Improved Precision Scaling for Simulating Coupled Quantum-Classical Dynamics

Sophia Simon,[1,*] Raffaele Santagati,[2,†] Matthias Degroote,[2] Nikolaj Moll,[2] Michael Streif,[2] and Nathan Wiebe[3,4,5,‡]

[1] *Department of Physics, University of Toronto, Toronto, Ontario M5S 1A7, Canada*
[2] *Quantum Lab, Boehringer Ingelheim, Ingelheim am Rhein 55218, Germany*
[3] *Department of Computer Science, University of Toronto, Toronto, Ontario M5S 2E4, Canada*
[4] *Pacific Northwest National Laboratory, Richland, Washington 99354, USA*
[5] *Canadian Institute for Advanced Studies, Toronto, Ontario M4G 4G8, Canada*

(Received 27 September 2023; accepted 12 February 2024; published 12 March 2024)

In this paper, we present a superpolynomial improvement in the precision scaling of quantum simulations for coupled quantum-classical systems. Such systems are found in, e.g., molecular-dynamics simulations within the Born-Oppenheimer approximation. By employing a framework based on the Koopman–von Neumann formulation of classical mechanics, we express the Liouville equation of motion as unitary dynamics and utilize phase kickback from a dynamical quantum simulation to calculate the quantum forces acting on classical particles. This approach allows us to simulate the dynamics of these classical particles without the overheads associated with measuring gradients and solving the equations of motion on a classical computer, resulting in a superpolynomial advantage at the price of increased space complexity. We demonstrate that these simulations can be performed in both microcanonical and canonical ensembles, enabling the estimation of thermodynamic properties from the prepared probability density.

DOI: 10.1103/PRXQuantum.5.010343

I. INTRODUCTION

We are accustomed to thinking of nature in terms of binaries. Specifically, we often speak of dynamics as if they were either purely quantum or purely classical. In reality, many models of physical interest actually share features with both quantum and classical matter. As a particular example, molecular dynamics (MD) is often formulated in this way, wherein the nuclei are assumed to follow Newton's equations but the electrons follow the Schrödinger equation. In other cases, we may treat an electromagnetic field as a time-dependent classical field and the particles interacting with it as quantum. In both cases, neither a fully quantum model nor a fully classical model can be used to address the problem efficiently. Quantum computers have long been known to provide, under reasonable complexity-theoretic conjectures, exponential advantages for simulating certain quantum systems [1–4]. Recently, this has been extended to show that quantum computers can offer exponential advantages even for systems that are classical [5]. However, when examining systems that straddle this line, such as the MD example considered above, the situation is not as clear, because such simulation methods are comparatively underexplored.

Descriptions within the Born-Oppenheimer (BO) approximation, where the wave functions of the nuclei can be considered independent of the wave functions of the electrons, play an important role in the chemical and pharmaceutical industries. These are often used to compute thermodynamic quantities of the chemical systems under study, such as the entropy or the free energy [6–10]. In fact, in classical computational chemistry, thermodynamic averages can be obtained by combining MD simulations with the use of thermostats to go beyond microcanonical ensembles [11–13]. MD simulations introduce another approximation on top of the BO approximation by treating the nuclei as classical particles but retaining the quantum description of the electrons. Recent works have explored the study of MD on fault-tolerant quantum computers, e.g., via force calculations, to update the coordinates of the classical nuclei [14–20]. Some of these works go beyond the Born-Oppenheimer approximation. In contrast to approaches where the full system is treated quantum mechanically, BO models have some advantages, namely, that the scale of the quantum and the classical dynamics

*sophia.simon@mail.utoronto.ca
†raffaele.santagati@boehringer-ingelheim.com
‡nathan.wiebe@utoronto.ca

Published by the American Physical Society under the terms of the Creative Commons Attribution 4.0 International license. Further distribution of this work must maintain attribution to the author(s) and the published article's title, journal citation, and DOI.

are naturally separated and that classical noise and external forces can be easily applied to the system without needing an expensive fully quantum description.

Recent work [20,21] has analyzed the cost of computing interatomic forces on quantum computers, which can be used to update the classical nuclear positions on a classical computer iteratively. In those works, the cost of an ϵ-approximate gradient evaluation has been found to scale with the error tolerance ϵ as $O(1/\epsilon)$. This approach is even less practical if we consider that, for many practical applications, we need to compute the properties of the chemical systems in the canonical ensemble [8,10,11,22]. A key problem in simulating MD in the canonical ensemble is to prepare the classical distribution of the nuclei (their probability density function) according to the Boltzmann distribution at a certain temperature. Current quantum computing methods do not allow for efficient encoding and evolution of the probability density of the classical degrees of freedom on the quantum computer. In the 1930s, Koopman and von Neumann (KvN) proposed an empirical formulation of classical mechanics that incorporated a Hilbert space consisting of complex and square-integrable wave functions $\psi_{\mathrm{KvN}}(q,p)$, which depend on the position q and momentum p of the particle [23]. These wave functions were understood as probability densities, $\rho(p,q) = \psi^*_{\mathrm{KvN}}\psi_{\mathrm{KvN}}$, of finding the particle in a specific configuration (p,q) of the phase space [24]. Both ρ and ψ_{KvN} evolve according to the Liouville equation,

$$\partial_t \rho = -iL\rho, \tag{1}$$

where L is a Hermitian operator called the Liouvillian operator. This gives a natural encoding (and subsequent evolution) of classical probability densities into quantum states that follow Hamiltonian dynamics.

The KvN formalism has recently been exploited to propose new algorithms for simulating classical systems on quantum computers and for solving nonlinear partial differential equations [25,26]. In this work, we do not use quantum linear systems algorithms but we explicitly evolve the positions and momenta in time. We provide a specific procedure for the time propagation along with the cost of the block encodings involved, as well as a method for obtaining ensemble averages from our final state.

We implement the time evolution of a hybrid quantum-classical system by propagating the discretized phase-space density of N classical nuclei that interact with $\tilde{N}$ electrons, which are treated quantumly using the first-quantized simulation method of Refs. [27,28]. We present this result for the microcanonical ensemble [11], where the number of particles $N_{\mathrm{tot}} = N + \tilde{N}$, the volume V, and the energy E are conserved. By adding a thermostat to the evolution, which couples the nuclear phase-space density to a classical heat bath, we can impose a constant temperature T, allowing us to perform the simulation in the canonical ensemble [12]. The asymptotic gate complexity of the Liouvillian simulation algorithm is in

$$\widetilde{O}\left(\frac{N_{\mathrm{tot}}\mu^{2+o(1)}t^{1+o(1)}}{\widetilde{\gamma}\,\widetilde{\delta}\,\epsilon^{o(1)}}\log\left(\frac{1}{\xi}\right)\right), \tag{2}$$

where μ is an upper bound on the spectral norm of the Liouvillian operator, t is the evolution time, $\widetilde{\gamma}$ is a lower bound on the spectral gap of the electronic Hamiltonian over all configurations visited during the simulation, $\widetilde{\delta}$ is a lower bound on the overlap of the initial electronic state with the target electronic state over all configurations visited during the simulation, ϵ is the desired simulation precision, and ξ is an upper bound on the failure probability. Our result provides superpolynomially better scaling with ϵ than the existing approach of Ref. [20], which in turn suggests that the road blocks previously identified for MD therein may not be the obstacles that they were previously believed to be. Additionally, the asymptotic space complexity of the Liouvillian simulation algorithm is moderate, scaling linearly in N_{tot} and logarithmically in all other simulation parameters, including the grid spacing as well as the precision ϵ.

To tackle the problem of computing thermodynamic quantities, we introduce a second algorithm, which, given a quantum state encoding the discretized phase-space density of the system together with the heat bath (e.g., obtained using the first algorithm), can output an estimate of the free energy of the system. In contrast to classical methods and previous approaches for MD simulations on quantum computers [15–18], our approach exploits a fully coherent state preparation of the classical probability density in the canonical ensemble, avoiding sampling and enabling the direct estimation of thermodynamic properties [10,20].

The paper is structured as follows. In Sec. II, we introduce the main concepts for coupled quantum-classical dynamics in the Liouvillian picture. We show that a thermostat can be used to prepare the canonical ensemble and that we can implement this simulation on a quantum computer by discretizing the phase space. Our main results are presented in Sec. III, which includes precise statements of the computational problems as well as the asymptotic gate complexity of our algorithms to solve these problems. In Sec. IV, we give an overview of our quantum algorithms for simulating Liouvillian dynamics and estimating the free energy of a quantum-classical system. We conclude with a brief discussion in Sec. V. The proofs of the theorems and lemmas presented in the main text are given in the Appendices A–E.

II. PRELIMINARIES

In this section, we provide an overview of the fundamental definitions, from the Liouvillian formalism to the

mappings into qubit registers and the microcanonical and canonical ensembles, that are essential for the definition of the computational problems and the implementation of the algorithms.

A. Liouvillian formalism

The trajectories of N classical particles in three dimensions are governed by Newton's equations of motion:

$$F_{n,j} = m_n \ddot{x}_{n,j}, \quad n \in \{1, 2, \ldots, N\},\ j \in \{1, 2, 3\}, \tag{3}$$

where $F_{n,j}$ is the jth component of the force on the nth particle, $\ddot{x}_{n,j}$ is its acceleration, and m_n is its mass. In general, these differential equations are nonlinear and nonunitary, meaning that the time evolution of the positions of the particles cannot be directly implemented on a quantum computer. The Born-Oppenheimer approximation in MD turns the Hamiltonian problem of jointly evolving nuclei and electrons into an example of such a nonlinear and nonunitary time evolution. We overcome the issue by working with the Liouvillian formulation of classical mechanics instead, which is centered around the phase-space probability density $\rho(\{x_n\}, \{p_n\}, t)$ of the classical particles. The probability density depends on the positions $x_n \in \mathbb{R}^3$ and momenta $p_n \in \mathbb{R}^3$ of the particles as well as on time t. It is normalized according to $\int_{\mathbb{R}^{6N}} \rho(\{x_n\}, \{p_n\}, t)\, d\{x_n\} d\{p_n\} = 1$ and satisfies the following equation of motion:

$$\frac{\partial \rho}{\partial t} = -iL\rho, \tag{4}$$

where L is the Liouvillian operator, defined as

$$L := -i \sum_{n=1}^{N} \sum_{j=1}^{3} \left(\frac{\partial H}{\partial p_{n,j}} \partial_{x_{n,j}} - \frac{\partial H}{\partial x_{n,j}} \partial_{p_{n,j}} \right). \tag{5}$$

Here, H is the classical Hamiltonian of the system and $\partial_{x_{n,j}}$ ($\partial_{p_{n,j}}$) denotes the partial derivative with respect to the jth position (momentum) component of the nth particle.

The formal solution to Eq. (4) reads

$$\rho(t) = e^{-iLt} \rho(0). \tag{6}$$

Note that L is Hermitian, which implies that e^{-iLt} is unitary. The similarities to quantum mechanics become even more apparent when employing the KvN formulation of classical mechanics [24]. The idea is to introduce a complex wave function, $\psi_{\mathrm{KvN}}(\{x_n\}, \{p_n\}, t)$, which evolves according to the Liouville equation just like $\rho(\{x_n\}, \{p_n\}, t)$:

$$\frac{\partial \psi_{\mathrm{KvN}}}{\partial t} = -iL\psi_{\mathrm{KvN}}. \tag{7}$$

The phase-space density can then be recovered via the relation $\rho = \psi^*_{\mathrm{KvN}} \psi_{\mathrm{KvN}}$, which resembles the quantum mechanical calculation of probabilities from amplitudes. This works out because the Liouvillian contains only first-order derivatives, $\partial_{x_{n,j}}$ and $\partial_{p_{n,j}}$, meaning that the product of two wave functions, each satisfying the Liouville equation, also satisfies the Liouville equation. In contrast, the Schrödinger equation of quantum mechanics generically contains second-order derivatives, $\partial^2_{x_{n,j}}$, implying that the product of two solutions does not necessarily satisfy the Schrödinger equation.

Consideration of the KvN wave function ψ_{KvN} rather than the phase-space density ρ has one significant advantage: ψ_{KvN} can be easily encoded on a quantum computer, since it has the same properties as a "true" quantum wave function. For example, while ρ needs to be real valued and positive, ψ_{KvN} can take on complex values. Furthermore, ψ_{KvN} is normalized according to the 2-norm, i.e., $\int_{\mathbb{R}^{6N}} |\psi_{\mathrm{KvN}}|^2 d\{x_n\} d\{p_n\} = 1$, in contrast to ρ, which is normalized according to the 1-norm.

While the Liouvillian formalism can be applied to any classical system governed by a Hamiltonian, we will focus on MD within the Born-Oppenheimer approximation. In this setting, the nuclei are treated as classical particles, whereas the electrons are treated quantumly. More specifically, solving the electronic Schrödinger equation as a function of the nuclear positions yields potential-energy surfaces that determine the dynamics of the classical nuclei. We consider MD simulations in the microcanonical and the canonical ensemble, as discussed in the following subsections.

B. Evolution in the microcanonical (*NVE*) ensemble

The microcanonical (*NVE*) ensemble is a thermodynamic ensemble in which the number of nuclei N, the volume V, and the total energy E of the system are constants of motion. In order to prevent potential misunderstandings, let us recall that the phase-space representation of an *NVE* ensemble is usually considered to be a time-independent phase-space density ρ_{NVE} that is constant over all configurations with energy E and zero otherwise. This is often written in terms of a Dirac delta distribution:

$$\rho_{NVE}(\{x_n\}, \{p_n\}) \propto \delta(H(\{x_n\}, \{p_n\}) - E). \tag{8}$$

In the following, we will not assume that the phase-space density is given by $\rho_{NVE}(\{x_n\}, \{p_n\})$ when we talk about simulations in the *NVE* ensemble. Rather, we refer to constant-energy dynamics of a generic time-dependent phase-space density, which evolves according to Eq. (6).

The classical Hamiltonian of the nuclei in the *NVE* ensemble takes the following form:

$$H^{(NVE)}_{\mathrm{nuc}} := H^{(NVE)}_{\mathrm{class}}(\{x_n\}, \{p_n\}) + E_{\mathrm{el}}(\{x_n\}), \tag{9}$$

where

$$H_{\text{class}}^{(NVE)} := \sum_{n=1}^{N}\sum_{j=1}^{3}\frac{p_{n,j}^2}{2m_n} + \sum_{k=1}^{N-1}\sum_{n>k}^{N}\frac{Z_n Z_k}{\|x_n - x_k\|} \tag{10}$$

is the classical Hamiltonian of the nuclei without any electronic contributions. The mass and the atomic number of the nth nucleus are denoted by m_n and Z_n, respectively. Unless stated otherwise, we will use $\|\cdot\|$ to refer to the (induced) 2-norm. $H_{\text{nuc}}^{(NVE)}$ also depends on $E_{\text{el}}(\{x_n\})$, the ground-state energy of the following quantum Hamiltonian governing the dynamics of the electrons for fixed nuclear positions:

$$H_{\text{el}} := -\sum_{n=1}^{\tilde{N}}\sum_{j=1}^{3}\frac{\nabla_{n,j}^2}{2} + \sum_{n>k}^{\tilde{N}}\frac{1}{\|\tilde{x}_n - \tilde{x}_k\|} - \sum_{k,n=1}^{\tilde{N},N}\frac{Z_n}{\|\tilde{x}_k - x_n\|}, \tag{11}$$

where $\tilde{N}$ is the number of electrons, $\tilde{x}_n \in \mathbb{R}^3$ denotes the position of the nth electron, and $\nabla_{n,j} := \partial_{\tilde{x}_{n,j}}$ is the partial derivative operator with respect to the jth coordinate of the nth electron. Note that the total number of particles, $N_{\text{tot}} := N + \tilde{N}$, is also conserved in the *NVE* ensemble (the same is true for the *NVT* ensemble discussed in Sec. II C).

So far, we have worked with continuous position and momentum variables that can take on any real value. However, to simulate the time evolution of the phase-space density according to Eq. (6) on a quantum computer, we need to consider a finite discretized phase space. The idea is to restrict each position and momentum component of the nuclei to a finite set of

$$g_x := \frac{x_{\max}}{h_x} \in \mathbb{N}, \tag{12}$$

$$g_p := \frac{p_{\max}}{h_p} \in \mathbb{N} \tag{13}$$

values, respectively, where $x_{\max}$ ($p_{\max}$) is the maximum attainable value of any $x_{n,j}$ ($p_{n,j}$) and h_x (h_p) is the grid spacing. The choice of grid spacing depends on the smoothness of the phase-space density. A detailed error analysis regarding the grid spacing for real-space simulations can be found in Ref. [29]. Further bounds on the grid spacing and numerical results are presented in Ref. [20]. Since we are considering a finite simulation box, we must also specify the boundary conditions. For simplicity, we choose periodic boundary conditions.

Each grid point of the discretized phase space corresponds to a computational basis state of the form

$$|\{\bar{x}_{n,j}\},\{\bar{p}_{n,j}\}\rangle := \bigotimes_{n,j}\Big(|\bar{x}_{n,j}\rangle \otimes |\bar{p}_{n,j}\rangle\Big), \tag{14}$$

where $\bar{x}_{n,j} \in [g_x]$ and $\bar{p}_{n,j} \in [g_p]$ are integers such that $x_{n,j} = \bar{x}_{n,j}h_x - x_{\max}/2$ and $p_{n,j} = \bar{p}_{n,j}h_p - p_{\max}/2$. Thus, $|\bar{x}_{n,j}\rangle$ is a computational basis state on $\log g_x$ qubits specifying the value of the jth discrete position coordinate of the nth nucleus and analogously for $|\bar{p}_{n,j}\rangle$. The mapping to qubits to obtain the computational basis is shown in Fig. 1.

Given a classical $\left(g_x^{3N}g_p^{3N}\right)$-dimensional probability vector $\vec{\rho}_0$ encoding the discretized initial phase-space density ρ_0, the kth amplitude of the initial quantum state representing the associated KvN wave function can simply be chosen to be $\sqrt{(\vec{\rho}_0)_k}$. In other words, the quantum register is initially prepared in the state $|\rho_0\rangle := \sum_k \sqrt{(\vec{\rho}_0)_k}|k\rangle$, where $k \in [g_x^{3N}g_p^{3N}]$ enumerates the points of the discretized phase space. Note that this is just a convenient relabeling of the computational basis states $\{|\{\bar{x}_{n,j}\},\{\bar{p}_{n,j}\}\rangle\}$ introduced before. Depending on the choice of the initial phase-space density of the classical particles, one can use a number of different general-purpose state-preparation methods to prepare a quantum state encoding the initial phase-space density. For example, if the initial phase-space density is efficiently integrable, one can use the Grover-Rudolph algorithm to prepare the corresponding quantum state [30]. For sparse quantum states, this method scales quadratically with the number of qubits [31]. Another option is to use the approach developed in Ref. [32], based on quantum singular-value transformation (QSVT). The authors' method provides a qubit-efficient way of encoding functions with a well-behaved polynomial or Fourier expansion in the amplitudes of a quantum state.

In order to evolve the discretized quantum state that encodes our system on a quantum computer, we also need to discretize the Liouvillian operator defined in Eq. (5). This requires us to define discrete versions of the derivative operators appearing in the Liouvillian operator. Central-finite-difference schemes are a popular tool for discretizing derivatives [33]. In the quantum setting, the corresponding discrete operator can be defined as follows.

