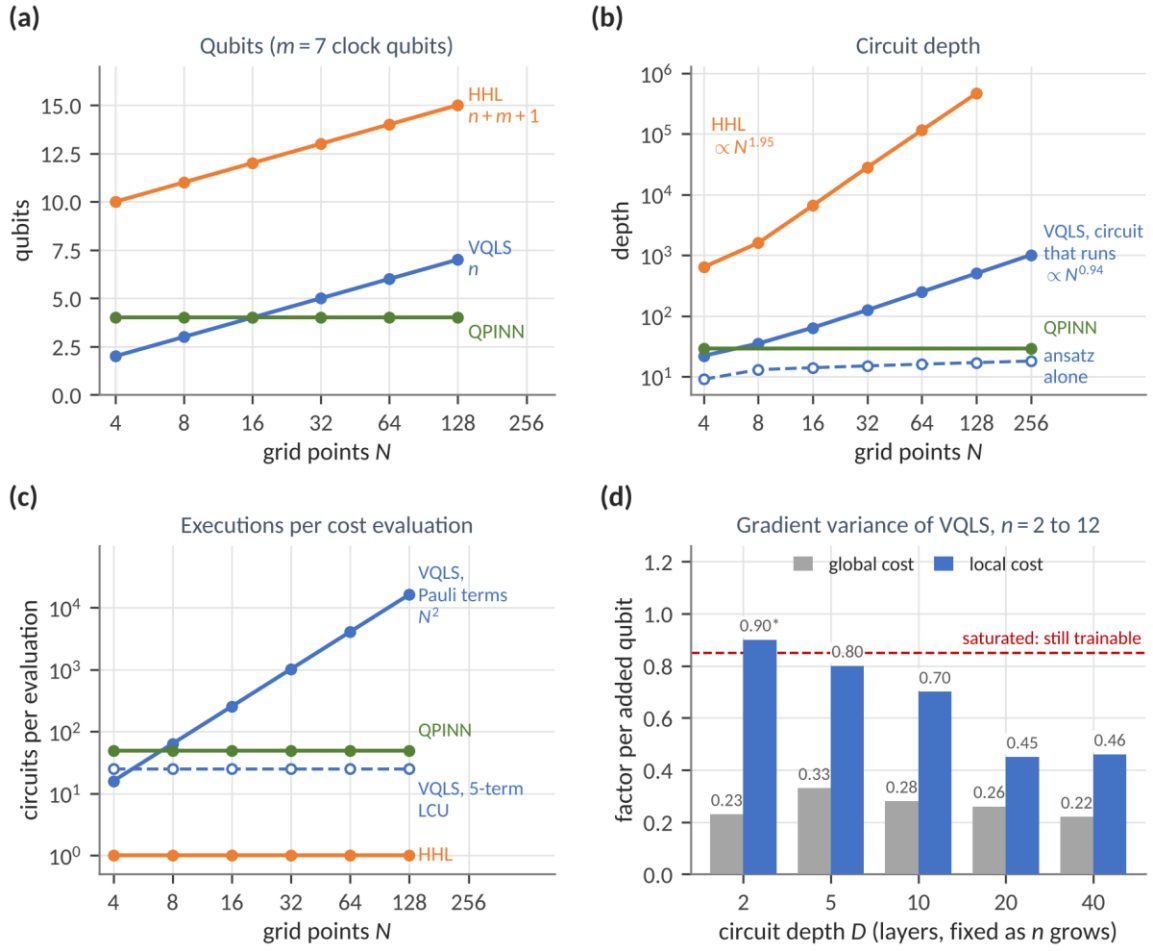


FIGURE 7. SCALING OF THE CIRCUIT SIZES WITH THE NUMBER OF GRID POINTS

(a) Qubits; (b) transpiled depth (HHL with dense unitary; VQLS ansatz and executed Hadamard test; QPINN); (c) circuits per cost evaluation; (d) decay factor of the VQLS gradient variance per added qubit at fixed depth D (for local and global cost).

4.2 Lubasch-type variational time stepping: trainability and resources

For the Burgers equation the cost function of the Lubasch-type algorithm [8] consists of three circuit families: $\langle U^T U \rangle$ and $\langle U^T A U \rangle$ for the linear terms and $\langle U^T D A U \rangle$ for the nonlinear term, where A is the adder and D the product with a duplicate of the state. The variance of the parameter-shift gradients sampled over structure-less circuit over

random parameters (Figure 8a) decreases roughly exponentially with the number of data qubits for all three families, and proportional to the nonlinearity of the term. Note that no warm start or TT structure is assumed so these are lower bounds. One and four ansatz layers behave alike: the observable of the Hadamard test is global, so depth does not remove the suppression. Gradient suppression, strongest for the nonlinear term, is thus the main scalability limit of the algorithm when it is not otherwise mitigated. Transpiled to the heavy-hex layout, the total depth of the nonlinear circuit grows from about 150 at (3 qubits, 1 layer) to about 3,500 at (9, 7) and its two-qubit depth from about 40 to about 1,000; the square lattice is shallower (Figure 8b).

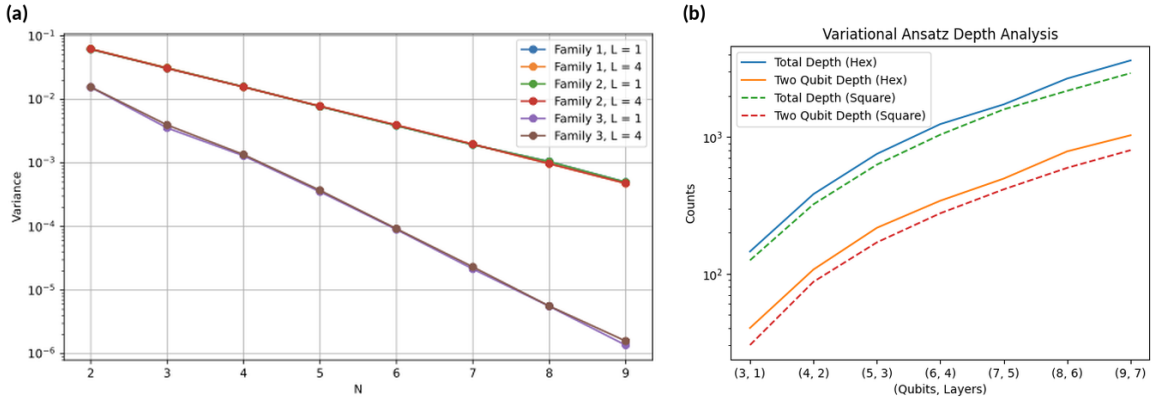


FIGURE 8. LUBASCH-TYPE VARIATIONAL CIRCUITS: GRADIENT VARIANCE AND HARDWARE RESOURCES

(a) Gradient variance of the three circuit families versus the number of data qubits for $L = 1$ and $L = 4$ layers. (b) Total depth and two-qubit depth of $\langle U^T DAU \rangle$ on heavy-hex (IBM Boston) and square-lattice (IBM Miami) layouts versus (qubits, layers).

4.3 Physics-informed quantum machine learning

In physics-informed quantum machine learning (PIQML) [29] a coordinate is encoded by a Chebyshev feature map, the solution is the expectation value of an observable $\hat{C}$ after a variational circuit, and the PDE residual and boundary terms form the loss. The gradient variance over 200 random parameter sets was computed for 3–6 qubits and 2–6 layers for three models: a reference circuit with the summed observable $\sum_j \quad Z_j$, a circuit with added entanglement and the same observable, and the reference circuit with the product observable $\bigotimes_j Z_j$ (Figure 9a). The variance decreases with the number of qubits in all three; added entanglement does not change it, whereas the product observable degrades it earlier. PIQML circuits are very shallow: a two-qubit depth of 6 at (3 qubits, 3 layers) and of about 70 at (9, 9), with total depths below 300 (Figure 9b).

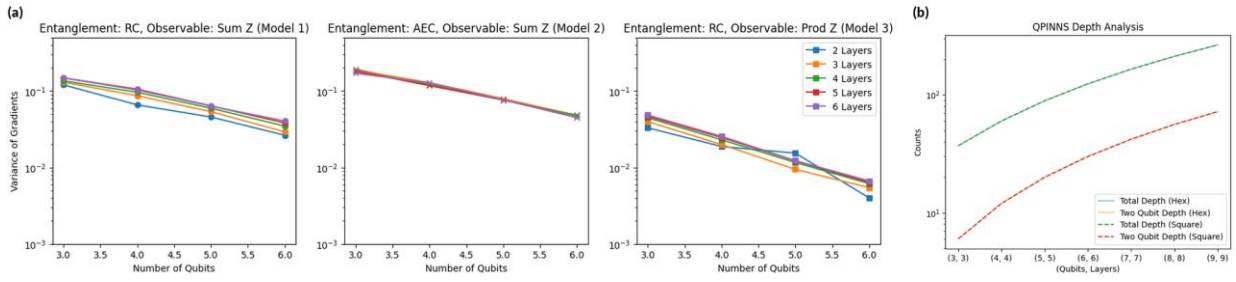


FIGURE 9. PHYSICS-INFORMED QUANTUM MACHINE LEARNING: GRADIENT VARIANCE AND HARDWARE RESOURCES

(a) Gradient variance versus number of qubits for the three models. (b) Total and two-qubit depth on heavy-hex and square-lattice layouts versus (qubits, layers).

4.4 Block encoding and QSVT

The quantum linear-systems pathway of [30] runs end to end from the differential equation through discretisation, Carleman linearisation for nonlinear problems, sparse block encoding and QSVT matrix inversion to observable estimation. For the 1D heat equation ($\nu = 0.01$, $\Delta t = 1$, Dirichlet left and Neumann right boundary) the QSVT

solution matches the exact implicit solution (Figure 10), and the polynomial degree needed for the inversion follows from the smallest singular value. For Burgers' equation ($\nu = 0.01$, seven grid points, Carleman order 2, $T = 0.3$) the same pathway reproduces evolution. Table 3 lists the resources; the circuits are deep because the QSVT polynomial of degree d calls the block encoding $O(d)$ times.

TABLE 3. HARDWARE RESOURCES OF THE BLOCK-ENCODING/QSVT PATHWAY [30]

	Heat, n = 3	Heat, n = 4	Burgers (Carleman 2), n = 5
Total qubits	7	8	11
Condition number / polynomial degree	8 / 559	12 / 1,275	8 / 559
Success probability (analytical / numerical)	0.0422 / 0.0422	0.0486 / 0.0482	0.0865 / 0.0865
Block encoding, 2q depth (heavy-hex / square)	311 / 255	550 / 396	5,687 / 3,892
QSVT, 2q depth (heavy-hex / square)	2.6e5 / 2.0e5	9.0e5 / 7.0e5	3.4e6 / 2.4e6

Heavy-hex: IBM Heron r3 (Boston); square: IBM Nighthawk r1 (Miami). 10^9 shots assumed for the numerical success probability.

