Quantum algorithm for the linear Vlasov equation with collisions

Abtin Ameri* and Erika Ye
Plasma Science and Fusion Center, Massachusetts Institute of Technology, Cambridge, Massachusetts 02139, USA

Paola Cappellaro
Research Laboratory of Electronics and Department of Nuclear Science and Engineering, Massachusetts Institute of Technology, Cambridge, Massachusetts 02139, USA

Hari Krovi
Riverlane Research, Cambridge, Massachusetts 02142, USA

Nuno F. Loureiro
Plasma Science and Fusion Center, Massachusetts Institute of Technology, Cambridge, Massachusetts 02139, USA

(Received 10 March 2023; revised 8 May 2023; accepted 18 May 2023; published 12 June 2023)

The Vlasov equation is a nonlinear partial differential equation that provides a first-principles description of the dynamics of plasmas. Its linear limit is routinely used in plasma physics to investigate plasma oscillations and stability. In this paper, we present a quantum algorithm that simulates the linearized Vlasov equation with and without collisions, in the one-dimensional electrostatic limit. Rather than solving this equation in its native spatial and velocity phase space, we adopt an efficient representation in the dual space yielded by a Fourier-Hermite expansion. For a given simulation time, the Fourier-Hermite representation is exponentially more compact, thus yielding a classical algorithm that can match the performance of a previously proposed quantum algorithm for this problem. This representation results in a system of linear ordinary differential equations (ODEs) which can be solved with well-developed quantum algorithms: a Hamiltonian simulation in the collisionless case, and quantum ODE solvers in the collisional case. In particular, we demonstrate that a quadratic speedup in system size is attainable.

DOI: 10.1103/PhysRevA.107.062412

I. INTRODUCTION

Plasma dynamics is difficult to simulate with present-day computers due to the broad range of timescales and length scales that are typically exhibited by nonlinear plasma phenomena. Indeed, direct numerical simulations covering scale ranges approaching those found in most real systems are beyond the capabilities of both current supercomputers and those predicted to exist in the near future. It is therefore natural to seek alternative computational platforms that can provide significant speedups. Quantum computers are a possible candidate since there exist quantum algorithms capable of outperforming their classical counterparts for a range of problems such as search via quantum walks [1], simulation of quantum mechanical systems (also called Hamiltonian simulation) [2–8], cryptography [9], and simulation of high-energy physics problems such as $(1+1)$-dimensional ϕ^4 theory [10], fermionic field theory [11], and conformal field theory [12]. Furthermore, industrial and research efforts to commercialize quantum computers have significantly advanced the technology, bringing us closer to implementing quantum algorithms with real-world applications.

Although quantum algorithms exist to simulate Hamiltonians and solve ordinary differential equations (ODEs), it is *a priori* not clear if they can be used to solve specific plasma physics problems, which are usually formulated in terms of coupled sets of nonlinear partial differential equations. Indeed, the field of designing quantum algorithms for nonlinear differential equations and, specifically, plasma physics, is still at an embryonic stage. So far, there has been a thorough exploration of quantum linear system algorithms (QLSAs) [13,14], linear ODEs [15–18], nonlinear ODEs [19–22], and certain linear partial differential equations [23,24]. These algorithms use a Hamiltonian simulation as a main subroutine [25]. The application of such algorithms to real-world problems, such as plasma dynamics, is nontrivial and, in some cases, their promised speedup is lost. Nonetheless, there has been some progress in this field over the past few years. Dodin and Startsev [26] performed a survey of various approaches that could be used to solve plasma dynamics on quantum computers. Among those, they discussed the possibility of using the Madelung transform to map the governing equations of a cold electron fluid to the Schrödinger equation. The Madelung transform was used by Zylberman *et al.* [27] to develop a quantum algorithm as well as a hybrid algorithm simulating fluids. Linear embedding approaches have also been investigated, yielding a potential speedup if nonlinearities are weak

*aameri@mit.edu

[21,28] or if the resulting system is sparse [29,30]. Other works have focused on developing quantum algorithms for wave problems in plasmas, namely the three-wave interaction [31] and extraordinary waves (an electromagnetic wave in a plasma where the wave's electric field is perpendicular to the background magnetic field) in cold plasmas [32]. As one might expect, both these wave problems can be mapped to a Schrödinger-type equation, which can be solved using Hamiltonian simulation algorithms. A more general quantum lattice representation for one-dimensional wave propagation in plasmas has also been developed [33] (but such an algorithm requires four qubits per node).

Ideally, one would like to have a quantum algorithm to solve a first-principles description of plasma physics as offered by the Vlasov equation. A first step in this direction is to understand if the linear limit of the Vlasov equation can be simulated efficiently on a quantum computer. This problem was explored by Engel, Smith, and Parker (henceforth referred to as ESP), where a quantum algorithm was developed that simulates the Landau damping of Langmuir waves [34], a quintessential plasma physics phenomenon whereby electrostatic waves in the plasma resonantly transfer energy to electrons and are thus damped [35]. By linearizing the collisionless Vlasov equation coupled to Ampère's law, ESP mapped the problem to a Hamiltonian simulation which can then be solved using recent algorithms [34]. Using a spatial grid of N_v points, ESP gave a quantum algorithm to simulate the linearized Vlasov evolution with gate complexity of $O[\text{polylog}\, N_v \log(1/\epsilon)]$, with ϵ being the error in the norm of the solution state. To extract the Landau damping rate, the electric field at different times must be obtained by sampling the final state (so as to obtain a time history from which the Landau damping rate can be computed in postprocessing). The number of times one needs to repeat the algorithm is independent of N_v and depends only on the absolute precision δ with which one needs to calculate the electric field, leading to an overall run time of $O[(1/\delta)\text{polylog}\, N_v]$, where $\delta \sim \epsilon$.

In this paper, we revisit this problem with a different mathematical approach. We find that, using the linearized Vlasov equation coupled to Poisson's equation and expanding the distribution function in velocity space using $M+1$ Hermite polynomials (with $N = M+1$ corresponding to the system size), we can obtain a system matrix which is exponentially smaller than that of ESP for the same precision ϵ. Thus, a classical algorithm solving this system can be as efficient as the quantum algorithm of ESP. Furthermore, the Fourier-Hermite representation allows for the inclusion of collisions without changing the system matrix structure. We also find that a quantum algorithm using this framework yields a quadratic speedup compared to classical ODE solvers, which are the fastest classical algorithms for solving this problem [36].

This paper is structured as follows. Section II provides a brief background on the Vlasov equation, Landau damping, and the Hermite representation, and rigorously defines the problem we aim to solve in addition to its key parameters. Section III discusses the performance of a quantum algorithm based on the Hermite formalism for both collisionless and collisional cases, and demonstrates that a quadratic speedup is possible. Section IV reviews the ESP formulation and performs a comparative error analysis of that system and ours. Finally, Sec. V summarizes our results and discusses their implications.

