Progress in Quantum Computing Research in Fluid Mechanics

Zhang Baiming (张柏铭)[1]

[1]3230100298, Department of Engineering Mechanics, School of Aeronautics and Astronautics, Zhejiang University, Hangzhou 310027

1 Attention

Although the course assignment requires typesetting within three pages, I really want to write this part more completely (three pages can't write anything at all) to showcase my actual work, so you can choose to deduct points at your discretion.

2 Preface

By enrolling in the course "Quantum Algorithms," my primary goal was to support the advancement of the research topic "Quantum Computing in Fluid Dynamics." Throughout this semester's study, I have not only solidified my understanding of quantum algorithms but also proposed innovative perspectives and algorithms for "Quantum Computing in Fluid Dynamics." Concurrently, I am collaborating with my mentor, Professor Xiong Shiying, on writing a book related to this subject. In this term paper, I will discuss some of the content I have contributed to. Considering that you, Professor Alexander Zenchuk, may not be a specialist in this field, and the specific algorithms are still under research and writing, not yet published. This paper will only include some brief review work that I have been involved in.

3 Acknowledgements

I would like to extend my special thanks to my supervisor, Xiong Shiying, for their guidance. I am also particularly grateful for the course "Quantum Algorithms" by Professor Alexander Zenchuk. I would also like to thank my teaching assistant Wu Junde for his diligent and responsible maintenance of the smooth progress of the course. This is an initial review of the application of quantum computing to fluid dynamics. The original text is in Chinese and has not been submitted for publication.

4 Abstract

This paper reviews the progress and challenges in the cutting-edge interdisciplinary research field of quantum computing in fluid dynamics. As a potential disruptive technology, quantum computing is expected to solve some of the real-world problems in the future. Fluid dynamics, being a highly challenging issue in classical physics and engineering applications, can serve as an example to demonstrate the practicality and superiority of quantum computing. At the same time, quantum computing can also bring new research paradigms to fluid dynamics.

The paper first elaborates on the characteristics of quantum computing in terms of quantum superposition and entanglement, pointing out the challenges in the preparation of initial states, quantum state evolution, and measurement in quantum computing for fluid dynamics. It then focuses on introducing quantum-classical hybrid algorithms, Hamiltonian simulation, and other quantum algorithms for fluid dynamics, as well as reviewing their hardware implementation progress on real quantum computers.

In summary, quantum computing in fluid dynamics is still in its infancy, facing many challenges in both quantum computing hardware and algorithms. Compared to traditional methods, although quantum computing has not yet shown superiority in simulating strongly nonlinear fluid dynamics problems, recent progress indicates its potential to efficiently simulate complex flows such as turbulence.

5 Introduction

Prominent physicist Richard Feynman proposed in 1982 that a new type of computer, built on the principles of quantum mechanics, could solve quantum simulation problems—such as quantum many-body problems—far more efficiently than classical computers, giving rise to the concept of quantum computing. On this foundation, in 1985, David Deutsch pointed out that by using the superposition of quantum states, one could achieve parallel computing by manipulating these superposed states, laying the theoretical groundwork for the accelerated capabilities of quantum computing. As a potential disruptive technology, quantum computing has flourished over the past half-century, gradually becoming one of the most notable research fields.

Feynman also noted that "turbulence is the most important unsolved problem in classical physics," and turbulence has numerous significant applications in fields such as aerospace, atmospheric science, and oceanography. Turbulence manifests as a highly disordered and chaotic state of fluid motion, involving the nonlinear interactions of countless vortices across many orders of magnitude in time and space scales. This multi-scale, nonlinear dynamic behavior makes the prediction and control of turbulence extremely difficult. Direct numerical simulation of high Reynolds number turbulence requires im-

mense computational resources, far beyond the capabilities of today's most powerful supercomputers, thus highlighting the immense potential of quantum computing in this field where computational power is desperately needed. However, since turbulence is a typical strongly nonlinear classical physics problem, not a quantum simulation problem, it requires the design of quantum computing algorithms specifically tailored to solve fluid dynamics problems.

In the emerging field of Quantum Computing for Fluid Dynamics (QCFD), quantum computing has the potential to address fundamental challenges in fluid dynamics, such as turbulence prediction, bringing a new paradigm to the field of fluid dynamics. Leveraging its exponential growth in information storage capacity and the advantage of quantum parallelism, quantum computing is expected to enable rapid simulation of complex flow problems with high Reynolds and Mach numbers in the future, which are crucial for important applications such as aircraft design and weather forecasting.

5.1 Principles of Quantum Computing

Quantum computers are primarily categorized into three types based on algorithms and physical mechanisms: the quantum circuit model, the adiabatic quantum computing model (including the quantum annealing model), and the topological quantum computing model. We will take the most common quantum circuit model as an example to briefly introduce the principles, basic concepts, and fundamental characteristics of quantum computing.