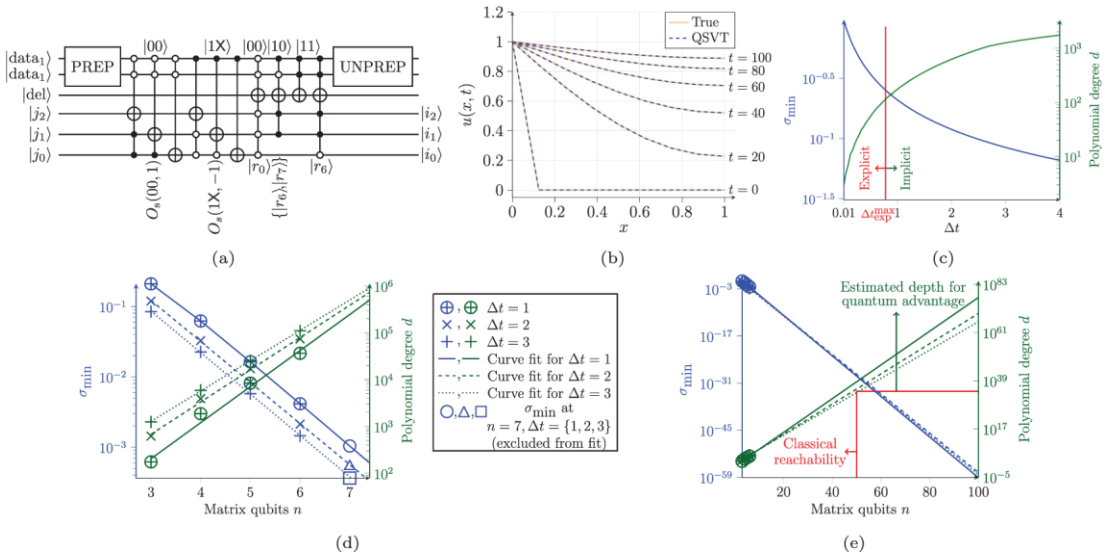


FIGURE 10. HEAT EQUATION BY BLOCK ENCODING AND QSVT

(a) Block-encoding circuit; (b) implicit QSVT solution with Dirichlet and Neumann boundaries against the exact solution; (c) minimum singular value and inverse polynomial degree versus time step; (d, e) scaling with the number of matrix qubits. From [30].

4.5 Which algorithm for devices of today

Table 4 compares the three approaches on one nonlinear problem, the Burgers equation at Re = 100 with about three qubits per field.

TABLE 4. ALGORITHMS FOR A THREE-QUBIT BURGERS PROBLEM (ANALYSIS BY A. SETTY, FZJ)

	PIQML	Lubasch-type VQA	Carleman + QSVT
Grid points	no restriction	8	7
Ansatz (qubits, layers)	(3, 3)	(3, 1)	–
Total qubits	3	7	11
Two-qubit depth, heavy-hex / square	6 / 6	40 / 30	3.4e6 / 2.4e6
Circuit evaluations	many (optimisation)	many (optimisation)	1 (success probability 0.087)

A circuit of g two-qubit gates runs without a gate error with probability about $(1-p)^g$. For $p = 10^{-2}$, the two-qubit fidelity of IBM Marrakesh, this is 8×10^{-5} for the 940 gates of HHL on 8 points, about 0.8 for the 23–24 gates of the VQLS and QPINN circuits, and zero in practice for the 10^6-deep QSVT circuits. This is why the project takes the Lubasch-type time stepping as its working choice for current devices: its circuits are shallow (two-qubit depth 30–40

here, adders of 31–73 two-qubit gates in Section 1), it needs no preparation of an arbitrary right-hand side because the previous solution is itself a circuit, and it treats nonlinear terms directly. PIQML circuits are even shallower but there is no solution guarantee, both classically and quantumly, while the QML does not have provable scaling advantage over the classical ML case. The price of both variational approaches is the number of circuit executions and an optimisation with cost function and gradients that can shrink with grid size (Figures 8 and 9), unless otherwise corrected as in Section 2. On fault-tolerant devices, where depth is cheap and executions are not, the coherent algorithms with the block encodings of Section 3 and the time marching KITE algorithm eventually become the better choice; where the break-even lies is an open question which depends on the error-correction overhead, which no source quantifies.

5 Variational quantum solution of the Grad–Shafranov equation

The Grad–Shafranov (GS) equation determines the magnetic flux $\psi(R, Z)$ of an axisymmetric plasma in ideal MHD equilibrium, as in tokamaks:

$$\Delta^*\psi = -\mu_0 R^2 \frac{dP}{d\psi} - F\frac{dF}{d\psi}, \qquad \Delta^* = \partial_R^2 - \frac{1}{R}\partial_R + \partial_Z^2. \tag{5}$$

It is a 2D nonlinear elliptic equation with position-dependent coefficients, a demanding step beyond the 1D benchmarks of the variational literature. In Mastorakis et al. [31], we apply the Lubasch-type algorithm to it with power-law profiles $P = P_0\psi^n$, $F = F_0\psi^m$: the linear Solov'ev case ($\Delta^*\psi = -R^2$) and a nonlinear case ($\Delta^*\psi = -2R^2\psi - \frac{3}{2}\psi^2$) on a rectangle with homogeneous, inhomogeneous and mixed Dirichlet–periodic boundaries. The R and Z indices occupy separate registers, so the shift operators keep their 1D adder form; only interior nodes are encoded, and Dirichlet edges are handled by neutralising terms that cancel the built-in periodicity of the shifts [32]. Nonlinear terms use bitwise products with copies of the state.

5.1 Picard and Weak cost functions

The **Picard** cost fits the trial state to a target built from one Gauss–Seidel update with the previous parameters, $C_p = \|\psi_{trial} - \psi_{target}\|^2$, refreshes the target after each sweep and eliminates the norm analytically. The **Weak** cost minimises the Ritz–Galerkin energy functional directly; because Δ^* is self-adjoint only with the weight $1/R$,

$$J(\psi) = \iint_\Omega \left[-\frac{1}{2}\psi\,\Delta^*\psi - \mu_0 R^2 P(\psi) - \frac{1}{2}F^2(\psi)\right]\frac{1}{R}\,dR\,dZ. \tag{6}$$

Weak has the same parameters in bra and ket and needs no outer loop, but up to 2^{2p} circuits per parameter update for a nonlinearity of degree p; Picard needs a constant number. Both use a real-amplitudes ansatz with 4 layers on $n_q = 4$ qubits (4×4 interior nodes, 20 parameters, 12 CNOT gates), the sequential optimiser SGEO [33], and 5×10^4 shots in noiseless simulation, with the infidelity against a finite-difference solution as a figure of merit.

5.2 Convergence and the hybrid strategy

The Weak cost is fast but Picard settles lower. Weak passes 10^{-2} after 40–80 parameter updates, Picard after 200–320; after 500–600 updates Picard is lower by factors 1.3–6.8, of which only 2.7 and 6.8 clearly exceed the run-to-run spread. A hybrid that starts with Weak and switches to Picard after three sweeps (about 60 updates), warm-starting trial and target, reaches 10^{-3} after about 260 updates instead of 440 for Picard alone, at the cost of a

transient of about 100 updates after the switch (Figure 11a). Weak alone reaches 10^{-3} after 180 updates but fluctuates, so the gain is one instance only; a switching criterion from the cost curve is still open.

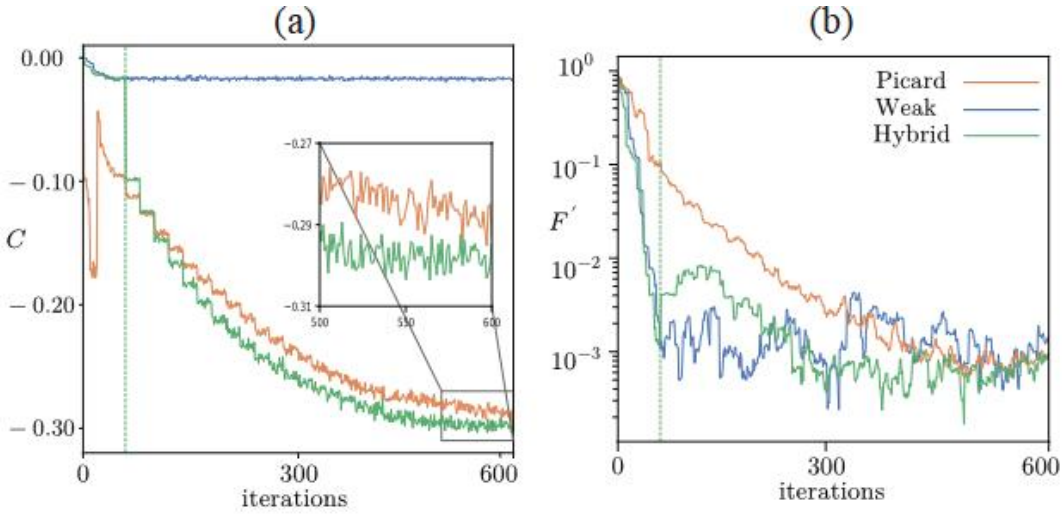


FIGURE 11. WEAK, PICARD AND HYBRID OPTIMISATION
(a) Cost and (b) infidelity for the nonlinear equation with boundary value 0.1 on all edges; the hybrid switches after 59 updates.

5.3 Resources of the boundary circuits

The neutralising circuits differ most between the formulations (Figure 12). Weak needs fewer qubits ($\frac{3}{2}n_q + 1$ against $2n_q + 1$, one state register instead of an old and a new one) but about 1.4 times the two-qubit gates and 1.5–1.6 times the depth of Picard at $n_q = 18$, because the $1/R$ weight requires multi-controlled Z gates. At $n_q = 18$ Picard needs 1,150 two-qubit gates on all-to-all trapped ions and 3,589 on the IBM Kingston coupling map; both formulations grow roughly as n_q^2. Trapped ions need one half to one third of the two-qubit gates of the superconducting map but with large clock speed overhead.