II. BACKGROUND

A. The Vlasov equation

The collisional Vlasov-Poisson system is given by

$$\frac{\partial f_s}{\partial t} + \mathbf{v}\cdot\nabla f_s + \frac{q_s}{m_s}\mathbf{E}\cdot\nabla_v f_s = C[f_s], \tag{1}$$

$$\epsilon_0 \nabla\cdot\mathbf{E} = \rho, \tag{2}$$

where ϵ_0 is the permittivity of free space, s denotes a species (ions or electrons) of charge q_s and mass m_s, and ρ denotes the charge density. The collision operator is represented by $C[\cdot]$ and $\mathbf{E}$ is the electric field. We are going to restrict ourselves to the electrostatic limit, so $\mathbf{E} = -\nabla\varphi$, where φ is the electrostatic potential.

The Vlasov-Poisson system describes the time evolution of the species' distribution function $f_s(\mathbf{x}, \mathbf{v}, t)$ in three-dimensional physical space and three-dimensional velocity space. The distribution function is defined such that the number of particles per unit volume in the vicinity of position $\mathbf{x}$ and velocity $\mathbf{v}$ is $f_s(\mathbf{x}, \mathbf{v}, t)d^3\mathbf{x}d^3\mathbf{v}$. It is normalized such that

$$\int f_s(\mathbf{x}, \mathbf{v}, t)d^3\mathbf{v} = n_s(\mathbf{x}, t), \tag{3}$$

where n_s is the species' number density. Therefore, the charge density is obtained from the distribution functions as

$$\rho = \sum_{s=i,e} q_s \int f_s d^3\mathbf{v}. \tag{4}$$

For simplicity, in this work we will limit ourselves to one spatial and one velocity dimensions—z and v, respectively—as in the textbook formulation of the Landau damping problem [37]. In this case, Eqs. (1) and (2) become

$$\frac{\partial f_s}{\partial t} + v\frac{\partial f_s}{\partial z} + \frac{q_s}{m_s}E\frac{\partial f_s}{\partial v} = C[f_s], \tag{5}$$

$$\epsilon_0\frac{\partial E}{\partial z} = \sum_{s=i,e} q_s \int f_s dv. \tag{6}$$

We can linearize Eqs. (5) and (6) about a Maxwellian background, and consider the case where the ions are stationary, meaning that their distribution function is not perturbed, so we need only consider the evolution of the electron distribution function. Then, $f \approx F_0 + g$, where we have dropped the species subscript for simplicity, $g \ll F_0$ is the perturbation to the distribution function, and F_0 is the (one-dimensional) Maxwellian distribution function

$$F_0(v) = \frac{1}{\sqrt{\pi}}\frac{n_0}{v_{\text{th}}}e^{-v^2/v_{\text{th}}^2}, \tag{7}$$

where n_0 is the background density and

$$v_{\text{th}} = \sqrt{\frac{2k_BT}{m}}, \tag{8}$$

is the electron thermal velocity, with T and k_B denoting the temperature and Boltzmann constant, respectively. We carry out the linearization by noting that $C[F_0] = 0$, ignoring

quadratic perturbative terms, and using the normalizations

$$\begin{aligned} z &\leftarrow \frac{z}{\sqrt{2}\lambda_D}, \\ t &\leftarrow \omega_p t, \\ v &\leftarrow \frac{v}{v_{\text{th}}}, \\ f &\leftarrow \frac{v_{\text{th}}}{n} f, \\ \varphi &\leftarrow \frac{e}{k_B T}\varphi, \end{aligned} \tag{9}$$

where e is the fundamental charge and

$$\omega_p = \sqrt{\frac{n_0 e^2}{\epsilon_0 m}}, \quad \lambda_D = \sqrt{\frac{\epsilon_0 k_B T}{n_0 e^2}}, \tag{10}$$

are, respectively, the plasma frequency and the Debye length. Thus, the linearized Vlasov-Poisson system becomes

$$\frac{\partial g_k}{\partial t} + ikvg_k + ikvF_0\varphi_k = C[g_k], \tag{11}$$

$$\varphi_k = \alpha \int_{-\infty}^{+\infty} g_k dv, \tag{12}$$

where we have also Fourier-transformed both equations, with k representing the Fourier wave number, and $\alpha = 2/k^2$ is the physics parameter. By choosing a different value of α satisfying $\alpha \geqslant -1$, different target problems can be described, where Eqs. (11) and (12) consider the evolution of only one species, and α prescribes the behavior of the other species (Boltzmann, isothermal, or no response) [38,39].

While the Vlasov equation is typically coupled to Poisson's equation in the electrostatic case, it can also be coupled to Ampère's law,

$$\frac{\partial E_k}{\partial t} = -J_k, \tag{13}$$

where J_k is the kth Fourier mode of the current. For the case of Langmuir waves, the linearized Ampère's law, coupled to the Vlasov equation in Fourier space, yields the following set of equations:

$$\frac{\partial g_k}{\partial t} + ikvg_k - vF_0E_k = C[g_k], \tag{14}$$

$$\frac{\partial E_k}{\partial t} = \int v g_k \, dv. \tag{15}$$

While Eqs. (11) and (12) and Eqs. (14) and (15) are two different formulations, they describe the same physics.

B. Landau damping

Landau damping is a textbook plasma phenomenon whereby wave energy is resonantly transferred to plasma particles. Despite what its name might suggest, Landau damping is a reversible process: In the absence of collisions, the total energy in a plasma is conserved throughout this process. During Landau damping, the plasma distribution function attains continuously finer structures in velocity space. In a collisionless plasma, these structures continue to develop until the end of the linear regime. In a collisional plasma, such structures are eventually smoothed out by collisions. This will become important when we perform the error analysis in Sec. IV.