Definition 1 (Discrete derivative operator). Let $\{|\bar{y}\rangle\}$ denote a complete set of computational basis states, representing the variable with respect to the derivative operator is applied, e.g., $x_{n,j}$ or $p_{n,j}$. The discrete derivative operator D_y of order $2d$ is defined as follows:

$$D_y := \frac{1}{h}\sum_{\bar{y}}\sum_{k=-d}^{d} c_{d,k}|\bar{y}-k\rangle\langle\bar{y}|, \tag{15}$$

where h is the user-specified grid spacing and the coefficients $\{c_{d,k}\}$ are given by [33]

$$c_{d,k} := \begin{cases} \dfrac{(-1)^{k+1}(d!)^2}{k(d-k)!(d+k)!}, & \text{if } k \neq 0 \\ 0, & \text{else.} \end{cases} \tag{16}$$

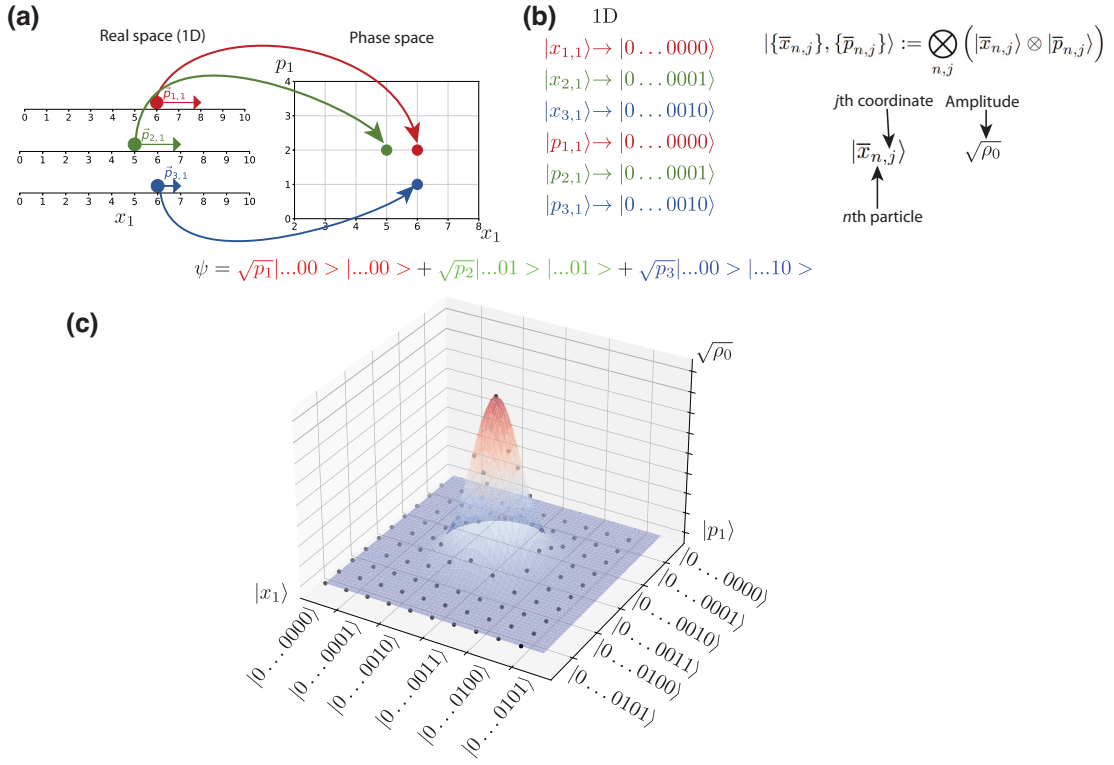

FIG. 1. Mapping from real space to phase space and to qubit registers for a one-dimensional (1D) problem. (a) Each configuration is stored as a point in the phase space. (b) Each point of the discretized phase space is associated with a computational basis state on the quantum computer. The computational basis states can be written as the tensor product of the states encoding the discretized positions and momenta of the individual particles. The amplitude of a computational basis state defines the corresponding probability for that point in the probability density function. (c) An example encoding of the discretized phase density of one particle in one spatial dimension.

In general, the higher the order $2d$ of the finite-difference scheme, the better is the error scaling with respect to the grid spacing (for more details, see Lemma 8 or Ref. [20]).

We are now ready to define the discretized Liouvillian operator in the *NVE* ensemble.

Definition 2 (Discretized Liouvillian operator for BO MD in the NVE ensemble). Let $H_{\mathrm{nuc}}^{(NVE)}$ be the BO Hamiltonian from Eq. (9). The discretized Liouvillian for simulations in the *NVE* ensemble is given by

$$L_{NVE} := -i\sum_{n=1}^{N}\sum_{j=1}^{3}\left(D_{x_{n,j}}\otimes\frac{\partial H_{\mathrm{nuc}}^{(NVE)}}{\partial p_{n,j}} - \frac{\partial H_{\mathrm{nuc}}^{(NVE)}}{\partial x_{n,j}}\otimes D_{p_{n,j}}\right), \tag{17}$$

where

$$\frac{\partial H_{\mathrm{nuc}}^{(NVE)}}{\partial p_{n,j}} = \sum_{\bar{p}_{n,j}}\frac{p_{n,j}}{m_n}|\bar{p}_{n,j}\rangle\langle\bar{p}_{n,j}| \tag{18}$$

$$\frac{\partial H_{\mathrm{nuc}}^{(NVE)}}{\partial x_{n,j}} = \sum_{n'\neq n}\sum_{\bar{x}_n}\sum_{\bar{x}_{n'}}\frac{-Z_nZ_{n'}}{\left(\|x_n-x_{n'}\|^2+\Delta^2\right)^{3/2}} \times (x_{n,j}-x_{n',j})|\bar{x}_n\rangle\langle\bar{x}_n|\otimes|\bar{x}_{n'}\rangle\langle\bar{x}_{n'}| + D_{n,j}^{\mathrm{el}} \tag{19}$$

are now diagonal matrices of dimension $g_x^{3N}g_p^{3N}$. Δ is a gap parameter introduced to regularize the Coulomb interaction and to avoid infinities in the simulation. $D_{x_{n,j}}$ is a discrete derivative operator of order $2d_x$ and $D_{p_{n,j}}$ is a discrete derivative operator of order $2d_p$. Furthermore,

$$D_{n,j}^{\mathrm{el}} := \frac{1}{h_x}\sum_{k=-d_e}^{d_e}\sum_{(n',j')\neq(n,j)}\sum_{\bar{x}_{n',j'}}\sum_{\bar{x}_{n,j}} c_{d_e,k} \times E_{\mathrm{el}}\left(\{x_{n',j'}\},x_{n,j}+kh_x\right)|\bar{x}_{n',j'}\rangle\langle\bar{x}_{n',j'}|\otimes|\bar{x}_{n,j}\rangle\langle\bar{x}_{n,j}| \tag{20}$$

is a central-finite-difference approximation of order $2d_e$ to $\partial E_{\mathrm{el}}/\partial x_{n,j}$. Note that we only show the quantum registers that are acted on in a nontrivial manner. For example,

$$\sum_{\bar{p}_{1,1}}|\bar{p}_{1,1}\rangle\langle\bar{p}_{1,1}| \equiv \left(\mathbb{1}_{x_{1,1}}\otimes\sum_{\bar{p}_{1,1}}|\bar{p}_{1,1}\rangle\langle\bar{p}_{1,1}|\right) \otimes\left(\mathbb{1}_{x_{1,2}}\otimes\mathbb{1}_{p_{1,2}}\right)\otimes\left(\mathbb{1}_{x_{1,3}}\otimes\mathbb{1}_{p_{1,3}}\right) \bigotimes_{n=2,j=1}^{N,3}\left(\mathbb{1}_{x_{n,j}}\otimes\mathbb{1}_{p_{n,j}}\right). \tag{21}$$

The reason for introducing $D_{n,j}^{\mathrm{el}}$ in the above definition is that we generally do not have an analytic expression for $\partial E_{\mathrm{el}}/\partial x_{n,j}$.

In Appendix C, we show that the spectral norm of L_{NVE} is upper bounded by μ_{NVE}, which is defined as follows:

$$\begin{aligned}\mu_{NVE} := 3N\frac{p_{\max}}{m_{\min}}\frac{2\,(\ln d_x+1)}{h_x} \\ + 6N^2\frac{2Z_{\max}^2 x_{\max}}{\Delta^3}\frac{2\,(\ln d_p+1)}{h_p} \\ + 3N\lambda\frac{2\ln(d_e+1)}{h_x h_p},\end{aligned} \tag{22}$$

where λ is an upper bound on the spectral norm of the discretized electronic Hamiltonian, as discussed in Lemma 3.

Since the discrete Liouvillian L_{NVE} is still Hermitian, we can use tools from Hamiltonian simulation [34,35] to efficiently implement the unitary $U_{NVE} := e^{-iL_{NVE}t}$ on a quantum computer to simulate the time evolution of the discretized phase-space density. Our quantum simulation algorithm is discussed in more detail in Sec. IV.

C. Evolution in the canonical (*NVT*) ensemble

By default, MD simulations are performed in the *NVE* ensemble. However, it is often desirable to perform simulations in the canonical (*NVT*) ensemble, where the temperature T rather than the energy is held constant. This is especially true when performing conformational searches of molecules, such as those required in drug design [36].

The Nosé-Hoover thermostat is a common choice in classical MD calculations to simulate dynamics in the *NVT* ensemble [37,38]. This thermostat is based on non-Hamiltonian equations of motion, meaning that there does not exist an underlying Hamiltonian governing the dynamics of the system. Therefore, it cannot be straightforwardly incorporated into the Liouvillian framework.

In contrast, the original Nosé thermostat is compatible with the Liouvillian framework, since the equations of motion can be derived from an extended-system Hamiltonian [12]. The idea is to introduce additional terms to the classical Hamiltonian that involve an extra degree of freedom, s, representing a heat bath. This effectively allows the kinetic energy of the nuclei to be exchanged with the energy of the bath, so that the system can be equilibrated to a user-specified temperature T. The extended-system Hamiltonian is defined as follows:

$$H_{\mathrm{nuc}}^{(NVT)} := H_{\mathrm{class}}^{(NVT)}\left(\{x_n\},\{p_n'\},s,p_s\right) + E_{\mathrm{el}}\left(\{x_n\}\right), \tag{23}$$

where

$$\begin{aligned}H_{\mathrm{class}}^{(NVT)} := \sum_{n=1}^{N}\sum_{j=1}^{3}\frac{p'^2_{n,j}}{2m_n s^2} + \sum_{k=1}^{N-1}\sum_{n>k}^{N}\frac{Z_n Z_k}{\|x_n-x_k\|^2} + \frac{p_s^2}{2Q} \\ + N_f\,k_B T\ln(s)\,.\end{aligned} \tag{24}$$

Here, p_s is the momentum variable conjugate to s and Q is an effective mass of s, which controls the coupling of the system to the heat bath. k_B is the Boltzmann constant and $N_f = 3N-K$ is equal to the number of degrees of freedom of the system, with K being the number of constraints. The heat bath modifies the kinetic energy term of the nuclei, while the Coulomb-potential term remains unaffected. In particular, $p'_{n,j}$ is the conjugate momentum variable to $x_{n,j}$ in the extended system. It is often called a "virtual" momentum variable and it is related to the real momentum variable $p_{n,j}$ of the physical system via the equation

$$p_{n,j} = \frac{p'_{n,j}}{s}. \tag{25}$$

The third term of $H_{\mathrm{class}}^{(NVT)}$ represents the kinetic energy of the heat bath, while the last term represents the potential energy of the heat bath. This potential-energy term ensures that the partition function $\mathcal{Z}$ associated with a microcanonical ensemble in the extended system gives rise to a canonical partition function when restricted to the real system [37,39]:

$$\begin{aligned}\mathcal{Z} &\propto \int d\{x_n\}\int d\{p_n'\} \\ &\quad\times\int ds\int dp_s\,\delta\left(H_{\mathrm{nuc}}^{(NVT)}\left(\{x_n\},\{p_n'\},s,p_s\right)-E_{\mathrm{ext}}\right) \\ &\propto \int d\{x_n\}\int d\{p_n\}e^{-H_{\mathrm{nuc}}^{(NVE)}(\{x_n\},\{p_n\})/(k_B T)},\end{aligned} \tag{26}$$

where E_{ext} is the conserved energy of the extended system.

To avoid confusion later on, let us briefly mention here that the phase-space version of an *NVT* ensemble is usually considered to be a time-independent phase-space density ρ_{NVT} that has the form of a Boltzmann distribution:

$$\rho_{NVT}\left(\{x_n\},\{p_n\}\right) \propto e^{-E(\{x_n\},\{p_n\})/(k_b T)}, \tag{27}$$

where E is the energy of the system (nuclei) for a given configuration. In the following, we will not assume that the phase-space density is given by $\rho_{NVT}\left(\{x_n\},\{p_n\}\right)$ when we talk about simulations in the *NVT* ensemble. Rather, we will refer to the constant-temperature dynamics of a generic time-dependent phase-space density obtained by evolving the joined probability density of the system and heat bath according to Eq. (6) and then integrating out the heat bath. However, if the dynamics of the extended system are ergodic, then we can mimic the behavior of

$\rho_{NVT}(\{x_n\},\{p_n\})$ in the sense that we can estimate thermodynamic properties such as the free energy of the system via coherent time averaging, as explained in more detail in Sec. IV A.

As with the *NVE* ensemble, we need to discretize the (now $(6N+2)$-dimensional) phase space to simulate the time evolution of the phase-space density $\rho\left(\{x_n\},\{p'_n\},s,p_s\right)$ according to Eq. (6) on a quantum computer. We do so by restricting each position and virtual momentum component of the nuclei as well as the bath variables s and p_s to a finite set of

$$g_x = \frac{x_{\max}}{h_x} \in \mathbb{N} \tag{28}$$

$$g_{p'} := \frac{p'_{\max}}{h_{p'}} \in \mathbb{N} \tag{29}$$

$$g_s := \frac{s_{\max}}{h_s} \in \mathbb{N} \tag{30}$$

$$g_{p_s} := \frac{p_{s,\max}}{h_{p_s}} \in \mathbb{N} \tag{31}$$

values, respectively, where $x_{\max}$ is the maximum attainable value of $x_{n,j}$ and h_x is the position grid spacing as before. Similarly, $p'_{\max}$, $s_{\max}$ and $p_{s,\max}$ are the maximum attainable values of $p'_{n,j}$, s, and p_s and $h_{p'}$, h_s, and h_{p_s} are the respective grid spacings. Since we are considering a finite simulation box, we also need to specify the boundary conditions. For simplicity, we again choose periodic boundary conditions.

The computational basis states in the *NVT* ensemble are of the form

$$|\{\bar{x}_{n,j}\},\{\bar{p}'_{n,j}\},s,p_s\rangle := \bigotimes_{n,j}\left(|\bar{x}_{n,j}\rangle\otimes|\bar{p}'_{n,j}\rangle\right)\otimes|\bar{s}\rangle\otimes|\bar{p}_s\rangle, \tag{32}$$

where $\bar{x}_{n,j}\in[g_x]$, $\bar{p}'_{n,j}\in[g_{p'}]$, $\bar{s}\in[g_s]$ and $\bar{p}_s\in[g_{p_s}]$ are integers such that $x_{n,j}=\bar{x}_{n,j}h_x - x_{\max}/2$, $p'_{n,j}=\bar{p}'_{n,j}h_{p'} - p'_{\max}/2$, $s=\bar{s}h_s$ and $p_s=\bar{p}_s h_{p_s} - p_{s,\max}/2$.

We again employ the KvN formalism to encode the (discretized) phase-space density in a quantum state on a quantum computer.

The discretized Liouvillian in the *NVT* ensemble is then defined as follows.

Definition 3 (Discretized Liouvillian operator for BO MD in the NVT ensemble). Let $H^{(NVT)}_{\text{nuc}}$ be the *NVT* Hamiltonian as defined in Eq. (23). The discretized Liouvillian for simulations in the *NVT* ensemble is given by

$$L_{NVT} := -i\sum_{n=1}^{N}\sum_{j=1}^{3}\left(D_{x_{n,j}}\otimes\frac{\partial H^{(NVT)}_{\text{nuc}}}{\partial p'_{n,j}} - \frac{\partial H^{(NVT)}_{\text{nuc}}}{\partial x_{n,j}}\otimes D_{p'_{n,j}}\right) - i\left(D_s\otimes\frac{\partial H^{(NVT)}_{\text{nuc}}}{\partial_{p_s}} - \frac{\partial H^{(NVT)}_{\text{nuc}}}{\partial_s}\otimes D_{p_s}\right), \tag{33}$$

where

$$\frac{\partial H^{(NVT)}_{\text{nuc}}}{\partial p'_{n,j}} = \sum_{\bar{p}'_{n,j}}\sum_{\bar{s}}\frac{p'_{n,j}}{m_n\,(s+s_{\min})^2}|\bar{p}'_{n,j}\rangle\langle\bar{p}'_{n,j}|\otimes|\bar{s}\rangle\langle\bar{s}|, \tag{34}$$

$$\frac{\partial H^{(NVT)}_{\text{nuc}}}{\partial x_{n,j}} = \sum_{n'\neq n}\sum_{\bar{x}_n}\sum_{\bar{x}_{n'}}\frac{-Z_nZ_{n'}}{\left(\|x_n-x_{n'}\|^2+\Delta^2\right)^{3/2}} \times (x_{n,j}-x_{n',j})|\bar{x}_n\rangle\langle\bar{x}_n|\otimes|\bar{x}_{n'}\rangle\langle\bar{x}_{n'}| + D^{\text{el}}_{n,j}, \tag{35}$$

$$\frac{\partial H^{(NVT)}_{\text{nuc}}}{\partial p_s} = \sum_{\bar{p}_s}\frac{p_s}{Q}|\bar{p}_s\rangle\langle\bar{p}_s|, \tag{36}$$

$$\frac{\partial H^{(NVT)}_{\text{nuc}}}{\partial s} = -\sum_{\bar{p}_{n,j}}\sum_{\bar{s}}\frac{2p'^2_{n,j}}{m_n\,(s+s_{\min})^3}|\bar{p}_{n,j}\rangle\langle\bar{p}_{n,j}|\otimes|\bar{s}\rangle\langle\bar{s}| + \sum_{\bar{s}}\frac{N_f\,k_BT}{s+s_{\min}}|\bar{s}\rangle\langle\bar{s}| \tag{37}$$

are now diagonal matrices of dimension $\eta := g_x^{3N}g_{p'}^{3N}g_sg_{p_s}$. As with the *NVE* Liouvillian, Δ is a gap parameter introduced to regularize the Coulomb interaction and to avoid infinities in the simulation. The bath variable cutoff $s_{\min}$ is introduced for the same reason. $D_{x_{n,j}}$ is a discrete derivative operator of order $2d_x$, $D_{p'_{n,j}}$ is a discrete derivative operator of order $2d_{p'}$, D_s is a discrete derivative operator of order $2d_s$, and D_{p_s} is a discrete derivative operator of order $2d_{p_s}$. They are constructed according to Definition 1. $D^{\text{el}}_{n,j}$ is again the finite-difference approximation to $\partial E_{\text{el}}/\partial x_{n,j}$. Note that we only show the quantum registers that are acted on in a nontrivial manner.

In Appendix C, we show that the spectral norm of L_{NVT} is upper bounded by μ_{NVT}, which is defined as follows:

$$\begin{aligned}\mu_{NVT} := {} & 3N\frac{p_{\max}}{m_{\min}s^2_{\min}}\frac{2\,(\ln d_x+1)}{h_x} + 6N^2\frac{2Z^2_{\max}x_{\max}}{\Delta^3}\frac{2\left(\ln d_p+1\right)}{h_p} + 3N\lambda\frac{2\ln(d_e+1)}{h_xh_p} \\ & + \frac{p_{s,\max}}{Q}\frac{2\,(\ln d_s+1)}{h_s} + 3N\frac{2p^2_{\max}}{m_{\min}s^3_{\min}}\frac{2\left(\ln d_{p_s}+1\right)}{h_{p_s}} + \frac{N_f\,k_BT}{s_{\min}}\frac{2\left(\ln d_{p_s}+1\right)}{h_{p_s}}.\end{aligned} \tag{38}$$

III. MAIN RESULTS

Our main result is an efficient quantum algorithm for simulating the time evolution of the discretized phase-space density of N nuclei within the Born-Oppenheimer approximation. The formal problem can be stated as follows.

Problem 1 (Simulating Liouvillian dynamics within the Born-Oppenheimer approximation). Let $L \in \{L_{NVE}, L_{NVT}\}$ be the discretized Liouvillian governing the dynamics of the discretized phase-space density associated with N classical nuclei in the *NVE* ensemble (Definition 2) or the *NVT* ensemble (Definition 3). Given a quantum state $|\rho_0\rangle$ encoding the initial discretized phase-space density, output a quantum state that is ϵ-close in ℓ^2 distance to $|\rho_t\rangle := e^{-iLt}|\rho_0\rangle$.

