Computers and Fluids 270 (2024) 106148

Contents lists available at ScienceDirect

Computers and Fluids

journal homepage: www.elsevier.com/locate/compfluid

Ensemble fluid simulations on quantum computers

Sauro Succi [a,*], Wael Itani [b], Claudio Sanavio [a], Katepalli R. Sreenivasan [b,c,d], René Steijl [e]

[a] Fondazione Istituto Italiano di Tecnologia, Center for Life Nano-Neuroscience at la Sapienza, Viale Regina Elena 291, 00161 Roma, Italy
[b] Tandon School of Engineering, New York University, Brooklyn, New York, NY 11201, United States of America
[c] Courant Institute of Mathematical Sciences, New York University, New York, NY 10012, United States of America
[d] Department of Physics, New York University, New York, NY 10003, United States of America
[e] James Watt School of Engineering, University of Glasgow, G12 8QQ Glasgow, United Kingdom

ARTICLE INFO

Keywords:
Fluid dynamics
Liouville
Quantum algorithm
Kinetic theory
Lattice
Navier–Stokes

ABSTRACT

We discuss the viability of ensemble simulations of fluid flows on quantum computers. The basic idea is to formulate a functional Liouville equation for the probability distribution of the flow field configuration and recognize that, due to its linearity, such an equation is in principle more amenable to quantum computing than the dynamic equations of fluid motion. After suitable marginalization and associated closure, the Liouville approach is shown to require several hundreds of logical qubits, hence calling for a major thrust in current noise correction and mitigation techniques.

1. Introduction

The extreme complexity of most problems in modern science and society poses a very steep challenge to our best theoretical and computational methods. As an example, even the most powerful supercomputers, reaching up to exascale operations (one billion billions floating point operations per second) pale in front of the task of predicting the weather on the planetary scale, based on the direct simulation of the equations of fluid motion [1]. Besides, this and similar problems are typically subject to various sources of uncertainty arising from the initial data and other parameters affecting the solution. As a result, each single case-study requires several realizations in order to accumulate sufficient statistical information (Ensemble Simulations), further reinforcing the quest of computational power.

Given that electronic computers are facing very stringent energy constraints, alternative simulation strategies are constantly sought. Among these, enormous efforts have been devoted in the last decade towards the development of quantum computers, using hardware devices capable of exploiting the ability of quantum systems to occupy a multitude of states at the same time (quantum entanglement). The immediate advantage is that a quantum system can *in principle* perform a multitude of parallel quantum computations, as opposed to classical computers which can only operate on binary states (bits). Lately, not a day goes by without hearing the last quantum computing breakthrough, but, leaving aside the hype [2], the fact remains that turning the immense potential of quantum computing into a concrete tool for scientific purposes remains a very challenging goal. The reasons are many, but, in a nutshell, the bottomline is that entanglement is very fragile and tends to crumble pretty quickly under the effects of environmental noise, which is extremely hard to avoid at any reasonable temperature—a problem know as "decoherence". Notwithstanding these major barriers, it is worth exploring the contributions that quantum computers can possibly make to the prospect of ensemble simulations of fluid flows.

2. Ensemble simulations

Ensemble simulations have gained popularity in the recent years, thanks to the availability of large supercomputers. The main idea is accumulate statistics over the many sources of uncertainties that are associated, for instance with weather forecasting, by running series of simulations with different initial conditions and/or parametric realizations [1,3].

To illustrate the idea we consider a set of nonlinear partial differential equations and discretize them on a grid with, say, G grid points. Let $u(t)$ be the set of unknowns after discretization, for instance the three-dimensional velocity field of a fluid flow; they obey a set of $O(G)$ first-order ODE's in (generalized) Langevin form, given by

$$\frac{du}{dt} = f(u;\lambda), \tag{1}$$

with initial conditions $u(0) = u_0$. In the above, λ stands for a set of parameters subject to various sources of uncertainty, thus acting

* Corresponding author.
E-mail addresses: sauro.succi@gmail.com (S. Succi), itani@nyu.edu (W. Itani), claudio.sanavio@iit.it (C. Sanavio), katepalli.sreenivasan@nyu.edu (K.R. Sreenivasan), rene.steijl@glasgow.ac.uk (R. Steijl).

https://doi.org/10.1016/j.compfluid.2023.106148
Received 11 April 2023; Received in revised form 29 July 2023; Accepted 3 November 2023
Available online 9 December 2023

like "noise" on the system. Ensemble simulations correspond to the generation of statistics of solutions upon changing initial conditions and/or perturbing the system parameters. Formally, this amounts to generating a probability distribution function (PDF) for the solutions $u(x,t)$, defined by:

$$p(u,t)|\delta u| = \frac{\delta t}{T}, \tag{2}$$

where δt is the time spent by the set of trajectories, spanning the time interval $[0 \le t \le T]$, in a volume of phase-space $|\delta u|$. Generating the trajectories is extremely demanding, since by construction each single simulation is set to stress the most advanced computer resources to their limit [3,4].

Quantum computing could help realizing an exponential speedup on each of these simulations. However, besides all standard concerns affecting quantum computing, two additional issues stand on the way of this program: *quantum mechanics is linear and energy-conserving, while the physics of fluids is mostly nonlinear and dissipative*.

Several ways around these problems are currently under exploration, based on various strategies, some of which resort to Carleman linearization of the fluid equations [5–8], while some others leverage nonlinear quantum ODE solvers [9]. In computational fluid dynamics applications of quantum computers, the use of a hybrid quantum/classical formulation is the most widely used approach to deal with the non-linearity of the governing equations of fluid dynamics [9–12], effectively by accounting for this in the classical part of the algorithm. All of these, however, focus on the solution of the dynamic equations of motion, with no focus on ensemble simulations.