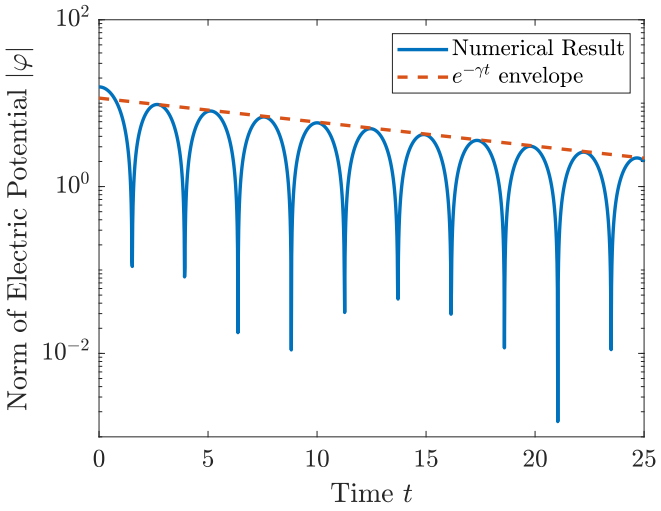

FIG. 1. The norm of the electric potential $|\varphi|$ as a function of time t. The blue solid line is obtained by numerically solving Eqs. (11) and (12) with $k = 0.4\sqrt{2}$, and $\alpha = 2/k^2$. The red dashed line is an $e^{-\gamma t}$ envelope, where $\gamma = 0.066\,237$. The parameters used here match those used in Ref. [34].

To calculate the Landau damping rate of a plasma wave, one typically tracks the time evolution of the amplitude of the electric potential (or, equivalently, the electric field) in the plasma. The amplitude can be fitted by an $e^{-\gamma t}$ envelope, where $\gamma > 0$ is the Landau damping rate. The density perturbation in a plasma can also be used to calculate the Landau damping rate, since the electric potential and density perturbation are proportional to each other, as per Eq. (12). This procedure is illustrated in Fig. 1, where the time evolution of the norm of the electric potential $|\varphi|$ is tracked. The peaks are fitted with an $e^{-\gamma t}$ envelope, and the Landau damping rate is recovered from the fit. The Landau damping rate can also be analytically calculated by solving the following dispersion relation,

$$1 + \frac{\sqrt{2}}{\pi k^2} \int_{-\infty}^{\infty} \frac{v e^{-v^2/2}}{v - \sqrt{2}\omega/k} dv = 0, \tag{16}$$

where the imaginary component of the frequency ω contains γ.

C. The Hermite representation

Working with the linear Vlasov-Poisson system, it is often convenient, and numerically advantageous, to express the distribution function as a series of Hermite polynomials [38,40–45],

$$g_k(v,t) = \sum_{m=0}^{\infty} \frac{H_m(v)F_0(v)}{\sqrt{2^m m!}} g_{m,k}(t), \tag{17}$$

where $H_m(v)$ denotes the Hermite polynomial of order m and $g_{m,k}(t)$ is its coefficient given by

$$g_{m,k}(t) = \int_{-\infty}^{+\infty} dv \frac{H_m(v)}{\sqrt{2^m m!}} g_k(v,t). \tag{18}$$

Adopting a Lenard-Bernstein-type collision operator for simplicity [46],

$$C[g_k] = \nu \frac{\partial}{\partial v}\left(\frac{1}{2}\frac{\partial}{\partial v} + v\right)g_k, \tag{19}$$

where $\nu \geqslant 0$ is the collision frequency, and introducing new variables defined as

$$\tilde{g}_{0,k} = g_{0,k}, \tag{20}$$

$$\tilde{g}_{m,k} = \frac{1}{\sqrt{1+\alpha}} g_{m,k}, \quad m \geqslant 1, \tag{21}$$

the Hermite-transformed equations become a coupled set of linear advection equations for the Hermite-Fourier coefficients, as follows,

$$\frac{dg_{0,k}}{dt} + ik\sqrt{\frac{1+\alpha}{2}}g_{1,k} = 0, \tag{22}$$

$$\frac{dg_{1,k}}{dt} + ik\left(g_{2,k} + \sqrt{\frac{1+\alpha}{2}}g_{0,k}\right) = 0, \tag{23}$$

$$\frac{dg_{m,k}}{dt} + ik\left(\sqrt{\frac{m+1}{2}}g_{m+1,k} + \sqrt{\frac{m}{2}}g_{m-1,k}\right) = -\nu m g_{m,k}, \quad m \geqslant 2, \tag{24}$$

where the tildes have been dropped for notational simplicity [47]. Provided that an infinite number of Hermite moments are kept, these equations are formally equivalent to Eqs. (11) and (12). For practical purposes, however, only a finite number of moments, $M+1$, must be retained. Since the equation for the Hermite moment of order m couples to the successive moment, $m+1$, a truncation is needed to integrate these equations on a computer. We adopt the customary closure $g_{M+1} = 0$, yielding the following system,

$$\frac{d\mathbf{g}_k}{dt} = A\mathbf{g}_k, \tag{25}$$

with the solution

$$\mathbf{g}_k(t) = e^{At}\mathbf{g}_k(t=0), \tag{26}$$

where $\mathbf{g}_k = [g_{0,k} \cdots g_{M,k}]^T$, $\mathbf{g}_k(t=0)$ is the initial condition, and A is a 3-sparse matrix containing the relevant coefficients,

$$A = -iH - \nu\,\mathrm{diag}([0, 0, 2, \ldots, M-1, M]), \tag{27}$$

with

$$H = k\begin{bmatrix} 0 & \sqrt{\frac{1+\alpha}{2}} & & & & \\ \sqrt{\frac{1+\alpha}{2}} & 0 & 1 & & & \\ & 1 & 0 & \sqrt{\frac{3}{2}} & & \\ & & \ddots & \ddots & \ddots & \\ & & & \sqrt{\frac{M-1}{2}} & 0 & \sqrt{\frac{M}{2}} \\ & & & & \sqrt{\frac{M}{2}} & 0 \end{bmatrix}. \tag{28}$$

The initial condition for the canonical Landau damping problem is $g_0(z, v, t=0) = \bar{g}_0 \cos(k_0 z)$, where $\bar{g}_0$ is a constant (an arbitrary amplitude). In Fourier-Hermite space, this initial condition is

$$g_{m,k}(t=0) = \delta_{m,0}\left(\delta_{k,k_0} + \delta_{k,-k_0}\right), \tag{29}$$

where we have chosen $\bar{g}_0 = \sqrt{2/\pi}$.

While M is in principle a free parameter, typically one wishes to have $M \gtrsim \omega/\nu$ in order to capture the physics, where ω is a characteristic frequency of the system, to ensure that the higher Hermite moments are properly dissipated by collisions. If $\nu = 0$, M will be a function of simulation time T. This will be relevant to the discussion in Sec. IV.

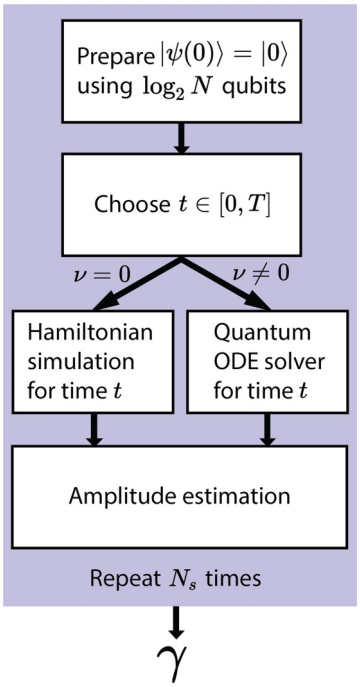

FIG. 2. Workflow diagram of the quantum algorithm in both the collisionless and collisional cases. Here, N is the system size, T is the simulation time, and N_s refers to the number of samples required at different times to calculate the Landau damping rate.

