QPURPOSE - QST-HACK2025 PROBLEMS AND BACKGROUND MATERIAL

GARD OLAV HELLE

Contents

1. Introduction

The dynamics of fluid mechanical systems are governed by systems of partial differential equations usually involving some version of the Navier-Stokes equations. Computational fluid dynamics (CFD) is concerned with producing precise simulations of such systems. A prime example is weather or climate prediction where the atmosphere is modelled as a thin fluid. Industries such as automotive, aerospace, civil engineering, wind energy, and defence rely heavily on CFD.

While the full Navier-Stokes equations are definitely out of reach on current quantum hardware, it is still highly interesting to find efficient methods to simulate more basic systems that still capture various key aspects of fluid flow.

2. Description of the problems

The project is divided into two parts that can be solved independently. The solution should be in the form of a program setting up a quantum circuit with Qiskit including classical pre- and post-processing as appropriate. The circuit should be tested with a noiseless simulator (e.g., aer-simulator).

2.1. **Quantum circuits for solving PDEs.** Let $T, d > 0$ and a smooth function $u_0\colon [0,d] \to \mathbb{R}$ be given. We will consider the problem of solving one of the partial differential equations

$$(A) \quad \frac{\partial u}{\partial t} + c\frac{\partial u}{\partial x} = 0 \qquad \text{(1d linear advection)}$$

$$(B) \quad \frac{\partial u}{\partial t} = \nu\frac{\partial^2 u}{\partial x^2} \qquad \text{(1d linear diffusion)}$$

Date: January 28, 2025.

for $u\colon [0,T] \times [0,d] \to \mathbb{R}$ subject to the initial conditions $u(0,x) = u_0(x)$ for all $x \in [0,d]$, and periodic boundary conditions $u(t,0) = u(t,d)$ for all $t \in [0,T]$. In (A) $c \in \mathbb{R}$ is the constant speed of the fluid flow and in (B) $\nu \in \mathbb{R}^+$ is the diffusion coefficient. In both cases u could be the concentration of some quantity being distributed in a fluid due to advection or diffusion, respectively.

Problem 1. Construct a quantum circuit that simulates either (A) the 1d linear advection equation or (B) the 1d linear diffusion equation. Explicitly, if $u(t,x)$ is the unique solution to the equation of choice, the circuit should implement a unitary U that takes $|\psi_0\rangle \mapsto |\psi_T\rangle \coloneqq U|\psi_0\rangle$, where $|\psi_0\rangle$ and $|\psi_T\rangle$ encode discrete approximations of $u_0(x) = u(0,x)$ and $u_T(x) \coloneqq u(T,x)$, respectively.

In both cases one can use Gaussian initial conditions of the form $u_0(x) = e^{-a(x-d/3)^2}$, appropriately normalized, with $a > 0$ sufficiently large to satisfy periodic boundary conditions. To get started one can take $d = 4$, $c = 1$, $\nu = 0.02$.

A natural approach is to discretize the problem via some choice of numerical scheme and encode it in a quantum circuit. Using finite differences, the advection equation can be turned into a Hamiltonian simulation problem, while the diffusion equation can be turned into a (Szegedy) quantum walk problem.

2.2. **Example: the diffusion equation.** The following example will be continued later in the document and serves as explanation of the accompanying sample code.

Consider the diffusion equation in (B) above. Let $\Delta t, \Delta x > 0$ and introduce the discrete approximation $v(t) = (u(t,x_0), u(t,x_1), \cdots, u(t,x_{N-1}))'$ for some N, where $x_j = j\Delta x$. If we introduce the finite difference approximations

$$\frac{\partial^2 u}{\partial x^2}(t,x) \approx \frac{u(t,x+\Delta x) - 2u(t,x) + u(t,x-\Delta x)}{\Delta x} \qquad \text{and} \qquad \frac{\partial u}{\partial t} \approx \frac{u(t+\Delta t,x) - u(t,x)}{\Delta t}$$

in the PDE, a few manipulations lead to the approximation

$$v(t+\Delta t) \approx (I + \alpha A)v(t) \qquad \text{where } \alpha = \nu\Delta t/(\Delta x)^2, \tag{1}$$