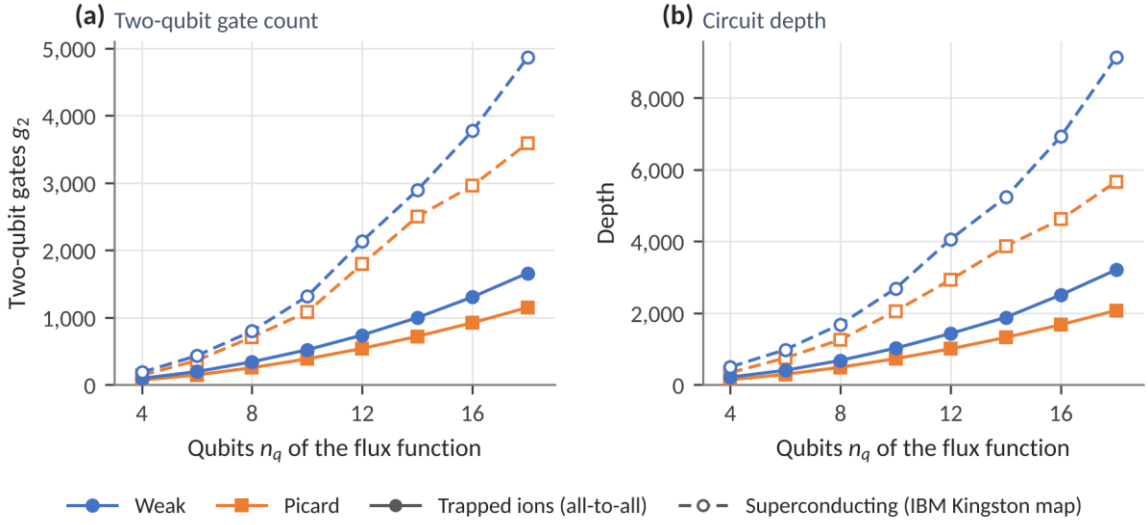


FIGURE 12. TWO-QUBIT GATES AND DEPTH OF THE NEUTRALISING CIRCUITS
(a) Two-qubit gate count and (b) depth versus the qubits of the flux function; solid: trapped ions, dashed: IBM Kingston map. Data of [31].

6 Cahn–Hilliard equation

The Cahn–Hilliard equation [34] describes how a binary mixture separates into two phases and how the domains then coarsen; it is the standard diffuse-interface model of multiphase flow. With constant mobility M and gradient-energy coefficient k it reads

$$\frac{\partial c}{\partial t} = M(\nabla^2 c^3 - \nabla^2 c - k\nabla^4 c). \tag{7}$$

Compared with Burgers it has a fourth derivative and a cubic nonlinearity, so it is a harder test of the same Lubasch-type algorithm. Da Costa Jesus [35] discretises it with central finite differences and one explicit Euler step, which gives the cost function

$$C = \|c_{t+dt} - (1 + dt\,\hat{O})c_t\|^2, \qquad \hat{O} = \nabla^2 D - \nabla^2 - k\nabla^4, \tag{8}$$

where D generates the diagonal of c^2 on auxiliary registers. With the adder (shift) A, the derivatives are

$$\nabla^2 = \frac{A + A^\dagger - 2I}{dx^2}, \qquad \nabla^4 = \frac{A^2 + (A^\dagger)^2 - 4A - 4A^\dagger + 6I}{dx^4}. \tag{9}$$

The cubic term needs two extra registers, so n problem qubits require $3n + 1$ qubits plus the auxiliary qubits of the adder (n-2). The cost then splits into overlaps $\langle\psi_{t+dt}|\psi_t\rangle$, $\langle\psi_{t+dt}|A|\psi_t\rangle$, $\langle\psi_{t+dt}|AA|\psi_t\rangle$, $\langle\psi_{t+dt}|D^2|\psi_t\rangle$ and $\langle\psi_{t+dt}|AD^2|\psi_t\rangle$ and their adjoint-shift counterparts, eight circuits in total, each evaluated by Hadamard interferometry (Figure 13). The operator D is a CNOT cascade followed by the inverse ansatz of the previous step on each auxiliary register; the shift acts on the main register and propagates through the CNOT ladder, so the adder is not needed on the auxiliary registers.

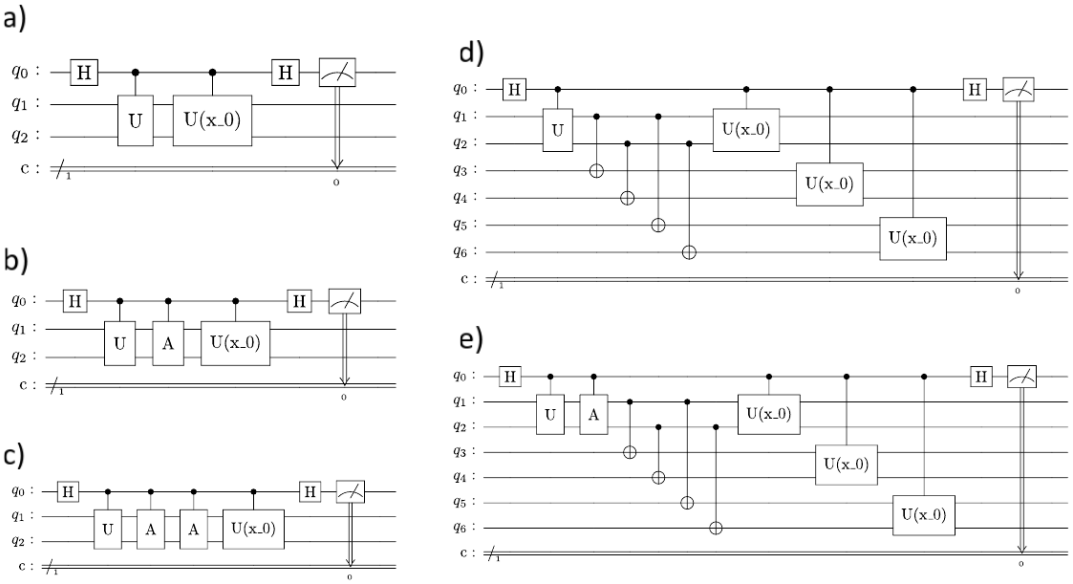


FIGURE 13. HADAMARD-TEST CIRCUITS FOR THE CAHN–HILLIARD COST FUNCTION

Two problem qubits. (a) $\langle\psi_{t+dt}|\psi_t\rangle$, (b) $\langle\psi_{t+dt}|A|\psi_t\rangle$, (c) $\langle\psi_{t+dt}|AA|\psi_t\rangle$, (d) $\langle\psi_{t+dt}|D^2|\psi_t\rangle$ and (e) $\langle\psi_{t+dt}|AD^2|\psi_t\rangle$; every circuit with A also has its counterpart with $A^\dagger$. From [35].

6.1 Results on two qubits

The test uses two problem qubits, periodic boundaries, the initial concentration $c = sin^2(\pi x/L)$, $M = 1$ and $k = 0.01$, a single-layer ansatz and SGEO [33], which needs two measurements of the eight circuits per parameter. The quality metric is the RMSE against an explicit finite-difference solution. The uncertainty of the cost at the solution falls with the number of shots, and the error grows with the time step, as expected for an explicit scheme (Table 6, Figure 14a, b). With 10,000 shots and $dt = 0.001$ the solution was advanced 30 time steps; the RMSE stays below

0.05 throughout (at most about 0.03) and does not drift upwards (Figure 14c), which shows that the algorithm handles the cubic nonlinearity and the fourth derivative.

TABLE 5. RMSE OF ONE CAHN–HILLIARD TIME STEP AGAINST FINITE DIFFERENCES [35]

Time step	500 shots	1,000 shots	5,000 shots	10,000 shots
$dt = 0.001$	0.042	0.024	0.014	0.009
$dt = 0.01$	0.066	0.085	0.016	0.020

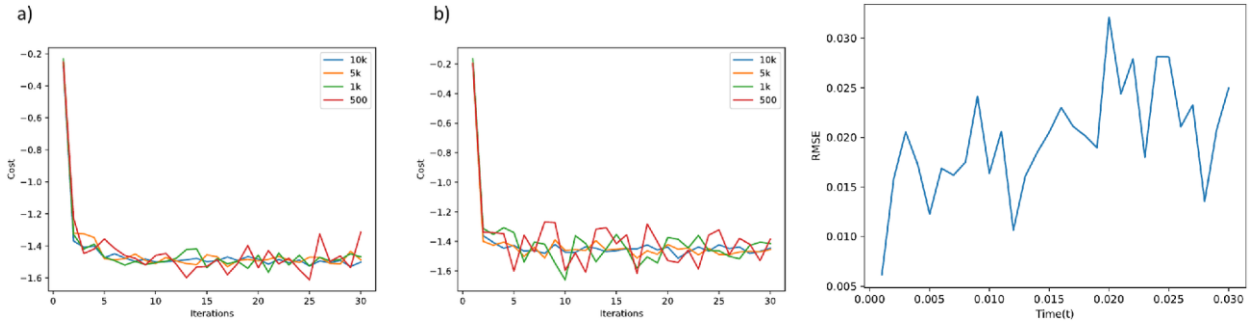


FIGURE 14. CAHN–HILLIARD RESULTS ON TWO QUBITS

(a, b) Cost versus SGEO iterations for $dt = 0.001$ and $dt = 0.01$ with 500 to 10,000 shots; (c) RMSE over 30 time steps for $dt = 0.001$ and 10,000 shots. From [35].

6.2 Gate requirements for larger grids

The deepest part of the algorithm, the nonlinear component with the adder, was transpiled for a 1D grid with a brick-layer ansatz of n qubits and $n - 1$ layers on all-to-all, heavy-hex and square-lattice connectivity (Figure 15). The two-qubit gate count grows faster than linearly; at 10 problem qubits it is about 5,700 on heavy-hex, 4,000 on the square lattice and 1,900 with all-to-all connectivity. The ansatz depth dominates the whole circuit, so more efficient ansätze are the main lever.

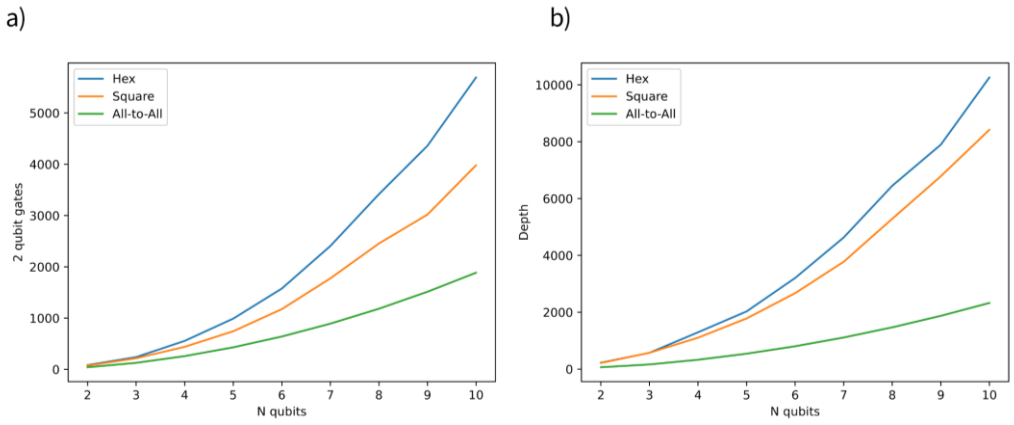


FIGURE 15. TWO-QUBIT GATES AND DEPTH OF THE NONLINEAR CAHN–HILLIARD CIRCUIT

(a) Two-qubit gates and (b) circuit depth versus problem qubits for heavy-hex, square-lattice and all-to-all connectivity; brick-layer ansatz with $n - 1$ layers. From [35].