III. THE QUANTUM ALGORITHM

In this section we will show that a quantum algorithm solving Eq. (25) yields a quadratic speedup with respect to system size compared to the most efficient classical algorithms. Figure 2 shows a workflow diagram of the proposed quantum algorithm.

A. Collisionless case and Hamiltonian simulation

When $\nu = 0$ we have a collisionless system where $A = -iH$ with H a Hermitian matrix. Such a system conserves total energy and thus is governed by unitary time evolution generated by H:

$$\frac{d\mathbf{g}_k}{dt} = -iH\mathbf{g}_k. \tag{30}$$

Since this is equivalent to a Schrödinger equation, it can be simulated on a quantum computer using Hamiltonian simulation techniques. An efficient quantum algorithm for simulating an s-sparse Hamiltonian has a query complexity

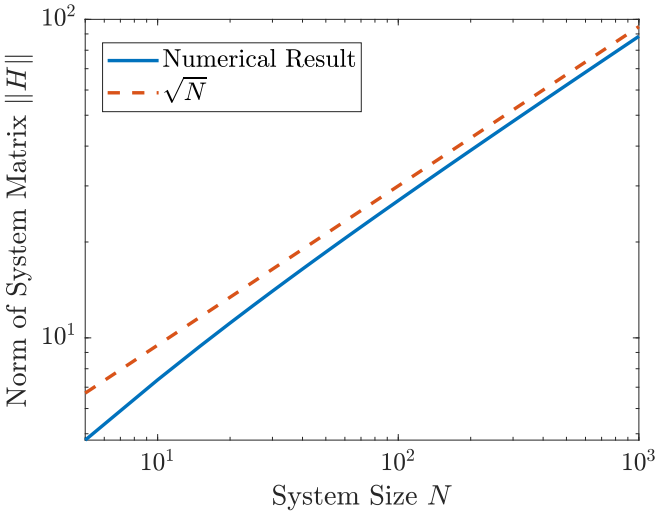

FIG. 3. Numerically obtained norm of the system matrix $\|H\|$ [Eq. (28)] as a function of the system size $N = M + 1$, where M is the number of Hermite moments. The orange dashed line is a $\sqrt{N}$ scaling.

$O(s\|H\|T + \frac{\log(1/\epsilon)}{\log\log(1/\epsilon)})$ [7], where s is the sparsity, $\|H\|$ is the norm, T is the final simulation time, and ϵ is the error in the norm of the solution state. In the Hermite system, the system matrix has sparsity $s = 2$. Furthermore, the norm of the system scales with the system size $N = M + 1$ as

$$\|H\| \sim O(\sqrt{N}). \tag{31}$$

This is also shown in Fig. 3, where we plot the norm of the system matrix $\|H\|$ as a function of system size N and observe the $O(\sqrt{N})$ scaling.

B. Collisional case and quantum ODE solvers

For the more realistic scenario where $\nu \neq 0$, unitary time evolution is lost due to the presence of the collision terms. In this case, we need to solve an ODE rather than simulate a Hamiltonian. We use the results from Ref. [18], which state the following. Suppose A is an s-sparse matrix with oracle access to its entries. Then Eq. (25) can be solved to produce a quantum state proportional to the solution in time $O[\mathcal{G}\,T\,\|A\|\,\mathcal{C}(A)]$, where $\mathcal{G} = \max_{t\in[0,T]}\|\mathbf{g}_k(t)\|/\|\mathbf{g}_k(T)\|$ measures the solution decay and $\mathcal{C}(A) = \sup_{t\in[0,T]}\|\exp(At)\|$. For our system, the matrix A has negative log norm and so $\mathcal{C}(A) \leqslant 1$. Furthermore, since $M \sim \omega/\nu$, the addition of collisions does not change the $O(\sqrt{N})$ scaling of $\|A\|$ as long as $\nu \ll k$. To see this, we can approximate ω by choosing $M = 2$ and solving for the dispersion relation, giving us

$$\omega = \sqrt{\frac{1+\alpha}{2}}k. \tag{32}$$

Typically, $\alpha = O(1)$, so $\omega \sim k$. To maintain the $O(\sqrt{N})$ scaling of $\|A\|$, we demand that the off-diagonal terms of Eq. (27) dominate the diagonal terms, yielding the following constraint:

$$\nu\sqrt{M} \ll k. \tag{33}$$

Knowing that $M \sim \omega/\nu \sim k/\nu$, we obtain $\nu \ll k$. This is satisfied in most plasmas of interest because ν, which is normalized by the plasma frequency ω_p, is very small compared to k, which is normalized by Debye length λ_D. Thus, even in the presence of collisions, we can still obtain a quadratic speedup in system size. However, the time dependence of the algorithm suffers. This is because collisions cause the norm of the solution to decay exponentially, meaning that for large T, $\|\mathbf{g}_k(0)\|/\|\mathbf{g}_k(T)\|$ scales as $e^{O(\nu)T}$. To obtain $\mathbf{g}_k(T)$ efficiently, we can restrict the simulation time, which we will discuss in Sec. III C.

C. Oracle representation, state preparation, and output

The Hamiltonian simulation and quantum ODE solvers require access to the system matrix A through an oracle O_A. The oracle has the following action,

$$O_A|i, j, 0\rangle = |i, j, A_{ij}\rangle, \tag{34}$$

where A_{ij} denotes the element in the ith row and jth column of A. The representation of O_A is as follows:

$$\begin{aligned} O_A = {} & (|1,2\rangle\langle 1,2| + |2,1\rangle\langle 2,1|) \otimes \left|-ik\sqrt{\frac{1+\alpha}{2}}\right\rangle\langle 0| \\ & + \sum_{l=2}^{M} |l, l+1\rangle\langle l, l+1| \otimes \left|-ik\sqrt{\frac{l}{2}}\right\rangle\langle 0| \\ & + \sum_{l=3}^{M+1} |l, l-1\rangle\langle l, l-1| \otimes \left|-ik\sqrt{\frac{l-1}{2}}\right\rangle\langle 0| \\ & + \sum_{l=3}^{M+1} |l, l\rangle\langle l, l| \otimes |(l-1)\nu\rangle\langle 0|. \end{aligned} \tag{35}$$

This oracle can be block-encoded using the methods in Ref. [48].