Our algorithm requires access to an initial electronic state-preparation oracle $\widetilde{U}_I$, the precise definition of which is given later in Definition 9. The main feature of $\widetilde{U}_I$ is that it prepares an initial electronic state $|\phi_0\{x_n\}\rangle$ that has nontrivial overlap with the ground state $|\widetilde{\psi}_0\{x_n\}\rangle$ of a discretized version of the electronic Hamiltonian from Eq. (11). To be more specific, let $0 < \widetilde{\delta} \leq 1$. Then,

$$\widetilde{U}_I|\{\overline{x}_n\}\rangle|0\rangle = |\{\overline{x}_n\}\rangle|\phi_0\{x_n\}\rangle, \tag{39}$$

where $\left|\langle\widetilde{\psi}_0\{x_n\}|\phi_0\{x_n\}\rangle\right| \geq \widetilde{\delta}$ for all nuclear configurations visited during the simulation. In other words, $\widetilde{\delta}$ is a lower bound on the overlap of the initial electronic state with the true electronic ground state over all nuclear grid points associated with a nonzero amplitude at some point during the simulation.

Unless stated otherwise, we use "log" to refer to the binary logarithm. Furthermore, we write $O\left(z^{o(1)}\right)$ to indicate subpolynomial scaling with respect to the parameter z and we use the $\widetilde{O}$ notation to hide subdominant logarithmic factors. With this in mind, we present our first result below.

Theorem 1 (Complexity of Born-Oppenheimer Liouvillian simulation). There exists a quantum algorithm that solves Problem 1 with success probability $\geq 1 - \xi$ using

$$\widetilde{O}\left(\frac{N_{\text{tot}}\, d\mu^{2+o(1)} t^{1+o(1)}}{\widetilde{\gamma}\,\widetilde{\delta}\,\epsilon^{o(1)}} \log\left(\frac{1}{\xi}\right)\right)$$

Toffoli gates, where d is the maximum order of the finite-difference schemes used, $\mu \in \{\mu_{NVE}, \mu_{NVT}\}$ is an upper bound on the spectral norm of the discretized Liouvillian $L \in \{L_{NVE}, L_{NVT}\}$, and $\widetilde{\gamma}$ is a lower bound on the spectral gap of the discretized electronic Hamiltonian over all phase-space grid points that are associated with a nonzero amplitude at some point during the simulation. Additionally,

$$\widetilde{O}\left(\frac{N d\mu^{1+o(1)} t^{1+o(1)}}{\widetilde{\delta}\,\epsilon^{o(1)}} \log\left(\frac{1}{\xi}\right)\right)$$

queries to the initial electronic state-preparation oracle $\widetilde{U}_I$ are needed.

Theorem 1 is proved in Appendix C. In comparison to gradient-based approaches [20], which, in the worst case scale exponentially with the evolution time (see Appendix E), our approach scales polynomially in time t and subpolynomially with error ϵ.

We also show how to use our Liouvillian simulation algorithm to estimate the free energy of the nuclei in the *NVT* ensemble, assuming that the dynamics of the extended system are ergodic. Usually, we are interested in the free energy when the system reaches equilibrium. The thermostat allows us to reach thermal equilibrium and then estimate thermodynamic properties such as the free energy of the classical system. A conceptual challenge that arises in the computation of the free energy is the definition of macrostates in the probability distribution. Specifically, we envision these microstates to be hypercubes in phase space and we define the probability of finding the entire system within this hypercube to be p_i. With this in mind, the definition of the discrete free energy is given below.

Definition 4 (Free energy). Let p_i denote the probability of a classical system being in the ith (discrete) microstate and let E_i be the energy associated with the ith microstate. Let

$$\mathcal{S}_G := -k_B \sum_j p_j \ln p_j \tag{40}$$

be the Gibbs entropy of the system, where k_B is the Boltzmann constant. Furthermore, let

$$\mathcal{U} := \sum_j p_j E_j \tag{41}$$

be the internal energy of the system. The free energy F of the system is then given by

$$\mathcal{F} := \mathcal{U} - T\mathcal{S}_G, \tag{42}$$

where T is the temperature of the system.

In our case, the energies $\{E_i\}$ are the eigenvalues of $H_{\text{nuc}} := H_{\text{kin}} + H_{\text{pot}} + H_{E_{\text{el}}}$, where

$$H_{\text{kin}} := \sum_{n,j} \sum_{\overline{p}'_{n,j}} \sum_{\overline{s}} \frac{p'^2_{n,j}}{m_n(s + s_{\min})^2} |\overline{p}'_{n,j}\rangle\langle\overline{p}'_{n,j}| \otimes |\overline{s}\rangle\langle\overline{s}|, \tag{43}$$

$$H_{\text{pot}} := \sum_{n' \neq n} \sum_{\overline{x}_n} \sum_{\overline{x}_{n'}} \frac{Z_n Z_{n'}}{\left(\|x_n - x_{n'}\|^2 + \Delta^2\right)^{1/2}} |\overline{x}_n\rangle\langle\overline{x}_n| \times \otimes |\overline{x}_{n'}\rangle\langle\overline{x}_{n'}|, \tag{44}$$

$$H_{E_{\text{el}}} := \sum_{\{\overline{x}_n\}} E_{\text{el}}\left(\{x_n\}\right) |\{\overline{x}_n\}\rangle\langle\{\overline{x}_n\}|. \tag{45}$$

In Appendix D, we show that the spectral norm of H_{nuc} is upper bounded by α_{nuc}, which is defined as follows:

$$\alpha_{\text{nuc}} := 3N\frac{p'^2_{\max}}{m_{\min}s^2_{\min}} + N^2\frac{Z^2_{\max}}{\Delta} + \lambda. \tag{46}$$

We now give a formal definition of the free-energy-estimation problem.

Problem 2 (Estimation of the free energy of a phase-space distribution). Let L_{NVT} be the discretized Liouvillian operator in the *NVT* ensemble as in Definition 3. Given a quantum state $|\rho_0\rangle$ encoding the initial discretized phase-space density of the system and the heat bath, output an ϵ-precise estimate of the free energy $\mathcal{F}$ of the system after time t, i.e., estimate $\mathcal{F}$ associated with

$$\rho_{\text{sys}}(t) := \text{Tr}_{\text{bath}}\left(e^{-iL_{NVT}t}|\rho_0\rangle\langle\rho_0|e^{iL_{NVT}}\right). \tag{47}$$

Note that the above problem description allows the free energy to be a time-dependent quantity. If the dynamics of the extended system (nuclei + heat bath) are ergodic, then we can obtain an estimate of the equilibrium free energy via coherent time averaging (for more details, see Sec. IV A).

The next theorem provides upper bounds on the complexity of estimating the free energy associated with the nuclear phase-space density.

Theorem 2 (Estimation of the free energy). Let $\eta := g_x^{3N}g_{p'}^{3N}g_s g_{p_s}$ be the number of grid points of the discretized phase space and assume that $\log\left(\eta^2/\epsilon\right) \leq \eta$. Then there exists a quantum algorithm that solves Problem 2 with success probability at least $1-\xi$ using

$$\widetilde{O}\left(\left(\frac{\eta^{o(1)}N_{\text{tot}}d\mu_{NVT}^{2+o(1)}t^{1+o(1)}}{\widetilde{\gamma}\,\widetilde{\delta}\,\epsilon^{1+o(1)}}\left(\alpha_{\text{nuc}} + \frac{\eta(k_bT)^{1.5+o(1)}}{\sqrt{\epsilon}}\right) + \frac{N_{\text{tot}}\alpha_{\text{nuc}}\lambda}{\epsilon^2}\right)\log\left(\frac{1}{\xi}\right)\right)$$

Toffoli gates. Additionally,

$$\widetilde{O}\left(\frac{\eta^{o(1)}Nd\mu_{NVT}^{1+o(1)}t^{1+o(1)}}{\widetilde{\delta}\,\epsilon^{1+o(1)}}\log\left(\frac{1}{\xi}\right)\left(\alpha_{\text{nuc}} + \frac{\eta(k_bT)^{1.5+o(1)}}{\sqrt{\epsilon}}\right)\right)$$

queries to the initial electronic state-preparation oracle $\widetilde{U}_I$ are needed.

Theorem 2 is proved in Appendix D. Although the scaling with η may seem challenging at first, as it implies, in the worst case, exponential scaling with the number of nuclei, this is actually a reasonable expectation, because estimation of the free energy is an NP-hard problem [40]. However, for many practical problems, the phase space can be coarse grained, which reduces the complexity considerably and makes the problem more manageable.

IV. OVERVIEW OF THE ALGORITHM

As mentioned before, the (discretized) Liouvillian L is Hermitian, meaning that the time-evolution operator e^{-iLt} of the (discretized) phase-space density is unitary. Hence, we can use Hamiltonian simulation algorithms to implement e^{-iLt} on a quantum computer [34,35,41,42]. The main idea is to split the overall Liouvillian $L = L_{\text{class}} + L_{\text{el}}$ into a classical electron-independent part and an electronic part as defined below.

Definition 5 (Electronic Liouvillian). Let $D^{\text{el}}_{n,j}$ be the finite-difference approximation to $\partial E_{\text{el}}/\partial x_{n,j}$ as in Definition 2. In the *NVE* ensemble, the electronic Liouvillian acting on the nuclei is given by

$$L^{(NVE)}_{\text{el}} := i\sum_{n=1}^{N}\sum_{j=1}^{3} D^{\text{el}}_{n,j}\otimes D^{1}_{p_{n,j}}, \tag{48}$$

where $D^1_{p_{n,j}}$ is a second-order discrete derivative approximation to $\partial_{p_{n,j}}$. In the *NVT* ensemble, the electronic Liouvillian acting on the nuclei is given by

$$L^{(NVT)}_{\text{el}} := i\sum_{n=1}^{N}\sum_{j=1}^{3} D^{\text{el}}_{n,j}\otimes D^{1}_{p'_{n,j}}, \tag{49}$$

where $D^1_{p'_{n,j}}$ is a second-order discrete derivative approximation to $\partial_{p'_{n,j}}$.

The reason for restricting $D_{p_{n,j}}$ and $D_{p'_{n,j}}$ to be second-order discrete derivatives in the above definition is related to their implementation, as explained in more detail in Appendix B.

Definition 6 (Classical Liouvillian). Let $L \in \{L_{NVE}, L_{NVT}\}$ be the discretized Liouvillian in the *NVE* ensemble (Definition 2) or the *NVT* ensemble (Definition 3). Let $L_{\text{el}} \in \{L^{(NVE)}_{\text{el}}, L^{(NVT)}_{\text{el}}\}$ be the electronic Liouvillian from Definition 5. The classical Liouvillian is then given by

$$L_{\text{class}} := L - L_{\text{el}}. \tag{50}$$

We simulate $U_{L_{\text{class}}} := e^{-iL_{\text{class}}t}$ and $U_{L_{\text{el}}} := e^{-iL_{\text{el}}t}$ separately and then recombine them using a $2k$th-order Trotter-Suzuki product formula [34,41,42].

Definition 7 ($2k$th-order Trotter-Suzuki product formula). Let $L = \sum_{\gamma=1}^{\Gamma} L_\gamma$ be an operator consisting of Γ Hermitian summands and $t \geq 0$. Then, the following recursion defines $\mathcal{S}_{2k}(t)$, the Trotter-Suzuki product formula of order $2k$:

$$\mathcal{S}_2(t) := e^{L_1\frac{t}{2}}\cdots e^{L_\Gamma\frac{t}{2}}e^{L_\Gamma\frac{t}{2}}\cdots e^{L_1\frac{t}{2}} \tag{51}$$

$$\mathcal{S}_{2k}(t) := \mathcal{S}^2_{2k-2}(u_k t)\mathcal{S}_{2k-2}((1-4u_k)t)\mathcal{S}^2_{2k-2}(u_k t), \tag{52}$$

where

$$\frac{1}{3} \leq u_k := \frac{1}{\left(4 - 4^{\frac{1}{2k-1}}\right)} \leq \frac{1}{2} \quad \forall\, k \in \mathbb{N}, k \geq 2. \tag{53}$$

Using results from Ref. [34], we show in Appendix C that the cost of our algorithm depends on the spectral norm of the nested commutator of L_{class} and L_{el}. We derive upper bounds μ'_{NVE} and μ'_{NVT} on the nested commutator in the *NVE* and the *NVT* ensemble that provide better scaling with respect to the number of nuclei N than μ_{NVE} [Eq. (22)] and μ_{NVT} [Eq. (38)]. However, we use the looser bounds μ_{NVE} and μ_{NVT} in the main theorems to keep the statements as simple as possible.

Our algorithm requires several different quantum registers. In the *NVE* ensemble, we use two types of registers. First, we have a nuclear register for encoding the nuclear phase-space density. The basis states of this register are given in Eq. (14). The simulation of $U_{L_{\text{el}}}$ requires a second register, which we call the electronic register, as it is used to encode the electronic wave function. This register can be treated like an ancilla register in the sense that it is only used to compute $D^{\text{el}}_{n,j}$ during the simulation of $U_{L_{\text{el}}}$. At the end of the algorithm, the electronic register is uncomputed. In the *NVT* ensemble, we have a third register for the bath variables s and p_s. The computational basis states of the nuclear register together with the bath register are given in Eq. (32).

Figure 2 summarizes our quantum algorithm for *NVE* and *NVT* Liouvillian simulations. The corresponding pseudocode is presented in Algorithm 1. The subroutines for evolving the phase-space density under the classical Liouvillian L_{class} and the electronic Liouvillian L_{el} are summarized in Algorithms 2 and 3, respectively.

Let us now explain these subroutines in more detail. To implement $U_{L_{\text{class}}}$, we first construct a block encoding of L_{class} according to the following definition.

Definition 8 (Block encoding ([35], Definition 24)). Let A be an s-qubit operator, $\alpha \in \mathbb{R}$ a normalization constant, $\epsilon \in \mathbb{R}$ the allowable error, and $a \in \mathbb{N}$ the number of ancilla qubits. Then, we define that the $(s+a)$-qubit unitary U is an (α, a, ϵ) block encoding of A if

$$\left\| A - \alpha \left(\langle 0|^{\otimes a} \otimes \mathbb{1}\right) U \left(|0\rangle^{\otimes a} \otimes \mathbb{1}\right) \right\| \leq \epsilon. \tag{54}$$

In other words, we embed L_{class} inside a larger unitary matrix. If $\left(\langle 0|^{\otimes a} \otimes \mathbb{1}\right) U \left(|0\rangle^{\otimes a} \otimes \mathbb{1}\right)$ is Hermitian, then we call U a Hermitian block encoding.

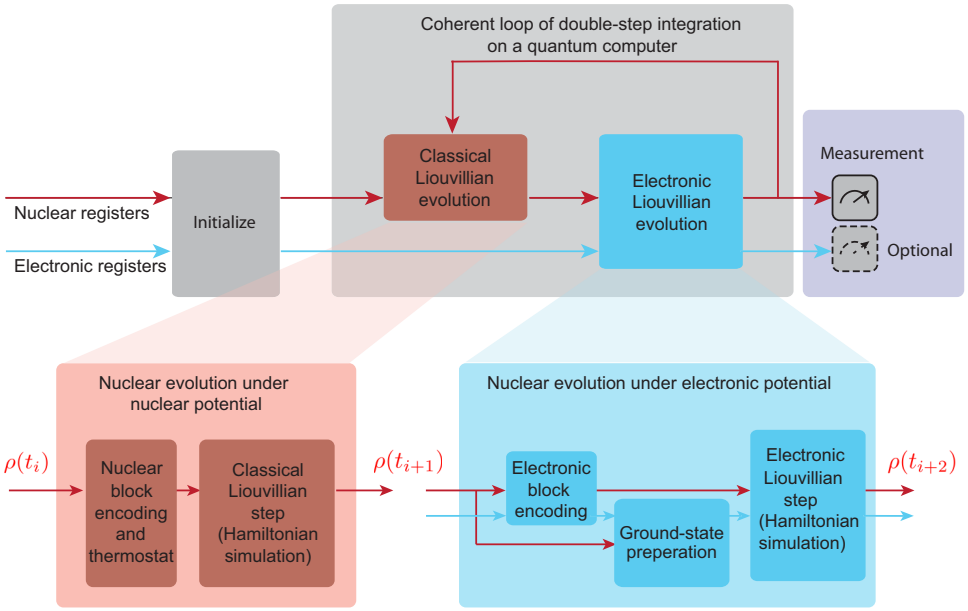

FIG. 2. The Liouvillian algorithm scheme. The red arrows represent the nuclear registers, while the electronic registers are in blue. After initializing the registers into the chosen initial conditions, the time evolution is carried out by iterating a two-step integration. The double-step integration originates from alternating the classical nuclear Liouvillian evolution $(t_i + 1)$ with the electronic one $(t_i + 2)$. In the classical nuclear Liouvillian evolution, the classical nuclear block encoding is used to implement the evolution under the classical NVE or NVT Liouvillians. The electronic Liouvillian-evolution step takes care of the implementation of the electronic block encodings, preparing the electronic ground state given the updated nuclear coordinates and applying the nuclear evolution due to the electronic wave-function contribution. After the required number of integration steps is achieved, one can decide whether to measure the output states, use them for other computations, or, for the nuclear register, use it for the estimation of the free energy.

ALGORITHM 1. Liouvillian MD simulation.

Input: Quantum state $|\rho_0\rangle$ encoding initial phase space density of nuclei (+ heat bath for NVT ensemble).
Input parameters for constructing the Liouvillian operator:
$t, \epsilon, \xi, k, N, \tilde{N}, \{m_n\}_{n=1}^N, \{Z_n\}_{n=1}^N, x_{\max}, p_{\max}/p'_{\max}, h_x, h_p/h_{p'}, d_x, d_p/d_{p'}, d_e, \Delta, B, h_{\mathrm{el}}, \widetilde{\delta}, \widetilde{\gamma}, \chi$.
Simulations in the NVT ensemble require additional input parameters: $N_f, T, Q, s_{\min}, h_s, h_{p_s}, d_s, d_{p_s}$.
Output: An ϵ-precise approximation to $e^{-iLt}|\rho_0\rangle$ with success probability $\geq 1-\xi$.

1. Compute a set of time steps $\{t_i\}_{i=1}^{N_{\mathrm{exp}}}$ using the recursive definition of higher-order Trotter product formulas with $N_{\mathrm{exp}} = 2\times 5^{k-1}+1$ being the total number of Trotter exponentials [34, 42]. Additionally, determine the set of indices I_{class} for which t_i corresponds to evolutions under the classical Liouvillian;
2. Initialize the nuclear position and momentum registers (and the bath register for NVT simulations), the electronic register, as well as an ancilla register;
3. $\epsilon' \leftarrow \epsilon/N_{\mathrm{exp}}$;
4. $\xi' \leftarrow \xi/N_{\mathrm{exp}}$;

for $1 \leq i \leq N_{exp}$ **do**
  if $i \in I_{class}$ **then**
    Apply `ClassLiouvillianEv` $\left(t_i, \epsilon', \xi', N, \{m_n\}_{n=1}^N, \{Z_n\}_{n=1}^N, x_{\max}, p_{\max}/p'_{\max}, h_x, h_p/h_{p'}, d_x, d_p/d_{p'}, \Delta\right)$ to the nuclear position and momentum registers (and the bath register in the case of NVT simulations), as well an ancilla register. Simulations in the NVT ensemble also take $N_f, T, Q, s_{\min}, h_s, h_{p_s}, d_s, d_{p_s}$ as input;
  else
    Apply
    `ElectronicLiouvillianEv` $\left(t_i, \epsilon', \xi', N, \tilde{N}, \{Z_n\}_{n=1}^N, x_{\max}, p_{\max}/p'_{\max}, h_x, h_p/h_{p'}, d_e, \Delta, B, h_{\mathrm{el}}, \widetilde{\delta}, \widetilde{\gamma}, \chi\right)$ to the nuclear position and momentum registers, the electronic register and an ancilla register;
  end
end

The following two lemmas show that we can block encode L_{class} efficiently. Both lemmas are proved in Appendix A.