In this brief note, we sketch a potential strategy which offers two major assets at the outset. First, it captures by construction all the statistical information that is sought on the system dynamics (statistical dynamics). Second, it does not resort to any linearization of the dynamic equations, but starts directly from an inherently linear representation of the corresponding probability distribution function (PDF).

The passage from Newtonian dynamics to statistical dynamics is a standard topic in statistical physics, where it is known as Liouville formulation of classical N-body problem. This formalism is elegant and conducive to very valuable approximations, mostly at the level of one-body effective kinetic equations, the most outstanding examples of which are the Boltzmann and Fokker–Planck equations.

Unfortunately, at least on classical computers, working with Liouville equations is unviable since the N-body distribution function lives in a N-dimensional space, with N of the order of the number of grid points of the dynamic simulation, hence easily in the order of billions or more for current supercomputer simulations.

This looks like a "medicine-is-worse-than-the-disease" scenario, and it is therefore of interest to explore what quantum computing could possibly contribute to ease up the difficulty.

3. Functional Liouville equation

By virtue of the Liouville theorem (Fig. 1), the N-point PDF associated with the (nonlinear) Langevin equations obeys a linear Liouville Fokker–Planck kinetic equation (LFPE) of the form

$$\partial_t p_N + \sum_{i=1}^{N} \partial_{u_i}[f(u)p_N - D\partial_{u_i} p_N] = 0, \tag{3}$$

where D is the diffusion coefficient associated with the noise in the (linear) Langevin equation (1).[1] Since the Liouville equation is linear by construction, it can operate under the same framework as quantum

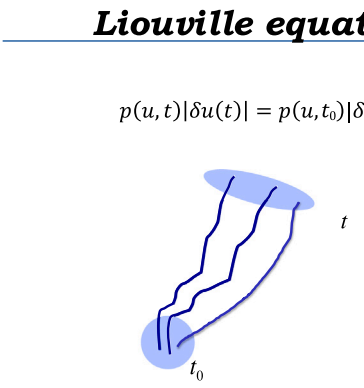

Fig. 1. Geometrical interpretation of the Liouville equation. The cloud of points representing various realizations of the system at $t = t_0$ evolves each along its own trajectory dictated by the dynamic equation $\dot{u} = f(u)$, with initial conditions $u(t_0) = u_0$. As time unfolds, the cloud changes its shape but not its volume (if $divu \neq 0$) and consequently the probability distribution $p(u,t)$ is invariant along the trajectory $dp/dt = 0$, leading to the Liouville equation.

mechanics and in particular, it can benefit of quantum linear-algebra solvers [13].

Before proceeding further, it is worth mentioning that higher-dimensional probabilistic approaches to quantum computing for nonlinear dissipative systems have been recently considered in the literature [14,15]. In these papers, the authors refer to linear representation as opposed to linearized approximation such as Carleman embedding, which is the same perspective taken here. However, the two approaches differ considerably in their scope and mathematical formulation. Indeed [14,15] transform nonlinear hyperbolic PDE's in $(d+1)$ dimensions into a linear transport equation for the corresponding PDF in $(2d+1)$ dimensions, for which no closure is required. In addition, Navier–Stokes dissipative equations are only marginally mentioned. Our approach is very different, in that the proposed Liouville equation lives in $O(G)$ dimensions where G, being the number of grid points in the classical simulations, is in principle in the order of billions, meaning by this that the closure problem is absolutely central. This said, the scope of this paper is not to solve the closure problem, for if one could, there would be no need of quantum computing in the first place. The point we wish to make in this paper is different; namely that quantum computing could potentially handle higher-order closures beyond reach of classical computers, hence provide potential new insights into the physics of turbulence correlators.

3.1. Taming the dimensional curse

The Liouville equation is very elegant but operationally unfeasible, since it lives in a ultra-dimensional space with as many dimensions as the number of grid sites where the direct simulations are performed; as already stated, this number is easily in excess of many billions for present-day supercomputers. This is the so called dimensional curse, affecting many problems in modern science and engineering.

The main merit of the Liouville equation, though, is that it opens up lower-dimensional approximations which can often capture the essence of the physical problem at hand. The technical procedure is called *marginalization* and consists of deriving equations for lower-order marginals of the original N-body PDF. Formally, this is obtained by projecting out the unwanted/unnecessary variables by integrating them out, as follows:

$$P_M(u_1 \dots u_M) = \int P_N(u_1 \dots u_N) du_{M+1} \dots du_N, \tag{4}$$

[1] Caveat: Since diffusion results from linearizing $f(u;\lambda)$ around a reference value λ_0, the corresponding diffusion coefficient is generally dependent on the actual flow field, that is, $D = D(u)$.

where $1 \le M \le N$ defines the order of the marginal distribution. By applying the above definition to the N-body Liouville equation (3), one readily obtains

$$\partial_t p_M + \sum_{i=1}^{M} \partial_{u_i}[F(u)p_M - D\partial_{u_i} p_M] = 0, \tag{5}$$

where

$$F(u) \equiv F(u_1 \dots u_M) = \int f(u_1 \dots u_N) p_N(u_1 \dots u_N) du_{M+1} \dots du_N \tag{6}$$

is the effective M-body force. Here we have assumed no-flux boundary conditions. From the above relation, it is immediately clear that the explicit expression of $F(u)$ generally depends on the unknown N-body PDF, signaling a much expected closure problem. This is, of course, a key issue for the success of the whole program, but in the following we shall proceed by assuming that a plausible closure can be found. In light of the major advances in statistical mechanics and the theory of coarse-graining this is, after all, a plausible assumption (though its precise realization has often defined major efforts).