One of the key features of quantum computing is the superposition of states. A quantum bit (qubit) is the unit of information in a quantum computer. It can be in the "0" state $|0\rangle = [1, 0]^{\mathrm{T}}$ or the "1" state $|1\rangle = [0, 1]^{\mathrm{T}}$, and it can also be in a superposition state $|\psi\rangle = a|0\rangle + b|1\rangle$, where a and b are complex coefficients. Therefore, a superposition state is any linear combination of the "0" and "1" states, meaning a qubit can be in both "0" and "1" states simultaneously. $|a|^2$ and $|b|^2$ represent the probabilities of $|\psi\rangle$ being in the $|0\rangle$ and $|1\rangle$ states, respectively, and $|a|^2 + |b|^2 = 1$, meaning $|\psi\rangle$ is a unit vector. A qubit can collapse to a definite "0" or "1" state upon measurement.

The Bloch sphere in Figure 1(a) provides a geometric representation of a single qubit state, facilitating an intuitive understanding of a single quantum state. A qubit can be physically realized through a two-state quantum system, such as the ground state and the first excited state of an electron in a hydrogen atom, the +1/2 and -1/2 components of proton spin in any direction, or the left and right circular polarizations of light.

The computational basis vectors for two qubits are the direct product combinations of the individual qubit basis vectors, i.e., $|00\rangle$, $|10\rangle$, $|01\rangle$, and $|11\rangle$, which can be generalized to the case of n qubits. Thus, the superposition state of n qubits can be described

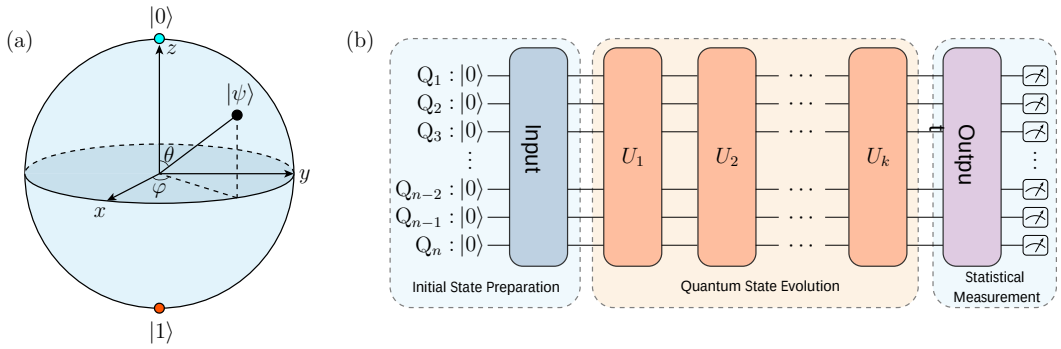

Figure 1: (a) The Bloch sphere representation of a single qubit state, which can be described using three-dimensional Cartesian coordinates $x = \sin\theta\cos\varphi$, $y = \sin\theta\sin\varphi$, $z = \cos\theta$, providing a geometric image of quantum states. Any single qubit state $|\psi\rangle = \cos\frac{\theta}{2}|0\rangle + e^{i\varphi}\sin\frac{\theta}{2}|1\rangle$ can be represented by a point on the sphere, with the $|0\rangle$ state at the north pole and the $|1\rangle$ state at the south pole. (b) The computational process of a quantum computer is divided into three steps: initial state preparation, quantum state evolution, and statistical measurement, with quantum gates $U_1, U_2, \ldots, U_k$ all being unitary transformations.

by a 2^n-dimensional Hilbert space $\mathbb{C}^{2^n}$ (i.e., a complex vector space), given by

$$|\psi\rangle = \sum_{q_1=0}^{1}\sum_{q_2=0}^{1}\cdots\sum_{q_n=0}^{1} c_{\text{binary}(q_n q_{n-1}\cdots q_2 q_1)} \bigotimes_{j=1}^{n} |q_j\rangle, \tag{1}$$

$\text{binary}(q_n q_{n-1}\cdots q_1)$ represents an n-bit binary number, $q_j \in \{0, 1\}$, and $c_0, c_1, \ldots, c_{2^n-1}$ are complex coefficients satisfying the normalization condition $\sum_{j=0}^{2^n-1}|c_j|^2 = 1$. Consequently, in a quantum computer, an n-qubit register in a superposition state contains all numbers from 0 to $2^n - 1$, each existing with a certain probability.

In classical computers, an n-bit register can only store one n-bit binary number; whereas in quantum computers, an n-bit quantum register can simultaneously store 2^n n-bit binary numbers, as shown in Figure 2(a). The amount of information stored in a quantum register grows exponentially with n, and upon measurement (i.e., reading out), the superposition collapses, yielding one of the n-bit numbers.

Another significant feature of quantum computing is the entanglement of states. If the state of multiple qubits cannot be expressed in the form of a direct product, these qubits are said to be in an entangled state. For example, $|\psi\rangle_A = \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)$ and $|\psi\rangle_B = \frac{1}{\sqrt{2}}(|01\rangle + |10\rangle)$ are entangled states, while $|\psi\rangle_C = \frac{1}{\sqrt{2}}(|00\rangle + |01\rangle) = \frac{1}{\sqrt{2}}|0\rangle \otimes (|0\rangle + |1\rangle)$ is not an entangled state. When multiple qubits are in an entangled state, measuring the state of some qubits will affect the measurement of the others. For instance, when measuring $|\psi\rangle_A$, measuring the state of one qubit will cause the other qubit's state to be the same, as shown in Figure 2(b); when measuring $|\psi\rangle_B$, measuring the state of one

qubit will cause the other qubit's state to be the opposite. In contrast, when measuring $|\psi\rangle_C$, no matter how the state of the right qubit is measured, the state of the left qubit is always $|0\rangle$. Leveraging the properties of entangled states in quantum computing allows for the design of efficient quantum state evolution and measurement algorithms.