Lemma 1 (Block encoding of the discretized classical NVE Liouvillian). There exists an $(\alpha_{NVE}, a_{NVE}, \epsilon)$ block encoding of the discretized classical Liouvillian $L_{\mathrm{class}}^{(NVE)}$ with normalization constant

$$\alpha_{NVE} \in O\left(N\frac{p_{\max}}{m_{\min}}\frac{\ln d_x}{h_x} + N^2\frac{Z_{\max}^2 x_{\max}}{\Delta^3}\frac{\ln d_p}{h_p}\right)$$

and a number of ancilla qubits

$$a_{NVE} \in O\left(\log\left(\frac{\alpha_{NVE}}{\epsilon}\right) + \log d\right),$$

where $d := \max\{d_x, d_p\}$. This block encoding can be implemented using

$$\widetilde{O}\left(N\log\left(\frac{g\alpha_{NVE}}{\epsilon}\right) + \log^{\log 3}\left(\frac{\alpha_{NVE}}{\epsilon}\right) + d\log g\right)$$

Toffoli gates, where $g := \max\{g_x, g_p\}$.

Lemma 2 (Block encoding of the discretized classical NVT Liouvillian). There exists an $(\alpha_{NVT}, a_{NVT}, \epsilon)$ block encoding of the discretized classical Liouvillian $L_{\mathrm{class}}^{(NVT)}$ with normalization constant

$$\alpha_{NVT} \in O\left(N\frac{p'_{\max}}{m_{\min}s_{\min}^2}\frac{\ln d_x}{h_x} + N^2\frac{Z_{\max}^2 x_{\max}}{\Delta^3}\frac{\ln d_{p'}}{h_{p'}} + \frac{p_{s,\max}}{Q}\frac{\ln d_s}{h_s} + \left(N\frac{p'^2_{\max}}{m_{\min}s_{\min}^3} + \frac{N_f k_B T}{s_{\min}}\right)\frac{\ln d_{p_s}}{h_{p_s}}\right)$$

and a number of ancilla qubits

$$a_{NVT} \in O\left(\log\left(\frac{\alpha_{NVT}}{\epsilon}\right) + \log d\right),$$

where $d := \max\{d_x, d_{p'}, d_s, d_{p_s}\}$. This block encoding can be implemented using

$$\widetilde{O}\left(N\log\left(\frac{g\alpha_{NVT}}{\epsilon}\right) + \log^{\log 3}\left(\frac{\alpha_{NVT}}{\epsilon}\right) + d\log g\right)$$

Toffoli gates, where $g := \max\{g_x, g_{p'}, g_s, g_{p_s}\}$.

We then apply QSVT to the block encoding of L_{class} to efficiently approximate the exponential $e^{-iL_{\mathrm{class}}t}$ [35]. The idea behind QSVT is to perform polynomial transformations of the singular values of a block-encoded matrix. In our case, we implement polynomial approximations of $\cos(L_{\mathrm{class}})$ and $-i\sin(L_{\mathrm{class}})$ that can then be added to simulate $e^{-iL_{\mathrm{class}}t}$.

ALGORITHM 2. `ClassLiouvillianEv`: Evolution under the classical Liouvillian.

Input: $t, \epsilon, \xi, N, \{m_n\}_{n=1}^N, \{Z_n\}_{n=1}^N, x_{\max}, p_{\max}/p'_{\max}, h_x, h_p/h_{p'}, d_x, d_p/d_{p'}, \Delta$.
Simulations in the NVT ensemble require additional input parameters: $N_f, T, Q, s_{\min}, h_s, h_{p_s}, d_s, d_{p_s}$.
Output: An ϵ-precise approximation to $e^{-iL_{\text{class}}t}$ with success probability $\geq 1-\xi$.

1. Construct an $\epsilon/(2t)$-precise block-encoding of L_{class} as shown in Appendix A;
2. Apply the QSVT based Hamiltonian simulation algorithm from [35] to the block-encoding of L_{class} to obtain an ϵ-precise block-encoding of $e^{-iL_{\text{class}}t}$;
3. Use $O\left(\log\left(\frac{1}{\xi}\right)\right)$ rounds of fixed-point amplitude amplification to boost the success probability to at least $1-\xi$;

The implementation of $U_{L_{\text{el}}}$ is more difficult because we do not generally have an analytic expression for $E_{\text{el}}(\{x_n\})$, which would be required for constructing a block encoding of L_{el} and subsequently using QSVT. In principle, one could use quantum phase estimation on the electronic Hamiltonian H_{el} to extract the ground-state energies at the different nuclear positions and construct a block encoding of D^{el}_{nj} from these (numerical) values. However, the associated computational cost is in $O(1/\epsilon)$, where ϵ is the desired precision of the simulation. The computational cost of our algorithm, on the other hand, is only in $O\left(1/\epsilon^{o(1)}\right)$, which provides a superpolynomial improvement over $O(1/\epsilon)$ scaling.

One important feature of the electronic Liouvillian L_{el} is that all summands commute with each other. The evolution operator associated with L_{el} can thus be decomposed as

ALGORITHM 3. `ElectronicLiouvillianEv`: Evolution under the electronic Liouvillian.

Input: $t, \epsilon, \xi, N, \tilde{N}, \{Z_n\}_{n=1}^N, x_{\max}, p_{\max}/p'_{\max}, h_x, h_p/h_{p'}, d_e, \Delta, B, h_{\text{el}}, \delta, \gamma, \chi$.
Output: An ϵ-precise approximation to $e^{-iL_{\text{el}}t}$ with success probability $\geq 1-\xi$.
Initialize the electronic register in the $|0\rangle$ state;
for $1 \leq n \leq N$ **do**
  for $1 \leq j \leq 3$ **do**
    Fourier transform the quantum register of the j-th momentum coordinate of the n-th nucleus:

$$\text{QFT}|\overline{p}_{n,j}\rangle = \frac{1}{\sqrt{g_p}} \sum_{l_{n,j}} e^{2\pi i \overline{p}_{n,j} l_{n,j}/g_p} |l_{n,j}\rangle$$

    $|\overline{x}_{n,j}\rangle \leftarrow |\overline{x}_{n,j} - d_e\rangle$;
    for $-d_e \leq k \leq d_e$ **do**

$$t_{c_{d_e,k}, l_{n,j}} \leftarrow \frac{c_{d_e,k}}{h_x} \frac{\sin(2\pi l_{n,j}/g_p)}{h_p} t$$

      Apply the state preparation oracle U_I from Definition 15 to the entire nuclear positions register $|\{x_n\}\rangle$ and the electronic register:

$$U_I|\{x_n\}\rangle|0\rangle = |\{x_n\}\rangle|\phi_0\{x_n\}\rangle$$

      Apply the ground state preparation algorithm from [43] to $|\{x_n\}\rangle|\phi_0\{x_n\}\rangle$ using a block-encoding of $H_{\text{el}}(\{x_n\})$;
      Apply $\exp\left(-iH_{\text{el}}(\{x_n\})\, t_{c_{d_e,k}, l_{n,j}}\right)$ to the electronic register;
      Uncompute the (approximate) electronic ground state;
      $|\overline{x}_{n,j}\rangle \leftarrow |\overline{x}_{n,j} + 1\rangle$;
    end
    $|\overline{x}_{n,j}\rangle \leftarrow |\overline{x}_{n,j} - d_e\rangle$;
    Apply QFT^{-1} to the Fourier transformed momentum register to switch back to the $|\overline{p_{n,j}}\rangle$ basis;
  end
end
Use $O\left(\log\left(\frac{1}{\xi}\right)\right)$ rounds of fixed-point amplitude amplification to boost the success probability to at least $1-\xi$;

follows:

$$e^{-iL_{\mathrm{el}}t} = e^{-i\left(i\sum_{n=1}^{N}\sum_{j=1}^{3} D^{\mathrm{el}}_{n,j}\otimes D^{1}_{p_{n,j}}\right)t} = \prod_{n,j} e^{D^{\mathrm{el}}_{n,j}\otimes D^{1}_{p_{n,j}}\,t}. \tag{55}$$

For simplicity, we use $p_{n,j}$ to refer to either a real or a virtual-momentum variable. Let us now explain how to implement a single exponential appearing in Eq. (55). First, we diagonalize the discrete (virtual) momentum derivative operator $D^{1}_{p_{n,j}}$ by applying the quantum Fourier transform (QFT) to the $|p_{n,j}\rangle$ register. Next, we shift the quantum register associated with the nuclear-position coordinate $x_{n,j}$ according to the finite-difference scheme of $D^{\mathrm{el}}_{n,j}$ and prepare the electronic ground state controlled by the entire nuclear-positions register. The electronic ground states are prepared using techniques from Ref. [43] together with the following initial-state-preparation oracle.

Definition 9 (Initial electronic state-preparation oracle). Let $U_{H_{\mathrm{el}}\{x_n\}}$ be a Hermitian block encoding of $H_{\mathrm{el}}(\{x_n\})$ for fixed nuclear positions and let $|\widetilde{\psi}_0\{x_n\}\rangle$ be the ground state of $(\langle 0|\otimes\mathbb{1})\,U_{H_{\mathrm{el}}\{x_n\}}\,(|0\rangle\otimes\mathbb{1})$. Furthermore, let $0<\delta\leq 1$. The electronic state-preparation oracle U_I is defined via its action on the nuclear-positions register $|\{\bar{x}_n\}\rangle$:

$$U_I|\{\bar{x}_n\}\rangle|0\rangle = |\{\bar{x}_n\}\rangle|\phi_0\{x_n\}\rangle, \tag{56}$$

where $|\phi_0\{x_n\}\rangle$ is an initial electronic state that is promised to satisfy $\left|\langle\psi_0\{x_n\}|\phi_0\{x_n\}\rangle\right|\geq\delta$ for all nuclear configurations. We write $\widetilde{U}_I$ to refer to a variant of the initial electronic state-preparation oracle where $\left|\langle\widetilde{\psi}_0\{x_n\}|\phi_0\{x_n\}\rangle\right|\geq\widetilde{\delta}$ with $\widetilde{\delta}\geq\delta$ for all nuclear configurations visited during the simulation. This means that $\widetilde{U}_I$ depends implicitly on the initial nuclear phase-space density.

While numerous strategies exist for addressing overlap problems in quantum algorithms for the electronic structure problem (see, e.g., Refs. [44–46]), the overlap issues remain a fundamental problem facing all quantum algorithms within the space and remain an active area of research. Providing an explicit implementation of the initial electronic state-preparation oracle is hence beyond the scope of this work.

Controlled by the entire nuclear-positions register as well as the Fourier transformed momentum register, we then apply $\exp\left(-iH_{\mathrm{el}}\{x_n\}t_{c_k,l_{n,j}}\right)$ to the electronic register holding the corresponding electronic ground state, where

$$t_{c_k,l_{n,j}} := \frac{c_{d_e,k}}{h_x}\frac{\sin\left(2\pi l_{n,j}/g_p\right)}{h_p}t \tag{57}$$

is a rescaled time variable depending on the finite-difference coefficients $\{c_{d_e,k}\}$ of $D^{\mathrm{el}}_{n,j}$ and the Fourier transform variables $\{l_{n,j}\}$, as explained in more detail in Appendix B. Next, we uncompute the Fourier-transformed momentum register as well as the electronic register and repeat the above procedure for each stencil point of the finite-difference formula of $D^{\mathrm{el}}_{n,j}$.

The above method requires access to a discretized electronic Hamiltonian. Instead of utilizing a grid discretization as for the nuclei, we use a finite set of basis functions to discretize the Hilbert space of the electrons. In particular, we choose B plane waves as basis functions, which take the following form in (three-dimensional) position space:

$$\phi_b(r) := \frac{1}{\sqrt{\Omega}}e^{-ik_b\cdot r}. \tag{58}$$

r is a vector in position space and $k_b = 2\pi b/\Omega^{1/3}$ is a wave vector in reciprocal space, where b is a vector in $\mathbb{Z}^3$ constrained to the cube $G := \left[-(B^{1/3}-1)/2, (B^{1/3}-1)/2\right]^3$. Furthermore, $\Omega\in\Theta\left(B\,h_{\mathrm{el}}^3\right)$ is the computational cell volume, where $1/h_{\mathrm{el}}$ is the grid spacing in reciprocal space.

The electronic basis states in first quantization can then be written as $|b_0\rangle|b_1\rangle\cdots|b_{\tilde{N}-1}\rangle$, where each $|b_j\rangle$ is a qubit register of size $\lceil\log B\rceil$ specifying the index $b\in[B]$ of the basis function occupied by electron j. The main advantage of using a plane-wave expansion of the electronic Hamiltonian is that all terms in the Hamiltonian can be obtained from the nuclear-position registers and the plane-wave momenta through coherent arithmetic on the quantum computer. It is shown in Ref. [28] that the first-quantized electronic Hamiltonian in the plane-wave basis takes the following form:

$$\begin{aligned} H^{(\mathrm{pw})}_{\mathrm{el}}(\{x_n\}) :=& \sum_{j=1}^{\tilde{N}}\sum_{b\in G}\frac{\|k_b\|^2}{2}|b\rangle\langle b|_j \\ &- \frac{4\pi}{\Omega}\sum_{n=1}^{N}\sum_{j=1}^{\tilde{N}}\sum_{\substack{b,c\in G\\ b\neq c}}\left(Z_n\frac{e^{ik_{c-b}\cdot x_n}}{\|k_{b-c}\|^2}\right)|b\rangle\langle c|_j \\ &+ \frac{2\pi}{\Omega}\sum_{i\neq j=1}^{\tilde{N}}\sum_{b,c\in G}\sum_{\substack{\nu\in G_0\\ (b+\nu)\in G\\ (c-\nu)\in G}}\frac{1}{\|k_\nu\|^2} \\ &\times|b+\nu\rangle\langle b|_i|c-\nu\rangle\langle c|_j, \end{aligned} \tag{59}$$

where $|b\rangle\langle b|_j$ acts nontrivially only on the register associated with electron j and similarly for the other terms. Furthermore, $G_0 := \left[-B^{1/3}, B^{1/3}\right]^3\subset\mathbb{Z}^3\backslash\{(0,0,0\}$. Unless stated otherwise, we will write H_{el} to refer to $H^{(\mathrm{pw})}_{\mathrm{el}}$ in the following.

The next lemma shows that H_{el} can be efficiently block encoded.

Lemma 3 (Block encoding the electronic Hamiltonian ([28], Lemma 1 rephrased)). There exists a Hermitian $(\lambda, a_{\mathrm{el}}, \epsilon)$ block encoding of the discretized electronic Hamiltonian H_{el} with normalization constant

$$\lambda \in O\left(\frac{\tilde{N}}{h_{\mathrm{el}}^2} + \frac{N\tilde{N}Z_{\max}}{h_{\mathrm{el}}} + \frac{\tilde{N}^2}{h_{\mathrm{el}}}\right) \tag{60}$$

and a number of ancilla qubits

$$a_{\mathrm{el}} \in O\left(\log\left(\frac{N\tilde{N}B}{\epsilon}\right)\right). \tag{61}$$

This block encoding can be implemented using

$$O\left(N + \tilde{N} + \log\left(\frac{B}{\epsilon}\right)\right) \tag{62}$$

Toffoli gates.

Let us briefly discuss the space complexity of the Liouvillian simulation algorithm. On the one hand, we need $O\left(N \log\left(g_x g_p\right)\right)$ qubits to represent the phase-space density of the nuclei. We need another $O\left(\tilde{N}\log(B)\right)$ qubits to represent the wave function of the electrons. On the other hand, we require a certain number of ancilla qubits for the various block encodings described above. Since Hamiltonian simulation via QSVT requires only an additional two ancilla qubits (see Lemma 4), we find that the overall space complexity is in

$$O\left(N\log(g) + \tilde{N}\log(B) + \log\left(\frac{\alpha}{\epsilon}\right) + \log(d)\right), \tag{63}$$

where $g = \max\{g_x, g_{p'}, g_s, g_{p_s}\}$, $\alpha \in \{\alpha_{NVE}, \alpha_{NVT}\}$, and $d = \max\{d_x, d_{p'}, d_s, d_{p_s}\}$. This scaling is very moderate given that it is linear in the total particle number and logarithmic in all other simulation parameters.

The complexity of our algorithm depends on several user-supplied parameters, which are summarized in Table I.

A. Estimation of the free energy

Let us now discuss how to estimate the free energy of the nuclei after time t (see Definition 4) using our Liouvillian simulation algorithm. At this stage, we do not assume that the nuclei are in thermal equilibrium. First, we apply $U_{L_{NVT}}$ to the initial discretized phase-space density of the nuclei and the heat bath to evolve them for time t. The main idea is to estimate the Gibbs entropy and the internal energy associated with $|\rho_t\rangle = e^{-iL_{NVT}t}|\rho_0\rangle$ separately and then add the results classically to estimate the free energy. This means that we require at least two separate simulations. In Appendix D, we show how to reduce the problem of estimating the Gibbs entropy of the nuclei to that of estimating the von Neumann entropy of a density

TABLE I. The input parameters that determine the complexity of our quantum algorithm for simulating *NVE* and *NVT* Liouvillian dynamics in the Born-Oppenheimer approximation.

Description	NVE	NVT
Evolution time	t	t
Desired precision	ϵ	ϵ
Failure probability	ξ	ξ
Order of the Trotter product formula	k	k
Number of nuclei and electrons	$N, \tilde{N}$	$N, \tilde{N}$
Mass of the lightest nucleus	$m_{\min}$	$m_{\min}$
Maximum atomic number over all nuclei	$Z_{\max}$	$Z_{\max}$
Maximum value of a component of the nuclear-position vectors	$x_{\max}$	$x_{\max}$
Maximum value of a component of the (virtual) momentum vectors	$p_{\max}$	$p'_{\max}$
Grid spacing for a component of the discretized variables	h_x, h_p	$h_x, h_{p'}, h_s, h_{p_s}$
Order of the finite-difference scheme used for approximating derivatives	d_x, d_p, d_e	$d_x, d_{p'}, d_s, d_{p_s}, d_e$
Gap parameter to regularize the Coulomb potential	Δ	Δ
Number of plane-wave basis functions in the electronic Hamiltonian	B	B
Inverse grid spacing for a component of the electronic wave number	h_{el}	h_{el}
Lower bound on the overlap of the initial and true electronic ground state	$\widetilde{\delta}$	$\widetilde{\delta}$
Lower bound on the spectral gap of H_{el} during the simulation	$\widetilde{\gamma}$	$\widetilde{\gamma}$
Upper bound on the higher-order derivatives of the electronic energy	χ	χ
Number of phase-space grid points	...	η
Number of degrees of freedom of the physical system	...	N_f
Temperature of the heat bath	...	T
Mass parameter associated with the heat bath	...	Q
Minimum value of the bath position variable	...	$s_{\min}$
Maximum value of the bath momentum variable	...	$p_{s,\max}$

matrix ρ'_{sys} obtained from $|\rho_t\rangle$ by tracing out the bath register and removing the off-diagonal elements. This allows us to employ Theorem 13 of Ref. [47]. The authors' algorithm requires access to a purification of the density matrix of the system, which in our case is essentially just $|\rho_t\rangle$. From that purification, we first construct a block encoding of ρ'_{sys} and then use QSVT to transform the singular values ρ_i of ρ'_{sys} via a polynomial approximation to $\ln(1/\rho_i)$. The resulting operator is then applied to the purification of ρ'_{sys}. Lastly, we use amplitude estimation to obtain an estimate of

$$\mathcal{S}_G = -k_b \operatorname{Tr}\left(\rho'_{\text{sys}} \ln \rho'_{\text{sys}}\right). \tag{64}$$

Next, let us discuss how to estimate the internal energy associated with $|\rho_t\rangle$. First, note that a classical system can be described by a density matrix ρ and a Hamiltonian H, both of which are diagonal in the computational basis. The internal energy of a classical system can thus be computed as follows:

$$\mathcal{U} = \operatorname{Tr}(\rho H). \tag{65}$$

In our case, we have that $\rho \equiv \rho'_{\text{sys}}$ and $H \equiv H_{\text{nuc}} = H_{\text{kin}} + H_{\text{pot}} + H_{E_{\text{el}}}$. In Appendix D, we show how to efficiently block encode each of the three terms as given in Eqs. (43)–(45). The idea is then to use the Hadamard test to estimate $\operatorname{Tr}\left(\rho'_{\text{sys}} H_{\text{kin}}\right)$, $\operatorname{Tr}\left(\rho'_{\text{sys}} H_{\text{pot}}\right)$, and $\operatorname{Tr}\left(\rho'_{\text{sys}} H_{E_{\text{el}}}\right)$ individually and combine the results classically.