I is the identity and A is the $N \times N$ tridiagonal matrix with -2 on the main diagonal and $+1$ on the two adjacent diagonals (including the upper right and lower left corners due to the periodic boundary conditions). For example for $N = 5$ one has

$$A = \begin{pmatrix} -2 & 1 & 0 & 0 & 1 \\ 1 & -2 & 1 & 0 & 0 \\ 0 & 1 & -2 & 1 & 0 \\ 0 & 0 & 1 & -2 & 1 \\ 1 & 0 & 0 & 1 & -2 \end{pmatrix} \tag{2}$$

Equation (1) then suggest the following scheme. Set $w_0 = v(0)$ and define w_k recursively by $w_{k+1} = Bw_k$, where $B \coloneqq I + \alpha A$. Then w_k approximates $v(k\Delta t)$ for all k, in particular

$$v(T) \approx w_M = B^M w_0 \qquad \text{for } T = M\Delta t, \tag{3}$$

which is a discrete approximation of $u_T(x) = u(T,x)$. The quality of the approximation depends on Δx, Δt and M. As a minimal criterion for stability one should require

$$1 \geq \frac{2\nu\Delta t}{(\Delta x)^2}.$$

In that case, the matrix $B = I + \alpha A$ is stochastic, that is, each column consists of non-negative real numbers whose sum is 1. This will be important for the quantum implementation later.

Remark. It is important to note that B is not unitary and we must therefore block encode it in a unitary matrix. As a consequence the quantum representatives of $v(0)$ and $v(T)$ will be of the form

$$|\psi_0\rangle = |0\rangle^{\otimes a} \otimes \left(\sum_{j=0}^{N-1} v(0)_j \,|j\rangle\right) \qquad \text{and} \qquad |\psi_T\rangle = |0\rangle^{\otimes a} \otimes \left(\sum_{j=0}^{N-1} v(T)_j \,|j\rangle\right) + |\gamma\rangle\,,$$

where a is the number of ancilla qubits needed for the encoding and $|\gamma\rangle$ is a garbage state. This means that B is encoded on the subspace $|0\rangle^{\otimes a} \otimes \mathbb{C}^{2^n}$. Further details are given in the section on block encoding.

2.3. **Extracting information via measurements.** The solution to the first part should be in the form of a quantum circuit implementing a unitary U that takes $|\psi_0\rangle \mapsto |\psi_T\rangle = U\,|\psi_0\rangle$, where $|\psi_0\rangle\,, |\psi_T\rangle$ encode discrete approximations of the initial condition $u_0(x)$ and the final desired state $u_T(x) = u(T, x)$.

Problem 2. These problems involve extracting interesting information from the output of the previous problem via measurements.

(A) Construct a quantum circuit and a measurement strategy that outputs an approximation to

$$S := \int_{d/4}^{3d/4} u(T, x)dx$$

when the initial state is $|\psi_T\rangle$.

(B) Construct a quantum circuit and a measurement strategy that outputs 1 if

$$\max_{x\in[0,d]} u_T(x) > \tau,$$

for a given threshold value $\tau \geq 0$, and otherwise outputs 0.

The information we attempt to extract in the above problems has practical value in weather or climate applications. The quantity in (A) is simply the mean value of u over a sub-domain $[d/4, 3d/4]$. The threshold problem in part (B) has relevance for prediction of extreme weather events.

2.4. **Extra: adding complexity to the PDEs.** In case the problems in (1) are too simple, here are some alternative equations of slightly higher complexity.

- The $1d$ advection-diffusion equation

$$\frac{\partial u}{\partial t} + c\frac{\partial u}{\partial x} = \nu\frac{\partial^2 u}{\partial x^2}. \tag{4}$$

- The $2d$ advection equation

$$\frac{\partial u}{\partial t} + C \cdot \nabla u = 0. \tag{5}$$

- The $2d$ diffusion equation

$$\frac{\partial u}{\partial t} = \nu \Delta u. \tag{6}$$

In the two last equations $u\colon [0, T] \times [0, d]^2 \to \mathbb{R}$ is defined on a two-dimensional spatial domain, $C = (c_1, c_2) \in \mathbb{R}^2$ and $\nu \geq 0$ as before.

2.5. **Exact solutions.** For comparison and testing purposes the exact solution of the advection-diffusion (4) is given. By setting $\nu = 0$ or $c = 0$, respectively, the exact solution of the advection or diffusion equation is obtained.