Due to the linear nature of Hilbert space, a unitary transformation on a superposition state $|\psi\rangle = \frac{1}{\sqrt{2^n}} \sum_{x=0}^{2^n-1} |x\rangle$ can be represented as

$$U_f|\psi, 0\rangle = U_f \frac{1}{\sqrt{2^n}} \sum_{x=0}^{2^n-1} |x, 0\rangle = \frac{1}{\sqrt{2^n}} \sum_{x=0}^{2^n-1} U_f|x, 0\rangle = \frac{1}{\sqrt{2^n}} \sum_{x=0}^{2^n-1} |x, f(x)\rangle, \tag{2}$$

where $|\psi, 0\rangle \equiv |\psi\rangle \otimes |0\rangle$. Therefore, in quantum computing, a single unitary transformation can produce the function values corresponding to all computational basis from 0 to $2^n - 1$, i.e., all x's $|f(x)\rangle$. This property is known as quantum parallelism. In classical computers, calculating $f(x)$ for all x requires 2^n iterations or 2^n processors working in parallel, whereas in quantum computers, this can be accomplished with a single transformation, as illustrated in Figure 2(c).

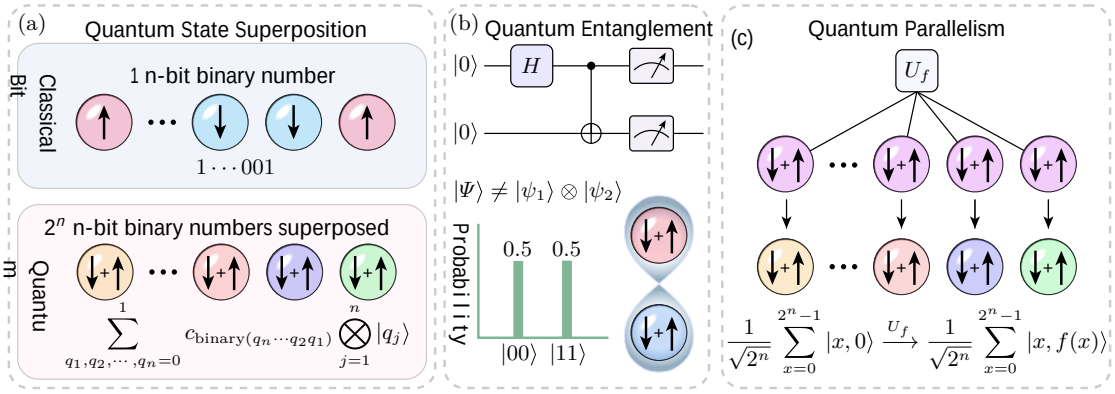

Figure 2: (a) In classical computers, an n-bit register holds one number from 0 to $2^n - 1$; whereas in quantum computers, an n-qubit register in a superposition state holds all numbers from 0 to $2^n - 1$ simultaneously, each existing with a certain probability. (b) The simplest quantum circuit for preparing a two-qubit entangled state: the output state $(|00\rangle + |11\rangle)/\sqrt{2}$ cannot be written as a direct product of two quantum states, meaning the two qubits are in an entangled state; measuring the state of one qubit in the entanglement will cause the state of the other qubit to be the same. (c) Quantum computing has a parallel-like characteristic, where a single unitary transformation on a superposition of quantum states can produce function values corresponding to all values from 0 to $2^n - 1$. Different or the same colors of qubits represent the presence or absence of entanglement between them, respectively.

Additionally, the evolution of quantum states in quantum computing is reversible. Quantum gates can transform the state of qubits, achieving various logical operations.

To maintain the orthogonality and normalization of the eigenstates of the Hilbert space in which the quantum states reside, the operator U that transforms the quantum states should be a unitary matrix, i.e., $UU^\dagger = I$, where I is the identity matrix, and $U^\dagger$ is the conjugate transpose of U.

Unitary transformations are reversible, i.e., $U|\psi\rangle = |\varphi\rangle$, and $U^\dagger|\varphi\rangle = |\psi\rangle$, hence quantum logic gates are also reversible. The input state, after passing through a quantum logic gate equivalent to a U transformation, becomes the output state, and the output state, after passing through a gate equivalent to a $U^\dagger$ transformation, returns to the input state. In classical computers, "AND" and "OR" gates are irreversible, whereas quantum logic gates are reversible.