Next, let us consider a generic observable $A(u_1 \dots u_N)$, whose average value is given by

$$\langle A \rangle(t) = \frac{1}{Z(t)} \int_{-\infty}^{+\infty} p(u,t) A(u) du, \tag{7}$$

where $Z(t) = \int_{-\infty}^{+\infty} p(u,t) du$ and u, as before, is a shorthand for $(u_1 \dots u_N)$.

If the dependence on each of the N independent variables u_j is irreducible, the average of u_j depends on the full N-body PDF $p_N(u_1 \dots u_N)$, with no room for marginalization. But this is rarely the case in classical physics. For instance, if $K = \sum_{j=1}^{N} u_j^2$ is the total kinetic energy of the fluid, its average depends only on the one-point distribution

$$\langle K \rangle(t) = \sum_{j=1}^{N} \int u_j^2 p_N(u) du = \sum_{j=1}^{N} \int u_j^2 p_1(u_j) du_j, \tag{8}$$

where $p_1(u_j)$ results from integrating out all variables u_1 to u_N, but u_j.

Likewise, if all we need is the value of the average velocity field at the space slice x_j, the 1-point PDF will suffice

$$\langle u_j \rangle(t) = \int u_j p_1(u_j, t) du_j. \tag{9}$$

By the same token, two-point observables require two-point PDFs, and so on, at all higher orders. As a result, the physics of many-body systems is often handled via effective equations for low order marginals, where low means order one or two. Clearly, the effects of higher order correlations are taken into account only via approximate closures. For a scalar PDE with spatial connectivity z, it is plausible to expect that correlators up to order z would be relevant, so that effective equations for marginals of order z should capture the essential physics. For discretized versions of three-dimensional fluid equations the number of interacting variables around each lattice site is of the order of tens (see below), meaning that the corresponding marginal Liouville equation lives in tens of dimensions. For classical computers this is an extremely challenging task, to say the least, but on a quantum computers with the order of a few hundreds functional qubits, the task might be viable, as detailed below. The count goes as follows. The N-body PDF associated with a discrete grid with G lattice sites, each hosting F fields discretized over a set of n discrete values, takes on n^{GF} discrete values. The number of qubits to represent the fully N-body discrete PDF is then

$$q = G\ F\ log_2 n. \tag{10}$$

Given that G is in the order of many billions for present-day supercomputer simulations, this requires multi-billions logical bits, a number which appears to be totally unrealistic in any foreseeable future. Fortunately, marginalization presents a much more optimistic picture.

If each field on a discrete grid with G lattices sites is connected to $z \ll G$ neighbors, the lowest order irreducible marginal is of order $M = zF$ and the qubit count now reduces to

$$q = z\ F\ log_2 n. \tag{11}$$

This is still very demanding but vastly simpler than (10) since for the three-dimensional Navier–Stokes equations, the parameter zF is on the order of tens.

This shows that marginalization stands good chances to circumvent the dimensional curse on a quantum computer.

Before proceeding, it is worth mentioning that the Monte Carlo method forms a key aspect for the simulation of fluid dynamics using the Liouville-based approach outlined above, with specific regard to the sampling of the solution (or certain features/moments of the PDF). It is in this Monte Carlo sampling that the previous work by Xu et al. [16] is shown to have the potential for a quadratic speed-up relative to classical approaches. Specifically, the number of samples (or evaluations) for a desired accuracy ϵ scales as $1/\epsilon$ for the proposed quantum algorithm, while the scaling is $1/\epsilon^2$ using the classical Monte Carlo sampling [17–19]. This quadratic complexity improvement in terms of number of samples/evaluations has also been noted in further quantum algorithms using the Monte Carlo approach, e.g. quantum algorithms in computational finance. For the Liouville-based quantum algorithm proposed here, it can therefore be expected that this kind of quadratic improvement in terms of sampling will occur as well.

4. Practical examples

In the following, we provide details of the Liouville formulation for the case of the Burgers equation, describing a one-dimensional pressure-free fluid, and for the three-dimensional Navier–Stokes equations.

4.1. The Burgers-Liouville equation

The Burgers equation describing one-dimensional pressureless fluids, reads as

$$\partial_t u + u\partial_x u = \nu \partial_{xx} u, \tag{12}$$

where ν is the kinematic viscosity.

A simple centered-finite difference scheme gives

$$\dot{u}_j = -u_j(u_{j+1} - u_{j-1}) + \nu(u_{j+1} - 2u_j + u_{j-1}) \equiv \sum_{k=-1}^{1} B_{j,k}(u_j) u_{j+k} \equiv f_j(u), \tag{13}$$

where the space step is made unity for simplicity and $B_{j,j+k}$, $j = 1, \dots, N$, $k = -1, 0, 1$ is the (nonlinear) Burgers matrix. In explicit form, we have:

$$B_{j,j-1} = \nu - \frac{1}{4}u_j,\ B_{j,j} = -2\nu - \frac{1}{4}u_j,\ B_{j,j+1} = \nu + \frac{1}{4}u_j. \tag{14}$$

The N-point Burgers–Liouville equation takes the following form:

$$\partial_t p_N + \sum_{j=1}^{N} \partial_{u_j}[\sum_{k=-1}^{1} B_{j,j+k} u_{j+k}] p_N. \tag{15}$$

Here $p_N \equiv p(u_1 \dots u_N, t)$ is the N-body PDF associated with the spatial grid of $G = N$ points (see Fig. 2).