Let us now assume that the dynamics of the extended system (nuclei plus heat bath) are ergodic, i.e., that the extended system samples all phase-space points associated with energy E_{ext}. This allows us to estimate the equilibrium free energy via coherent time averaging. More specifically, we first prepare the following time-averaged density matrix:

$$\overline{\rho} := \frac{1}{t}\int_0^t e^{-iL_{NVT}\tau}|\rho_0\rangle\langle\rho_0|e^{iL_{NVT}\tau}d\tau, \tag{66}$$

where $|\rho_0\rangle$ is an initial phase-space density of the extended system that has support only on configurations with energy E_{ext}. Operationally speaking, the above density matrix can be prepared by sampling $t' \in [0, t]$ uniformly at random and applying $e^{-iL_{NVT}t'}$ to $|\rho_0\rangle$. If t is sufficiently large, then expectation values of observables estimated with $\overline{\rho}$ are approximately equal to expectation values computed with ρ_{NVT} [48].

Note that having access to an initial phase-space density describing a microcanonical ensemble in the extended system, i.e., $\rho_0 \propto \delta\left(H_{\text{nuc}}^{(NVT)} - E_{\text{ext}}\right)$, allows us to directly prepare the corresponding Boltzmann distribution over the nuclear variables by tracing out the bath variables (see Eq. (26)). This implies that we could estimate the free energy as an ensemble average without having to perform any time evolution.

V. CONCLUSIONS

Our main achievement is a new approach for efficiently simulating coupled quantum-classical dynamics on fault-tolerant quantum computers that provides a superpolynomial improvement in the precision scaling over previous work. The upper bounds on the computational costs of our algorithm for the evolution of a classical phase-space density scale polynomially with the 1-norm of the Liouvillian and with the simulation time t. This is in stark contrast to earlier gradient-based approaches [20], which, as we show in Appendix E, can scale under worst-case assumptions exponentially with the evolution time. The presented Liouvillian simulation algorithm illustrates the value of incorporating classical dynamics into quantum simulations coherently on fault-tolerant quantum computers and paves the way for simulating coupled quantum-classical systems. We apply the approach to the simulation of molecular systems in both the microcanonical and canonical ensembles and to the estimation of thermodynamic quantities, such as the free energy.

To make our algorithms applicable to practical problems, challenges and limitations remain. For example, preparing the classical system in the canonical ensemble requires it to thermalize. In classical simulations, it is possible to have clear indicators of thermalization [49–51], while it is unclear how to estimate those indicators within our approach without sampling. Also, the computational cost of the free-energy estimation scales exponentially with the number of particles because of the growth of the phase space [40]. Another challenge is the preparation of a quantum state encoding the initial phase-space density of the classical particles, since preparing an arbitrary quantum state can take time that scales exponentially with the number of qubits in the worst case. Additionally, artifacts affecting classical simulation [52] will likely influence our simulation methodology.

Compared to state-of-the-art classical MD, where many refinements have been developed over the years, our approach is still rudimentary. Several solutions could be explored for solving the above issues and adapted to complement our approach; e.g., exploiting adaptive time steps to improve the computational costs or using multiple coupled thermostats to allow correct thermalization [53].

Future work must optimize our resource scalings, which are based on loose bounds. In terms of quantum circuit design, numerous improvements are possible. One example is the extensive use of products of block encodings to simulate the parts of the Liouvillian. These costs could likely be brought down by designing a specific block encoding in a single step. Similarly, it is an open question whether combining the classical and electronic Liouvillians in a higher-order Trotter formula is the most efficient choice. Exploiting the fractional query model [54] or multiproduct formulas may lead to polylogarithmic scaling

with the error tolerance rather than the subpolynomial scalings obtained using high-order Trotter formulas. Another important task is the identification of a first potential application, along with assessing its run time and qubit count. This will aid in pinpointing bottlenecks within the proposed algorithm, as well as enable a comparison with alternative approaches. It would also be interesting to move beyond the Born-Oppenheimer approximation by adapting our algorithm to include excited electronic states.

Looking forward, a larger question emerges about the role that quantum computers may play in the simulation of classical or quantum-classical dynamics. While our research strengthens our understanding of the advantages that quantum may provide for simulating such hybrid dynamics, it is still necessary to fully explore the nature of the limitations and opportunities that quantum computers face or provide when simulating both types of dynamical systems [5,25,26,55]. Our belief is that further study of such applications will unveil a host of new use cases for quantum computers that lie outside of purely quantum simulations and, in turn, will reveal that quantum computation is much more broadly applicable to simulation than previously thought.

ACKNOWLEDGMENTS

S.S. acknowledges support from a Research Award from Google Inc. and Natural Sciences and Engineering Research Council of Canada (NSERC) Discovery Grants, as well as support from Boehringer Ingelheim. N.W.'s work on this project was supported by the "Embedding Quantum Computing into Many-Body Frameworks for Strongly Correlated Molecular and Materials Systems" project, which is funded by the U.S. Department of Energy (DOE), Office of Science, Office of Basic Energy Sciences, the Division of Chemical Sciences, Geosciences, and Biosciences. We thank Benjamin Ries, Aniket Magarkar, Thomas Fox, and Rodrigo Ochoa for their insights and stimulating discussions on MD applications. We thank Christofer Tautermann and Clemens Utschig-Utschig for useful discussions and feedback. Additionally, we thank Ryan Babbush and Tom O'Brien for their comments on the manuscript.

APPENDIX A: EVOLUTION UNDER THE CLASSICAL LIOUVILLIAN

We simulate $e^{-iL_{\text{class}}t}$ using qubitization and/or QSVT [35,56], which requires us to prepare a block encoding of L_{class} according to Definition 8. These block encodings are implemented using the linear-combination-of-unitaries (LCU) framework [57]. First, for an arbitrary matrix A, we decompose it into an LCU, $A = \sum_{j=0}^{2^a-1} \alpha_j U_j$, where $\alpha_j \geq 0\ \forall j$. This linear combination can then be implemented using the following two unitary operations, defined via their action on an ancilla register initialized to $|0\rangle^{\otimes a}$ and some quantum state $|\psi\rangle$:

$$\text{PREP}|0\rangle^{\otimes a}|\psi\rangle := \sum_j \sqrt{\frac{\alpha_j}{\alpha}}|j\rangle|\psi\rangle, \tag{A1}$$

$$\text{SEL}|j\rangle|\psi\rangle := |j\rangle U_j|\psi\rangle, \tag{A2}$$

where $\alpha := \sum_j \alpha_j$ is a normalization constant. This allows us to implement A probabilistically in the sense that

$$\frac{A}{\alpha} = \left(\langle 0|^a \otimes \mathbb{1}\right) \text{PREP}^\dagger \cdot \text{SEL} \cdot \text{PREP} \left(|0\rangle^a \otimes \mathbb{1}\right). \tag{A3}$$

Once we have such a block encoding of the classical Liouvillian L_{class}, we can use QSVT to construct a polynomial approximation of $e^{-iL_{\text{class}}t}$ [35,56]. The corresponding query complexity of block-Hamiltonian—or in our case, block-Liouvillian—simulation based on qubitization and/or QSVT is stated below.

Lemma 4 (Robust block-Hamiltonian simulation [35, 58]). Let $t \in \mathbb{R}_{\geq 0}$, $\epsilon \in (0,1)$ and let U be an $(\alpha, a, \epsilon/2t)$ block encoding of the Hamiltonian H. Then, we can implement an ϵ-precise Hamiltonian simulation unitary V that is an $(1, a+2, \epsilon)$ block encoding of e^{iHt}, with probability of success at least $1-\xi$, with

$$O\left(\log\left(\frac{1}{\xi}\right)\left(\alpha t + \log\left(\frac{1}{\epsilon}\right)\right)\right) \tag{A4}$$

uses of U or its inverse and three uses of controlled-U or its inverse, using $O\left(\log\left(1/\xi\right)\left(a\alpha t + a\log\left(1/\epsilon\right)\right)\right)$ two-qubit gates and using $O(1)$ ancilla qubits.

A proof of Lemma 4 with constant success probability is given in Ref. [35]. Using (fixed-point) amplitude amplification, we can boost the success probability to $1-\xi$ at the expense of a multiplicative factor of $\log\left(1/\xi\right)$ [58].

The following lemma bounds the error propagation of the PREP subroutine, which will be useful for our discussion of the overall block-encoding error of the classical Liouvillian.

Lemma 5 (Error propagation of PREP*).* Let $A = \sum_{j=0}^{2^a-1} \alpha_j U_j$ be an LCU with $\alpha_j \geq 0\ \forall j$. Let $|\text{PREP}\rangle := \sum_j \sqrt{\alpha_j/\alpha}|j\rangle$ with $\alpha := \sum_j \alpha_j$ be the quantum state prepared by the PREP subroutine as defined in Eq. (A1). Let $|\widetilde{\text{PREP}}\rangle := \sum_j c_j|j\rangle$ be an $\epsilon/(2\alpha\sqrt{2^a})$-precise approximation to $|\text{PREP}\rangle$ prepared by $\widetilde{\text{PREP}}$ such that $\||\widetilde{\text{PREP}}\rangle - |\text{PREP}\rangle\| \leq \epsilon/(2\alpha\sqrt{2^a})$. Given access to the unitary $\text{SEL} := \sum_j |j\rangle\langle j| \otimes U_j$, we can implement an ϵ-precise block encoding of A.

Proof. First, recall that for any $v \in \mathbb{C}^{2^a}$, it holds that $\|v\|_1 = \sum_{j=0}^{2^a-1} |v_j| \leq \sqrt{2^a}\,\|v\|_2 \equiv \sqrt{2^a}\,\|v\|$. Using this inequality and the triangle inequality, one finds that

$$\begin{aligned}
&\left\| \alpha \left(\langle 0|^a \otimes \mathbb{1}\right) \widetilde{\text{PREP}}^\dagger \cdot \text{SEL} \cdot \widetilde{\text{PREP}} \left(|0\rangle^a \otimes \mathbb{1}\right) - A \right\| \\
&= \left\| \alpha \left(\langle 0|^a \otimes \mathbb{1}\right) \widetilde{\text{PREP}}^\dagger \cdot \text{SEL} \cdot \widetilde{\text{PREP}} \left(|0\rangle^a \otimes \mathbb{1}\right) - \alpha \left(\langle 0|^a \otimes \mathbb{1}\right) \text{PREP}^\dagger \cdot \text{SEL} \cdot \text{PREP} \left(|0\rangle^a \otimes \mathbb{1}\right) \right\| \\
&= \alpha \left\| \sum_j c_j c_j^* U_j - \sum_j \frac{\alpha_j}{\alpha} U_j \right\| \leq \alpha \sum_j \left| c_j c_j^* - c_j \sqrt{\frac{\alpha_j}{\alpha}} + c_j \sqrt{\frac{\alpha_j}{\alpha}} - \frac{\alpha_j}{\alpha} \right| \|U_j\| \\
&\leq \alpha \sum_j |c_j| \left| c_j^* - \sqrt{\frac{\alpha_j}{\alpha}} \right| + \alpha \sum_j \left| \sqrt{\frac{\alpha_j}{\alpha}} \right| \left| c_j - \sqrt{\frac{\alpha_j}{\alpha}} \right| \leq 2\alpha \sum_j \left| c_j - \sqrt{\frac{\alpha_j}{\alpha}} \right| \\
&\leq 2\alpha\sqrt{2^a} \sqrt{\sum_j \left| c_j - \sqrt{\frac{\alpha_j}{\alpha}} \right|^2} = 2\alpha\sqrt{2^a}\, \| |\widetilde{\text{PREP}}\rangle - |\text{PREP}\rangle \| \leq \epsilon.
\end{aligned} \tag{A5}$$

■

Lemmas 1 and 2 provide upper bounds on the complexity of block encoding the classical Liouvillian in the *NVE* and *NVT* ensemble, respectively. We prove these lemmas by explicitly constructing a block encoding of the relevant L_{class}. The general idea is to block encode each term of L_{class} separately and then combine all the smaller block encodings to obtain a block encoding of the overall classical Liouvillian. To do so, we need to know how to multiply two block encodings.

Lemma 6 (Product of block-encoded matrices ([35], Lemma 30)). If U is an (α, a, δ) block encoding of an s-qubit operator A and V is a (β, b, ϵ) block encoding of an s-qubit operator B, then $(I_a \otimes U)(I_b \otimes V)$ is an $(\alpha\beta, a+b, \alpha\epsilon + \beta\delta)$ block encoding of AB.

A linear combination of block encodings can be constructed using the concept of a state-preparation pair.

Definition 10 (State-preparation pair ([35], Definition 28)). Let $y \in \mathbb{C}^m$ and $\|y\|_1 \leq \beta$. The pair of unitaries (P_L, P_R) is called a (β, b, ϵ)-state-preparation pair if $P_L|0\rangle^{\otimes b} = \sum_{j=0}^{2^b-1} c_j |j\rangle$ and $P_R|0\rangle^{\otimes b} = \sum_{j=0}^{2^b-1} d_j |j\rangle$ such that $\sum_{j=0}^{m-1} \left| \beta c_j^* d_j - y_j \right| \leq \epsilon$ and for all $j \in \{m, \ldots, 2^b - 1\}$ we have $c_j^* d_j = 0$.

Note that b in the above definition is chosen such that $2^b \geq m$. This is necessary to accommodate all m entries of y. In general, m does not need to be a power of 2. The condition $c_j^* d_j = 0$ for all $j \in \{m, \ldots, 2^b - 1\}$ ensures that we are limited to an m-dimensional subspace of the 2^b-dimensional space of the b register.

For our purposes, it will always be true that $P_L = P_R$ in which case we call P_L a state-preparation unitary.

Lemma 7 (Linear combination of block-encoded matrices (improved version of Lemma 29, which has appeared in Ref. [35])). Let $A = \sum_{j=0}^{m-1} y_j A_j$ be an s-qubit operator and $\epsilon \in \mathbb{R}_{>0}$. Suppose that (P_L, P_R) is a (β, b, ϵ_1)-state-preparation pair for y, $W = \sum_{j=0}^{m-1} |j\rangle\langle j| \otimes U_j + ((I - \sum_{j=0}^{m-1} |j\rangle\langle j|) \otimes I_a \otimes I_s$ is an $(s+a+b)$-qubit unitary such that $\forall j \in \{0, 1, \ldots, m-1\}$ we have that U_j is an (α, a, ϵ_2) block encoding of A_j. Then, we can implement an $(\alpha\beta, a+b, \alpha\epsilon_1 + \beta\epsilon_2)$ block encoding of A, with a single use of W, P_R, and $P_L^\dagger$.

Proof. We adapt the proof from Ref. [35] by showing that $\widetilde{W} := \left(P_L^\dagger \otimes I_a \otimes I_s\right) W (P_R \otimes I_a \otimes I_s)$ is an $(\alpha\beta, a+b, \alpha\epsilon_1 + \beta\epsilon_2)$ block encoding of A:

$$\begin{aligned}
&\left\| A - \alpha\beta \left(\langle 0|^{\otimes b} \otimes \langle 0|^{\otimes a} \otimes I\right) \widetilde{W} \left(|0\rangle^{\otimes b} \otimes |0\rangle^{\otimes a} \otimes I\right) \right\| \\
&= \left\| A - \alpha \sum_{j=0}^{m-1} \beta \left(c_j^* d_j\right) \left(\langle 0|^{\otimes a} \otimes I\right) U_j \left(|0\rangle^{\otimes a} \otimes I\right) \right\| \\
&\leq \alpha\epsilon_1 + \left\| A - \alpha \sum_{j=0}^{m-1} y_j \left(\langle 0|^{\otimes a} \otimes I\right) U_j \left(|0\rangle^{\otimes a} \otimes I\right) \right\| \\
&\leq \alpha\epsilon_1 + \sum_{j=0}^{m-1} y_j \left\| A_j - \alpha \left(\langle 0|^{\otimes a} \otimes I\right) U_j \left(|0\rangle^{\otimes a} \otimes I\right) \right\| \\
&\leq \alpha\epsilon_1 + \sum_{j=0}^{m-1} y_j \epsilon_2 \\
&\leq \alpha\epsilon_1 + \beta\epsilon_2.
\end{aligned}$$

■

Additionally, we require a bound on the coefficients of higher-order central-finite-difference approximations arising from the discretized derivative operators D_x and D_p

for the *NVE* ensemble and D_x, $D_{p'}$, D_s, and D_{p_s} for the *NVT* ensemble. The following result provides a higher-order bound on centered-difference formulas that can be used for our purposes.

Lemma 8 (Higher-order central-finite-difference approximation [33]). Let d be an integer and let $\{x_k\}$ with $k \in \{-d, -d+1, \ldots, d-1, d\}$ be a set of equally spaced points on the interval $[a, b]$, i.e.,

$$x_k = x_0 + h, \tag{A6}$$

for some $h > 0$. Furthermore, let $f \in C^{2d+1}[a, b]$ be a function on the interval $[a, b]$ that is $2d+1$ times continuously differentiable. Then, one can use a linear combination of $f(x_k)$ to construct a central-finite-difference formula of order $2d$ to approximate $f^{(1)}(x_0) = f'(x_0)$, i.e.,

$$f'(x_0) = \frac{1}{h}\sum_{k=-d}^{d} c_{d,k} f(x_k) + R_d(x_0), \tag{A7}$$

where

$$c_{d,k} := \begin{cases} \dfrac{(-1)^{k+1}(d!)^2}{k(d-k)!(d+k)!}, & \text{if } k \neq 0, \\ 0, & \text{else.} \end{cases} \tag{A8}$$

The remainder term can be bounded as follows:

$$R_d(x_0) \in O\left(\max_{x\in[a,b]} \left|f^{(2d+1)}(x)\right| \left(\frac{eh}{2}\right)^{2d}\right). \tag{A9}$$

In order to provide the cost of block encoding the result, we need to identify the sum of the coefficients for the finite-difference formula. Such a bound is given in the following lemma.

Lemma 9. The coefficients $c_{d,k}$ of the central-finite-difference formula as defined in Lemma 8 satisfy

$$\sum_{k=-d}^{d} |c_{d,k}| \leq 2(\ln d + 1). \tag{A10}$$

Proof. First, note that for $k \neq 0$, we have

$$\frac{(d!)^2}{(d-k)!(d+k)!} = \frac{d \times (d-1) \times \cdots \times (d-k+1)}{(d+k) \times (d+k-1) \times \cdots \times (d+1)} < 1. \tag{A11}$$

Thus,

$$|c_{d,k}| = \left|\frac{(-1)^{k+1}}{k}\frac{(d!)^2}{(d-k)!(d+k)!}\right| \leq \left|\frac{1}{k}\right|. \tag{A12}$$

This implies that

$$\sum_{k=-d}^{d} |c_{d,k}| \leq \sum_{\substack{k=-d \\ i\neq 0}}^{d} \left|\frac{1}{k}\right| = 2\sum_{k=1}^{d}\frac{1}{k} \leq 2(\ln d + 1). \tag{A13}$$

■

Finally, let us show how to construct a general inequality-testing circuit, which will be used repeatedly as a subroutine for block encoding L_{class}.

Lemma 10 (Inequality testing). Let a and b be arbitrary bit strings of length n. Using $n+2$ additional qubits and $5n-2$ Toffoli gates, one can construct a quantum circuit that outputs "0" if and only if $a \leq b$ and "1" otherwise.