The typical computational process of a quantum computer is shown in Figure 1(b), determined by a quantum algorithm, with different algorithms consisting of different unitary transformations. The initial state preparation circuit sets the input data into the desired initial state, which, after a series of quantum gate transformations, results in the output state, where $U_i, i = 1, 2, \ldots, k$ are unitary transformations, and finally, the result is obtained through statistical measurement. Since the measurement results in quantum computing are probabilistic, multiple repetitions of computation and measurement are required to achieve the desired level of precision.

The core operations of quantum computers are based on the principles of quantum mechanics, and directly and precisely manipulating and observing microscopic quantum states remains a formidable challenge. Therefore, auxiliary tools are relied upon to achieve this goal, with classical computers playing a crucial role. As a powerful auxiliary of modern technology, classical computers act as a bridge between the macroscopic and quantum worlds. In practical applications, classical and quantum computers are often closely integrated, allowing the user interface and data (such as control instructions, program code, input data, and output results) to remain within the classical domain.

The field of quantum computing is divided into two main branches: analog and digital quantum computing. Analog quantum computing is dedicated to using quantum systems to simulate the behavior of other quantum systems, which is crucial for understanding complex quantum phenomena and the development of new materials. It is commonly used to study complex systems in quantum physics, chemistry, and biology, such as molecular structures, phase transitions, and quantum coherence. Digital quantum computing aims to build a universal quantum computer capable of solving a variety of general problems, addressing issues in cryptography, materials science, drug design, and the simulation of classical mechanical systems through the design of quantum algorithms. Overall, analog and digital quantum computing each have their own prospects and challenges, but their common goal is to use the principles of quantum mechanics to solve problems that are intractable for classical computing.

In recent decades, several efficient digital quantum computing algorithms for specific

problems have been proposed, such as Shor's algorithm for factoring large numbers, Grover's algorithm for fast data search, a series of algorithms for solving linear systems of equations, and quantum walk methods for unstructured graphs. These algorithms have a theoretical complexity far lower than their classical counterparts and possess great potential for application. However, current quantum computing prototypes are still in the Noisy Intermediate-Scale Quantum (NISQ) stage and are not yet sufficient to run the above quantum algorithms on a scale with practical value and obtain results with sufficient accuracy.

Excitingly, current NISQ hardware has demonstrated an acceleration advantage over classical computing for some specific quantum simulation problems, such as Gaussian-Bose sampling, random circuit sampling, and random quantum walks. However, due to the limitations of current quantum hardware and algorithms, quantum supremacy with practical value has not yet been fully demonstrated on quantum machines.

5.2 Challenges of Quantum Computing in Fluid Dynamics

Applying quantum computing technology to the field of fluid dynamics, especially turbulence, which is widely considered one of the most challenging problems in classical physics, can not only bring a new research paradigm to fluid dynamics but also leverage the advantages of exponential growth in quantum computing storage space and quantum parallelism to achieve efficient numerical simulation of complex flow problems with high Reynolds and Mach numbers in engineering applications. At the same time, it can also provide a highly persuasive and practically quantum-supremacy case for the field of quantum computing, thereby promoting the cross-integration and development of the two disciplines.

Similar to the classical CFD process, most QCFD simulations can also be roughly divided into three parts: preprocessing, solver, and postprocessing. In QCFD preprocessing, it is necessary to encode flow field information into a state vector and prepare it as a quantum state, as shown in Figure 3. The required quantum initial state preparation is a common problem in solving differential equations, linear systems of equations, etc., using digital quantum computing methods.

Quantum computer initialization usually corresponds to the zero operation, which means we can easily obtain the quantum state $|0\rangle^{\otimes n}$, that is, the column vector $[1, 0, 0, \cdots, 0]^T$. To convert this zero state into an initial state corresponding to given initial conditions, it is necessary to write the corresponding unitary transformation and decompose it into basic quantum gates for operation. By applying a Hadamard gate to each quantum bit, we can obtain a balanced superposition of all computational bases, the maximum mixed state $[1, 1, \cdots, 1]^T/\sqrt{2^n}$. The initial state preparation process is equivalent to acting on this superposition state with a diagonal matrix formed by the initial conditions, decomposing the initial state preparation process into basic gate operations, which corresponds

to the unitary decomposition of this diagonal matrix. However, there is currently no effective decomposition algorithm that can make the corresponding quantum algorithm have logarithmic complexity for any initial state with respect to the total number of grid points 2^n, initial state condition number κ, and precision $1/\varepsilon$. The time complexity of accurately preparing an n-qubit initial state is usually as high as $\mathcal{O}(2^n)$. This initial state preparation problem is expected to be solved in the future with the help of Quantum Random Access Memory (QRAM) technology.

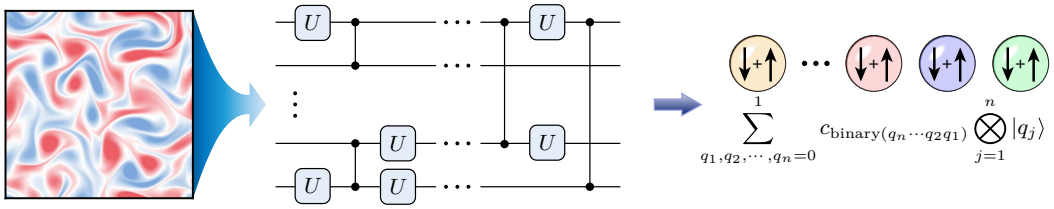

Figure 3: Encoding flow field information into a quantum state and decomposing the quantum initial state preparation process into polynomial complexity basic gate operations is one of the challenges faced by QCFD.