The initial condition for the Hermite system is given by Eq. (29). This is trivial to construct, and does not add to the complexity of the algorithm.

In general, one has access to all Hermite moments in the solution state. If one is interested only in computing the Landau damping rate, then it suffices to measure the zeroth Hermite moment $g_{0,k}$, from which the electric potential can be calculated,

$$\varphi_k = \alpha g_{0,k}. \tag{36}$$

To obtain the Landau damping rate, $g_{0,k}$ needs to be sampled at various times. To obtain $g_{0,k}$ at a particular time, amplitude estimation can be used [49]. This requires $O(1/\delta)$ repetitions of the algorithm to calculate $g_{0,k}$ to absolute precision δ. Once the amplitude is obtained, the algorithm must be repeated for the other sample times. Doing amplitude estimation efficiently at all times is contingent upon the electric potential decaying slowly. Furthermore, in the collisional case, the simulation time must be limited due to the solution norm decaying as $e^{-O(\nu)t}$. In general, one is interested in kinetic effects, which are captured when $\nu \ll \gamma$. This means that the collisions are occurring on a timescale which is longer than the timescales of interest (Landau damping, in this case). Thus, by choosing $T \sim 1/\gamma$, we can ensure that we recover the Landau damping

rate accurately without the decay of both the solution norm and the electric potential hindering the performance.

These repetitions required for extracting the Landau damping rate increase the complexity of the quantum algorithm with respect to error. Efficient Hamiltonian simulation algorithms to prepare the solution state scale as $O[\log(1/\epsilon)]$, but extracting a classical parameter (such as the Landau damping rate) by using amplitude estimation—assuming that $\delta \sim \epsilon$—turns this scaling into $O(1/\epsilon)$.

IV. ERROR ANALYSIS

A. Statement of the problem

Given a precision ϵ, we would like to determine the system size N required to obtain $g_{0,k}$ from Eq. (25), as well as the system size N_v required to obtain E_k from Eqs. (14) and (15). In the former case, the input parameters are the simulation time T [50], the Fourier coefficient k_0 for the initial condition, the collision frequency ν, and physics parameter α. For the ESP system, in addition to T and k_0, the velocity grid truncation value $v_{\max}$ is required (their algorithm only solves the collisionless case, $\nu = 0$).

B. The Hermite system

The error analysis for the Hermite system is nontrivial and depends on the function that is being Hermite-expanded. Typically, for the types of functions involved in this problem (decaying as e^{-v^2}), the convergence of the Hermite series is superexponential [51,52]. That is,

$$\epsilon = O(e^{-\beta M}), \tag{37}$$

where $\beta = \beta(k, T)$.

Since an exact analytical solution to Eqs. (11) and (12) is not known, for the purpose of the error analysis we analyze the case $\alpha = 0$ (as done in other works, e.g., Ref. [42]), which corresponds to the free-streaming equation

$$\frac{\partial g_k}{\partial t} + ikvg_k = 0, \tag{38}$$

with initial condition

$$g_k(v, 0) = e^{-v^2}, \tag{39}$$

whose exact solution is

$$g_k(v, t) = e^{-ikvt - v^2}. \tag{40}$$

The Hermite coefficients of this solution which solve Eq. (25) are given by

$$g_{m,k}(t) = \frac{e^{-k^2t^2/4}(-ikt)^m}{\sqrt{2^m m!}}. \tag{41}$$

The factorial in the denominator scales as $O(e^{m \log m})$ and results in a superexponential convergence as m increases. Figure 4 plots the convergence of the Hermite series for the zeroth moment $g_{0,k}$. An ODE solver is used to solve Eq. (25) and the solution is compared with Eq. (41) to obtain the relative error. The figure shows the predicted convergence.

This exponential convergence can also be demonstrated in the collisional case. Indeed, as shown in Ref. [45], seeking a steady-state solution to Eq. (24) with $m \gg 1$ results in a Hermite spectrum given by

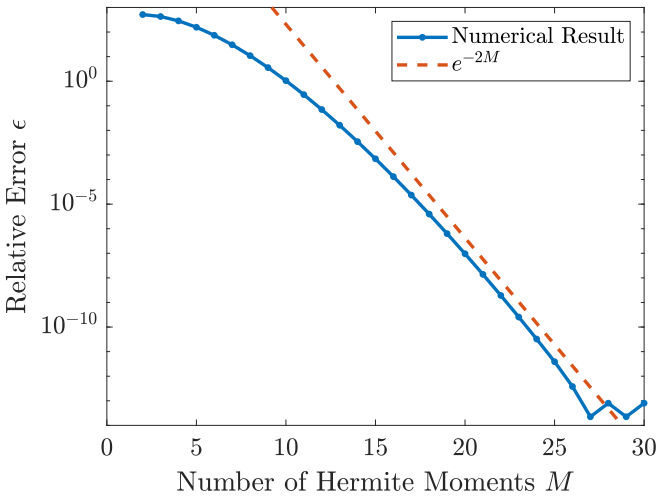

FIG. 4. The relative error ϵ as a function of the number of Hermite moments M. The blue solid line is the error obtained from solving Eq. (25) using an ODE solver with $\alpha = \nu = 0$, $k = 2$, and $T = 2.5$, and the dashed orange line is an e^{-2M} scaling.

$$|g_m| = O\left(\frac{1}{m^{1/4}} \exp\left[-\frac{1}{2}\left(\frac{m}{m_c}\right)^{3/2}\right]\right), \tag{42}$$

where m_c corresponds to the collisional cutoff (a function of the collision frequency and other physical inputs of the problem). While the convergence is polynomial with $1/m$ for small values of m, it becomes exponential asymptotically for larger values of m.

The fast convergence means that the system size $N = M + 1$ scales logarithmically with the error

$$N = O[\log(1/\epsilon)]. \tag{43}$$

C. The ESP system

Here, we will briefly overview the approach of ESP [34] and perform an error analysis. The reader is referred to their paper for further details.

Instead of using the Poisson equation, as we do in our formulation, ESP couple the linear (collisionless) Vlasov equation to Ampère's law, as in Eqs. (14) and (15) with $C[g_k] = 0$. They then restrict velocity space to the domain $[-v_{\max}, v_{\max}]$, where $v_{\max}$ is an input parameter, and approximate the integral in Eq. (15) using a Riemann sum on a velocity grid of N_v points. This, along with variable transformations, allows them to write the system as a Schrödinger-type equation,

$$\frac{d\boldsymbol{\psi}_k}{dt} = -iH\boldsymbol{\psi}_k, \tag{44}$$

where $\boldsymbol{\psi}_k$ stores the (Fourier-transformed) distribution function in velocity space, as well as the amplitude of the electric field.