Proof. Consider the circuit shown in Fig. 3, with bit strings a and b as inputs. The general strategy is to perform bit-wise comparisons starting with the most significant bits and to store the result in an additional qubit $|r\rangle$. To avoid overwriting the result of the previous bit comparison, we use an additional n qubits, $|c_0\rangle, \ldots, |c_{n-1}\rangle$, as a clock register. Furthermore, we need one ancilla qubit to implement triply-controlled-NOT (triple-CNOT) gates from Toffoli gates. Note that this ancilla qubit is not shown in the circuit diagram. We need one additional qubit to store the result of the inequality test. The state of that qubit is denoted $|r\rangle$ in Fig. 3. The circuit first compares the most significant bits, a_0 and b_0, using a CNOT gate. The second (triple-CNOT) gate fires only if, initially, $a_0 = 1$ and $b_0 = 0$, i.e., if $a_0 > b_0$. In this case, the last qubit, $|r\rangle$, which stores the result of the inequality test, gets flipped to $|1\rangle$. None of the remaining triple-CNOT gates fire, since the indicator state of the clock register, $|c_0\rangle = |1\rangle$, does not get transferred to the next clock qubit $|c_1\rangle$. The same is true for the case $a_0 < b_0$, i.e., if $a_0 = 0$ and $b_0 = 1$. However, in this case, not even the first triple-CNOT gate fires. The indicator state of the clock register gets swapped to $|c_1\rangle$ if and only if $a_0 = b_0$. This is repeated until $a_j \neq b_j$ for some $j \in [n]$. In the worst case, $j = n$. At the end of the inequality test, we uncompute intermediate results by applying all gates except for the triple-CNOT gates in reverse. Using the fact that a single triple-CNOT gate can be implemented with three Toffoli gates, we then find that the overall Toffoli count is equal to $5n-2$. ■

Now, we are ready to prove Lemmas 1 and 2.

1. Proof of Lemma 1

For convenience, let us restate Lemma 1 here.

Lemma 1 (Block encoding of the discretized classical NVE Liouvillian). There exists an $(\alpha_{NVE}, a_{NVE}, \epsilon)$ block encoding of the discretized classical Liouvillian $L_{\text{class}}^{(NVE)}$ with normalization constant

$$\alpha_{NVE} \in O\left(N\frac{p_{\max}}{m_{\min}}\frac{\ln d_x}{h_x} + N^2\frac{Z_{\max}^2 x_{\max}}{\Delta^3}\frac{\ln d_p}{h_p}\right)$$

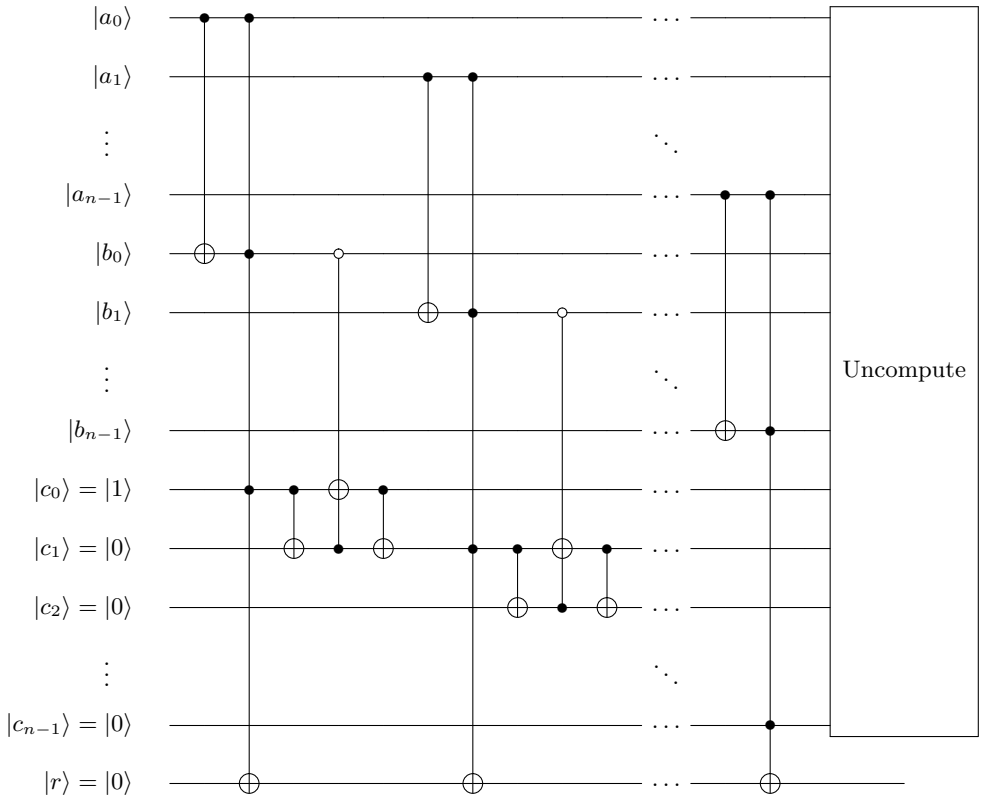

FIG. 3. The inequality-testing circuit $U_{\leq}$. The basic idea behind this circuit is to perform bit-by-bit comparison and use an ancillary qubit, $|r\rangle$, in which the outcome of the comparison is stored. The additional n-qubit register $|c\rangle$ is used to avoid overwriting the results of the other bit-comparison results. The "Uncompute" part consists of all previous gates applied in reverse except for the triple-CNOT gates, which are not uncomputed. If $a \leq b$, then the final output is $|r\rangle = |0\rangle$. Otherwise, $|r\rangle = |1\rangle$.

and a number of ancilla qubits

$$a_{NVE} \in O\left(\log\left(\frac{\alpha_{NVE}}{\epsilon}\right) + \log d\right),$$

where $d := \max\{d_x, d_p\}$. This block encoding can be implemented using

$$\widetilde{O}\left(N \log\left(\frac{g\alpha_{NVE}}{\epsilon}\right) + \log^{\log 3}\left(\frac{\alpha_{NVE}}{\epsilon}\right) + d \log g\right)$$

Toffoli gates, where $g := \max\{g_x, g_p\}$.

Proof. We block encode $L^{(NVE)}_{\text{class}}$ via several layers of "smaller" block encodings, which can be thought of as a block-encoding hierarchy. More precisely, we apply different PREP and SEL operations in a nested fashion as summarized in Fig. 4. Let us first give an overview of all the levels of the hierarchy before discussing the gate and space complexity.

At the lowest level, we have four types of block encodings:

(1) $U_{p_{n,j}}$, an $\left(\alpha_p, a_p, \epsilon_p\right)$ block encoding of $\sum_{\overline{p}_{n,j}=0}^{g_p-1} p_{n,j} |\overline{p}_{n,j}\rangle\langle\overline{p}_{n,j}|$

(2) $U_{D_{x_{n,j}}}$, an $\left(\alpha_{D_x}, a_{D_x}, \epsilon_{D_x}\right)$ block encoding of $D_{x_{n,j}}$

(3) $U_{V_{n,n',j}}$, an $(\alpha_V, a_V, \epsilon_V)$ block encoding of

$$\sum_{\overline{x}_{n,j}=0}^{g_x-1} [(x_{n,j} - x_{n',j})/\left(\|x_n - x_{n'}\|^2 + \Delta^2\right)^{3/2}] |\overline{x}_{n,j}\rangle\langle\overline{x}_{n,j}|$$

(4) $U_{D_{p_{n,j}}}$, an $\left(\alpha_{D_p}, a_{D_p}, \epsilon_{D_p}\right)$ block encoding of $D_{p_{n,j}}$

Using Lemma 6, we combine $U_{p_{n,j}}$ and $U_{D_{x_{n,j}}}$ as well as $U_{V_{n,n',j}}$ and $U_{D_{p_{n,j}}}$ to construct

(a) $U_{(pD_x)_{n,j}}$, an $\left(\alpha_p\alpha_{D_x}, a_p + a_{D_x}, \alpha_p\epsilon_{D_x} + \alpha_{D_x}\epsilon_p\right)$ block encoding of $\sum_{\overline{p}_{n,j}=0}^{g_p-1} p_{n,j} |\overline{p}_{n,j}\rangle\langle\overline{p}_{n,j}| \otimes D_{x_{n,j}}$

(b) $U_{(VD_p)_{n,n',j}}$, an $\left(\alpha_V\alpha_{D_p}, a_V + a_{D_p}, \alpha_V\epsilon_{D_p} + \alpha_{D_p}\epsilon_V\right)$ block encoding of

$$\sum_{\overline{x}_{n,j}=0}^{g_x-1} \frac{(x_{n,j} - x_{n',j})}{\left(\|x_n - x_{n'}\|^2 + \Delta^2\right)^{3/2}} |\overline{x}_{n,j}\rangle\langle\overline{x}_{n,j}| \otimes D_{p_{n,j}}$$

The next level of the hierarchy involves two different state-preparation unitaries:

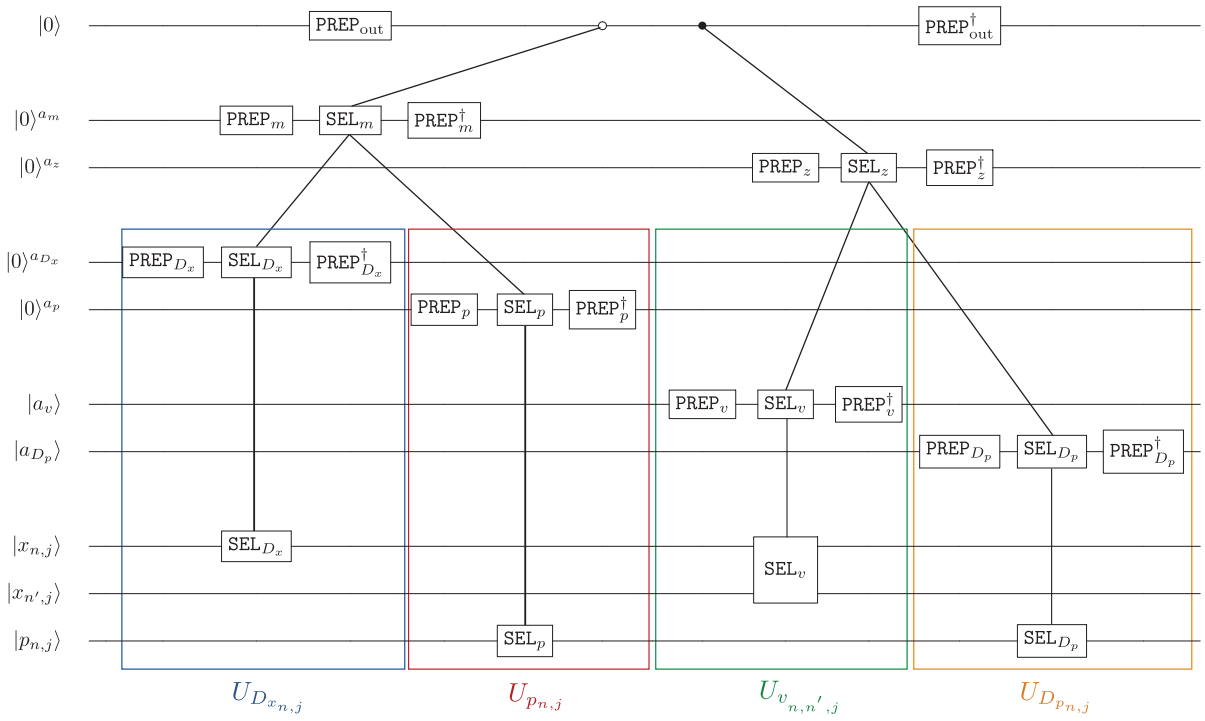

FIG. 4. The circuit diagram of the block-encoding hierarchy used for implementing a block encoding of $L^{(NVE)}_{\text{class}}$.

(a) PREP_m, an $(\alpha_m, a_m, \epsilon_m)$ state-preparation unitary that encodes the nuclear masses
(b) PREP_Z, an $(\alpha_Z, a_Z, \epsilon_Z)$ state-preparation unitary that encodes the atomic numbers of the nuclei

Application of Lemma 7 to PREP_m and $\left\{U_{(pD_x)_{n,j}}\right\}$ as well as to PREP_Z and $\left\{U_{(VD_p)_{n,n',j}}\right\}$ yields

(a) $U_{L^{(NVE)}_{\text{kin}}}$, an $\left(\alpha^{(NVE)}_{\text{kin}}, a^{(NVE)}_{\text{kin}}, \epsilon^{(NVE)}_{\text{kin}}\right)$ block encoding of $-i\sum_{n=1}^{N}\sum_{j=1}^{3}\sum_{\bar{p}_{n,j}=0}^{g_p-1}(p_{n,j}/m_n)|\bar{p}_{n,j}\rangle\langle\bar{p}_{n,j}|\otimes D_{x_{n,j}}$, where

$$\alpha^{(NVE)}_{\text{kin}} = \alpha_m\alpha_p\alpha_{D_x}, \tag{A14}$$
$$a^{(NVE)}_{\text{kin}} = a_m + a_p + a_{D_x}, \tag{A15}$$
$$\epsilon^{(NVE)}_{\text{kin}} = \alpha_m(\alpha_p\epsilon_{D_x} + \alpha_{D_x}\epsilon_p) + \alpha_p\alpha_{D_x}\epsilon_m \tag{A16}$$

(b) $U_{L^{(NVE)}_{\text{pot}}}$, an $\left(\alpha^{(NVE)}_{\text{pot}}, a^{(NVE)}_{\text{pot}}, \epsilon^{(NVE)}_{\text{pot}}\right)$ block encoding of

$$i\sum_{n=1}^{N}\sum_{n'>n}\sum_{j=1}^{3}\sum_{\bar{x}_{n,j}=0}^{g_x-1}[Z_nZ_{n'}(x_{n,j}-x_{n',j})/ \times\left(\|x_n - x_{n'}\|^2 + \Delta^2\right)^{3/2}]|\bar{x}_{n,j}\rangle\langle\bar{x}_{n,j}|\otimes D_{p_{n,j}},$$

where

$$\alpha^{(NVE)}_{\text{pot}} = \alpha_Z\alpha_V\alpha_{D_p}, \tag{A17}$$
$$a^{(NVE)}_{\text{pot}} = a_Z + a_V + a_{D_p}, \tag{A18}$$
$$\epsilon^{(NVE)}_{\text{pot}} = \alpha_Z(\alpha_V\epsilon_{D_p} + \alpha_{D_p}\epsilon_V) + \alpha_V\alpha_{D_p}\epsilon_Z \tag{A19}$$

Note that 7 requires all block encodings of the linear combination to have the same block-encoding normalization constant, which is true in both cases discussed above. More specifically, all $\left\{U_{(pD_x)_{n,j}}\right\}$ terms have the same normalization constant. Similarly, all $\left\{U_{(VD_p)_{n,n',j}}\right\}$ terms have the same normalization constant.

Lastly, we use Lemma 7 once more to combine $U_{L^{(NVE)}_{\text{kin}}}$ and $U_{L^{(NVE)}_{\text{pot}}}$. Since they have different normalization constants, we first need to renormalize both block encodings. The idea is as follows: if U is an (α, a, ϵ) block encoding of some matrix A, then U is also an $(\alpha\beta, a, \beta\epsilon)$ block encoding of the scaled matrix βA. This follows straight from Definition 8. Thus, $U_{L^{(NVE)}_{\text{kin}}}$ can also be viewed as an

$$\left(\alpha^{(NVE)}_{\text{kin}} + \alpha^{(NVE)}_{\text{pot}}, a^{(NVE)}_{\text{kin}} + a^{(NVE)}_{\text{pot}}, \frac{\alpha^{(NVE)}_{\text{kin}} + \alpha^{(NVE)}_{\text{pot}}}{\alpha^{(NVE)}_{\text{kin}}}\epsilon^{(NVE)}_{\text{kin}} + \frac{\alpha^{(NVE)}_{\text{kin}} + \alpha^{(NVE)}_{\text{pot}}}{\alpha^{(NVE)}_{\text{pot}}}\epsilon^{(NVE)}_{\text{pot}}\right) \tag{A20}$$

block encoding of

$$-i\frac{\alpha_{\rm kin}^{(NVE)}+\alpha_{\rm pot}^{(NVE)}}{\alpha_{\rm kin}^{(NVE)}}\sum_{n=1}^{N}\sum_{j=1}^{3}\sum_{\overline{p}_{n,j}=0}^{g_p-1}\frac{p_{n,j}}{m_n}|\overline{p}_{n,j}\rangle\langle\overline{p}_{n,j}|\otimes D_{x_{n,j}} \tag{A21}$$

and $U_{L_{\rm pot}^{(NVE)}}$ can be viewed as an

$$\left(\alpha_{\rm kin}^{(NVE)}+\alpha_{\rm pot}^{(NVE)}, a_{\rm kin}^{(NVE)}+a_{\rm pot}^{(NVE)}, \frac{\alpha_{\rm kin}^{(NVE)}+\alpha_{\rm pot}^{(NVE)}}{\alpha_{\rm kin}^{(NVE)}}\epsilon_{\rm kin}^{(NVE)} + \frac{\alpha_{\rm kin}^{(NVE)}+\alpha_{\rm pot}^{(NVE)}}{\alpha_{\rm pot}^{(NVE)}}\epsilon_{\rm pot}^{(NVE)}\right) \tag{A22}$$

block encoding of

$$i\frac{\alpha_{\rm kin}^{(NVE)}+\alpha_{\rm pot}^{(NVE)}}{\alpha_{\rm kin}^{(NVE)}}\sum_{n=1}^{N}\sum_{n'>n}\sum_{j=1}^{3}\sum_{\bar{x}_{n,j}=0}^{g_x-1}\frac{Z_nZ_{n'}(x_{n,j}-x_{n',j})}{\left(\|x_n-x_{n'}\|^2+\Delta^2\right)^{3/2}} \times|\bar{x}_{n,j}\rangle\langle\bar{x}_{n,j}|\otimes D_{p_{n,j}}. \tag{A23}$$

The following state-preparation unitary is used to recover the appropriate weighting of the two block encodings:

(a) $\text{PREP}_{\rm out}$, an $(\alpha_{\rm out}, a_{\rm out}, \epsilon_{\rm out})$ state-preparation unitary, which prepares the state $\sqrt{\alpha_{\rm kin}^{(NVE)}/(\alpha_{\rm kin}^{(NVE)}+\alpha_{\rm pot}^{(NVE)})}\,|0\rangle+\sqrt{\alpha_{\rm pot}^{(NVE)}/(\alpha_{\rm kin}^{(NVE)}+\alpha_{\rm pot}^{(NVE)})}|1\rangle$. Note that $\alpha_{\rm out}=1$ and $a_{\rm out}=1$.