In the equation solving of Quantum Computing for Fluid Dynamics (QCFD), due to the linear nature of quantum mechanics, transforming the temporal evolution of nonlinear fluid dynamics equations into the evolution of quantum states is also a significant challenge. Apart from quantum measurement operations, the unitary, linear, and reversible evolution operators of quantum mechanics can be implemented through a series of unitary quantum gates (quantum circuit model). Assuming the velocity field is encoded by the amplitude of the wave function, meaning the magnitude of the velocity components at all grid points is encoded by the amplitude of the wave function, the no-cloning theorem prohibits the use of temporary copies of these amplitudes, which prevents the computation of nonlinear terms such as u^2 in a quantum computer by storing a temporary copy $u' = u$ and then computing $u \times u'$.

Furthermore, due to the normalization constraint of the two-norm of quantum states, the magnitude of all amplitudes is less than 1 (unless there is only one non-zero amplitude with a magnitude of 1). After nonlinear operations, the two-norm of the quantum state will inevitably be less than 1, losing normalization, and thus the corresponding operator cannot be unitary. Additionally, since nonlinear evolution operators rotate the quantum state by an angle depending on the state itself, the orthogonality of the eigenstates cannot be guaranteed. Lastly, because real flows have viscous dissipation, the time evolution operator is typically non-unitary, thus it is often necessary to introduce auxiliary variables to expand the Hilbert space, thereby mapping non-unitary operators to unitary operators.

When dealing with nonlinear problems, some studies have attempted to design quantum algorithms based on linearization operations, such as Carleman linearization, transformation into the Fokker-Planck equation, the Koopman-von Neumann equation, and

the Liouville equation. However, these linearized equations require the original problem to be lifted to a higher dimension, and quantum algorithms can achieve effective acceleration only when the dimension after lifting is polynomial in the original problem's dimension, which usually corresponds to weakly nonlinear cases dominated by dissipation. For strongly nonlinear chaotic systems, these methods have been proven to have limitations and are difficult to achieve effective quantum acceleration.

In the post-processing of QCFD, although flow field information such as density, velocity, and temperature is encoded in the quantum state, it cannot be directly extracted by traditional hard disk reading methods and must be reconstructed through statistical measurement according to the principles of quantum mechanics. After each measurement, the quantum state collapses to an eigenstate of the measurement operator, so the process of initial state preparation and quantum state evolution must be repeated to obtain the quantum state needed for measurement. To reconstruct all flow field information through statistical measurement, a complete quantum state tomography is required, i.e., measuring the expectation values of each Pauli string (a sequence composed of Pauli operators I, X, Y, Z). Therefore, for n qubits, the complexity of measurement is as high as $\mathcal{O}(4^n)$. However, in practical flow problems, we usually do not need to focus on the detailed flow field information at all grid points, but are more concerned with some statistical quantities, such as precipitation probability and aircraft lift. Therefore, there is no need to extract all the vast amount of information stored in the quantum state. To reduce the cost of measurement, the quantum state can be pre-processed before measurement, such as by designing ingenious unitary transformations to concentrate the statistical quantities to be measured onto a few qubits, thus allowing precise measurement at a lower cost.

In summary, QCFD faces three major core challenges: initial state preparation, quantum state evolution, and measurement. Initial state preparation is a common problem in the intersection of scientific computing and quantum computing, while designing quantum state evolution algorithms specifically for nonlinear fluid dynamics equations is a unique challenge for QCFD. Additionally, although measurement is also a common problem in quantum computing, developing efficient fluid statistical measurement methods also requires the integration of fluid mechanics expertise.

6 Progress in Quantum Algorithms for Fluid Dynamics

With the rapid development of quantum computing software and hardware, a variety of quantum algorithms for simulating fluid motion have emerged in recent years, giving rise to the research field of Quantum Computing for Fluid Dynamics (QCFD).

Current QCFD algorithms are mainly divided into two categories: one is the quantum-classical hybrid algorithm, where quantum and classical computers each take on some

of the computational tasks; the other is the Hamiltonian simulation algorithm, which implements the complete computational process on a quantum computer.

6.1 Quantum-Classical Hybrid Method

6.1.1 Quantum Linear Equation Solver

As described in Section 5.2, QCFD faces challenges related to nonlinearity and non-unitarity, and the limitations of quantum circuit depth in the NISQ (Noisy Intermediate-Scale Quantum) era hardware have led to the adoption of quantum-classical hybrid methods in most existing research in QCFD. In these methods, the quantum computer is responsible for performing computational tasks for which there are already efficient quantum algorithms, and its output is passed to the classical computer to complete the computational parts that are not yet suitable for quantum algorithms.