The advection-diffusion equation can be solved exactly via Fourier analysis. Let the spatial domain be $[0, d]$ and impose periodic boundary conditions. If

$$u_0(x) = \sum_{k \in \mathbb{Z}} a_k e^{i\omega kx} \qquad \text{where } \omega = 2\pi/d,$$

then the exact solution is given by

$$u(t, x) = \sum_{k \in \mathbb{Z}} a_k e^{-\nu\omega^2 k^2 t - ic\omega kt} e^{i\omega kx}. \tag{7}$$

3. Background material

The example in section 2.2 introduced a numerical approximation scheme for the diffusion equation. It has the form $w_M = B^M w_0$ (see equation (3)) for a certain matrix B depending on various parameters. Here M denotes the number of time steps. In the case of the advection equation, one can via suitable finite differences arrive at a numerical scheme of the form

$$v(T) \approx e^{iMC} v(0)$$

for a certain Hermitian matrix C. Implementation of this scheme is a special instance of the Hamiltonian simulation problem.

The common theme of the two methods is that we are given a Hermitian matrix D and we want to implement a function $f(D)$ of that matrix. In the first case, $f(x) = x^M$ and in the second case $f(x) = e^{iMx}$. This can be achieved via a general quantum algorithm called the Quantum Singular Value Transform (QSVT). The purpose of the following sections is to provide background material and references for QSVT, and continue the example in 2.2.

3.1. **Block encoding.** The notion of a block encoding is a key underlying concept of the QSVT algorithm. A unitary block encoding of a matrix A is the realization of A as a block of a unitary matrix

$$U = \begin{pmatrix} A & * \\ * & * \end{pmatrix},$$

where the stars denote unimportant matrices. More precisely, the Hilbert space $\mathcal{H}$ on which U is defined admits an orthogonal decomposition $\mathcal{H} = \mathcal{H}_1 \oplus \mathcal{H}_2$ such that $\langle\psi| U |\phi\rangle = \langle\psi| A |\phi\rangle$ for all $\psi, \phi \in \mathcal{H}_1$. Equivalently $U |\phi\rangle = A |\phi\rangle + |\gamma\rangle$ for some $|\gamma\rangle \in \mathcal{H}_2$, whenever $|\phi\rangle \in \mathcal{H}_1$. The subspace on which A is encoded is conveniently recorded in terms of the orthogonal projection

$$\Pi \colon \mathcal{H} \to \mathcal{H} \qquad \text{given by} \qquad \Pi(|\phi\rangle + |\gamma\rangle) = |\phi\rangle \tag{8}$$

for all $\phi \in \mathcal{H}_1$ and $\gamma \in \mathcal{H}_2$.

Remark. A unitary block-encoding of A exists if and only if $||A|| \leq 1$ (spectral norm). If the relevant matrix fails to satisfy this bound, one must work with a rescaled version A/α and keep track of the scaling constant α instead.

In practice this is achieved by decomposing the Hilbert space as $H = \mathbb{C}^{2^r} \otimes \mathbb{C}^{2^n}$ and encode A on the subspace $|0\rangle^{\otimes r} \otimes \mathbb{C}^{2^n}$. In that case

$$U(|0\rangle \otimes |\psi\rangle) = |0\rangle \otimes A |\psi\rangle + \sum_{j=1}^{2^r-1} |j\rangle \otimes |\phi_j\rangle$$

for certain states $|\phi_j\rangle \in \mathbb{C}^{2^n}$. By measuring the first register and post-selecting $|0\rangle$, we have obtained a normalized version of $A|\psi\rangle$ in the second register. Moreover, the probability of obtaining $|0\rangle$ in the measurement is precisely $||A|\psi\rangle||^2$, so by sampling one can estimate the amplitude as well.

- The paper `https://arxiv.org/abs/2203.10236` [2] is a good introduction to block encoding and provided explicit algorithms for the type of matrices encountered in the above numerical schemes.
- The survey paper `https://arxiv.org/abs/2310.03011` [3, 10] contains a section on block encodings and further references.

3.2. **The Quantum Singular Value Transform.** The QSVT algorithm [5] takes as input a block encoding U of A, and outputs a block encoding of $p(A)$ for a real polynomial $p\colon \mathbb{R} \to \mathbb{R}$ satisfying two basic requirements.