Using $\text{PREP}_{\rm out}$ together with $U_{L_{\rm kin}^{(NVE)}}$ and $U_{L_{\rm pot}^{(NVE)}}$, we can construct $U_{L_{\rm class}^{(NVE)}}$, an $(\alpha_{NVE}, a_{NVE}, \epsilon_{NVE})$ block encoding of $L_{\rm class}^{(NVE)}$, where

$$\alpha_{NVE}=\alpha_{\rm kin}^{(NVE)}+\alpha_{\rm pot}^{(NVE)}, \tag{A24}$$

$$a_{NVE}=a_{\rm kin}^{(NVE)}+a_{\rm pot}^{(NVE)}+1, \tag{A25}$$

$$\epsilon_{NVE}=\frac{\alpha_{\rm kin}^{(NVE)}+\alpha_{\rm pot}^{(NVE)}}{\alpha_{\rm kin}^{(NVE)}}\epsilon_{\rm kin}^{(NVE)}+\frac{\alpha_{\rm kin}^{(NVE)}+\alpha_{\rm pot}^{(NVE)}}{\alpha_{\rm pot}^{(NVE)}}\epsilon_{\rm pot}^{(NVE)} + \left(\alpha_{\rm kin}^{(NVE)}+\alpha_{\rm pot}^{(NVE)}\right)\epsilon_{\rm out}. \tag{A26}$$

We have $\epsilon_{NVE}\le\epsilon$ if

$$\epsilon_{\rm kin}^{(NVE)}\le\frac{\alpha_{\rm kin}^{(NVE)}}{\alpha_{\rm kin}^{(NVE)}+\alpha_{\rm pot}^{(NVE)}}\frac{\epsilon}{3}, \tag{A27}$$

$$\epsilon_{\rm pot}^{(NVE)}\le\frac{\alpha_{\rm pot}^{(NVE)}}{\alpha_{\rm kin}^{(NVE)}+\alpha_{\rm pot}^{(NVE)}}\frac{\epsilon}{3}, \tag{A28}$$

$$\epsilon_{\rm out}\le\frac{1}{\alpha_{\rm kin}^{(NVE)}+\alpha_{\rm pot}^{(NVE)}}\frac{\epsilon}{3}. \tag{A29}$$

It follows from Eqs. (A16) and (A19) that the above error bounds can be achieved by ensuring that

$$\epsilon_p\le\frac{1}{\alpha_m\alpha_{D_x}}\frac{\alpha_{\rm kin}^{(NVE)}}{\alpha_{\rm kin}^{(NVE)}+\alpha_{\rm pot}^{(NVE)}}\frac{\epsilon}{9}, \tag{A30}$$

$$\epsilon_{D_x}\le\frac{1}{\alpha_m\alpha_p}\frac{\alpha_{\rm kin}^{(NVE)}}{\alpha_{\rm kin}^{(NVE)}+\alpha_{\rm pot}^{(NVE)}}\frac{\epsilon}{9}, \tag{A31}$$

$$\epsilon_m\le\frac{1}{\alpha_p\alpha_{D_x}}\frac{\alpha_{\rm kin}^{(NVE)}}{\alpha_{\rm kin}^{(NVE)}+\alpha_{\rm pot}^{(NVE)}}\frac{\epsilon}{9}, \tag{A32}$$

$$\epsilon_V\le\frac{1}{\alpha_Z\alpha_{D_p}}\frac{\alpha_{\rm pot}^{(NVE)}}{\alpha_{\rm kin}^{(NVE)}+\alpha_{\rm pot}^{(NVE)}}\frac{\epsilon}{9}, \tag{A33}$$

$$\epsilon_{D_p}\le\frac{1}{\alpha_Z\alpha_V}\frac{\alpha_{\rm pot}^{(NVE)}}{\alpha_{\rm kin}^{(NVE)}+\alpha_{\rm pot}^{(NVE)}}\frac{\epsilon}{9}, \tag{A34}$$

$$\epsilon_Z\le\frac{1}{\alpha_V\alpha_{D_p}}\frac{\alpha_{\rm pot}^{(NVE)}}{\alpha_{\rm kin}^{(NVE)}+\alpha_{\rm pot}^{(NVE)}}\frac{\epsilon}{9}. \tag{A35}$$

Let us now show how to implement the four basic block encodings, starting with $U_{p_{n,j}}$. Since this is a block encoding of a diagonal matrix, we can use the simplest form of the alternating-sign trick [54]. To explain this trick in more detail, let us consider a single computational basis state associated with the momentum variable $p_{n,j}$. We require two additional ancilla registers to implement this trick. The overall input state can then be written as $|0\rangle^{\otimes a_P}|\overline{p}_{n,j}\rangle|0\rangle$, with all remaining quantum registers being suppressed for ease of notation. Let $U_{\le}$ be the inequality-testing unitary from Lemma 10 (see Fig. 3). The circuit in Fig. 5 then evolves the initial state as follows:

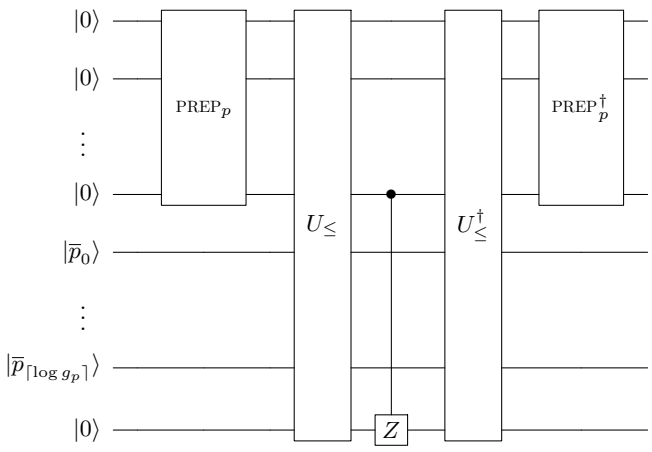

FIG. 5. The circuit for implementing the alternating-sign trick. PREP_p prepares a uniform superposition over all computational basis states $|j\rangle$ of the ancilla register where $j \in [2^{a_p}]$. The inequality test $U_{\leq}$ from Fig. 3 is applied to the ancilla register and the input variable $|p\rangle$ expressed in binary. The result of the inequality test is stored in the bottom qubit ("1" if $j > p$ and "0" otherwise). Next, we apply a Z gate controlled by the least significant qubit of the ancilla register to the output qubit of the inequality test. As long as $j \leq p$, the controlled-Z (CZ) gate acts as the identity gate. However, when $j > p$, the CZ gate introduces a minus sign for every second computational basis state of the ancilla register. This creates an alternating sequence of ± 1 such that the contributions of all $j > p$ cancel each other out. Finally, we uncompute the ancilla qubits.

$$
\begin{aligned}
&|0\rangle|\overline{p}_{n,j}\rangle|0\rangle \\
&\xrightarrow{\text{PREP}_p} \frac{1}{\sqrt{2^{a_p}}}\sum_{j=0}^{2^{a_p}-1}|j\rangle|\overline{p}_{n,j}\rangle|0\rangle \\
&\xrightarrow{U_{\leq}} \frac{1}{\sqrt{2^{a_p}}}\left(\sum_{j\leq\overline{p}_{n,j}}|j\rangle|\overline{p}_{n,j}\rangle|0\rangle+\sum_{j>\overline{p}_{n,j}}|j\rangle|\overline{p}_{n,j}\rangle|1\rangle\right) \\
&\xrightarrow{CZ} \frac{1}{\sqrt{2^{a_p}}}\left(\sum_{j\leq\overline{p}_{n,j}}|j\rangle|\overline{p}_{n,j}\rangle|0\rangle+\sum_{j>\overline{p}_{n,j}}(-1)^j|j\rangle|\overline{p}_{n,j}\rangle|1\rangle\right) \\
&\xrightarrow{U_{\leq}^{\dagger}} \frac{1}{\sqrt{2^{a_p}}}\left(\sum_{j\leq\overline{p}_{n,j}}|j\rangle+\sum_{j>\overline{p}_{n,j}}(-1)^j|j\rangle\right)|\overline{p}_{n,j}\rangle|0\rangle \\
&\xrightarrow{\text{PREP}_p^{\dagger}} \frac{1}{2^{a_p}}\sum_k\left(\sum_{j\leq\overline{p}_{n,j}}(-1)^{k\cdot j}+\sum_{j>\overline{p}_{n,j}}(-1)^j(-1)^{k\cdot j}\right) \\
&\quad\times|k\rangle|\overline{p}_{n,j}\rangle|0\rangle.
\end{aligned}
$$

As usual with block encodings, we postselect on $|k\rangle = |0\rangle$. If $\overline{p}_{n,j}$ is even, this yields the following (unnormalized) state:

$$\frac{\overline{p}_{n,j}}{2^{a_p}}|0\rangle|\overline{p}_{n,j}\rangle|0\rangle, \tag{A36}$$

as desired. If $\overline{p}_{n,j}$ is odd, then we obtain

$$\frac{\overline{p}_{n,j}+1}{2^{a_p}}|0\rangle|\overline{p}_{n,j}\rangle|0\rangle. \tag{A37}$$

The number of PREP ancilla qubits a_p determines the precision, ϵ_p, of $U_{p_{n,j}}$. In particular,

$$\left\| U_{p_{n,j}} - \frac{1}{\alpha_p}\sum_{\overline{p}_{n,j}}\overline{p}_{n,j}|\overline{p}_{n,j}\rangle\langle\overline{p}_{n,j}| \right\| \leq \frac{1}{2^{a_p}}. \tag{A38}$$

Note that a_p should be equal to the number of qubits used to represent a single momentum variable (otherwise, the inequality test does not work). Increasing a_p therefore requires us to (temporarily) blow up the momentum values as well. The above calculation shows that $U_{p_{n,j}} := \text{PREP}_p^{\dagger}\cdot\text{SEL}_p\cdot\text{PREP}_p$, where

$$\text{PREP}_p|0\rangle := \sum_{j=0}^{2^{a_p}-1}\frac{1}{\sqrt{2^{a_p}}}|j\rangle, \tag{A39}$$

$$\text{SEL}_p := U_{\leq}^{\dagger}\cdot\text{CZ}\cdot U_{\leq}, \tag{A40}$$

provides an $(\alpha_p, a_p, \epsilon_p)$ block encoding of $\sum_{\overline{p}_{n,j}=0}^{g_p-1}p_{n,j}|\overline{p}_{n,j}\rangle\langle\overline{p}_{n,j}|$, in which $\alpha_p \in O(p_{\max})$ and $a_p \in O(\log(p_{\max}/\epsilon_p))$. The precision, ϵ_p, is determined by the overall error tolerance ϵ, as shown in Eq. (A30), which implies that

$$a_p \in O\left(\log\left(\alpha_m\alpha_{D_x}\frac{\alpha_{\text{kin}}^{(NVE)}+\alpha_{\text{pot}}^{(NVE)}}{\alpha_{\text{kin}}^{(NVE)}}\frac{p_{\max}}{\epsilon}\right)\right). \tag{A41}$$

Since PREP_p only requires Hadamard gates but no Toffoli gates, the Toffoli complexity of $U_{p_{n,j}}$ is equal to the Toffoli complexity of SEL_p, which is in $O(a_p)$ due to the inequality testing.

The implementation of $U_{V_{n,n',j}}$, an $(\alpha_V, a_V, \epsilon_V)$ block encoding of

$$\sum_{\overline{x}_{n,j}=0}^{g_x-1}\frac{(x_{n,j}-x_{n',j})}{\left(\|x_n-x_{n'}\|^2+\Delta^2\right)^{3/2}}|\overline{x}_{n,j}\rangle\langle\overline{x}_{n,j}|, \tag{A42}$$

is also based on the alternating-sign trick. Using $\text{PREP}_V|0\rangle := \sum_{l=0}^{2^{a_V}-1}(1/\sqrt{2^{a_V}})|l\rangle$, we test the following inequality:

$$l^2\left(\|\overline{x}_n-\overline{x}_{n'}\|^2+\overline{\Delta}^2\right)^3 \leq (\overline{x}_{n,j}-\overline{x}_{n',j})^2, \tag{A43}$$

where $\overline{\Delta}\in\mathbb{N}$ such that $\Delta = \overline{\Delta}h_x$. Note that $\alpha_V \in O\left(x_{\max}/\Delta^3\right)$. The number of ancilla qubits a_V is again

determined by the allowable error. More specifically, $a_V \in O\left(\log\left(x_{\max}/\Delta^3\epsilon_V\right)\right)$. The precision ϵ_V is determined by the overall error tolerance ϵ as shown in Eq. (A33), which implies that

$$a_V \in O\left(\log\left(\alpha_Z\alpha_{D_p}\frac{\alpha_{\text{kin}}^{(NVE)}+\alpha_{\text{pot}}^{(NVE)}}{\alpha_{\text{pot}}^{(NVE)}}\frac{x_{\max}}{\Delta^3\epsilon}\right)\right). \quad \text{(A44)}$$

To determine the correct sign, we also need to test $x_{n,j} \le x_{n',j}$, which has Toffoli complexity in $O(\log(g_x))$. The advantage of testing Eq. (A43) rather than

$$l \le \frac{(\bar{x}_{n,j} - \bar{x}_{n',j})}{\left(\|\bar{x}_n - \bar{x}_{n'}\|^2 + \overline{\Delta}^2\right)^{3/2}} \quad \text{(A45)}$$

directly is that we do not have to calculate fractions containing square roots. However, the inequality test in Eq. (A43) does require us first to compute the left- and right-hand sides of the inequality using $O(1)$ quantum Karatsuba multiplications. This can be done using $O\left((a_V)^{\log 3}\right)$ Toffoli gates, whereas the inequality test itself requires only $O(a_V)$ Toffolis [59].

Next, let us explain how to construct a block encoding $U_{D_{x_{n,j}}}$ of the discrete derivative operator $D_{x_{n,j}}$ of order $2d_x$. The idea is to apply a linear combination of $2d_x$ unitary adders to the $|\bar{x}_{n,j}\rangle$ register as shown in Fig. 6.

Let

$$\text{PREP}_{D_x}|0\rangle := \sum_{k=-d_x}^{d_x}\sqrt{\frac{c_{d_x,k}}{c_{d_x}}}|k\rangle, \quad \text{(A46)}$$

$$\text{SEL}_{D_x} := \sum_{k=-d_x}^{d_x}|k\rangle\langle k| \otimes \sum_{\bar{x}}|\bar{x}-k\rangle\langle\bar{x}|, \quad \text{(A47)}$$

where $\{c_{d_x,k}\}$ are the finite-difference coefficients as given in Lemma 8 and $c_{d_x} := \sum_{k=-d_x}^{d_x}|c_{d_x,k}|$. Then, $U_{D_{x_{n,j}}} := \text{PREP}_{D_x}^{\dagger}\text{SEL}_{D_x}\text{PREP}_{D_x}$ is an $(\alpha_{D_x}, a_{D_x}, \epsilon_{D_x})$ block encoding of $D_{x_{n,j}}$, where

$$\alpha_{D_x} = \frac{c_{d_x}}{h_x} \le \frac{2(\ln d_x + 1)}{h_x} \quad \text{(A48)}$$

(see Lemma 9) and $a_{D_x} \in O(\log(d_x))$. The error ϵ_{D_x} stems solely from the state-preparation error associated with PREP_{D_x}. Lemma 5 implies that we need to prepare the state $\text{PREP}_{D_x}|0\rangle$ within error

$$\frac{\epsilon_{D_x}}{\alpha_{D_x}\sqrt{d_x}} \in O\left(\frac{h_x\epsilon_{D_x}}{\ln(d_x)\sqrt{d_x}}\right). \quad \text{(A49)}$$

Such a general quantum state preparation has Toffoli cost in $O\left(d_x\log\left(\ln(d_x)\sqrt{d_x}/h_x\epsilon_{D_x}\right)\right)$. By Eq. (A31), this is in

$$O\left(d_x\log\left(\alpha_m\alpha_p\frac{\alpha_{\text{kin}}^{(NVE)}+\alpha_{\text{pot}}^{(NVE)}}{\alpha_{\text{kin}}^{(NVE)}}\frac{\ln(d_x)\sqrt{d_x}}{h_x}\frac{1}{\epsilon}\right)\right). \quad \text{(A50)}$$

A single unitary adder requires $O(\log(g_x))$ Toffolis where g_x is again the number of grid points for a single position coordinate. Additionally, we need $O(\log d_x)$ Toffolis to implement a controlled version of the adder controlled by the PREP_d register.

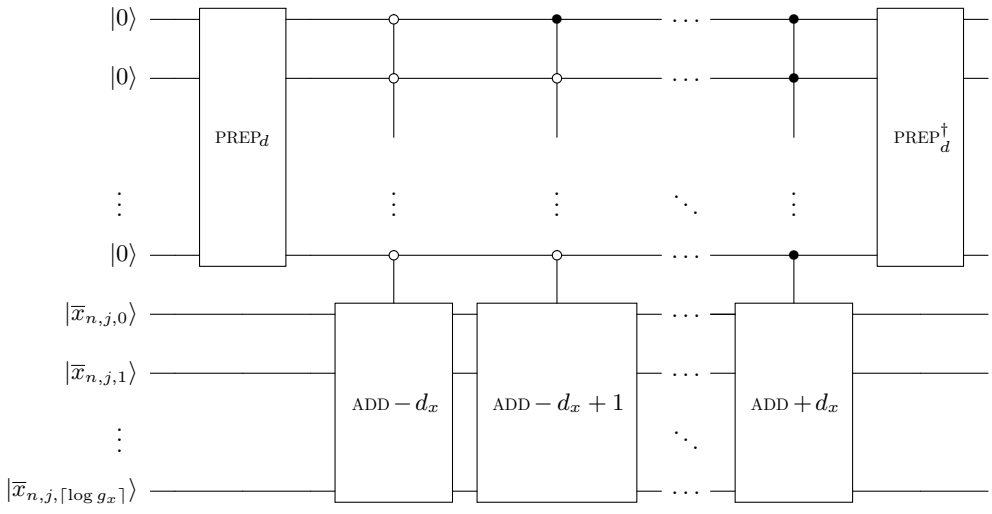

FIG. 6. The circuit for implementing a central-finite-difference operator of order $2d_x$. ADD $+j$, with j being an integer, is a unitary adder of the form $\sum_{\bar{x}}|\bar{x}-j\rangle\langle\bar{x}|$.

In total, we therefore require

$$O\left(d_x\left(\log d_x+\log g_x + \log\left(\alpha_m\alpha_p\frac{\alpha_{\text{kin}}^{(NVE)}+\alpha_{\text{pot}}^{(NVE)}}{\alpha_{\text{kin}}^{(NVE)}}\frac{\ln(d_x)\sqrt{d_x}}{h_x}\frac{1}{\epsilon}\right)\right)\right) \tag{A51}$$

Toffolis to implement $U_{D_{x_{n,j}}}$.

We implement $U_{D_{p_{n,j}}}$, an $\left(\alpha_{D_p}, a_{D_p}, \epsilon_{D_p}\right)$ block encoding of the discrete derivative operator $D_{p_{n,j}}$ of order $2d_p$, using exactly the same strategy as for $U_{D_{x_{n,j}}}$. This implies that

$$\alpha_{D_p} \leq \frac{2\left(\ln d_p+1\right)}{h_p} \tag{A52}$$

and $a_{D_p} \in O\left(\log\left(d_p\right)\right)$. As before, the error ϵ_{D_p} stems solely from the state-preparation error of PREP_{D_p}. The Toffoli cost associated with preparing the state $\text{PREP}_{D_p}|0\rangle$ within sufficiently small error is then in

$$O\left(d_p\log\left(\alpha_Z\alpha_V\frac{\alpha_{\text{kin}}^{(NVE)}+\alpha_{\text{pot}}^{(NVE)}}{\alpha_{\text{kin}}^{(NVE)}}\frac{\ln(d_p)\sqrt{d_p}}{h_p}\frac{1}{\epsilon}\right)\right). \tag{A53}$$

In total, we then require

$$O\left(d_p\left(\log d_p+\log g_p + \log\left(\alpha_Z\alpha_V\frac{\alpha_{\text{kin}}^{(NVE)}+\alpha_{\text{pot}}^{(NVE)}}{\alpha_{\text{pot}}^{(NVE)}}\frac{\ln(d_p)\sqrt{d_p}}{h_p}\frac{1}{\epsilon}\right)\right)\right) \tag{A54}$$

Toffolis to implement $U_{D_{p_{n,j}}}$.

Now that we have shown how to implement the four basic block encodings, $U_{p_{n,j}}$, $U_{D_{x_{n,j}}}$, $U_{V_{n,n',j}}$, and $U_{D_{p_{n,j}}}$, we can combine them. More specifically, we multiply $U_{p_{n,j}}$ and $U_{D_{x_{n,j}}}$ to obtain $U_{(pD_x)_{n,j}}$. This can be done at no extra Toffoli cost by simply keeping the ancilla qubits separate and applying the two block encodings consecutively. The same is true when multiplying $U_{V_{n,n',j}}$ and $U_{D_{p_{n,j}}}$ to obtain $U_{(VD_p)_{n,n',j}}$.