In a time step of a quantum-classical hybrid algorithm, it is first necessary to use a quantum algorithm to transform the initial quantum state $|\psi_0\rangle$ into an intermediate quantum state $|\psi_1\rangle$, and then measure $|\psi_1\rangle$ to obtain an observable O_1. Then, a classical algorithm is used to calculate O_2 based on O_1, which usually involves a lot of strong nonlinear operations, and O_2 is used to reconstruct a new quantum state $|\psi_2\rangle$. This new state will serve as the initial state $|\psi_0\rangle$ for the next time step, in order to carry out the quantum state evolution within the next time step.

For n qubits, the transition from the quantum state $|\psi_1\rangle$ to the observable O_1 requires quantum measurement. Since the quantum state cannot be reused, the preparation circuit for $|\psi_1\rangle$ must be run multiple times, and the measurement complexity is $\mathcal{O}(4^n)$. On the other hand, reconstructing the quantum state $|\psi_2\rangle$ from the observable O_2 involves the process of initial state preparation described in Section 5.2, and the complexity of preparing any initial state precisely is $\mathcal{O}(2^n)$. Therefore, the exchange of quantum and classical data within each time step significantly increases the computational complexity of the quantum-classical hybrid algorithm, which, given current storage technology, may result in the entire computational process not being effectively accelerated.

A typical quantum-classical hybrid algorithm integrates a quantum linear system solver into traditional CFD algorithms. The core idea is to use current efficient quantum algorithms to solve sparse linear systems of equations, such as the Harrow-Hassidim-Lloyd (HHL) algorithm, unitary matrix linear combination (LCU), quantum singular value transformation, random methods, iterative methods, and quantum adiabatic methods, etc.

Currently, the most complex optimal discrete quantum adiabatic algorithm has been reduced to a computational complexity of $\mathcal{O}(\log(N)\kappa\log(1/\varepsilon))$, where N is the matrix size, ε is the error, and κ is the matrix condition number. These quantum linear solvers are embedded as independent modules into CFD algorithms, while other nonlinear oper-

ations such as matrix assembly and the computation of non-homogeneous column vectors are still performed on classical computers. Another similar method is to integrate quantum algorithms for solving the Poisson equation into traditional CFD algorithms, where nonlinear operations in source term computation also need to be executed on classical computers.

6.1.2 Quantum Lagrangian Vortex Method

The Lagrangian Vortex Method (LVM) is a class of numerical methods that simulate fluid motion by tracking discrete vortex elements representing vorticity. It has been widely applied in aerodynamics, ocean engineering, and climate simulation. Although LVM excels in handling complex boundary conditions and strong nonlinear flows, its vortex element initialization and update process are computationally intensive.

Since the connection between quantum states and LVM can be analogized through the path dependence of vortex flux and Berry phase, quantum algorithms for LVM are expected to significantly improve the computational efficiency of vortex element initialization and evolution. In the quantum vortex method, initial state preparation requires extracting vortex element positions from a given velocity field and encoding them into quantum states. This problem can be formulated as finding a set of directed curves (points in two dimensions) representing vorticity lines with a strength of $\hbar/(2\pi)$ in a given velocity field $\boldsymbol{u}$ and an intensity parameter $\hbar > 0$, approximating the vorticity field $\boldsymbol{\omega} = \boldsymbol{\nabla} \times \boldsymbol{u}$. Weissmann et al. proposed implicitly representing vortex lines through the zero set of a complex function field ψ, that is, by solving the intersection of the equal value surfaces of $\mathrm{Re}(\psi) = 0$ and $\mathrm{Im}(\psi) = 0$ (as shown in Figure 4), converting the vortex extraction problem into an optimization problem.

$$\min_{\|\psi\|=1} \langle \psi | \hat{H} | \psi \rangle \tag{3}$$

Here, the operator $\hat{H}$ is determined by the given velocity field. Solving this optimization problem using classical methods is usually very time-consuming, and the acceleration potential of quantum computing provides a possible solution.

For the subsequent evolution of vortex elements, the computational complexity of the interactions between vortex elements in the classical Lagrangian Vortex Method (LVM) grows exponentially with the number of vortex elements. Therefore, if the exponential speedup advantage of quantum computing can be utilized, it may be possible to break through the computational bottleneck of traditional LVM and achieve larger-scale fluid simulations. In the quantum LVM, considering the nonlinear characteristics of vortex motion, a linearization approximation can be made for the quantum states corresponding to the vortex elements, thus achieving time-step advancement. This approximation is similar to the idea of dynamic mode decomposition.

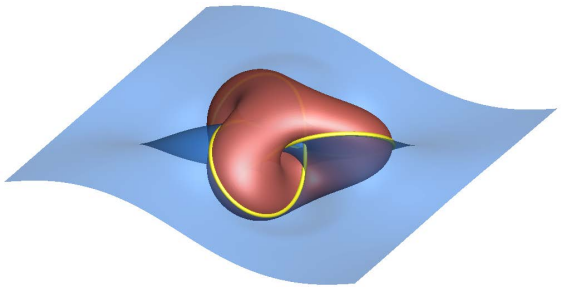

Figure 4: Vortex filaments are extracted from the zero contour of the complex scalar field ψ. The semi-transparent red surface represents the contour $\mathrm{Re}(\psi) = 0$, the semi-transparent blue surface represents the contour $\mathrm{Im}(\psi) = 0$, and the yellow curve is the intersection of $\mathrm{Re}(\psi) = 0$ and $\mathrm{Im}(\psi) = 0$.