6.3 2D implementation

We explore a 2 dimensional implementation for the Cahn-Hilliard equation. To that end, we consider an ansatz where each dimension is encoded into a specific set of qubits. Therefore we will require n_x qubits to represent the discretization in x and n_y qubits to represent the discretization in y. The discretization of the derivatives acts on each specific subset of qubits. This way, when computing derivatives across the x dimension we act with the adder

on the x qubits and vice versa. Given that we only have laplacians, we do not require cross dimensional derivatives and therefore the adder gate is only applied to a specific subset of qubits.

Taking a central finite difference discretization and periodic boundary conditions, we need to measure $\langle\psi_{t+dt}|\psi_t\rangle$, $\langle\psi_{t+dt}|A(x/y)|\psi_t\rangle$, $\langle\psi_{t+dt}|AA(x/y)|\psi_t\rangle$, $\langle\psi_{t+dt}|D^2|\psi_t\rangle$ and $\langle\psi_{t+dt}|A(x/y)D^2|\psi_t\rangle$ where the adder is applied either to the x qubits or the y qubits and their adjoint-shift counterparts, for a total of 14 circuits. The deepest circuit will correspond to the shift in the non-linearity and will scale with $3*(n_x+n_y)+1+n_x/n_y-2$ qubits.

We apply these circuits to a grid with 4 points in the x dimension, and 4 points in the y dimension ($n_x=n_y=2$) and further simplify the problem by considering k=0. We consider a brick layer ansatz with 4 layers, for a total of 17 parameters (16 angles and the extra norm). This means that the ansatz is actually bigger than the grid size which once again motivates the search for more efficient ansatze. For initial random data, we will observe the separation of the 2 phases into regions of positive value and of negative value. This can be seen in Figure 16.

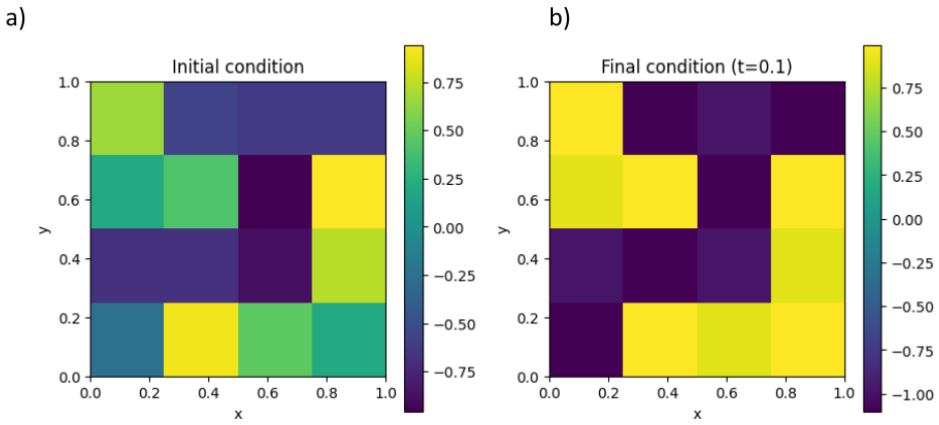


FIGURE 16. EVOLUTION OF THE 2D CAHN–HILLIARD EQUATION
(a) Initial condition (b) Evolution after 10 steps with dt=0.01. The phases are separating into 2 regions, one positive and one negative. From [35].

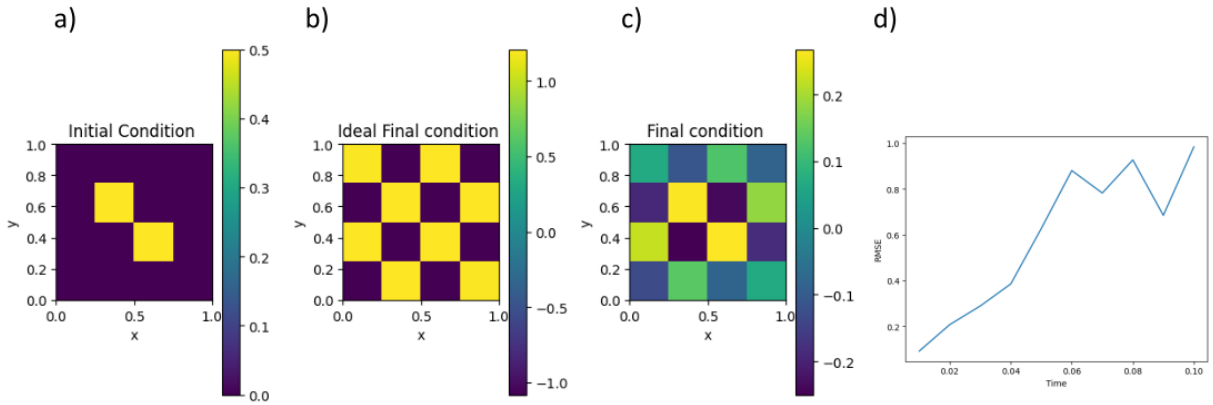


FIGURE 17. EVOLUTION OF THE 2D CAHN–HILLIARD EQUATION WITH SHOTS
(a) Initial condition (b) Ideal evolution after 10 steps with dt=0.01. The phases are separating into 2 regions, one positive and one negative. c) Evolution under shot noise. The phase separation is visible, although the norm is severely affected by the noise. d) Evolution of RMSE over time. As can be seen, the RMSE quickly grows before stabilizing. From [35].

We further investigate how the shot noise affects this evolution. We simplify the initial conditions by considering an empty grid with values of ½ in the corners of the inner square and 0 everywhere else. We consider 100k shots per measurement and minimize the cost using the SGEO optimizer[33] by performing 60 iterations. We maintain dt=0.01. The results can be seen in Figure 17.

Although the results from figure 17 show a strong impact of shot noise, the final solution can be measured directly from a quantum measurement. Because we have phase separation between the regions, the final solution will be well approximated by an equal superposition of amplitudes which differ by a phase factor (either +1 or -1). This can be measured directly by applying a hadamard gate on every qubit of the ansatz. The results of said measurement can be found in Figure 18. We can see that the highest probability corresponds to the ideal measurement, motivating a future hardware implementation.

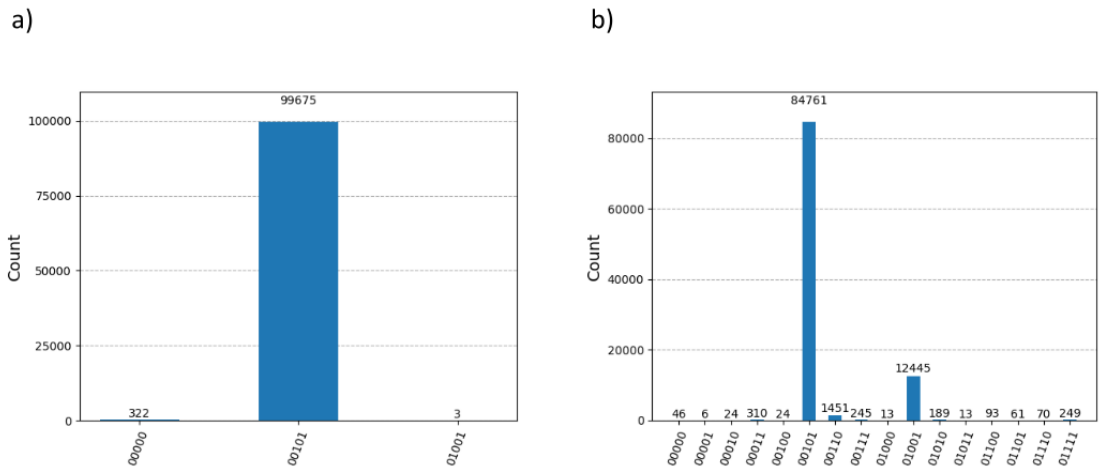


FIGURE 18. FINAL MEASUREMENT OF THE 2D CAHN–HILLIARD EQUATION WITH SHOTS
(a) Final measurement for the ideal parameters (b) Final measurement for the parameters extracted under shot noise. Although the shot noise has degraded the solution, we can still see that the most probable result, '00101', is the correct one. From [35].

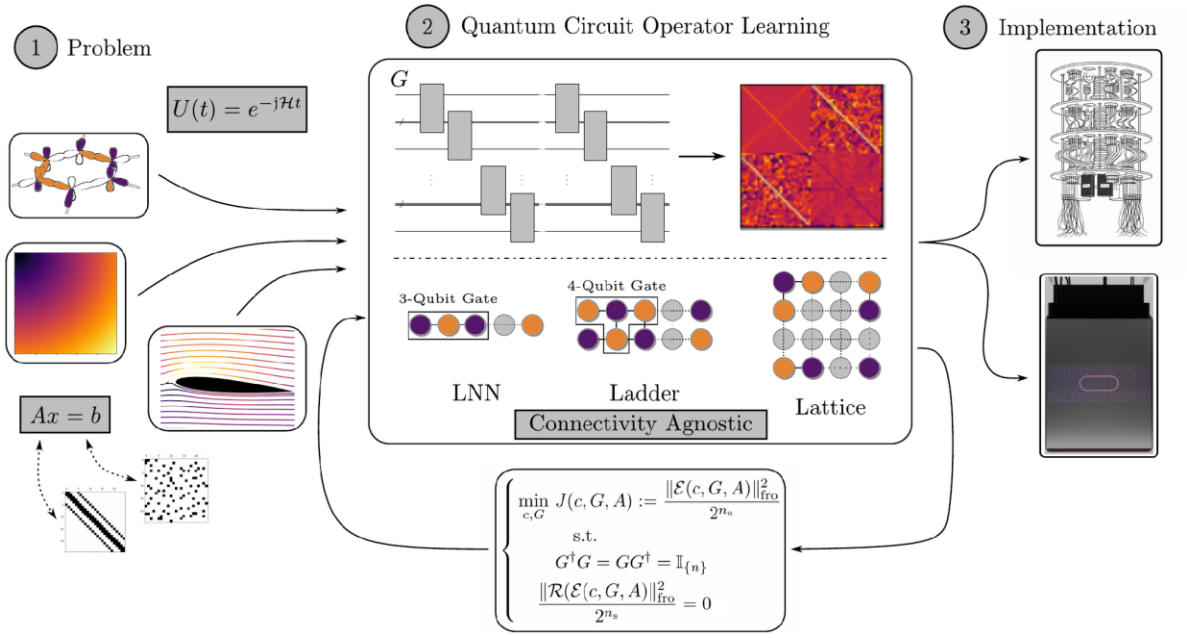
7 Operator-learning circuits

The algorithms in the sections above require input operators that can be decomposed into circuits. A generic n-qubit unitary needs exponentially many gates, and analytic constructions such as Trotter steps or structured block encodings ignore the connectivity of the device. Operator learning [36] makes the gate layout an input and learns the gates (Figure 16): the circuit G is a product of l layers of m arbitrary r-qubit gates on chosen qubit lists (chain, ladder, lattice or long-range, including the multi-qubit gates of neutral atoms and ions). A unitary target is approximated by G itself; any other operator is block-encoded with one ancilla, $A \approx c\, Tr_{anc}[PG]$, with success probability $p = \|PGP\|_F^2 / 2^{n_s}$. Training minimises

$$J = \frac{\|E\|_F^2}{2^{n_s}} + \rho\, \frac{\|R(E)\|_F^2}{2^{n_s}} + \mu c^2, \qquad E = A - c\, Tr_{anc}[PG], \tag{10}$$

where the five-point-stencil term R smooths the residual and suppresses outliers, which matters for amplitude encoding, and μ stabilises the success probability. Gates are updated by backpropagation with an SVD projection onto the unitary group and L-BFGS, in a hierarchical order (most significant qubits first). A training iteration costs $O\big(klm(2^{3r} + 2^{2n}2^{r})\big)$, exponential in the number of qubits, because the operator is held in full.

