ParCFD2024
35^{th} International Conference on Parallel Computational Fluid Dynamics
Sep. 02-04, 2024, Bonn, Germany

Solving Fluid Dynamics Equations with Differentiable Quantum Circuits

Smit Chaudhary[a,*], Giorgio Tosti Balducci[a], Oleksandr Kyriienko[b], Panagiotis Kl. Barkoutsos[a], Lorenzo Cardarelli[a] and Antonio A. Gentile[a]

[a] Pasqal SAS, 7 Rue L. de Vinci, Massy, 91300, France
[b] University of Sheffield, School of Mathematical and Physical Sciences, Western Bank, Sheffield, S10 2TN, UK

ARTICLE INFO[†]

Keywords:
Fluid Dynamics;
Quantum Computing;
Physics-Informed Neural Networks

ABSTRACT

Differentiable quantum circuits (DQCs) are the hybrid quantum-classical alternative to Physics-Informed Neural Networks (PINNs). The latter ones have been introduced from the machine learning community to avoid the curse of dimensionality in mesh-based computational fluid dynamics (CFD) solvers, and allow for seamless inclusion of information from available data. The adoption of quantum circuits is motivated by enabling access to highly expressive feature maps, which might be key in capturing intricate solutions to selected fluid dynamics problems. In this work, we discuss the potential of DQCs and its recent extensions to address paradigmatic CFD use cases.

1. Introduction

Recent advances in the domain of CFD allowed for better solutions of fluid dynamics problems and their respective partial differential equations (PDEs). Many of CFD's most successful techniques use finite differences/elements/volume methods or spectral methods to solve some form of the Navier-Stokes equations [1]. Despite the high fidelity that these methods can reach, their reliance on meshes or modes exposes them to the curse of dimensionality, i.e., high computational costs in multi-scale PDEs involving multiple equations in 2D and 3D geometries.

Recently, PINNs have been proposed as an alternative paradigm to solve PDEs [2]. In essence, PINNs approximate the PDE solution with the output of a neural network (NN), trained to minimize loss terms directly derived from the equations. PINNs offer an edge with respect to standard supervised learning (SL), as in principle, they do not require any sample of the solution, be that analytical, numerical or experimental. Thanks to their efficiency and flexibility, PINNs found applications also in CFD problems [3].

[†] This paper is part of the ParCFD 2024 Proceedings. The DOI of this document is 10.34734/FZJ-2025-02445 and of the Proceedings 10.34734/FZJ-2025-02175.

*Corresponding author
smit.chaudhary@pasqal.com (S. Chaudhary); giorgio.tosti-balducci@pasqal.com (G. Tosti Balducci); o.kyriienko@sheffield.ac.uk (O. Kyriienko); pbarkoutsos@gmail.com (P.Kl. Barkoutsos); lorenzo.cardarelli@pasqal.com (L. Cardarelli); andrea.gentile@pasqal.com (A.A. Gentile)
ORCID(s): 0000-0003-0243-6513 (S. Chaudhary); 0000-0001-9242-676X (G. Tosti Balducci); 0000-0002-6259-6570 (O. Kyriienko); 0000-0001-9428-913X (P.Kl. Barkoutsos); 0000-0001-5010-5167 (L. Cardarelli); 0000-0002-1763-9746 (A.A. Gentile)

Using NNs as a solution approximator is an obvious choice due to the popularity of neural architecture, however any universal, differentiable, trainable model can be used in their stead. One proposal replaces NNs with Differentiable Quantum Circuits (DQCs) [4], as represented in Fig. 1. The approach is variational (which makes it more viable on near-term quantum hardware [4]), and offers exact differentiation [5], removing the need of numerical differentiation entirely. Finally, DQC offers efficient ways to encode problem features (e.g., coordinates), leveraging the exponentially large Hilbert space (w.r.t. the number of qubits) accessed by the quantum circuits [4, 6]. Therefore, their execution can be more energy-efficient, when compared to GPU training of NNs.

2. Methods

Consider the following generic differential problem

$$\frac{\partial u}{\partial t} + \mathcal{N}[u] = 0, \quad t \in [0, T], \, x \in \Omega, \tag{1}$$

$$u(t = 0, x) = g(x), \quad x \in \Omega, \tag{2}$$

$$\mathcal{B}[u] = 0, \quad t \in [0, T], \, x \in \partial\Omega, \tag{3}$$

where $\mathcal{N}$ is a generic nonlinear operator of u and its x-derivatives, $g(x)$ is an initial condition and $\mathcal{B}[u]$ is a boundary operator.