Since $\dot{u}_j$ depends on the triplet (u_{j-1}, u_j, u_{j+1}), the lowest order irreducible marginal is the three-point PDF p_3, which is defined by integrating out all independent variables but three, u_{j-1}, u_j, u_{j+1}. This gives:

$$p_3(u_{j-1}, u_j, u_{j+1}) = \int_{\infty}^{+\infty} p(u_1 \dots u_N) du_1 \dots du_{j-2} du_{j+2} \dots du_N. \tag{16}$$

The corresponding 3-point kinetic equation takes the form:

$$\partial_t p_3 + \partial_{u_{j-1}}[\sum_{k=-1}^{1} B_{j-1,k} U_{j-1+k}] p_3 + \partial_{u_j}[\sum_{k=-1}^{1} B_{j,k} U_{j+k}] p_3$$

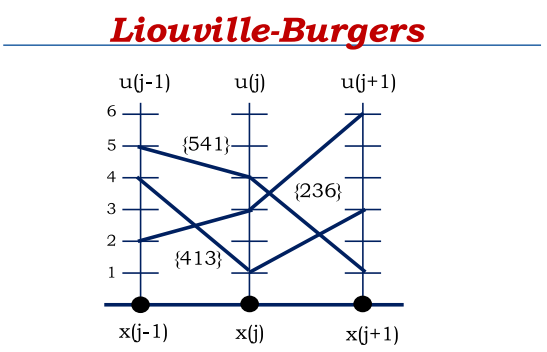

Fig. 2. Geometrical interpretation of the Liouville–Burgers equation. Each of the three independent variables u_j, u_{j-1}, u_j takes up to $n = 6$ values, hence the discrete Liouville–Burgers equation takes values on a set of three-points paths labeled by three integers n_{j-1}, n_j, n_{j-i}, each varying between 1 and 6. The figure reports the paths {413}, {236} and {541}.

$$+\partial_{u_{j+1}}[\sum_{k=-1}^{1} B_{j+1,k}U_{j+1+k}]p_3 = 0. \tag{17}$$

In the above we have defined

$$U_j(t) = \int u_j p(u,t) du_1 \ldots du_{j-2} du_{j+2} du_N, \tag{18}$$

which is a generally unknown function of u_{j-1}, u_j, u_{j+1} (as well as u outside this 3-point stencil). Hence a suitable expression for U_j as a function of u_{j-1}, u_j, u_{j+1} is required. This is a specific instance of the more general closure problem, which consists of formulating the effective equation for the lower order marginal PDF solely in terms of the lower order set of independent variables, in this case u_{j-1}, u_j, u_{j+1}. More general cases shall be discussed in the section devoted to the Navier–Stokes equations. Section 7 provides a further analysis of the kind of closures required and challenges involved.

With reference to the expression (11), we have $z = 3$ and $F = 1$, hence the corresponding qubit count gives

$$q = 3\, log_2 n. \tag{19}$$

Current quantum computers feature up to $q \sim 500$ *nominal* qubits [20], Here it should be noted that this represents an estimate for the storage of the discretized Liouville–Burgers system, without accounting for additional qubits required in an actual quantum computer implementation related to the Monte-Carlo sampling and the time-integration approach. As sketched later in Section 5, the time-integration method used can be expected to require an additional number of qubits $\propto log_2(M)$, where M is the number of time slices. Also, *nominal* refers to implemented qubits with short coherence times. IBM's current 100 × 100 Challenge illustrates how using the ever-growing number of implemented qubits can be used to realize fewer *functional* qubits with a significantly improved fault-tolerance (and therefore increased achievable quantum circuit depth). For a reasonable resolution, say $n \sim 1000 \sim 2^{10}$, we obtain $q \sim 30$ from Eq. (19). Assuming that in the near future $O(100)$ functional qubits can be used with a further improvement in noise levels and decoherence relative the hardware in the 100 × 100 Challenge, it appears that proof-of-concept realizations will become viable within a few years.

4.2. Example 2: The Navier–Stokes Liouville equation

The Navier–Stokes governing the motion of compressible, dissipative fluids read as follows

$$\partial_t \rho + \partial_a(\rho u_a) = 0, \tag{20}$$

$$\partial_t(\rho u_a) + \partial_b(\rho u_a u_b + P\delta_{ab} - \sigma_{ab}) = 0, \tag{21}$$

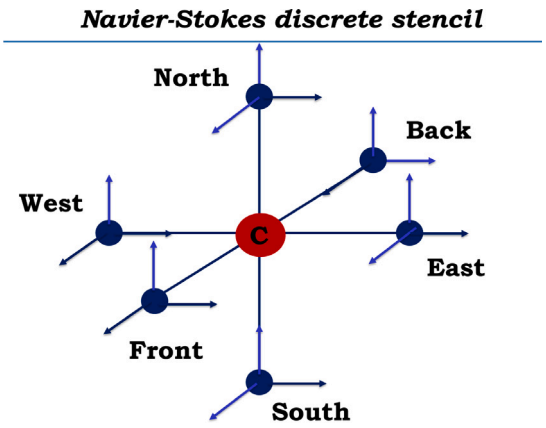

Fig. 3. Geometrical stencil for the discretization of the Navier–Stokes equations. Each lattice site hosts four fields, the scalar density ρ and the three components of the velocity field u, v, w. Each of the $4 \times 7 = 28$ stored in the nearest neighborhood of the center site C, namely East, West, North, South, Front, Back, represents an independent dimension in the two-body Liouville–Navier–Stokes equation. Not all of the 28 components are however directly coupled, depending on the chosen discretization scheme.