FIGURE 19. OPERATOR-LEARNING PIPELINE



A problem is discretised into an operator, learned as a circuit of local gates laid out for the connectivity of the device, and executed. Reproduced from [36].

7.1 Propagators and CFD operators

For the propagator of a transverse-field Ising chain with a longitudinal field (4–12 qubits), the learned circuit G_1 with the layout and gate count of a first-order Trotter step is 142 times more accurate at $\delta t = 0.01$ and gains one power of δt; G_2 (gate count of second-order Trotter) and G_3 (three-qubit gates) are 1.1×10^3 and 3.3×10^4 times more accurate, with errors flat in the system size (Figure 17). For the butadiene propagator on eight qubits the gain is only 22–28 %, which is attributed to less sparse generators. Two engineering operators test the non-unitary case with four-qubit gates and one ancilla (Figure 18, Table 7): the second-difference operator is reproduced to 10^{-12} for 4–5 system qubits with the intrinsic success probability 0.37, but the error jumps to 3×10^{-3} at six qubits even with hierarchical training (0.17 without) and to 2×10^{-2}–9×10^{-2} at nine; the dense 32×32 panel-method matrix of a NACA 0012 aerofoil [37], hard for tensor networks because of its many large singular values, is reproduced to 10^{-9}.

TABLE 6. LEARNED CIRCUITS: RESOURCES AND ACCURACY [36]

Operator	Qubits	Gates (CNOT estimate)	Error	Success prob.
Ising chain, G_1 / G_2 / G_3	8	7 / 14 two-qubit, 6 three-qubit (28 / 56 / 96)	142× / 1.1e3× / 3.3e4× below Trotter	1
Butadiene propagator	8	6 three- + 15 two-qubit (156)	22–28 % below Trotter	1
Second difference, $n_s = 6$, hierarchical / + regulariser	6 + 1	16 four-qubit (1,024)	3.1e−3 / 2.7e−4	0.35 / 0.07
NACA 0012 panel matrix, $n_s = 5$	5 + 1	12 four-qubit (768)	1.8e−9 / 1.1e−9	0.028 / 0.048

CNOT estimate 4^{r-1} per r-qubit gate (this report); the circuits were not transpiled. Error: relative Frobenius error; for the propagators the gain over Trotter at $\delta t = 0.01$.

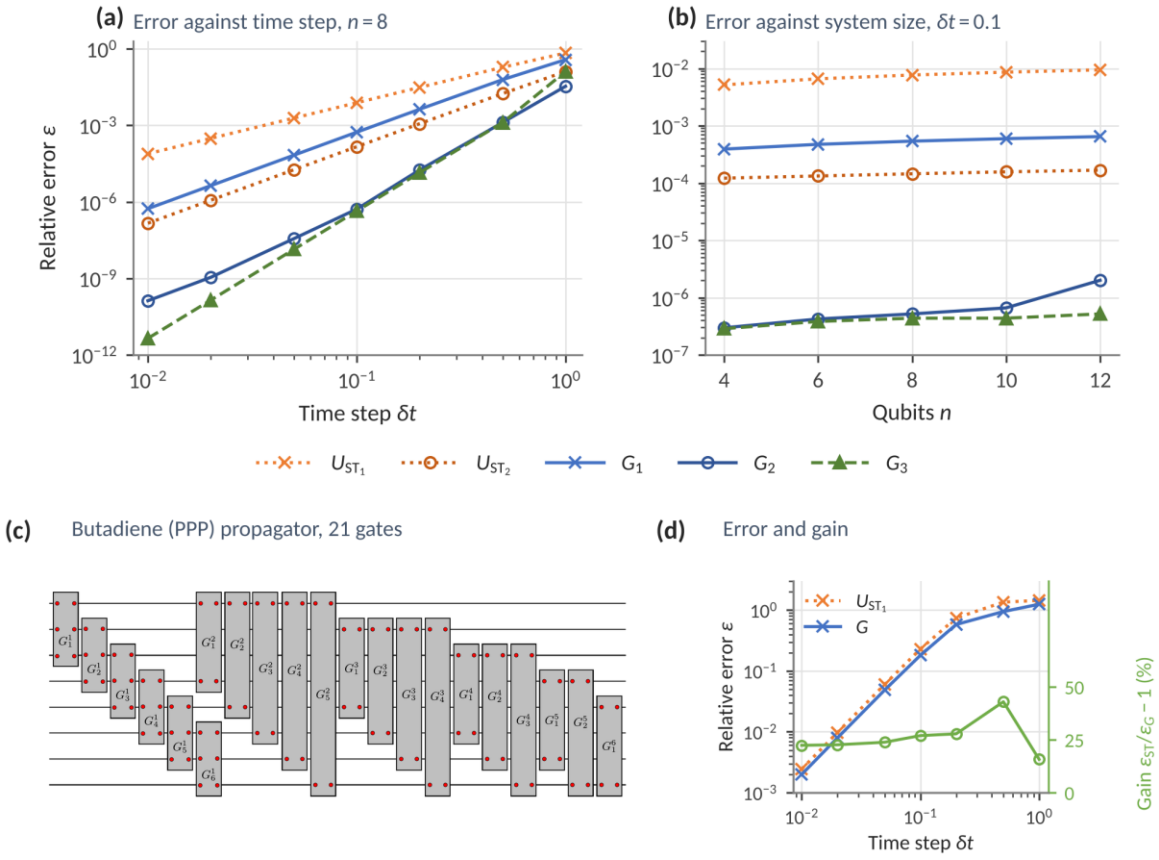


FIGURE 20. COMPRESSION OF PROPAGATORS

(a) Relative error of learned circuits G_1–G_3 and Trotter steps versus time step (Ising chain, $n = 8$); (b) versus system size at $\delta t = 0.1$; (c) learned butadiene ansatz; (d) butadiene error and gain. Replotted from [36].

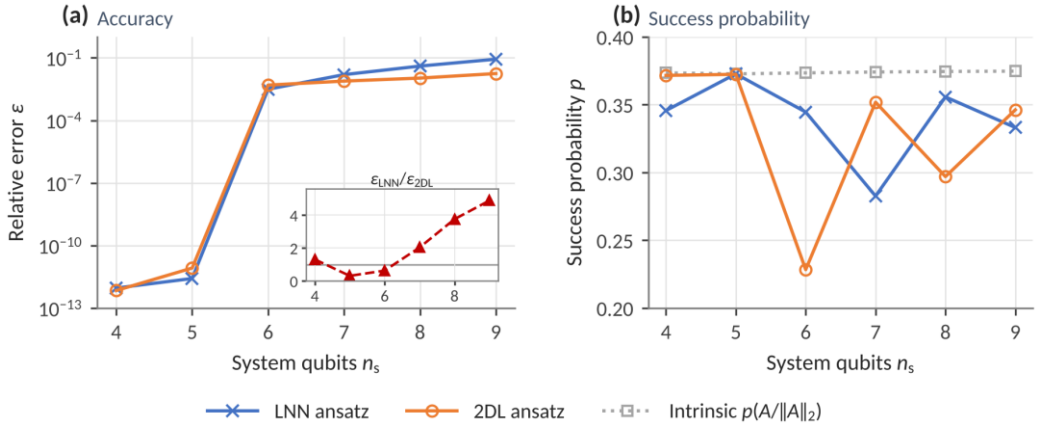


FIGURE 21. SECOND-DIFFERENCE OPERATOR LEARNED WITH FOUR-QUBIT GATES

(a) Relative error versus system qubits for chain (LNN) and 2D-lattice (2DL) layouts; (b) success probability and the intrinsic value. Replotted from [36].

Open questions: The training cost grows as 2^{2n} and scalable initialisation is still a concern; a tensor-network version would restrict the method to compressible operators. All results are exact, noiseless operators; the gates still have to be compiled to native gate sets, so the CNOT estimates are lower bounds. Accuracy and success probability trade off against each other.

8 Variational generalised linear phenomena

In Bengoechea et al. [43], we extend the variational algorithm of Lubasch and Over [8, 32] to the generic 1D second-order transport equation with five coefficients $a_1, \ldots, a_5$, which covers heat conduction with variable diffusivity, convection–diffusion, Burgers' equation and the wave equation. Each time step minimises a Ritz–Galerkin cost $J = \frac{1}{2}B(y,y) - F(y)$: the symmetric bilinear part B holds the implicit terms (diffusion, time derivatives), the linear part F the explicit convective term and the old time levels, which halves the terms of a least-squares cost. Each term is a Hadamard test that reads one ancilla. The field is amplitude-encoded with a classical norm; the ansatz is a real brick-layer circuit whose parameters grow linearly with n; a particle-swarm start and BFGS with warm starts do the optimisation.

8.1 Building blocks for band matrices

Every diagonal of a finite-difference band matrix becomes one circuit built from at most three quantum nonlinear processing units (QNPUs): a potential P (pointwise product with a field stored as a trained state; n Toffoli gates and n auxiliary qubits), an adder A (cyclic shift of the grid index; $n-2$ CNOT and $2n-2$ Toffoli gates with $n-2$ carry qubits) and an overlap. Upwind convection is written with two mask vectors of zeros and ones that select the backward or forward difference by the sign of the velocity, so UDS, LUDS and QUICK [44] need three, five and seven circuits, CDS two, and flux blending (FB) those of both schemes. Counting a Toffoli gate as 15 compiled gates, the blocks grow linearly with the number of qubits, $15n$ for P and $31n - 32$ for A, independent of the stencil width; the stencil enters only through the number of circuits (Figure 22, Table 9). The masks of nonlinear convection depend on the previous solution and must be obtained by tomography and training at every step, a cost outside these counts.

TABLE 7. CONVECTION SCHEMES AND THE COMPILED QNPU GATES OF THEIR CONVECTIVE TERMS [43]

Scheme	Order	Bandwidth	QNPU circuits	Gates, n = 4	n = 6	n = 12
UDS	1	3	3	484	758	1,580
CDS	2	3	2	304	488	1,040
LUDS	2	5	5	1,092	1,734	3,660
QUICK	3	7	7	1,516	2,402	5,060

Sums of the building-block counts (this report); the ansatz blocks, potential and mask preparation and boundary terms are not included.