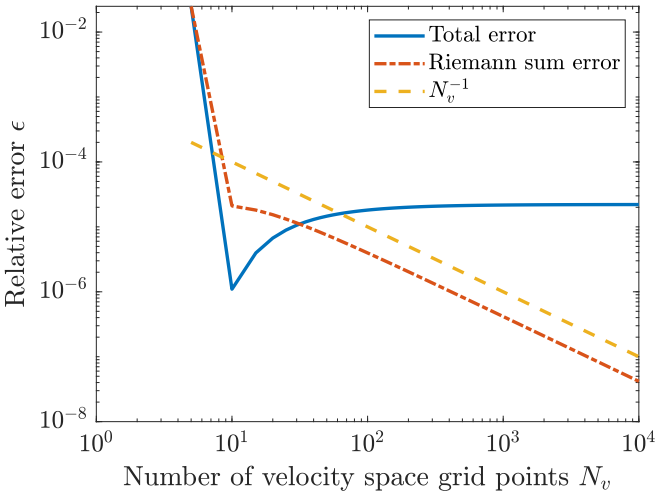

FIG. 5. The relative error in integrating $\int_{-\infty}^{\infty} e^{-v^2}\,dv$ as a function of the number of velocity space grid points N_v with $v_{\max} = 3$. The blue solid line is the total error, as shown in Eq. (45), the dashed-dotted orange line is the Riemann sum error, as shown by the second term on the right-hand side of Eq. (45), and the dashed yellow line is an N_v^{-1} scaling.

The error in the amplitude of the electric field originating from Eq. (15) is given by

$$\epsilon = O\left(e^{-v_{\max}^2} + \frac{2L_1 v_{\max}^2}{N_v}\right), \tag{45}$$

where $L_1 = \max_{v\in[-v_{\max},v_{\max}]} |dg_k/dv|$. The first term corresponds to the domain truncation error [52], and the second corresponds to the Riemann sum error [53]. Because L_1 is dependent on the amount of structure present in the perturbed distribution function, we can conclude that $L_1 = L_1(k,T)$ (as per the discussion in Sec. II B). As an example, Fig. 5 is a convergence plot for approximating $\int_{-\infty}^{\infty} e^{-v^2}\,dv$ as a domain-truncated Riemann sum, showing convergence of the error with respect to N_v. The Gaussian function is specifically chosen for the error analysis because g_k decays as a Gaussian as $|v| \to \infty$. We can see the total error [54] and the Riemann sum error, and we observe that the latter converges as N_v^{-1}, as predicted. The saturation of the total error as N_v increases is due to the domain truncation error (the difference between the blue and orange lines).

For simplicity, let us take the domain truncation error and the Riemann sum error to be of the same order:

$$e^{-v_{\max}^2} \sim \frac{2L_1 v_{\max}^2}{N_v} \sim \epsilon. \tag{46}$$

Then, to achieve a precision ϵ, we have the following expression for N_v:

$$N_v = O\left(\frac{\log(1/\epsilon)}{\epsilon}\right). \tag{47}$$

Thus, the ESP system size N_v scales polynomially with $1/\epsilon$.

D. Comparison

Comparing Eqs. (43) and (47), we see that the Hermite system can achieve the same precision ϵ with an exponentially smaller system size N compared to the ESP system. This means that a classical ODE solver—whose complexity scales as $O(N)$—implementing the Hermite system has the same performance as the quantum algorithm of ESP.

TABLE I. Gate complexities of the algorithms to estimate Landau damping discussed in this paper. Here, N is the Hermite system size, N_v is the ESP system size, ϵ is the absolute error in amplitude estimation, and $\theta \leqslant 1$ is the order of the classical ODE solver, with smaller values corresponding to higher-order solvers. Note that $N = O[\log(N_v)]$, so the classical Hermite algorithm matches the performance of the ESP quantum algorithm. The simulation time T is a constant and is omitted from the complexity analysis.

	Gate complexity	
Representation	Classical	Quantum
ESP [34]	$O(N_v/\epsilon^\theta)$	$O[\text{polylog}\,(N_v)/\epsilon]$
Hermite	$O(N/\epsilon^\theta)$	$O[\sqrt{N}\log(N)/\epsilon]$

While we have excluded the dependence of the system size on the simulation time T in the error analysis, we note that this dependence is different for the Hermite and ESP representations. Inspecting Eq. (40), we find that structures develop from the $\exp(-ikTv)$ term, meaning that the ESP representation would need $N_v \sim kT$ grid points to resolve such structures. On the other hand, from Eq. (41) we find that the peak of $|g_{M,k}(T)|$ occurs when $M = (kT)^2/2$; this sets the minimum resolution required to capture the physics. Thus, in the long simulation time limit, the ESP system size has better dependence on T. However, this does not affect the results of this paper as long as ϵ is not held fixed. This is because the ESP system size's dependence on T and ϵ goes as $N_v \sim T\log(1/\epsilon)/\epsilon$, whereas for the Hermite system size the dependence is $N \sim T^2\log(1/\epsilon)$. Thus, we still have $N = O(\log N_v)$, so the Hermite system size remains exponentially smaller than the ESP system size. It is important to note that the system size's dependence on T exists only in the collisionless case. Upon adding collisions, the number of Hermite moments required to capture the fine structures is dictated by $M \gtrsim \omega/\nu$, which is independent of T.

V. DISCUSSION AND CONCLUSION

In this paper we have demonstrated two main results. First, using a Hermite representation of velocity space, we can obtain a system that is exponentially smaller than the one obtained via finite-difference discretization, as proposed in the work of ESP [34], for the same error ϵ. This implies that a classical implementation of the Hermite approach will have a similar performance to that of ESP's quantum algorithm. Second, a quantum algorithm for the Hermite formulation can yield a quadratic speedup compared to classical algorithms that solve the same system of equations. An exponential speedup, however, does not seem possible with currently known methods due to the large norm of the matrices involved. Table I summarizes the complexities of the algorithms discussed in this paper.

The problem analyzed in this paper is somewhat nuanced in that system size is conflated with the computation error. This is because the resolution required in velocity (or Hermite) space is strictly a function only of the error one wishes to achieve in the computation of the Landau damping rate (physically, running the computation for longer times so that a longer-in-time decay stage is obtained leads to more phase mixing and thus a finer-scale structure in velocity space—or, correspondingly, a structure at higher Hermite moments). However, this only applies when the minimum resolution required to capture the relevant physics is achieved (e.g., for the collisional case, $M \gtrsim \omega/\nu$). Similarly, in more general problems in plasma physics, there are minimal requirements imposed on velocity- and position-space grid sizes, set by the need to resolve specific physics processes (for example, one typically wishes to simulate a system of a given size L, but is forced to resolve kinetic-scale physics happening at scales below, say, the ion Larmor radius ρ_i. Frequently, $L/\rho_i \gg 1$, implying, therefore, a very large number of grid points in position space before one can even consider the error convergence with respect to system size. A comparable situation in velocity space is one where there is a superthermal particle population, in addition to a Maxwellian bulk). Therefore, the scaling with system size of quantum algorithms for plasma problems is of intrinsic interest.