where ρ is the density, u_a, $(a = x, y, z)$ the flow velocity, $P = P(\rho)$ the fluid pressure and σ_{ab} the dissipative tensor. In discrete form and with a reference cell centered in C and connected to six neighbors E, W, N, S, F, B (East, West, North, South, Front, Back) along the x,y,z axes respectively, we write:

$$\begin{aligned}
\frac{d\rho_C}{dt} &= \frac{1}{2}(\rho_W u_W - \rho_E u_E) + \frac{1}{2}(\rho_B v_B - \rho_F v_F) + \frac{1}{2}(\rho_S w_S - \rho_N w_N) \\
&= 0 \\
\frac{d(\rho_C u_C)}{dt} &= \frac{1}{2}(\rho_W u_W u_W - \rho_E u_E u_E) + \frac{1}{2}(\rho_B u_B v_B - \rho_F u_F v_F) \\
&\quad + \frac{1}{2}(\rho_S u_S w_S - \rho_N u_N w_N) = \frac{P_W - P_E}{2} \\
\frac{d(\rho_C v_C)}{dt} &= \frac{1}{2}(\rho_W u_W v_W - \rho_E u_E v_E) + \frac{1}{2}(\rho_B v_B v_B - \rho_F v_F v_F) \\
&\quad + \frac{1}{2}(\rho_S v_S w_S - \rho_N v_N w_N) = \frac{P_B - P_F}{2} \\
\frac{d(\rho_C w_C)}{dt} &= \frac{1}{2}(\rho_W u_W w_W - \rho_E u_E w_E) + \frac{1}{2}(\rho_B v_B w_B - \rho_F v_F w_F) \\
&\quad + \frac{1}{2}(\rho_S w_S w_S - \rho_N w_N w_N) = \frac{P_S - P_N}{2}
\end{aligned}$$

In the above (u, v, w) are the three components of the velocity field, ρ is the density and we have taken the mesh spacing $\Delta_x = \Delta_y = \Delta_z = 1$ for simplicity.

In the equations above, the dissipative term of the stress tensor was omitted. Including these terms and using a centered 2nd-order accurate finite-difference scheme means that lattice sites outside the stencil shown in Fig. 3 are needed. However, all required lattice site would still be in the group of $5^3 = 125$ lattice sites centered around C, with a total number of sites included in the stencil that is still $O(10)$.

With reference to the expression (11), we now have $F = 4$ (density and three velocity components) and $z = 7$ (each grid site connected to six nearest neighbors) for the case without dissipative terms, hence the corresponding qubit count gives

$$q = 28\, log_2 n. \tag{22}$$

With $n = 10^3$, we have $q = 280$, much larger than for Burgers, but still within the nominal capabilities of current quantum hardware [20]. However, 280 exceeds the number of functional qubits with reduced noise levels that can be realized on near term hardware. Also, for three-dimensional Navier–Stokes demonstrations, n will typically be much greater than 1000. This shows that for proof-of-concept demonstrations for Navier–Stokes, significant further steps in quantum hardware development are still needed.

However, it should be noted that, depending on the specific discretization scheme, not every variable in the cell center C connects with every other variable in the cell neighborhood $\{W, E, B, F, S, N\}$. For instance, from the first equation above, ρ_C connects only to twelve and from the second, we see that u_C connects only to *seventeen*. For instance, from the second equation we see that u_C connects to all the seven neighboring densities and ten of the velocities u, v, w. As a result, the Liouville equation needs basically half of the $zF = 28$ dimensions considered above, which halves the count of the qubits as well (inclusion of the dissipative operator brings in an additional four).

This shows that different representations or altogether different formulations, such as lattice Boltzmann [21,22] methods, may lead to more favorable scalings, this being an interesting topic for future research.

5. Sketch of the quantum algorithm

As mentioned above, a number of different quantum computing strategies have been proposed in the recent past to simulate fluid problems on quantum computers. While all of these methods need to handle the nonlinearity issue, in the present case there is no such need, since the problem is linear from scratch. Therefore, one can resort to quantum linear solver algorithm [13], as detailed below.

We start by writing the 3-point Liouville equation (for the Burgers equation) in the explicit conservative form

$$\partial_t p + \partial_u(Up) + \partial_v(Vp) + \partial_w(Wp) = 0, \tag{23}$$

where we have set $u \equiv u_{j-1}$, $v \equiv u_j$ and $w \equiv u_{j+1}$ and (U, V, W) are three supposedly known functions of (u, v, w). Upon discretizing the three-dimensional functional space (u, v, w), (u_i, v_j, w_k), $i, j, k = 1, n$, we obtain a set of $G = n^3$ ODE's of the form

$$\dot{p}_{ijk} + U_{ii',jk}p_{i',jk} + V_{i,jj',k}p_{ij'k} + W_{ij,kk'}p_{ij,k'} = 0, \tag{24}$$

where U, V, W are the discrete matrices associated with the three flux terms.

A simple Euler forward time marching delivers:

$$p^{t+1} - (1 - Ldt)p^t = 0, \tag{25}$$

with the initial condition $p^0 = p_0$, t labeling discrete time, and p_0 being the initial condition, all spatial indices being suppressed for simplicity and L denotes the sum of the matrices, $L = U + V + W$. The above relations deliver a linear system for the unknown $p = p^{(0)}, p^{(1)} \dots p^{(M)}$, each component being an array of dimension n^3. In explicit form,

$$\begin{aligned} &p^{(0)} = p_0 \\ &p^{(m+1)} - (1 - Ldt)p^{(m)} = 0 \\ &m = 0, \dots M - 1, \end{aligned} \tag{26}$$

where $M = T/dt$ is the number of time slices. This is a linear system $Ap = b$, with a sparse lower-triangular matrix structure and b features just a single entry, p_0. It is still formally a linear system; as such, it can be handled by quantum linear solver algorithms, as shown by Berry and co-workers [23,24] for inhomogeneous linear differential equations with time independent coefficients. Using higher-order time-stepping methods, the algorithm in [23] has a computational complexity $\propto T^2 \ poly(log(G), 1/\epsilon)$. This time complexity represents the required number of calls to the quantum oracle implementing the linear system, while the complexity in space (qubit count) and time (circuit depth) of this quantum oracle was not addressed. Here, T is the considered time-span, G the grid size, and ϵ is the error tolerance. In subsequent work [24], this complexity was improved exponentially in terms of dependency on ϵ from $poly(1/\epsilon)$ to $poly(log(1/\epsilon))$. In more recent work [5], this approach was extended to dissipative nonlinear differential equations. For cases with a sufficiently small ratio of nonlinearity and forcing to the linear dissipation (as detailed in their work), the algorithm presented has a computational complexity $\propto T^2 \ q \ poly(log(T), log(G), log(1/\epsilon))/\epsilon$, where q represents decay of the solution, highlighting that time complexity increases for cases where there is a stronger decay of the (amplitudes defining the) solution relative to initial state. In terms of space complexity, the time-integration approach defined in Eq. (26) and used in previous works [23,24] implies a linear growth of the system size (Hilbert space) with T, and for storage on a quantum computer a $log_2(T)$ dependency of the number of qubits required.