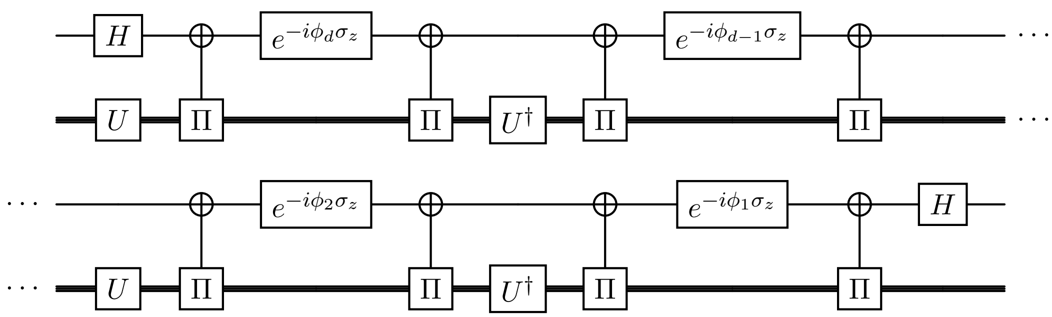
(1) p must be either even $p(-x) = p(x)$ or odd $p(-x) = -p(x)$
(2) p must satisfy $|p(x)| \leq 1$ for all $x \in [-1, 1]$

The following is a basic version of the QSVT.

Theorem 3.1. *Let U be a block encoding of A and let Π be the orthogonal projection onto the subspace on which A is encoded. For any polynomial p of degree d satisfying the above two requirements there is an angle sequence (only depending on p)*

$$\Phi = (\phi_1, \cdots, \phi_d)$$

such that the following circuit



provides a block encoding of $p^{SV}(A)$ with one additional ancilla qubit on the subspace $|0\rangle \otimes \mathrm{Image}\,\Pi$.

A few comments are in order.

(1) $p^{SV}(A)$ means that p is applied to A in the singular value sense (see [5] for a definition). If A is Hermitian this coincides with $p(A)$, and this is the case of interest to us.
(2) If we want to apply a more general function $f\colon \mathbb{R} \to \mathbb{R}$ satisfying the two requirements above, compute a polynomial approximation p to f and apply the theorem with p as above.
(3) The controlled Π gate acts as follows
$$|k\rangle|\psi\rangle \mapsto |k+1\rangle\,\Pi\,|\psi\rangle + |k\rangle\,(I - \Pi)\,|\psi\rangle \qquad k \in \{0, 1\},$$
i.e., it flips the first qubit conditioned on the state in the second register belonging to $\mathrm{Im}\,\Pi$. If we have block encoded A on the subspace $|0\rangle \otimes \mathbb{C}^{2^n} \subset \mathbb{C}^{2^r} \otimes \mathbb{C}^{2^n}$, then the controlled Π gate is just a controlled NOT gate with the following r qubits as controls.
(4) Quantum Signal Processing (QSP) tells us how to pass between polynomials and angle sequences. The general theory is not necessary to understand, since there is freely available software that computes the angle sequence associated with a given polynomial (see below).

The main references for QSVT and the computation of angle sequences are the following.

- https://arxiv.org/abs/1806.01838 is the original paper that introduced QSVT in 2019. It is comprehensive and includes many applications [5].
- https://arxiv.org/abs/2105.02859 is a more pedagogical introduction to QSVT [6].
- https://github.com/ichuang/pyqsp is a python package accompanying the above paper. It contains various methods for computing angle sequences (the sym_qsp metod is recommended for our purposes).
- https://qsppack.gitbook.io/qsppack provides a very good MATLAB package for computing angle sequences [4]. The website also contains a number of good examples.

3.3. **Example continued: the diffusion equation.** Recall from section 2.2 that the numerical scheme took the form $w_M = B^M w_0$, where w_k is the approximation at time step k. The matrix B has the form

$$B = \begin{pmatrix} a & b & 0 & 0 & b \\ b & a & b & 0 & 0 \\ 0 & b & a & b & 0 \\ 0 & 0 & b & a & b \\ b & 0 & 0 & b & a \end{pmatrix}, \tag{9}$$

here for $N = 5$, where the entries are computed from Δx, Δt and ν

$$a = 1 - \frac{2\nu\Delta t}{(\Delta x)^2} \qquad \text{and } b = \frac{\nu\Delta t}{(\Delta x)^2}.$$

We had to require $a \geq 0$ for numerical stability, and this ensures that the matrix is stochastic.