Our results also highlight the challenge of applying currently existing quantum algorithms to real-world problems. In applying quantum algorithms for a Hamiltonian simulation or differential equation solvers to the Vlasov equation, we encounter matrix norms that scale as the square root of the system size. In addition, extracting classical information such as the Landau damping parameter increases the complexity from $O[\log(1/\epsilon)]$ to $O(1/\epsilon)$.

We also note that it is possible to formulate the linear Vlasov equation as an eigenvalue problem and use a quantum phase estimation [55,56] and quantum eigenvalue solvers [57–61] to determine the smallest eigenvalue for the collisionless and collisional systems, respectively. However, in both cases, finding the smallest eigenvalue requires knowledge of the corresponding eigenvector. Since we do not know the eigenvector *a priori*, this approach is ineffective. Furthermore, Landau damping cannot be captured with an eigenvalue formulation, as it inherently requires an initial-value problem approach.

Possible extensions of this work include generalizing the system to higher dimensions, as well as extending the quantum algorithm to the fully nonlinear Vlasov equation. For the latter, in the regime of weak nonlinearity, it is possible that quantum approaches such as those proposed by Liu *et al.* [21] could lead to a speedup. Whether a quantum speedup can be obtained in a strongly nonlinear regime remains an open and important problem.

ACKNOWLEDGMENTS

A.A. thanks Alexander Engel and Scott Parker for providing great insight on their work and for the useful feedback on this paper, as well as Muni Zhou and Noah Mandell for helpful discussions on plasma physics and quantum algorithms. A.A. acknowledges the support of the Natural Sciences and Engineering Research Council of Canada (NSERC), PGSD3-557902-2021. The authors acknowledge support from the U.S. Department of Energy Grant No. DE-SC0020264.

[1] L. K. Grover, A fast quantum mechanical algorithm for database search, in *Proceedings of the Twenty-Eighth Annual ACM Symposium on Theory of Computing* (ACM Press, New York, 1996), pp. 212–219.

[2] S. Lloyd, Universal quantum simulators, Science **273**, 1073 (1996).

[3] D. W. Berry, G. Ahokas, R. Cleve, and B. C. Sanders, Efficient quantum algorithms for simulating sparse Hamiltonians, Commun. Math. Phys. **270**, 359 (2007).

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

[5] D. W. Berry, A. M. Childs, R. Cleve, R. Kothari, and R. D. Somma, Exponential improvement in precision for simulating sparse Hamiltonians, in *Proceedings of the Forty-Sixth Annual ACM Symposium on Theory of Computing* (ACM Press, New York, 2014), pp. 283–292.

[6] D. W. Berry, A. M. Childs, R. Cleve, R. Kothari, and R. D. Somma, Simulating Hamiltonian Dynamics with a Truncated Taylor Series, Phys. Rev. Lett. **114**, 090502 (2015).

[7] G. H. Low and I. L. Chuang, Optimal Hamiltonian Simulation by Quantum Signal Processing, Phys. Rev. Lett. **118**, 010501 (2017).

[8] G. H. Low and I. L. Chuang, Hamiltonian simulation by qubitization, Quantum **3**, 163 (2019).

[9] P. W. Shor, Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer, SIAM Rev. **41**, 303 (1999).

[10] S. P. Jordan, K. S. M. Lee, and J. Preskill, Quantum algorithms for quantum field theories, Science **336**, 1130 (2012).

[11] S. P. Jordan, K. S. Lee, and J. Preskill, Quantum algorithms for fermionic quantum field theories, arXiv:1404.7115.

[12] T. J. Osborne and A. Stottmeister, Quantum simulation of conformal field theory, arXiv:2109.14214.

[13] A. W. Harrow, A. Hassidim, and S. Lloyd, Quantum Algorithm for Linear Systems of Equations, Phys. Rev. Lett. **103**, 150502 (2009).

[14] A. M. Childs, R. Kothari, and R. D. Somma, Quantum algorithm for systems of linear equations with exponentially improved dependence on precision, SIAM J. Comput. **46**, 1920 (2017).

[15] D. W. Berry, High-order quantum algorithm for solving linear differential equations, J. Phys. A: Math. Theor. **47**, 105301 (2014).

[16] D. W. Berry, A. M. Childs, A. Ostrander, and G. Wang, Quantum algorithm for linear differential equations with exponentially improved dependence on precision, Commun. Math. Phys. **356**, 1057 (2017).

[17] A. M. Childs and J.-P. Liu, Quantum spectral methods for differential equations, Commun. Math. Phys. **375**, 1427 (2020).

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

[19] S. K. Leyton and T. J. Osborne, A quantum algorithm to solve nonlinear differential equations, arXiv:0812.4423.

[20] S. Lloyd, G. De Palma, C. Gokler, B. Kiani, Z.-W. Liu, M. Marvian, F. Tennie, and T. Palmer, Quantum algorithm for nonlinear differential equations, arXiv:2011.06571.

[21] J.-P. Liu, H. Ø. Kolden, H. K. Krovi, N. F. Loureiro, K. Trivisa, and A. M. Childs, Efficient quantum algorithm for dissipative nonlinear differential equations, Proc. Natl. Acad. Sci. USA **118**, e2026805118 (2021).

[22] C. Xue, W. Yu-Chun, and G. Guo, Quantum homotopy perturbation method for nonlinear dissipative ordinary differential equations, New J. Phys. **23**, 123035 (2021).

[23] Y. Cao, A. Papageorgiou, I. Petras, J. Traub, and S. Kais, Quantum algorithm and circuit design solving the Poisson equation, New J. Phys. **15**, 013021 (2013).

[24] P. C. S. Costa, S. Jordan, and A. Ostrander, Quantum algorithm for simulating the wave equation, Phys. Rev. A **99**, 012323 (2019).

[25] While the aforementioned quantum algorithms for differential equations rely on a Hamiltonian simulation, it is possible to solve nonlinear differential equations using variational quantum machine learning methods. We refer the reader to Ref. [62] for further details.

[26] I. Y. Dodin and E. A. Startsev, On applications of quantum computing to plasma simulations, Phys. Plasmas **28**, 092101 (2021).