The time complexity of a classical explicit algorithm for time-integration of linear differential equation would scale like

$$C_c \propto T \ G, \tag{27}$$

showing that the main advantage is the $poly(log(G))$ factor in the quantum algorithm, partly reabsorbed by the T^2 dependence on the time-span. Since $T \sim G^{1/3}$, for large grids with $G \sim 10^9$, there is a clear potential for a quantum speed-up

It should be noted that while the dependency of the computational complexity on the spatial size G can be brought under $poly(log(G))$ thanks to the exponential size of Hilbert space, the same is not possible with time. As discussed in the previous work cited above [5,24], a reduction in time complexity to $O(T)$ is feasible for Hamiltonian simulations of quantum dynamics, where the linear system of differential equations is homogeneous and the matrix defining the differential equation is anti-Hermitian, so that no decay takes place. For more general linear differential equations such as those arising in the current approach, the extent of decay of the solutions will be a key factor and requires further investigation in future work.

So far, the complexity discussion has mainly focussed on time complexity. In terms of a quantum circuit implementation, this shows how the number of time slices (during which independent quantum gate operations take place) increases as function of G, T, and further factors detailed above. The space complexity of the approach presented here has so far been mainly illustrated using Eqs. (10) and (11), where the extent of marginalization is crucial in obtaining an acceptable number of qubits. Clearly, the $log_2(n)$ dependency of the required number of qubits shows exponential improvement relative to a classical computer implementation, and for large values of n this is clearly significant. As mentioned earlier, the M-step time integration based on solution of linear systems of equations will require an additional number of qubits that can be expected to scale as $log_2(M)$. To summarize, for the Burgers–Liouville algorithm, the complexity analysis sketched here suggests that for $O(100)$ qubits on a future quantum computer with a much improved level of fault-tolerance, initial small-scale proof-of-concept realizations can be considered. A key challenge remains the required quantum circuit depth of such an implementation. This will be orders of magnitude larger than any quantum circuit currently realized, and out of reach for the hardware used in IBM's 100×100 Challenge.

6. Comparison with the dynamic approach

In the beginning of this paper, we have cautioned the reader about the practical unviability of the Liouville approach on classical computers, as the problem occupies a space with about thirty dimensions. Hence, on classical computers there is no option but solving repeated realizations of the fluid equations. It is therefore of interest to assess the cost of the dynamic approach on quantum computers. Current approach provide log scaling in the number of grid points and typically a quadratic scaling in time (when using algorithms such as [5]). With such a scaling, running a billion grid points (10^9) over a million time steps, would take about 10^{21} operations, hence that many dynamic degrees of freedom. By running an ensemble of 1000 simulations, say, this comes to 10^{24}. If the required number of qubits is estimated to be between $80 - 100$, this would still be much less than for the Liouville equation. The problem though is that time-marching for nonlinear partial differential equations on a quantum computers involves the reconstruction of the full quantum state at each time step (by virtue of

the no-cloning theorem), an operation which scales exponentially with the number of qubits. For the case of a billion grid points, requiring more than 30 qubits, this adds a dramatic slowdown, which is a major issue in quantum computing [25].

It thus appears that ensemble simulations of fluid flows on quantum computers are best performed via the Liouville approach, provided (i) a sensible closure can be worked out, and (ii) hundreds of reliable logical qubits can be used. Finally, we remark that a similar statement applies to basically any nonlinear field theory.

7. On the closure again

It is apparent that although the viability of the Liouville approach to quantum ensemble computing of fluid flows depends crucially on the availability of a suitable closure, finding an appropriate closure is a problem of its own regardless of quantum computing. In fact, if once could find a satisfactory solution to this problem there would be no need of quantum computing altogether! The main point of this paper is that a closure in tens of dimensions would still remain unviable on classical computers, while quantum computers would stand reasonable chances to handle it. Having clarified the point, let us nonetheless dig a bit deeper into the closure issue. Consider the following toy model: a nonlinear set of ODE's defined on a 3 lattice sites grid and forward nearest-neighbor connection:

$$\dot{u}_j = f(u_j, u_{j+1}), j = 1, 2, 3, \tag{28}$$

where periodicity is assumed. The corresponding 3-dimensional Liouville equation reads as follows

$$\partial_t P_{123} + \partial_1(f_{12}P_{123}) + \partial_2(f_{23}P_{123}) + \partial_3(f_{31}P_{123}) = 0 \tag{29}$$

where $P_{123} \equiv P(u_1, u_2, u_3)$ and $f_{ij} \equiv f(u_i, u_j)$. Next we marginalize to P_1 upon integrating upon $d2d3 \equiv du_2 du_3$. Assuming no-flux boundary conditions, this readily yields

$$\partial_t P_1 + \partial_1(\int f_{12}P_{123}d2d3) = 0 \tag{30}$$

indicating the need of a closure to evaluate the effective term

$$F_1 P_1 = \int f_{12}P_{123}d2d3 \tag{31}$$

To this purpose, we write

$$P_{123} = \langle \delta(u_1 - u_1(t))\delta(u_2 - u_2(t))\delta(u_3 - u_3(t)) \rangle \tag{32}$$