The accompanying code implements the numerical scheme via the following two steps:

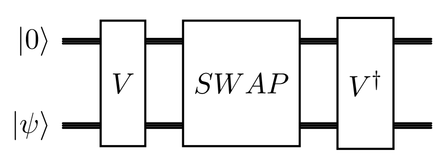
(1) block encode B in a circuit U
(2) apply the QSVT algorithm as given in theorem 3.1 with $p(x) = x^M$ and the block encoding U.

The second step is clear once we have constructed the block encoding, and that is the purpose of this section. We will employ a general method for symmetric stochastic matrices contained in the following proposition.

Proposition 3.2. *Let $A = (A_{ij})_{ij}$ be an $2^n \times 2^n$, symmetric and stochastic matrix where $0 \leq i, j < 2^n$. Assume that V is a unitary acting on two n qubit registries that implements*

$$|0\rangle\,|k\rangle \mapsto \sum_{j=0}^{2^n-1} \sqrt{a_{kj}}\,|j\rangle\,|k\rangle$$

for each $0 \leq k < N$. Then the following circuit is a block encoding of A on the subspace $|0\rangle \otimes \mathbb{C}^{2^n}$.



Taking this for granted, the remaining task is to implement V for the given matrix B. This will be achieved via three simple transformations.

(1) A state preparation gate G_{prep} on two qubits acting by

$$G_{prep}\colon\ |0\rangle \mapsto \sqrt{b}\,|0\rangle + \sqrt{a}\,|1\rangle + \sqrt{b}\,|2\rangle\,.$$

(2) The inverse $S^\dagger$ of the shift operator

$$S\colon\ |j\rangle \mapsto |j + 1 \pmod{2^n}\rangle \qquad \text{for all } 0 \leq k < 2^n.$$

(3) A modular adder Add acting on two n-qubit registries as

$$|k\rangle\,|j\rangle \mapsto |k+j \pmod{2^n}\rangle\,|j\rangle \qquad \text{for all } 0 \leq k, j < 2^n$$

The unitary $V \coloneqq \mathrm{Add} \circ (S^\dagger \otimes I) \circ (G_{prep} \otimes I)$ then implements

$$\begin{aligned} |0\rangle\,|k\rangle &\mapsto (\sqrt{b}\,|0\rangle + \sqrt{a}\,|1\rangle + \sqrt{b}\,|2\rangle) \otimes |k\rangle \mapsto (\sqrt{b}\,|{-1}\rangle + \sqrt{a}\,|0\rangle + \sqrt{b}\,|1\rangle) \otimes |k\rangle \\ &\mapsto (\sqrt{b}\,|k-1\rangle + \sqrt{a}\,|k\rangle + \sqrt{b}\,|k+1\rangle) \otimes |k\rangle \end{aligned}$$

for each k, where the computational basis states in are considered modulo 2^n. Inspection of B in (9) and the statement of proposition 3.2 show that this is the required transformation needed to block encode B.

References

[1] Daan Camps and Roel Van Beeumen. Fable: Fast approximate quantum circuits for block-encodings. *2022 IEEE International Conference on Quantum Computing and Engineering (QCE)*, pages 104–113, 2022.

[2] Daan Camps, Lin Lin, Roel Van Beeumen, and Chao Yang. Explicit Quantum Circuits for Block Encodings of Certain Sparse Matrices. *SIAM J. Matrix Anal. Appl.*, 45(1):801–827, 2024.

[3] Alexander M. Dalzell, Sam McArdle, Mario Berta, Przemyslaw Bienias, Chi-Fang Chen, Andr'as Gily'en, Connor T. Hann, Michael J. Kastoryano, Emil T. Khabiboulline, Aleksander Kubica, Grant Salton, Samson Wang, and Fernando G. S. L. Brandão. Quantum algorithms: A survey of applications and end-to-end complexities. 2023.

[4] Yulong Dong, Xiang Meng, K. Whaley, and Lin Lin. Efficient phase-factor evaluation in quantum signal processing. *Physical Review A*, 103, 04 2021.

[5] András Gilyén, Yuan Su, Guang Hao Low, and Nathan Wiebe. Quantum singular value transformation and beyond: exponential improvements for quantum matrix arithmetics. In *Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing*, STOC 2019, page 193–204, New York, NY, USA, 2019. Association for Computing Machinery.

[6] John M. Martyn, Zane M. Rossi, Andrew K. Tan, and Isaac L. Chuang. Grand unification of quantum algorithms. *PRX Quantum*, 2:040203, Dec 2021.