Although quantum computing theoretically offers more efficient potential for LVM, and there have been studies attempting to introduce quantum state evolution into the classical algorithm of vortex methods, the quantum LVM still faces a series of common challenges in QCFD, especially since LVM can represent the fluid field as a low-dimensional structure, but efficiently extracting this structural information from the quantum state still requires new methods.

6.2 Hamiltonian Simulation Method

Since the quantum-classical hybrid algorithm involves frequent data exchanges between quantum and classical hardware, and these exchanges often take much longer than the computation time for solving the equations themselves at each time step, it is difficult to effectively accelerate the entire computational process. Therefore, to avoid frequent measurements and initial state preparations in quantum computing, the development of end-to-end QCFD algorithms is particularly important. Since quantum computers operate based on the principles of quantum mechanics, that is, the evolution of quantum states follows the Schrödinger equation, this type of Hamiltonian simulation method first needs to transform the fluid problem into the form of the Schrödinger equation:

$$i\hbar \frac{\partial}{\partial t}|\psi\rangle = \hat{H}|\psi\rangle, \tag{4}$$

where $i = \sqrt{-1}$, $|\psi\rangle$ is the state vector satisfying the normalization condition, and $\hat{H}$ is the Hamiltonian. To ensure that the corresponding evolution operator $U = e^{-i\hat{H}t}$ is unitary, the Hamiltonian must be Hermitian, i.e., $\hat{H}^\dagger = \hat{H}$. This method essentially transforms a classical problem into a quantum mechanical system described by a specific Hamiltonian, and then uses a quantum computer to simulate this Hamiltonian system.

Schrödingerization of General Linear Partial Differential Equations

After spatial discretization of a general linear partial differential equation, a system of linear ordinary differential equations is obtained:

$$\frac{\mathrm{d}u}{\mathrm{d}t} = Au + b, \tag{5}$$

where u is the vector to be solved, A is the coefficient matrix, and b is the non-homogeneous vector. However, direct encoding in the spatial direction usually does not allow the two-norm of the state vector to correspond to the conserved quantity of the system, that is, the state vector cannot remain normalized, corresponding to energy concentration or dissipation phenomena in the system's evolution. Therefore, the operator A obtained is non-Hermitian, corresponding to a non-unitary time evolution operator. This makes it impossible to directly use the Hamiltonian simulation method on a quantum computer to perform time integration on this semi-discrete equation.

7 Summary and Outlook

This paper firstly outlines the basic principles of quantum computing, analyzes the main challenges of fluid dynamics in the field of quantum computing, and then introduces recent research progress in quantum computing for fluid dynamics (QCFD).

The research direction of QCFD is still in its infancy. There are three major challenges in terms of algorithms: how to efficiently and accurately prepare the initial quantum state, simulate nonlinear and non-unitary time evolution, and measure key fluid statistical quantities. In particular, there is currently no effective quantum algorithm that can demonstrate accelerated performance over the best classical algorithms in simulating strongly nonlinear fluid dynamics problems. The focus of existing quantum algorithms is on solving linear and weakly nonlinear problems. The development of efficient quantum algorithms for general nonlinear problems in the time evolution of QCFD is a key research direction.

In terms of QCFD hardware, the current quantum computing prototypes across various technological routes can achieve a minimum single-gate error rate of approximately 10^{-3}, which is much higher than the error rate of classical computers, 10^{-18}. Moreover, classical computers maintain a low error rate and employ various error correction mechanisms to ensure the accuracy of computational results and the stable operation of the system. Therefore, reducing the interaction of quantum computers with the environment, lowering noise to reduce error rates, and implementing quantum error correction/reduction algorithms are key to achieving deep quantum circuit fluid simulations.

At the same time, the lifetime of qubits in existing quantum computing prototypes (related to bit relaxation time and decoherence time) is relatively short, especially in