where brackets denote averaging over initial conditions. The above expression can be rewritten as

$$P_{123} = \langle (p_1 + q_1)(p_2 + q_2)(p_3 + q_3) \rangle \tag{33}$$

where p_i is the mean-field one-body contribution and q_i is the 3-body correlation on the ith side. Expanding, we obtain

$$\begin{aligned} P_{123} &= p_1 p_2 p_3 + (p_1 p_2 \langle q_3 \rangle + p_1 p_3 \langle q_2 \rangle + p_2 p_3 \langle q_1 \rangle) \\ &+ (p_1 q_2 \langle q_3 \rangle + p_2 \langle q_1 q_3 \rangle + p_3 \langle q_1 q_2 \rangle) + \langle q_1 q_2 q_3 \rangle \end{aligned} \tag{34}$$

On the assumption that $\langle q_i \rangle = \epsilon p_i$, the above expression can be ordered in powers of ϵ from order 0 to 3. At order zero, we obtain

$$F_1^{(0)} = \int f_{12}p_2 p_3 d2d3 = \int f_{12}p_2 d2 \tag{35}$$

which is self-consistent as it only depends on p_2, the one-body distribution at site 2. At first order, we obtain

$$F_1^{(1)} P_1 = p_1 \int f_{12}p_2 \langle q_3 \rangle d2d3 + p_1 \int f_{12}p_3 \langle q_2 \rangle d2d3 + \langle q_1 \rangle \int f_{12}p_2 p_3 d2d3 \tag{36}$$

The first two terms can be neglected by assuming $\int \langle q_2 \rangle d2 = \int \langle q_3 \rangle d3 = 0$, while the third delivers $q_1 \int f_{12}p_2 d2$. By taking $\langle q_1 \rangle \sim \epsilon p_1$ and accounting for the expression (35), we obtain

$$F_1^{(1)} = \epsilon F_1^{(0)} \tag{37}$$

It is now clear that by adopting the same procedure to the second and third order terms we obtain closed expressions for F_1 of order ϵ^2 and ϵ^3 respectively. In detail, at second order, we have

$$F_1^{(2)} P_1 = p_1 \int f_{12}\langle q_2 q_3 \rangle d2d3 + \int f_{12}p_2 \langle q_1 q_3 \rangle d2d3 + \int f_{12}p_3 \langle q_1 q_2 \rangle d2d3 \tag{38}$$

By taking $\langle q_2 q_3 \rangle = \epsilon^2 p_2 p_3$, the first yields $\epsilon^2 F_1^{(0)}$. By assuming $\int \langle q_3 \rangle d3 = \int \langle q_2 \rangle d2 = 0$, both second and third terms vanish. Finally at third order we have

$$F_1^{(3)} P_1 = \int f_{12}\langle q_1 q_2 q_3 \rangle d2d3 \tag{39}$$

which, by the usual assumptions, $\epsilon^3 F_1^0$. Of course, the assumptions made to close the Liouville equation at the first level are purely heuristic but they provide nonetheless an operational procedure to marginalize the Liouville equation. In fact, by identifying the many-body correlations q_i with fluctuations with zero mean and skewness, the above closures is tantamount to assuming a variance proportional to the mean field signal, namely $\langle q_i q_j \rangle = \epsilon^2 p_i p_j$, in which case only the even terms survive. Many other closures could be conceived, for instance, in analogy with turbulence modeling, one may want to assume that the variance of the fluctuations is not proportional to $p_i p_j$ but their gradients instead, i.e. $\langle q_i q_j \rangle = \epsilon^2 (p_i - p_j)/d_{ij}$, where d_{ij} is the distance between site i and j. This is of course a separate matter which goes beyond the scope of the present paper, the main point here being to illustrate potential operational closures of the Liouville–Navier–Stokes equations. The toy three lattice sites model becomes significantly more cumbersome for the case of realistic lattice sizes with trillions of grid point,

$$P_{1,2\ldots N} = \langle \prod_{i=1}^{N} (p_i + q_i) \rangle,$$

which involves long-range correlations of order up to N. However these are expected to decay with increasing powers of the corresponding order amplitude as well as of spatial separation. To this regard, it is reasonable to expect that the resort to tensor network techniques [26] may offer significant support to the task of marginalizing the Liouville–Navier–Stokes equation on realistic lattice sizes. Incidentally, they may also offer a new inroad to turbulence modeling, regardless of quantum computing [27].

8. Beyond fluids

Finally, it is worth pointing out that the Liouville formulation described in this paper applies to virtually any nonlinear field theory, at least in principle. Notable examples in point are the Kardar–Parisi–Zhang equation for nonlinear growth of interfaces [28], the Kuramoto–Sivashinski equation governing chemical turbulence [29] and the Korteweg–De Vries family of nonlinear dissipative equations [30]. Despite their different nature, all of these equations share the same feature of being first order in time and higher order in space. As a result, the corresponding Liouville equations present different connectivities, hence different computational complexity, but still fall within the general framework discussed in this paper.

9. Summary

Summarizing, we have estimated the viability of the functional Liouville formulation for ensemble simulations of fluid flows on quantum computers. The present analysis refers to a blue-sky scenario whereby a quantum algorithm capable of logarithmic scaling with the number of static degrees of freedom is available *and* running on *ideal* quantum computers, with no appreciable decoherence and/or noise problems.

In actual practice, quantum computing ensemble simulations of the Navier–Stokes equations demand hundreds of noiseless *logical* qubits.

Given that current quantum computing typically works only up to a few logical qubits, say of the order ten, the target appears pretty much into the future. This is no invitation to surrender, but just a realistic appraisal of the current state of affairs [31].