[27] J. Zylberman, G. Di Molfetta, M. Brachet, N. F. Loureiro, and F. Debbasch, Quantum simulations of hydrodynamics via the Madelung transformation, Phys. Rev. A **106**, 032408 (2022).

[28] A. Engel, G. Smith, and S. E. Parker, Linear embedding of nonlinear dynamical systems and prospects for efficient quantum algorithms, Phys. Plasmas **28**, 062305 (2021).

[29] I. Joseph, Koopman–von Neumann approach to quantum simulation of nonlinear classical dynamics, Phys. Rev. Res. **2**, 043102 (2020).

[30] I. Joseph, Y. Shi, M. Porter, A. Castelli, V. Geyko, F. Graziani, S. Libby, and J. DuBois, Quantum computing for fusion energy science applications, Phys. Plasmas **30**, 010501 (2023).

[31] Y. Shi, A. R. Castelli, X. Wu, I. Joseph, V. Geyko, F. R. Graziani, S. B. Libby, J. B. Parker, Y. J. Rosen, L. A. Martinez *et al.*, Simulating non-native cubic interactions on noisy quantum machines, Phys. Rev. A **103**, 062608 (2021).

[32] I. Novikau, E. A. Startsev, and I. Y. Dodin, Quantum signal processing for simulating cold plasma waves, Phys. Rev. A **105**, 062444 (2022).

[33] G. Vahala, J. Hawthorne, L. Vahala, A. K. Ram, and M. Soe, Quantum lattice representation for the curl equations of Maxwell equations, Radiat. Eff. Defects Solids **177**, 85 (2022).

[34] A. Engel, G. Smith, and S. E. Parker, Quantum algorithm for the Vlasov equation, Phys. Rev. A **100**, 062315 (2019).

[35] L. Landau, On the vibrations of the electronic plasma, in *The Collected Papers of L. D. Landau*, edited by D. ter Haar (Pergamon Press, London, 1965), pp. 445–460.

[36] R. L. Burden, J. D. Faires, and A. M. Burden, *Numerical Analysis* (Cengage Learning, Boston 2015).

[37] F. F. Chen, *Introduction to Plasma Physics* (Springer, Berlin, 2012).

[38] A. Kanekar, A. A. Schekochihin, W. Dorland, and N. F. Loureiro, Fluctuation-dissipation relations for a plasma-kinetic Langevin equation, J. Plasma Phys. **81**, 305810104 (2015).

[39] T. Adkins and A. A. Schekochihin, A solvable model of Vlasov-kinetic plasma turbulence in Fourier–Hermite phase space, J. Plasma Phys. **84**, 905840107 (2018).

[40] T. P. Armstrong, Numerical studies of the nonlinear Vlasov equation, Phys. Fluids **10**, 1269 (1967).

[41] F. C. Grant and M. R. Feix, Fourier-Hermite solutions of the Vlasov equations in the linearized limit, Phys. Fluids **10**, 696 (1967).

[42] G. Hammett, M. Beer, W. Dorland, S. Cowley, and S. Smith, Developments in the gyrofluid approach to tokamak turbulence simulations, Plasma Phys. Controlled Fusion **35**, 973 (1993).

[43] S. E. Parker and D. Carati, Renormalized Dissipation in Plasmas with Finite Collisionality, Phys. Rev. Lett. **75**, 441 (1995).

[44] H. Sugama, T.-H. Watanabe, and W. Horton, Collisionless kinetic-fluid closure and its application to the three-mode ion temperature gradient driven system, Phys. Plasmas **8**, 2617 (2001).

[45] A. Zocco and A. A. Schekochihin, Reduced fluid-kinetic equations for low-frequency dynamics, magnetic reconnection, and electron heating in low-beta plasmas, Phys. Plasmas **18**, 102309 (2011).

[46] A. Lenard and I. B. Bernstein, Plasma oscillations with diffusion in velocity space, Phys. Rev. **112**, 1456 (1958).

[47] The damping term for the $m = 1$ equation is manually switched off because the collision operator must conserve momentum, corresponding to Eq. (23) [39].

[48] C. Sünderhauf, E. Campbell, and J. Camps, Block-encoding structured matrices for data input in quantum computing, arXiv:2302.10949.

[49] G. Brassard, P. Hoyer, M. Mosca, and A. Tapp, Quantum amplitude amplification and estimation, Contemp. Math. **305**, 53 (2002).

[50] The linear Vlasov equation exhibits a phenomenon known as recursion, where, for a long simulation time, the velocity space cascade towards finer structures reverses. This phenomenon is purely numerical and requires one to choose the velocity space resolution to be high enough (or the simulation time to be small enough) to prevent it. For the error analysis, we will assume that this is indeed the case, so as to avoid the problem of recursion.

[51] J. P. Boyd, Asymptotic coefficients of Hermite function series, J. Comput. Phys. **54**, 382 (1984).

[52] J. P. Boyd, *Chebyshev and Fourier spectral methods* (Courier Corporation, North Chelmsford, MA, 2001).

[53] L. Rall, Numerical integration and the solution of integral equations by the use of Riemann sums, SIAM Rev. **7**, 55 (1965).

[54] The reason why the total error dips below the Riemann sum error for lower values of N_v is because the domain truncation underestimates the error, while the Riemann sum overestimates it. Thus, the two errors can partly cancel each other out. Note that this only occurs at the beginning, where the Riemann sum error is large enough. As the Riemann sum error decreases, we see that the it stays below the total error.

[55] A. Y. Kitaev, Quantum measurements and the Abelian stabilizer problem, arXiv:quant-ph/9511026.

[56] M. Nielsen and I. Chuang, *Quantum Computation and Quantum Information: 10th Anniversary Edition* (Cambridge University Press, Cambridge, UK, 2010).

[57] D. S. Abrams and S. Lloyd, Quantum Algorithm Providing Exponential Speed Increase for Finding Eigenvalues and Eigenvectors, Phys. Rev. Lett. **83**, 5162 (1999).

[58] P. Jaksch and A. Papageorgiou, Eigenvector Approximation Leading to Exponential Speedup of Quantum Eigenvalue Calculation, Phys. Rev. Lett. **91**, 257902 (2003).

[59] H. Wang, L.-A. Wu, Y.-X. Liu, and F. Nori, Measurement-based quantum phase estimation algorithm for finding eigenvalues of non-unitary matrices, Phys. Rev. A **82**, 062303 (2010).

[60] A. Daskin, A. Grama, and S. Kais, A universal quantum circuit scheme for finding complex eigenvalues, Quantum Inf. Process. **13**, 333 (2014).

[61] C. Shao, Computing eigenvalues of diagonalizable matrices on a quantum computer, ACM Trans. Quantum Comput. **3**, 1 (2022).

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