In the DQC methodology, we approximate the solution as $u(x) \approx u_\theta(x)$, via the expectation value of an observable $\hat{C}$:

$$u_\theta(x) = \langle u_\theta(x) | \hat{C} | u_\theta(x) \rangle, \tag{4}$$

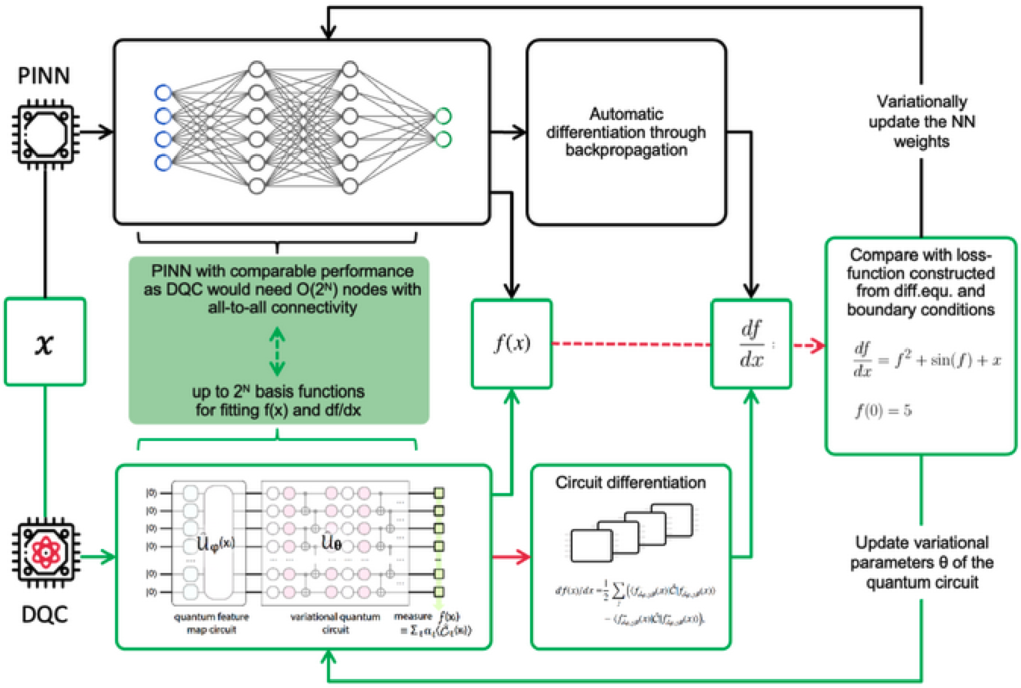

Figure 1: Diagram of the DQC algorithm.

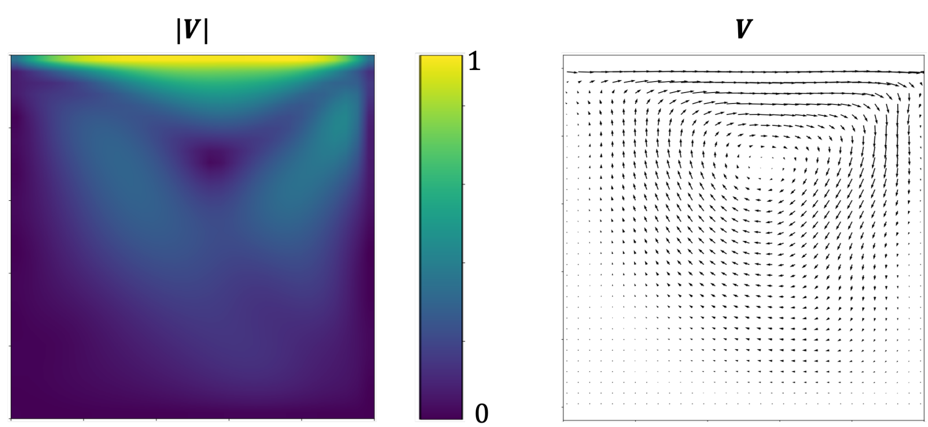

Figure 2: Lid-driven flow in a square cavity solved with DQC. On the left is the total velocity magnitude and on the right the velocity profile.

measured on the output state $|u_\theta(x)\rangle$ of a Quantum Neural Network (QNN), a circuit described by (trainable) θ parameters and encoding the features x as [7]

$$|u_\theta(x)\rangle = U_\theta U_\varphi(x)\,|0\rangle\,. \tag{5}$$

θ's are trained to minimize a loss function evaluating the residuals of Eqs. 1-3, when $u_\theta(x) \to u_(x)$, with $U_\varphi(x)$ and U_θ unitary operations.

3. Results

Figure 2 shows the solution of the lid-driven cavity flow problem obtained with DQC. We present the solution of the incompressible Navier-Stokes equation in its non-dimensional, steady state form

$$(\mathbf{V}\cdot\boldsymbol{\nabla})\mathbf{V} = -\boldsymbol{\nabla} p + \frac{1}{Re}\nabla^2\mathbf{V}, \tag{6}$$