Declaration of competing interest

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

Data availability

No data was used for the research described in the article.

Acknowledgments

The authors have benefited from valuable discussions with many colleagues, particularly S.S. Bharadwaj, D. Buaria, P. Coveney, S. Das Sarma, N. Defenu, A. Di Meglio, M. Grossi, A. Mezzacapo, S. Ruffo, A. Solfanelli and T. Weaving. S.S. and C.S. acknowledge financial support form the Italian National Centre for HPC, Big Data and Quantum Computing (CN00000013).

References

[1] Tennie F, Palmer TN. Quantum computers for weather and climate prediction: The good, the bad and the noisy. 2022, arXiv:2210.17460v1 [quant-ph].
[2] Das Sarma S. Quantum computing has a hype problem. MIT Technol Rev 2022. https://www.technologyreview.com/2022/03/28/1048355/quantum-computing-has-a-hype-problem/.
[3] Navarra A, Tribbia J, Klus S. Estimation of koopman transfer operators for the equatorial Pacific SST. J Atmos Sci 2021;1227. http://dx.doi.org/10.1175/JAS-D-20-0136.1.
[4] Bhati AP, Wan S, Alfe D, et al. Pandemic drugs at pandemic speed: infrastructure for accelerating COVID-19 drug discovery with hybrid machine learning-and physics-based simulations on high-performance computers. Interface Focus 2021;11(6):20210018.
[5] Liu JP, Kolden HO, Krovi HK, et al. Efficient quantum algorithm for dissipative nonlinear differential equations. Proc Natl Acad Sci USA 2021;118(35):e2026805118.
[6] Mezzacapo A, Sanz M, Lamata L, et al. Quantum simulator for transport phenomena in fluid flows. Sci Rep 2015;5(1):1–7.
[7] Itani W, Succi S. Analysis of carleman linearization of lattice Boltzmann. Fluids 2022;7:24. http://dx.doi.org/10.3390/fluids7010024.
[8] Sanavio C, Succi S. Quantum lattice Boltzmann-Carleman algorithm. 2023, arXiv:2310.17973.
[9] Gaitan F. Finding flows of a Navier–Stokes fluid through quantum computing. Npj Quant Inf 2020;60.
[10] Budinski L. Quantum algorithm for the Navier–Stokes equations by using the streamfunction–vorticity formulation and the lattice Boltzmann method. 2022, arXiv:2103.03804v2 [quant:ph].
[11] Steijl R. Quantum algorithms for fluid simulations. 2019, http://dx.doi.org/10.5722/intechopen.86685.
[12] Bharadwaj SS, Sreenivasan KR. Quantum computing of fluid dynamics. Pramana Indian Acad Sci Conf Aer 2020;3:77–96.
[13] Harrow AW, Hassidim A, Lloyd S. Quantum algorithm for linear systems of equations. Phys Rev Lett 2009;103(15):150502.
[14] Jin Shi, Liu Nana, Yu Yue. Time complexity analysis of quantum difference methods for linear high dimensional and multiscale partial differential equations. J Comput Phys 2022;471. Paper (111641).
[15] Jin Shi, Liu Nana, Yu Yue. Time complexity analysis of quantum algorithms via linear representations for nonlinear ordinary and partial differential equations. J Comput Phys 2023;487:112149.
[16] Xu G, Daley AJ, Givi P, Somma RD. Turbulent mixing simulation via a quantum algorithm. AIAA J 2018;56(2):687–99.
[17] Montanaro A. Quantum speedup of Monte Carlo methods. Proc R Soc A 2015;471:20150301.
[18] Suzuki Y, Uno S, Raymond R, Tanaka T, Onodera T, Yamamoto N. Amplitude estimation without phase estimation. Quantum Inf Process 2020;19.
[19] Herbert S. Quantum Monte Carlo integration: The full advantage in minimal circuit depth. Quantum 2022;6:823.
[20] https://www.ibm.com/quantum/roadmap.
[21] Succi S. The lattice Boltzmann equation for fluid dynamics and beyond. Oxford U.P.; 2001.
[22] Benzi R, Succi S, Vergassola M. The lattice Boltzmann equation: Theory and applications. Phys Rep 1992;222(3):145.
[23] Berry DW. High-order quantum algorithm for solving linear differential equations. J Phys A 2014;47:105301.
[24] Berry DW, Childs AM, Ostrander A, Wang G. Quantum algorithm for linear differential equations with exponentially improved dependence on precision. Comm Math Phys 2017;356:1057–81.
[25] Lubasch M, Joo J, Moinier P, Kiffner M, Jaksch D. Variational quantum algorithms for nonlinear problems. Phys Rev A 2021;101(1):010301(R).
[26] Biamonte J, Bergholm V. Tensor networks in a nutshell. 2023, arXiv:1708.00006.
[27] Gourianov N, Lubasch M, Dolgov S, van den Berg QY, Babaee H, Givi P, Kiffner M, Jaksch D. A quantum inspired approach to exploit turbulence structures. 2022, arXiv:2106.05782v3 [physics.flu-dyn] 4 2022, see also https://ora.ox.ac.uk/objects/uuid:9e3f4786-ad68-4913-9a0d-e9b1e108128f.
[28] Kardar M, Parisi G, Zhang YC. Dynamic scaling of growing interfaces. Phys Rev Lett 1986;56(9):889–92.
[29] Kuramoto Y. Diffusion-induced chaos in reaction systems. Progress Theor Phys Supple 1978;64:346–67.
[30] Zabusky NJ, Kruskal MD. Interaction of solitons in a collisionless plasma and the recurrence of initial states. Phy Rev Lett 1965;15:240.
[31] Succi S, Itani W, Sreenivasan K, Steigl R. Quantum computing for fluids: Where do we stand? Europhys Lett 2023;144(1):10001.