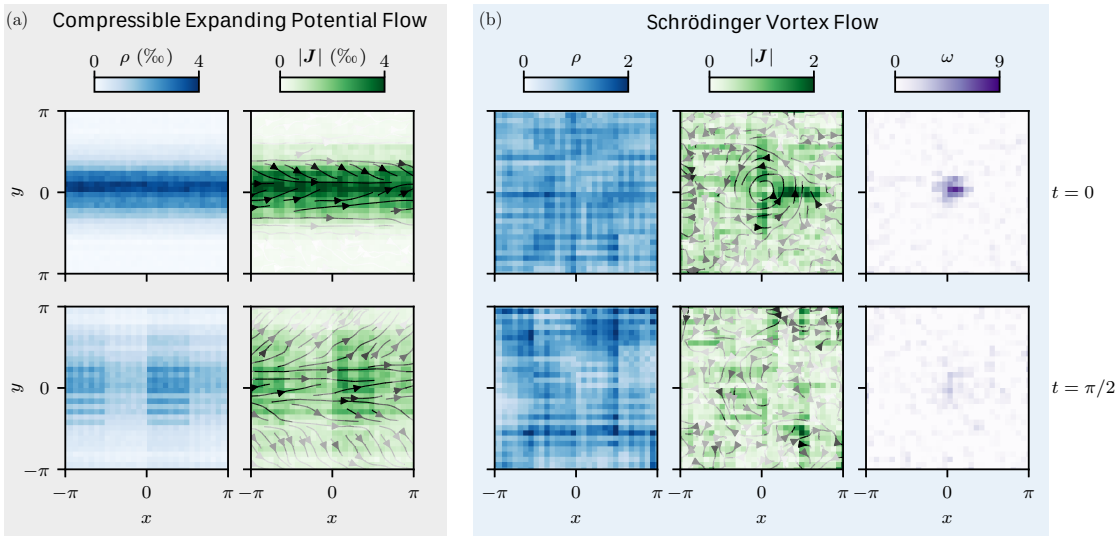

Figure 5: This figure shows the preliminary images that have been realized on a superconducting quantum computer, where experiments based on Hamiltonian simulation have been conducted for (a) compressible gradually expanding potential flow and (b) Schrödinger vortex flow. The measurements from top to bottom represent the results at times $t = 0$ and $\pi/2$, including the density field (blue), momentum field (green), and vorticity field (purple), with streamlines drawn in the momentum field.

superconducting quantum computers, where the lifetime of qubits is usually only a few hundred microseconds. This limits the depth of circuits in experiments. Therefore, it is necessary to enhance the stability and scalability of qubits in the future so that quantum computers can handle more complex algorithms and larger datasets. Additionally, the physical architecture of quantum computers may undergo significant changes, such as using new types of superconducting materials or topological insulators to construct qubits, to improve operational speed and reduce energy consumption.

8 References

1. R. P. Feynman. Simulating physics with computers. *Int. J. Theor. Phys.*, 21:467-488, 1982.

2. D. Deutsch. Quantum theory, the Church-Turing principle and the universal quantum computer. *Proc. R. Soc. London Ser. A-Math. Phys. Eng. Sci.*, 400:97-117, 1985.

3. R. P. Feynman, R. B. Leighton, and M. Sands. *The Feynman Lectures on Physics, Vol. II: The New Millennium Edition: Mainly Electromagnetism and Matter.* Basic Books, 2015.

4. G. Chiribella, G. M. D'Ariano, and P. Perinotti. Quantum circuit architecture. *Phys. Rev. Lett.*, 101:060401, 2008.

5. T. Albash and D. A. Lidar. Adiabatic quantum computation. *Rev. Mod. Phys.*, 90:015002, 2018.

6. A. Das and B. K. Chakrabarti. Colloquium: Quantum annealing and analog quantum computation. *Rev. Mod. Phys.*, 80:1061-1081, 2008.

7. C. Nayak, S. H. Simon, A. Stern, M. Freedman, and S. Das Sarma. Non-abelian anyons and topological quantum computation. *Rev. Mod. Phys.*, 80:1083-1159, 2008.

8. I. Buluta and F. Nori. Quantum simulators. *Science*, 326:108-111, 2009.

9. J. I. Cirac and P. Zoller. Goals and opportunities in quantum simulation. *Nat. Phys.*, 8:264-266, 2012.

10. I. M. Georgescu, S. Ashhab, and F. Nori. Quantum simulation. *Rev. Mod. Phys.*, 86:153-185, 2014.

11. S. McArdle, S. Endo, A. Aspuru-Guzik, S. C. Benjamin, and X. Yuan. Quantum computational chemistry. *Rev. Mod. Phys.*, 92:015003, 2020.

12. A. J. Daley, I. Bloch, C. Kokail, S. Flannigan, N. Pearson, M. Troyer, and P. Zoller. Practical quantum advantage in quantum simulation. *Nature*, 607:667-676, 2022.

13. H.-J. Shao, Y.-X. Wang, D.-Z. Zhu, Y.-S. Zhu, H.-N. Sun, S.-Y. Chen, C. Zhang, Z.-J. Fan, Y. Deng, X.-C. Yao, Y.-A. Chen, and J.-W. Pan. Antiferromagnetic phase transition in a 3D fermionic Hubbard model. *Nature*, 632:267-272, 2024.

14. D. Bernstein and T. Lange. Post-quantum cryptography. *Nature*, 549:188-194, 2017.

15. S. G. Louie, Y.-H. Chan, F. H. da Jornada, Z. Li, and D. Y. Qiu. Discovering and understanding materials through computation. *Nat. Mater.*, 20:728-735, 2021.

16. R. Santagati, A. Aspuru-Guzik, R. Babbush, M. Degroote, L. González, E. Kyoseva, N. Moll, M. Oppel, R. M. Parrish, N. C. Rubin, M. Streif, C. S. Tautermann, H. Weiss, N. Wiebe, and C. Utschig-Utschig. Drug design on quantum computers. *Nat. Phys.*, 20:549-557, 2024.