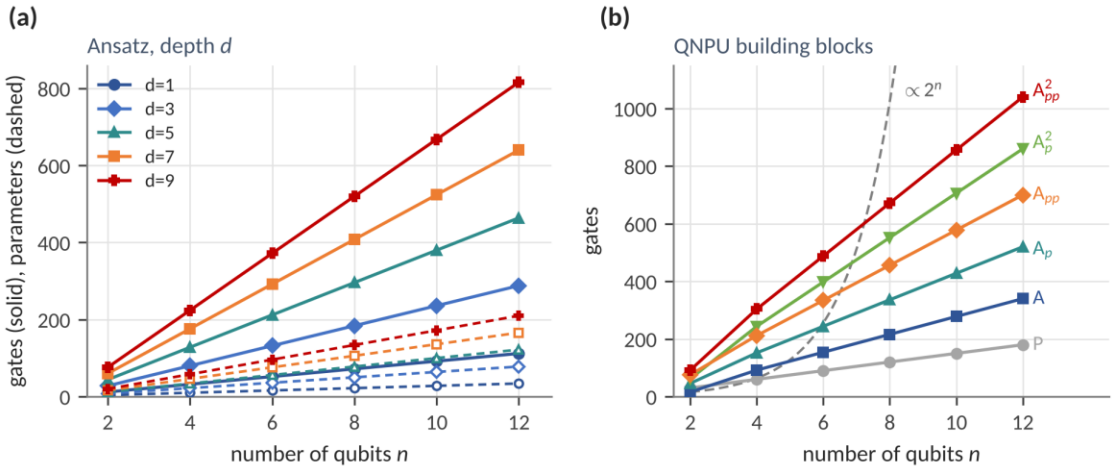


FIGURE 22. GATE COUNTS OF THE ANSATZ AND OF THE QNPU BUILDING BLOCKS

(a) Compiled gates (solid) and angles (dashed) of the brick-layer ansatz for depths 3–9; (b) compiled gates of the potential, the adder and their composites; dashed grey: growth of a classical sweep over 2^n points. Replotted from [43].

8.2 Test problems

Six 1D problems were emulated noiselessly with exact expectation values on 16 ($n = 4$) and 64 ($n = 6$) interior points and compared with finite differences of the same discretisation; the first is D2.1 case 3.1.1 [1]. With 4 qubits the time-dependent runs agree to 10^{-7}–10^{-5}; with 6 qubits only to 10^{-3}–2×10^{-2} (Table 10, Figure 23), which the authors attribute to the ansatz and the optimiser. In steady advection–diffusion the upwind schemes reproduce the classical solution at cell Péclet numbers 0.3 and 3; at 30 the oscillating classical CDS solution cannot be followed, and FB with $\beta = 0.5$ converges at Pe = 3 where CDS does not, for 24–101 % more optimisation steps than UDS. The optimisation is expensive: the steady UDS case needs 3,000–4,000 BFGS steps for 16 points, 11–18 times the classical reference, i.e. at least 6.5×10^5 circuit executions before shots. The largest circuit, the nonlinear UDS block at $n = 6$ has 23 qubits.

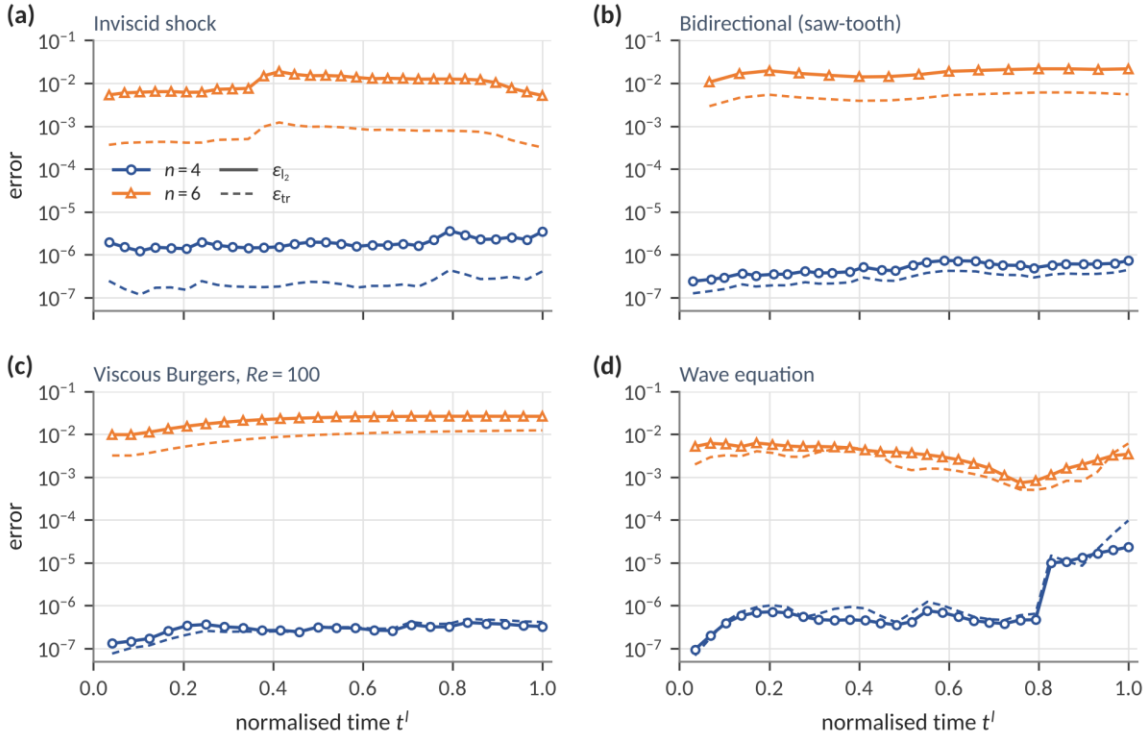


FIGURE 23. ERROR AGAINST THE CLASSICAL SOLUTION FOR FOUR UNSTEADY PROBLEMS

ℓ_2 error (solid) and trace distance (dashed) for $n = 4$ (circles) and $n = 6$ (triangles): (a) inviscid shock, (b) saw-tooth, (c) Burgers at Re = 100, (d) wave equation. Replotted from [43].

Open questions: The loss of accuracy going from 4 to 6 qubits, shot noise and hardware, masks without knowledge of the previous field, implicit upwind schemes and more than one dimension. The same building blocks appear also in the hardware run of Section 1, whose nonlinear products are formed in a QNPU block.

9 Optimizing circuit depth for VQA

Circuits for VQAs typically grow in depth by repeating a core structure called layer. Each additional layer introduces new variational parameters, potentially increasing the circuit's expressive power. However, greater depth also leads to accumulated noise, which can offset the benefits of these additional parameters. Consequently, reducing circuit depth while preserving the implemented quantum transformation has become an important research goal, both within and beyond VQAs.

9.1 Depth optimization of CNOT ladder circuits

A CNOT ladder circuit consists of a sequence of consecutive CNOT gates, where each gate shares a qubit with the next one. Ansatz circuits of VQAs admit multiple designs [45, 46], but a CNOT ladder structure is commonly used [47]. A descending 5-qubit CNOT ladder is shown in the first step of Fig.24

CNOT depth of a CNOT ladder consisting of n register qubits is equal to $n-1$ for a unitary implementation. Reducing the CNOT depth of a circuit and ideally keeping it constant is an active research field [48-51]. Instead of re-arranging quantum gates in the circuit of the same width, there is also the option of adding auxiliary qubits or, more drastically, adding mid-circuit measurements and classically controlled operations. In Ref. [52] CNOT depth is kept constant and equal to 2 regardless of the register qubit count, while CNOT count, and measurement count are smaller than existing approaches [48-51]. In Fig. 24 we diagrammatically present how a 5-qubit descending CNOT ladder is transformed from a unitary to a non-unitary circuit step by step.

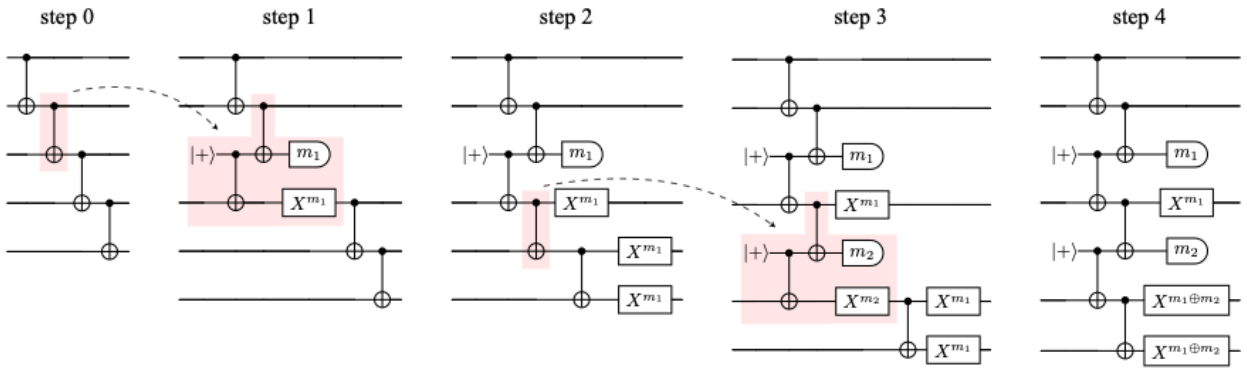


FIGURE 24. STEP BY STEP TRANSITION FROM A UNITARY TO A NON-UNITARY CIRCUIT

Step 1 shows the first substitution of a CNOT gate with its measurement-based equivalent. In step 2 the commutation of the conditional gate takes place. In step 3 the second substitution of CNOT is shown. Finally, in step 4 the commutation of the last gate takes place. Replotted from [52].