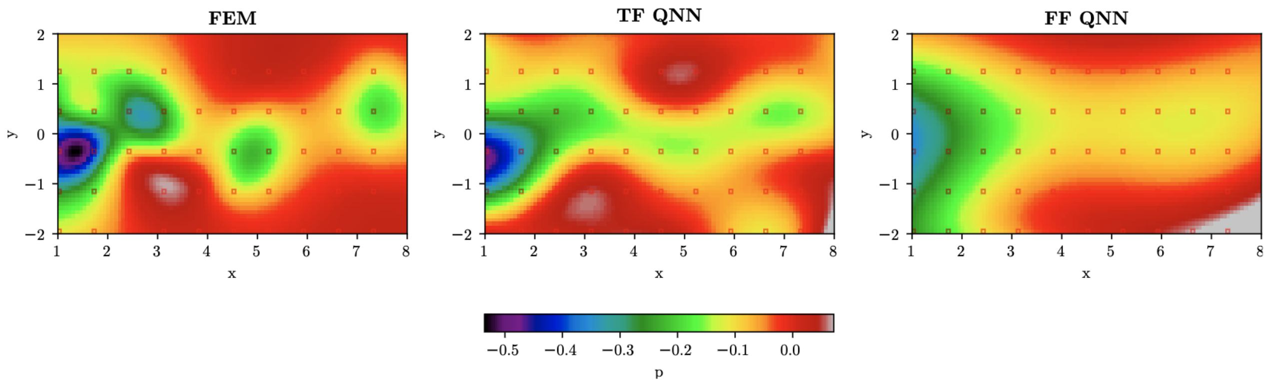

Figure 3: Pressure field downstream from an obstacle at $x = 0$. On left is the reference FEM result. In middle is the prediction of a Trainable Frequency (TF) QNN. On the right is the prediction of a Fixed Frequency (FF) QNN.

where $\mathbf{V} = (v_x, v_y)^T$ is the 2D velocity vector, p is the pressure and Re is the Reynolds number. The QNN surrogate for $\mathbf{V}$ and p employed 4 qubits for both x and y coordinates, and used 8 layers of the so-called *hardware efficient ansatz* [8] as U_θ. The solution for Re=150 matches closely the benchmark obtained via a finite element method (FEM) solver.

The spectrum of frequencies accessible to the feature map in the quantum circuit can be augmented by including in U_φ additional trainable variational parameters [9]. Figure 3 shows the pressure field downstream from a cylinder for 2D time-dependent Navier-Stokes equations.

A QNN approximating the stream function $\widetilde{\psi}$ is used to derive the two components of the velocity $\mathbf{V}$ and to satisfy the continuity relation automatically. Another QNN is used to approximate the pressure field p. Each QNN employs 8 layers of the ansatz. The results for Re=100 show a noticeable improvement compared to a fixed frequency version of the QNNs.

4. Conclusions

In this work, we applied the DQC algorithm to fluid dynamics PDEs. We presented promising results obtained both with various feature map architectures. Ongoing development aims to also include inductive biases [10], targeting, e.g., irrotational flows.

References

[1] T. J. Chung, Computational Fluid Dynamics, Cambridge University Press, 2002. doi:10.1017/CBO9780511606205.

[2] M. Raissi, P. Perdikaris, G. E. Karniadakis, Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations, Journal of Computational Physics 378 (2019) 686–707. doi:10.1016/j.jcp.2018.10.045.

[3] S. Cai, Z. Mao, Z. Wang, M. Yin, G. E. Karniadakis, Physics-informed neural networks (PINNs) for fluid mechanics: a review, Acta Mechanica Sinica 37 (12) (2021) 1727–1738. doi:10.1007/s10409-021-01148-1.

[4] O. Kyriienko, A. E. Paine, V. E. Elfving, Solving nonlinear differential equations with differentiable quantum circuits, Physical Review A 103 (5) (2021) 052416. doi:10.1103/PhysRevA.103.052416.

[5] D. Wierichs, J. Izaac, C. Wang, C. Y.-Y. Lin, General parameter-shift rules for quantum gradients, Quantum 6 (2022) 677. doi:10.22331/q-2022-03-30-677.

[6] A. E. Paine, V. E. Elfving, O. Kyriienko, Physics-informed quantum machine learning: Solving nonlinear differential equations in latent spaces without costly grid evaluations (2023). arXiv:2308.01827.

[7] D. Seitz, N. Heim, J. P. Moutinho, R. Guichard, V. Abramavicius, A. Wennersteen, G.-J. Both, A. Quelle, C. de Groot, G. V. Velikova, V. E. Elfving, M. Dagrada, Qadence: a differentiable interface for digital and analog programs, IEEE Software (2025) 1–14. doi:10.1109/MS.2025.3536607.

[8] 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. doi:10.1038/nature23879.

[9] B. Jaderberg, A. A. Gentile, Y. A. Berrada, E. Shishenina, V. E. Elfving, Let quantum neural networks choose their own frequencies, Physical Review A 109 (4) (2024) 042421. doi:10.1103/PhysRevA.109.042421.

[10] A. Ghosh, A. A. Gentile, M. Dagrada, C. Lee, S.-H. Kim, H. Cha, Y. Choi, B. Kim, J.-I. Kye, V. E. Elfving, Harmonic (Quantum) Neural Networks, arXiv:2212.07462 (2022). arXiv:2212.07462.