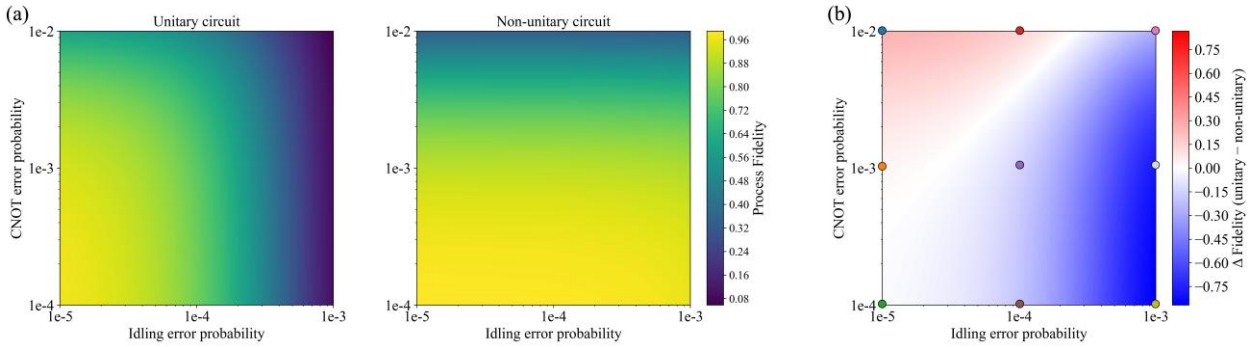


FIGURE 25. COMPARISON BETWEEN A UNITARY AND A NON-UNITARY CIRCUIT

In panel (a) two plots are presented for the unitary and the non-unitary version of the CNOT ladder. Both have the same range in terms of idling error probability and CNOT error probability: 1e-5 to 1e-3 and 1e-4 to 1e-2, respectively. The colorbar corresponds to both plots and indicates the lower bound of process fidelity of the entire circuit, which appears to be at its highest for the unitary circuit when idling error probability is low, and for the non-unitary circuit when CNOT error probability is low. In panel (b) the difference of the process fidelities is shown, Δ Fidelity, where positive (red) indicates an advantage for the unitary circuit and negative (blue) for the non-unitary one. Replotted from [52].

In Fig. 25, we compare the conventional CNOT ladder, which is represented as a unitary transformation, shown in the first step of Fig. 24, with the corresponding non-unitary transformation in the last step. In particular, we consider the case of 50 register qubits and present in panel (a) two plots whose color indicates the lower bound of the process fidelity of the entire circuit. On the x-axis we have the idling error probability p_{idle} ranging from 10^{-5} to 10^{-3}, while on the y-axis the CNOT error probability p_{CNOT} ranging from 10^{-4} to 10^{-2}. We assume that each conditional single-qubit gate has an error probability equal to $p_{\text{con}} = p_{\text{CNOT}}/10$. Following the same reasoning, we assume that the qubit initialization and measurement error probabilities are also one order of magnitude smaller than p_{CNOT}, i.e., $p_{\text{in}} = p_{\text{CNOT}}/10$ and $p_{\text{meas}} = p_{\text{CNOT}}/10$. We observe that the unitary circuit performs the best for low idling error, while the opposite is true for the non-unitary circuit. In order to compare them quantitatively, we present in Fig. 25(b) another plot but now the colorbar corresponds to the Δ Fidelity, i.e., the process fidelity of the unitary circuit minus the non-unitary one. Positive values, represented with red color, imply a better performance for the unitary circuit, and negative values, represented with blue color, imply a better performance for the non-unitary ones.

9.2 Minimum Toffoli depth for the multi-controlled Toffoli gate via teleportation

Quantum algorithms often rely on complex operations that in practice need to be decomposed into simpler gate sets to be implemented. An example of such a complex operation is the multi-controlled Toffoli (MCT) gate, which is the generalization of the Toffoli gate over more than three qubits. MCT gates appear on several algorithms, VQA included, e.g., through adder circuits [53].

Several MCT decompositions have been suggested in the literature. In the work by Nie *et al.* [54] the MCT gate is decomposed on a circuit with a Toffoli depth equal to $20\log_2 n$ requiring a single ancilla qubit. Another MCT decomposition comes from the work by Khattar and Gidney [55], where the Toffoli depth is equal to $2n - 3$ for a single ancilla qubit and approximately $4\log_2 n$ with two ancilla qubits. In the more recent work by Dutta *et al.* [56], more decompositions were derived, alongside a lower bound of $\lceil \log_2(n) \rceil$ for the Toffoli depth, which, we show in Ref. [57] that it can be surpassed. In Fig. 25(a) we present how the Toffoli depth scales over a range of twenty control qubits for different decompositions, and in Fig. 25(b) the corresponding Toffoli count. In Fig. 25(c) we compare the additional qubits, also known as ancilla qubits, required for each implementation.

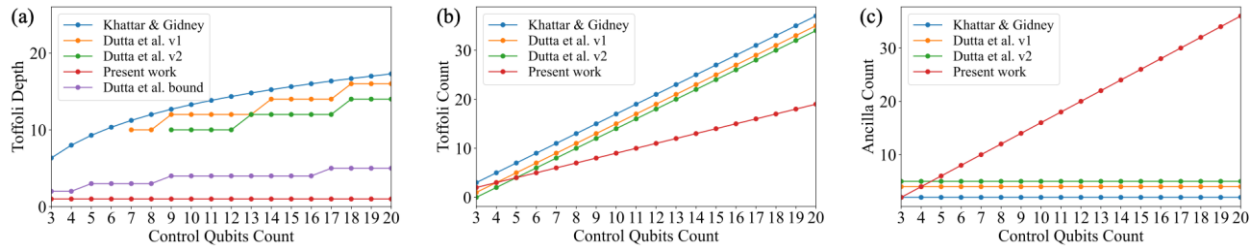


FIGURE 25. COMPARISON AMONG MCT DECOMPOSITIONS

In panels (a)-(c) we plot the Toffoli depth, the Toffoli count, and the ancilla count, respectively, against the number of control qubits in an MCT gate decomposition. The color code is the same for all three panels. With blue we have the corresponding values for the decomposition by Khattar and Gidney [55] With orange the corresponding values for the protocol derived by Dutta et al. [56] that uses four ancilla qubit and with green the corresponding protocol by the same authors that uses five ancilla qubits. With red we have the corresponding values for the teleportation-based protocol proposed in this work. With purple there is a trend appearing only on panel (a) corresponding to the lower bound of Toffoli depth derived by Dutta et al. [56]. Replotted from [57].

The MCT decomposition proposed in Ref. [57] is based on gate teleportation [58, 59]. In particular, the gate teleportation protocol is consecutively applied until all local gates to correspond to Toffoli gates. In Fig. 26, the MCT gate with 5 register qubits is presented under this decomposition in tow paradigms: the conventional monolithic quantum computing paradigm and the distributed quantum paradigm.

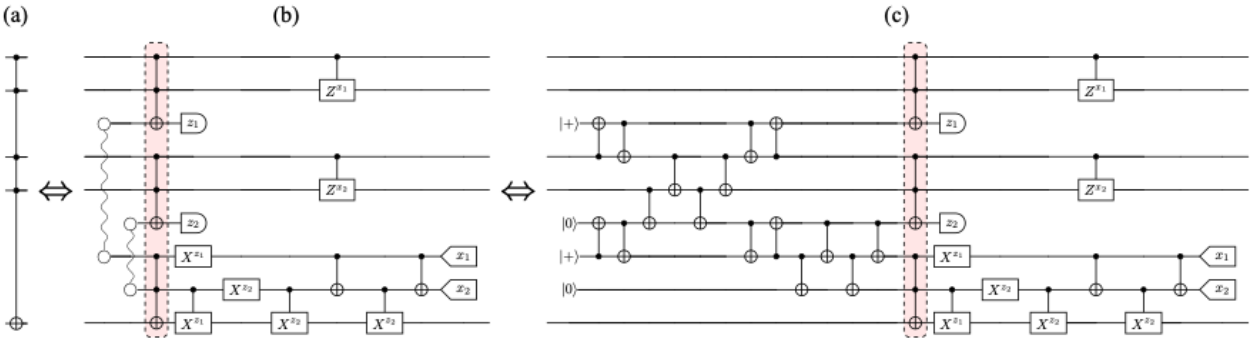


FIGURE 26. TELEPORTATION-BASED DECOMPOSITION OF AN MCT GATE.

An MCT gate with 5 register qubits: (a) as a unitary operation; (b) through the teleportation-based decomposition in the DQC paradigm and; (c) through the teleportation-based decomposition in the MQC paradigm. Replotted from [57].

Open questions: The effects of coherent errors and realistic measurement and feedforward latencies on the fidelity gains from depth reduction require further investigation. Additionally, entanglement distribution, ancilla overhead, and idling noise should be assessed together to determine whether shallow depth translates into reduced execution time and improved fidelity in complete CFD algorithms.

10 Summary

While no algorithm yet solves large fluid-dynamics problems of practical size, this report has mapped out some of the main advances and limitations at the gate-level picture. Variational circuits are shallow and run on a superconducting processor using a large number of measurement shots; coherent algorithms have a stated asymptotic cost but need depths that only fault-tolerant devices may be able to run. The cost of the variational route trades away circuit depth to circuit executions and optimisation without convergence guarantee. Table 10 states the main result and the limitations of each section.

TABLE 8. RESULT AND LIMIT OF EACH SECTION

Sect.	Main result	Limiting factors
1	Viscous Burgers advanced five steps on IBM Marrakesh, final MSE 2.1 %; adder $13N - 14 \rightarrow 7N - 11$ two-qubit gates; norm within 1 % with mitigation	Few qubits, one dimension, one device
2	Success probability of the nonlinear step 15–30 % on up to 10^7 points instead of $< 10^{-4}$	Classical TT copy whose cost grows with Re; noiseless, one step
3	Index-mapping circuits 4–15× shallower on heavy-hex and square lattices	Subnormalisation unchanged; three circuits, transpilation only
4	Variational circuits: two-qubit depth 30–40 for three-qubit Burgers vs $2.4–3.4 \times 10^6$ for QSVT; HHL depth 1,604 on 8 points	Gradients shrink with qubits, most for the nonlinear term; many executions
5	Grad–Shafranov on four qubits to infidelity about 10^{-3}; hybrid Weak/Picard strategy	Ansatz larger than the problem; noiseless
6	Cahn–Hilliard (cubic, fourth order) on two qubits, RMSE < 0.05 over 30 steps with 10,000 shots, 2 Dimensions	Shot noises; explicit step; gates grow quickly with n
7	Learned circuits up to 3×10^4 times more accurate than Trotter; CFD operators to 10^{-9}–10^{-12} precision	Training cost $\propto 2^{2n}$; error grows beyond six qubits
8	QNPU blocks linear in n ($15n$, $31n - 32$); errors 10^{-7}–10^{-5} at four qubits	10^{-3}–2×10^{-2} at six qubits; $\geq 10^5$ executions for a steady case
9	Circuit depth optimization for CNOT and MCT gates	Extra ancilla qubits and feed-forward measurements.

Open issues. (i) Trainability: gradient variances fall exponentially with qubit number, and the variational errors grow from 4 to 6 qubits (Sections 4 and 9). (ii) Shots and noise: apart from one hardware study all results are noiseless, and the mitigation assumes global depolarising noise. (iii) Fault-tolerant cost: state preparation, readout, error correction and the Hamiltonian simulation of Section 8 are not incorporated. (iv) Platforms: no neutral-atom resource count yet. (v) Benchmarks have only three D2.1 cases are used so far, all in one or two dimensions.

Next steps. Running the Grad–Shafranov, Cahn–Hilliard and QNPU circuits on hardware with the optimisations and mitigation of Section 1 and count shots per time step; combining the diagonal block encoding of Section 2 with the coherent permutation of Section 3; extending the variational solvers to two dimensions and to further D2.1 cases (unsteady heat conduction with boundaries, curvilinear cylinder flow, Rayleigh–Bénard convection).

10 References

[1] QCFD consortium (TUHH, ENG, UHH), Deliverable D2.1: Core benchmark CFD set, revision 0 (2026); datasets at www.fdr.uni-hamburg.de/communities/qcfd/.

[2] A. W. Harrow, A. Hassidim and S. Lloyd, Quantum algorithm for linear systems of equations, Phys. Rev. Lett. 103, 150502 (2009).

[3] A. Gilyén, Y. Su, G. H. Low and N. Wiebe, Quantum singular value transformation and beyond: exponential improvements for quantum matrix arithmetics, Proc. 51st ACM STOC, 193–204 (2019).

[4] J. Gonzalez-Conde, D. Lewis, S. S. Bharadwaj and M. Sanz, Quantum Carleman linearization efficiency in nonlinear fluid dynamics, Phys. Rev. Res. 7, 023254 (2025).

[5] H. Krovi, Improved quantum algorithms for linear and nonlinear differential equations, Quantum 7, 913 (2023).

[6] C. Bravo-Prieto, R. LaRose, M. Cerezo, Y. Subaşı, L. Cincio and P. J. Coles, Variational quantum linear solver, Quantum 7, 1188 (2023).

[7] O. Kyriienko, A. E. Paine and V. E. Elfving, Solving nonlinear differential equations with differentiable quantum circuits, Phys. Rev. A 103, 052416 (2021).

[8] M. Lubasch, J. Joo, P. Moinier, M. Kiffner and D. Jaksch, Variational quantum algorithms for nonlinear problems, Phys. Rev. A 101, 010301(R) (2020).

[9] D. Jaksch, P. Givi, A. J. Daley and T. Rung, Variational quantum algorithms for computational fluid dynamics, AIAA J. 61, 1885–1894 (2023).

[10] A. Javadi-Abhari et al., Quantum computing with Qiskit, arXiv:2405.08810 (2024).

[11] J. D. da Costa Jesus, A. Setty, T. Calarco, D. Jaksch, F. A. Cárdenas-López and F. Motzoi, Time evolution of nonlinear dynamics on a quantum processor, arXiv:2608.13041 (2026).

[12] D. Maslov, Advantages of using relative-phase Toffoli gates with an application to multiple control Toffoli optimization, Phys. Rev. A 93, 022311 (2016).

[13] P. Siegl, G. S. Reese, T. Hashizume, N.-L. van Hülst and D. Jaksch, Tensor-programmable quantum circuits for solving differential equations, Phys. Rev. Res. 8, 013052 (2026).

[14] A. Sarma, T. W. Watts, M. Moosa, Y. Liu and P. L. McMahon, Quantum variational solving of nonlinear and multidimensional partial differential equations, Phys. Rev. A 109, 062616 (2024).

[15] P. Siegl, N.-L. van Hülst, M. Mandelt Buxadé, T. Hashizume and D. Jaksch, Efficient treatment of non-linearity in quantum computational fluid dynamics using hybrid tensor networks, manuscript (2026); dataset doi:10.25592/uhhfdm.19340.

[16] A. Termanova et al., Tensor quantum programming, New J. Phys. 26, 123019 (2024).

[17] A. M. Childs and N. Wiebe, Hamiltonian simulation using linear combinations of unitary operations, Quantum Inf. Comput. 12, 901–924 (2012).

[18] I. V. Oseledets, Constructive representation of functions in low-rank tensor formats, Constr. Approx. 37, 1–18 (2013).

[19] N.-L. van Hülst et al., Quantum-inspired simulation of 2D turbulent Rayleigh–Bénard convection, manuscript (2026).

[20] B. D. Clader, A. M. Dalzell, N. Stamatopoulos, G. Salton, M. Berta and W. J. Zeng, Quantum resources required to block-encode a matrix of classical data, IEEE Trans. Quantum Eng. 3, 1–23 (2023).

[21] C. Sünderhauf, E. Campbell and J. Camps, Block-encoding structured matrices for data input in quantum computing, Quantum 8, 1226 (2024).

[22] D. Camps, L. Lin, R. Van Beeumen and C. Yang, Explicit quantum circuits for block encodings of certain sparse matrices, SIAM J. Matrix Anal. Appl. 45, 801–827 (2024).

[23] A. Setty, Block encoding of sparse matrices via coherent permutation, Phys. Rev. Res. 8, 033282 (2026).

[24] D. F. Crouse, On implementing 2D rectangular assignment algorithms, IEEE Trans. Aerosp. Electron. Syst. 52, 1679–1696 (2016).

[25] QCFD WP6, Quantum and classical solvers for the 1D heat equation: complete analysis and response to review, internal project study (2026).

[26] M. Raissi, P. Perdikaris and G. E. Karniadakis, Physics-informed neural networks, J. Comput. Phys. 378, 686–707 (2019).

[27] M. Cerezo, A. Sone, T. Volkoff, L. Cincio and P. J. Coles, Cost function dependent barren plateaus in shallow parametrized quantum circuits, Nat. Commun. 12, 1791 (2021).

[28] J. R. McClean, S. Boixo, V. N. Smelyanskiy, R. Babbush and H. Neven, Barren plateaus in quantum neural network training landscapes, Nat. Commun. 9, 4812 (2018).

[29] A. Setty, R. Abdusalamov and F. Motzoi, Self-adaptive physics-informed quantum machine learning for solving differential equations, Mach. Learn.: Sci. Technol. 6, 015002 (2025).

[30] A. Setty, A quantum linear systems pathway for solving differential equations, J. Phys. A: Math. Theor. 59, 185303 (2026).

[31] E. Mastorakis, M. Umer and D. G. Angelakis, Quantum variational approaches to plasma equilibrium: cost function design for the Grad–Shafranov equation, manuscript (2026).

[32] P. Over, S. Bengoechea, T. Rung, F. Clerici, L. Scandurra, E. de Villiers and D. Jaksch, Boundary treatment for variational quantum simulations of partial differential equations on quantum computers, Comput. Fluids 288, 106508 (2025).

[33] M. Umer, E. Mastorakis and D. G. Angelakis, Efficient estimation and sequential optimization of cost functions in variational quantum algorithms, Quantum Sci. Technol. 10, 035022 (2025).

[34] J. W. Cahn and J. E. Hilliard, Free energy of a nonuniform system. I. Interfacial free energy, J. Chem. Phys. 28, 258–267 (1958).
[35] J. D. da Costa Jesus, Cahn–Hilliard equation with the variational algorithm, FZJ internal note (October 2026).
[36] P. Over, S. Bengoechea, L. Borello Busilacchi, M. Kiffner, T. Rung and A. A. Michailidis, Operator learning for efficient quantum computation, arXiv:2606.20184 (2026).
[37] J. L. Hess and A. M. O. Smith, Calculation of potential flow about arbitrary bodies, Prog. Aerosp. Sci. 8, 1–138 (1967).
[38] P. Over, S. Bengoechea, P. Brearley, S. Laizet and T. Rung, Quantum algorithm for the advection–diffusion equation by direct block encoding of the time-marching operator, Phys. Rev. A 112, L010401 (2025).
[39] D. Fang, L. Lin and Y. Tong, Time-marching based quantum solvers for time-dependent linear differential equations, Quantum 7, 955 (2023).
[40] P. Brearley and S. Laizet, Quantum algorithm for solving the advection equation using Hamiltonian simulation, Phys. Rev. A 110, 012430 (2024).
[41] S. Bengoechea, P. Over and T. Rung, Quantum time-marching algorithms for solving linear transport problems including boundary conditions, Int. J. Numer. Methods Eng. 127, e70326 (2026).
[42] G. H. Low and I. L. Chuang, Hamiltonian simulation by qubitization, Quantum 3, 163 (2019).
[43] S. Bengoechea, P. Over, D. Jaksch and T. Rung, Toward variational quantum algorithms for generalized linear and nonlinear transport phenomena, AIAA J. 64, 585–604 (2026).
[44] B. P. Leonard, Simple high-accuracy resolution program for convective modelling of discontinuities, Int. J. Numer. Methods Fluids 8, 1291–1318 (1988).
[45] A. Kandala, A. Mezzacapo, K. Temme, M. Takita, M. Brink, J.M. Chow, J.M. Gambetta, Hardware-efficient variational quantum eigensolver for small molecules and quantum magnets, Nature 549 (7671) (2017) 242–246.
[46] S. Sim, P.D. Johnson, A. Aspuru–Guzik, Expressibility and entangling capability of parameterized quantum circuits for hybrid quantum–classical algorithms, Adv. Quantum Technol. 2 (12) (2019).
[47] M.R. Geller, Z. Zhou, Efficient error models for fault-tolerant architectures and the Pauli twirling approximation, Phys. Rev. A 88 (2013) 012314.
[48] Y. Quek, E. Kaur, M.M. Wilde, Multivariate trace estimation in constant quantum depth, Quantum 8 (2024) 1220.
[49] E. Bäumer, V. Tripathi, D.S. Wang, P. Rall, E.H. Chen, S. Majumder, A. Seif, Z.K. Minev, Efficient long-range entanglement using dynamic circuits, PRX Quantum 5 (2024) 030339.
[50] E. Bäumer, S. Woerner, Measurement-based long-range entangling gates in constant depth, Phys. Rev. Res. 7 (2025) 023120.
[51] M. Remaud, V. Vandaele, Ancilla-free quantum adder with sublinear depth, in: Reversible Computation, Springer Nature Switzerland, 2025, p. 137–154.
[52] S. Tserkis, M. Umer, and D. G. Angelakis, Depth optimization of CNOT ladder circuits, Phys. Lett. A 599 (2026) 132286.
[53] V. Vedral, A. Barenco, and A. Ekert, Quantum networks for elementary arithmetic operations, Phys. Rev. A 54, 147 (1996).
[54] J. Nie, W. Zi, and X. Sun, Quantum circuit for multi-qubit Toffoli gate with optimal resource, arXiv:2402.05053 (2024).
[55] T. Khattar and C. Gidney, Rise of conditionally clean ancillae for efficient quantum circuit constructions, Quantum 9, 1752 (2025).
[56] S. Dutta, S. Wang, A. Baksi, A. Chattopadhyay, and S. Maitra, Exact space-depth trade-offs in multicontrolled Toffoli decomposition, Phys. Rev. A 111, 052611 (2025).
[57] S. Tserkis, M. Umer, E. Mastorakis and D. G. Angelakis, Minimum Toffoli depth for the multi-controlled Toffoli gate via teleportation, arXiv: 2604.25861 (2026).
[58] J. Eisert, K. Jacobs, P. Papadopoulos, and M. B. Plenio, Optimal local implementation of nonlocal quantum gates, Phys. Rev. A 62, 052317 (2000).
[59] M. Sarvaghad-Moghaddam and M. Zomorodi, A general protocol for distributed quantum gates, Quantum Inf. Process. 20 (2021).