Next, let us explain how to implement PREP_m, an $(\alpha_m, a_m, \epsilon_m)$ state-preparation unitary, which we define as follows:

$$\text{PREP}_m|0\rangle := \sum_{n=1}^{N}\sqrt{\frac{1/m_n}{\alpha_m}}|n\rangle\otimes\frac{1}{\sqrt{3}}\sum_{j=1}^{3}|j\rangle, \tag{A55}$$

where

$$\alpha_m = \sum_{n,j}\frac{1}{m_n} \leq \frac{3N}{m_{\min}}. \tag{A56}$$

The above definition implies that $a_m = \lceil\log N\rceil + \lceil\log 3\rceil$. It follows from Lemma 5 that we need to prepare $\text{PREP}_m|0\rangle$ within error

$$\frac{\epsilon_m}{\alpha_m\sqrt{N}} \in O\left(\frac{m_{\min}\epsilon_m}{N\sqrt{N}}\right). \tag{A57}$$

Such a general quantum state preparation has Toffoli cost in Ref. [60]

$$O\left(N\log\left(\frac{N}{m_{\min}\epsilon_m}\right)\right). \tag{A58}$$

By Eq. (A32), this is in

$$O\left(N\log\left(\alpha_p\alpha_{D_x}\frac{\alpha_{\text{kin}}^{(NVE)}+\alpha_{\text{pot}}^{(NVE)}}{\alpha_{\text{kin}}^{(NVE)}}\frac{N}{m_{\min}}\frac{1}{\epsilon}\right)\right). \tag{A59}$$

We use PREP_m together with $U_{(pD_x)_{n,j}}$ to implement $U_{L_{\text{kin}}^{(NVE)}}$. This can be done efficiently with the help of two additional ancilla registers, which we call "SWAP registers" [28]. Controlled by the PREP_m register, we swap the appropriate position and momentum variables into the two SWAP registers. This allows us to apply the block encodings $U_{p_{n,j}}$ and $U_{D_{x_{n,j}}}$ only once (to the SWAP registers holding the appropriate position and momentum variables) rather than $3N$ times (to each individual position and momentum variable). However, we do require a total of $O\left(N\log(g)\right)$ SWAP operations, where $g=\max\{g_x,g_p\}$, implying $O\left(N\log(g)\right)$ Toffolis.

$U_{L_{\text{pot}}^{(NVE)}}$ can be implemented following the same strategy. We have

$$\text{PREP}_Z|0\rangle := \sum_{n=1}^{N}\sqrt{\frac{Z_n}{Z}}|n\rangle\otimes\sum_{n'=1}^{N}\sqrt{\frac{Z'_n}{Z}}|n'\rangle\otimes\frac{1}{\sqrt{3}}\sum_{j=1}^{3}|j\rangle, \tag{A60}$$

an $(\alpha_Z, a_Z, \epsilon_Z)$ state-preparation unitary with

$$\alpha_Z = \sum_{n,n',j}Z_nZ_{n'} \leq 3N^2Z_{\max}^2. \tag{A61}$$

The above definition implies that $a_Z = 2\lceil\log N\rceil + \lceil\log 3\rceil$. Importantly, it is a product state, meaning that

$\sum_{n=1}^{N}\sqrt{Z_n/Z}|n\rangle$, $\sum_{n'=1}^{N}\sqrt{Z'_n/Z}|n'\rangle$, and $\sum_{j=1}^{3}|j\rangle$ can be prepared individually. It follows from Lemma 5 that we need to prepare $\text{PREP}_Z|0\rangle$ within error

$$\frac{\epsilon_Z}{\alpha_Z N}\in O\left(\frac{\epsilon_Z}{N^3 Z_{\max}^2}\right). \tag{A62}$$

Preparation of such a product state has Toffoli cost in $O\left(N\log\left(NZ_{\max}/\epsilon_Z\right)\right)$. By Eq. (A35), this is in

$$O\left(N\log\left(\alpha_V\alpha_{D_p}\frac{\alpha_{\text{kin}}^{(NVE)}+\alpha_{\text{pot}}^{(NVE)}}{\alpha_{\text{pot}}^{(NVE)}}\frac{NZ_{\max}}{\epsilon}\right)\right). \tag{A63}$$

We use PREP_Z together with $U_{(VD_p)_{n,j}}$ to implement $U_{L_{\text{pot}}^{(NVE)}}$. This can be done efficiently with the help of seven SWAP registers, six for the six nuclear-position variables appearing in $(x_{n,j}-x_{n',j})/\left(\|x_n-x_{n'}\|^2+\Delta^2\right)^{3/2}$ and one for the nuclear-momentum variable of $D_{p_{n,j}}$. As before, this allows us to apply the block encodings $U_{V_{n,n',j}}$ and $U_{D_{p_{n,j}}}$ only once (to the SWAP registers holding the appropriate position and momentum variables) rather than $3N^2$ times and $3N$ times, respectively. However, we again require a total of $O\left(N\log\left(g\right)\right)$ SWAP operations, implying $O\left(N\log\left(g\right)\right)$ Toffolis. To ensure that we exclude terms where the nuclei are the same, i.e., $n=n'$, and also avoid double counting, we perform an inequality test on $|n\rangle$ and $|n'\rangle$ and store the result in an ancilla qubit. The corresponding Toffoli cost is in $O\left(\log N\right)$.

Lastly, we use the $(\alpha_{\text{out}},a_{\text{out}},\epsilon_{\text{out}})$ state-preparation unitary

$$\begin{aligned}\text{PREP}_{\text{out}}|0\rangle :=&\sqrt{\frac{\alpha_{\text{kin}}^{(NVE)}}{\alpha_{\text{kin}}^{(NVE)}+\alpha_{\text{pot}}^{(NVE)}}}|0\rangle\\&+\sqrt{\frac{\alpha_{\text{pot}}^{(NVE)}}{\alpha_{\text{kin}}^{(NVE)}+\alpha_{\text{pot}}^{(NVE)}}}|1\rangle\end{aligned} \tag{A64}$$

together with $U_{L_{\text{kin}}^{(NVE)}}$ and $U_{L_{\text{pot}}^{(NVE)}}$ to construct $U_{L_{\text{class}}^{(NVE)}}$, an $(\alpha_{NVE},a_{NVE},\epsilon_{NVE})$ block encoding of $L_{\text{class}}^{(NVE)}$. As mentioned before, $\alpha_{\text{out}}=1$ and $a_{\text{out}}=1$. It follows from Lemma 5 that we need to prepare $\text{PREP}_{\text{out}}|0\rangle$ within error ϵ_{out}. Such a general quantum state preparation on one qubit has Toffoli cost in $O\left(\log\left(1/\epsilon_{\text{out}}\right)\right)$. By Eq. (A29), this is in

$$O\left(\log\left(\frac{\alpha_{\text{kin}}^{(NVE)}+\alpha_{\text{pot}}^{(NVE)}}{\epsilon}\right)\right). \tag{A65}$$

Combining all of the previous results, we find that

$$\alpha_{NVE}\in O\left(N\frac{p_{\max}}{m_{\min}}\frac{\ln d_x}{h_x}+N^2\frac{Z_{\max}^2x_{\max}}{\Delta^3}\frac{\ln d_p}{h_p}\right). \tag{A66}$$

Furthermore, ensuring $\epsilon_{NVE}\le\epsilon$ requires

$$\begin{aligned}a_{NVE}=&\,a_p+a_V+a_{D_x}+a_{D_p}+a_m+a_Z+1\\\in&\,O\left(\log\left(\frac{\alpha_{NVE}}{\epsilon}\right)+\log\left(d_x\right)+\log\left(d_p\right)+\log\left(N\right)\right)\\\in&\,O\left(\log\left(N\frac{p_{\max}}{m_{\min}}\frac{\ln d_x}{h_x}+N^2\frac{Z_{\max}^2x_{\max}}{\Delta^3}\frac{\ln d_p}{h_p}\right)\right.\\&\left.+\log\left(\frac{d}{\epsilon}\right)\right)\end{aligned} \tag{A67}$$

block-encoding ancilla qubits, where $d:=\max\{d_x,d_p\}$, and

$$\widetilde{O}\left(N\log\left(\frac{g\alpha_{NVE}}{\epsilon}\right)+\log^{\log 3}\left(\frac{\alpha_{NVE}}{\epsilon}\right)+d\log\left(g\right)\right) \tag{A68}$$

Toffoli gates. ■

2. Proof of Lemma 2

For convenience, let us restate Lemma 2 here.

Lemma 2 (Block encoding of the discretized classical NVT Liouvillian). There exists an $(\alpha_{NVT},a_{NVT},\epsilon)$ block encoding of the discretized classical Liouvillian $L_{\text{class}}^{(NVT)}$ with normalization constant

$$\begin{aligned}\alpha_{NVT}\in O\left(\right.&N\frac{p'_{\max}}{m_{\min}s_{\min}^2}\frac{\ln d_x}{h_x}+N^2\frac{Z_{\max}^2x_{\max}}{\Delta^3}\frac{\ln d_{p'}}{h_{p'}}\\&+\frac{p_{s,\max}}{Q}\frac{\ln d_s}{h_s}\\&\left.+\left(N\frac{p'^2_{\max}}{m_{\min}s_{\min}^3}+\frac{N_f k_B T}{s_{\min}}\right)\frac{\ln d_{p_s}}{h_{p_s}}\right)\end{aligned}$$

and a number of ancilla qubits

$$a_{NVT}\in O\left(\log\left(\frac{\alpha_{NVT}}{\epsilon}\right)+\log d\right),$$

where $d:=\max\{d_x,d_{p'},d_s,d_{p_s}\}$. This block encoding can be implemented using

$$\widetilde{O}\left(N\log\left(\frac{g\alpha_{NVT}}{\epsilon}\right)+\log^{\log 3}\left(\frac{\alpha_{NVT}}{\epsilon}\right)+d\log g\right)$$

Toffoli gates, where $g:=\max\{g_x,g_{p'},g_s,g_{p_s}\}$.

Proof. Lemma 2 can be proved analogously to Lemma 1 via a modified block-encoding hierarchy. Here, we only give a brief summary of the construction. At the lowest level, we now have nine types of block encodings that we can express as functions of α_{NVT}. Note that the resulting upper bounds on the individual block encodings are somewhat looser than the corresponding bounds used in the proof of Lemma 1. However, this does not affect the

overall Toffoli or ancilla complexity. With this in mind, we list the lowest-level block encodings below:

(1) $U_{p'_{n,j}}$, an $\left(\alpha_{p'}, a_{p'}, \epsilon_{p'}\right)$ block encoding of $\sum_{\overline{p}_{n,j}=0}^{g_{p'}-1} \sum_{\overline{s}=0}^{g_s-1} (p'_{n,j}/(s+s_{\min})^2)|\overline{p}'_{n,j}\rangle\langle\overline{p}'_{n,j}| \otimes |\overline{s}\rangle\langle\overline{s}|$, where

$$\alpha_{p'} \in O\left(\frac{p'_{\max}}{m_{\min} s_{\min}^2}\right), \tag{A69}$$

$$a_{p'} \in O\left(\log\left(\frac{\alpha_{NVT}}{\epsilon}\right)\right), \tag{A70}$$

$$\epsilon_{p'} \in O\left(\frac{\epsilon}{\alpha_{NVT}}\right). \tag{A71}$$

It can be efficiently implemented using the alternating-sign trick together with quantum Karatsuba multiplication. The resulting Toffoli cost is in $O\left(\left(a_{p'}\right)^{\log 3}\right)$, which is dominated by the cost of implementing quantum Karatsuba multiplication with $a_{p'}$ qubits [59].

(2) $U_{D_{x_{n,j}}}$, an $\left(\alpha_{D_x}, a_{D_x}, \epsilon_{D_x}\right)$ block encoding of $D_{x_{n,j}}$, where

$$\alpha_{D_x} \in O\left(\frac{\ln d_x}{h_x}\right), \tag{A72}$$

$$a_{D_x} \in O\left(\log d_x\right), \tag{A73}$$

$$\epsilon_{D_x} \in O\left(\frac{\epsilon}{\alpha_{NVT}}\right). \tag{A74}$$

It can be efficiently implemented via a linear combination of unitary adders. The associated Toffoli cost is in

$$\widetilde{O}\left(d_x\left(\log d_x + \log g_x + \log\left(\frac{\alpha_{NVT}}{\epsilon}\right)\right)\right), \tag{A75}$$

which includes the cost of implementing a controlled-unitary adder on $\log g_x$ qubits and the cost of preparing a state encoding the $2d_x$ coefficients of the central-finite-difference formula of order $2d_x$.

(3) $U_{V_{n,n',j}}$, an $(\alpha_V, a_V, \epsilon_V)$ block encoding of $\sum_{\overline{x}_{n,j}=0}^{g_x-1} (x_{n,j} - x_{n',j})/\left(\|x_n - x_{n'}\|^2 + \Delta^2\right)^{3/2} |\overline{x}_{n,j}\rangle\langle\overline{x}_{n,j}|$, where

$$\alpha_V \in O\left(\frac{x_{\max}}{\Delta^3}\right), \tag{A76}$$

$$a_V \in O\left(\log\left(\frac{\alpha_{NVT}}{\epsilon}\right)\right), \tag{A77}$$

$$\epsilon_V \in O\left(\frac{\epsilon}{\alpha_{NVT}}\right). \tag{A78}$$

It can be efficiently implemented using the alternating-sign trick together with quantum Karatsuba multiplication. The associated Toffoli cost is in $O\left((a_V)^{\log 3}\right)$, which is dominated by the cost of implementing quantum Karatsuba multiplication with a_V qubits.

(4) $U_{D_{p'_{n,j}}}$, an $\left(\alpha_{D_{p'}}, a_{D_{p'}}, \epsilon_{D_{p'}}\right)$ block encoding of $D_{p'_{n,j}}$, where

$$\alpha_{D_{p'}} \in O\left(\frac{\ln d_{p'}}{h_{p'}}\right), \tag{A79}$$

$$a_{D_{p'}} \in O\left(\log d_{p'}\right), \tag{A80}$$

$$\epsilon_{D_{p'}} \in O\left(\frac{\epsilon}{\alpha_{NVT}}\right). \tag{A81}$$

It can be efficiently implemented via a linear combination of unitary adders. The associated Toffoli cost is in

$$\widetilde{O}\left(d_{p'}\left(\log d_{p'} + \log g_{p'} + \log\left(\frac{\alpha_{NVT}}{\epsilon}\right)\right)\right), \tag{A82}$$

which includes the cost of implementing a controlled-unitary adder on $\log g_{p'}$ qubits and the cost of preparing a state encoding the $2d_{p'}$ coefficients of the central-finite-difference formula of order $2d_{p'}$.

(5) U_{p_s}, an $\left(\alpha_{p_s}, a_{p_s}, \epsilon_{p_s}\right)$ block encoding of $\sum_{\overline{p}_s} p_s |\overline{p}_s\rangle\langle\overline{p}_s|$, where

$$\alpha_{p_s} \in O\left(p_{s,\max}\right), \tag{A83}$$

$$a_{p_s} \in O\left(\log\left(\frac{\alpha_{NVT}}{\epsilon}\right)\right), \tag{A84}$$

$$\epsilon_{p_s} \in O\left(\frac{\epsilon}{\alpha_{NVT}}\right). \tag{A85}$$

It can be efficiently implemented using the alternating-sign trick. The associated Toffoli cost is in $O\left(a_{p_s}\right)$.

(6) U_{D_s}, an $\left(\alpha_{D_s}, a_{D_s}, \epsilon_{D_s}\right)$ block encoding of D_s, where

$$\alpha_{D_s} \in O\left(\frac{\ln d_s}{h_s}\right), \tag{A86}$$

$$a_{D_s} \in O\left(\log d_s\right), \tag{A87}$$

$$\epsilon_{D_s} \in O\left(\frac{\epsilon}{\alpha_{NVT}}\right). \tag{A88}$$

It can be efficiently implemented via a linear combination of unitary adders. The associated Toffoli cost is in

$$\widetilde{O}\left(d_s\left(\log d_s + \log g_s + \log\left(\frac{\alpha_{NVT}}{\epsilon}\right)\right)\right), \tag{A89}$$

which includes the cost of implementing a controlled-unitary adder on $\log g_s$ qubits and the

cost of preparing a state encoding the $2d_s$ coefficients of the central-finite-difference formula of order $2d_s$.

(7) $U_{s,n,j}$, an $(\alpha_s, a_s, \epsilon_s)$ block encoding of $-\sum_{\overline{p}_{n,j}=0}^{g_p-1} \sum_{\overline{s}=0}^{g_s-1} 2p_{n,j}^2/(s+s_{\min})^3 |\overline{p}_{n,j}\rangle\langle\overline{p}_{n,j}| \otimes |\overline{s}\rangle\langle\overline{s}|$, where

$$\alpha_s \in O\left(\frac{p_{\max}^2}{m_{\min} s_{\min}^3}\right), \tag{A90}$$

$$a_s \in O\left(\log\left(\frac{\alpha_{NVT}}{\epsilon}\right)\right), \tag{A91}$$

$$\epsilon_s \in O\left(\frac{\epsilon}{\alpha_{NVT}}\right). \tag{A92}$$

It can be efficiently implemented using the alternating-sign trick together with quantum Karatsuba multiplication. The associated Toffoli cost is in $O\left((a_s)^{\log 3}\right)$, which is dominated by the cost of implementing quantum Karatsuba multiplication with a_s qubits.

(8) $U_{1/s}$, an $\left(\alpha_{1/s}, a_{1/s}, \epsilon_{1/s}\right)$ block encoding of $\sum_{\overline{s}=0}^{g_s-1} (1/s+s_{\min})|\overline{s}\rangle\langle\overline{s}|$, where

$$\alpha_{1/s} \in O\left(\frac{1}{s_{\min}}\right), \tag{A93}$$

$$a_{1/s} \in O\left(\log\left(\frac{\alpha_{NVT}}{\epsilon}\right)\right), \tag{A94}$$

$$\epsilon_{1/s} \in O\left(\frac{\epsilon}{\alpha_{NVT}}\right). \tag{A95}$$

It can be efficiently implemented using the alternating-sign trick together with quantum Karatsuba multiplication. The associated Toffoli cost is in $O\left(\left(a_{1/s}\right)^{\log 3}\right)$, which is dominated by the cost of implementing quantum Karatsuba multiplication with $a_{1/s}$ qubits.

(9) $U_{D_{ps}}$, an $\left(\alpha_{D_{ps}}, a_{D_{ps}}, \epsilon_{D_{ps}}\right)$ block encoding of D_{p_s}, where

$$\alpha_{D_{ps}} \in O\left(\frac{\ln d_{p_s}}{h_{p_s}}\right), \tag{A96}$$

$$a_{D_{ps}} \in O\left(\log d_{p_s}\right), \tag{A97}$$

$$\epsilon_{D_{ps}} \in O\left(\frac{\epsilon}{\alpha_{NVT}}\right). \tag{A98}$$

It can be efficiently implemented via a linear combination of unitary adders. The associated Toffoli cost is in

$$\widetilde{O}\left(d_{p_s}\left(\log d_{p_s} + \log g_{p_s} + \log\left(\frac{\alpha_{NVT}}{\epsilon}\right)\right)\right), \tag{A99}$$

which includes the cost of implementing a controlled-unitary adder on $\log g_{p_s}$ qubits and the cost of preparing a state encoding the $2d_{p_s}$ coefficients of the central-finite-difference formula of order $2d_{p_s}$.

We then use Lemmas 6 and 7 to combine the above block encodings. This requires the following state-preparation unitaries:

(a) PREP$_m$, an $(\alpha_m, a_m, \epsilon_m)$ state-preparation unitary that encodes the nuclear masses, where

$$\alpha_m \in O\left(\frac{N}{m_{\min}}\right), \tag{A100}$$

$$a_m \in O\left(\log N\right), \tag{A101}$$

$$\epsilon_m \in O\left(\frac{\epsilon}{\alpha_{NVT}}\right). \tag{A102}$$

The Toffoli cost of this state-preparation unitary is in

$$O\left(N \log\left(\frac{\alpha_{NVT}}{\epsilon}\right)\right). \tag{A103}$$

(b) PREP$_Z$, an $(\alpha_Z, a_Z, \epsilon_Z)$ state-preparation unitary that encodes the atomic numbers of the nuclei, where

$$\alpha_Z \in O\left(N^2 Z_{\max}^2\right), \tag{A104}$$

$$a_Z \in O\left(\log N\right), \tag{A105}$$

$$\epsilon_Z \in O\left(\frac{\epsilon}{\alpha_{NVT}}\right). \tag{A106}$$

The Toffoli cost of this state-preparation unitary is in

$$O\left(N \log\left(\frac{\alpha_{NVT}}{\epsilon}\right)\right). \tag{A107}$$

(c) PREP$_{\text{out}}^{NVT}$, an $(\alpha_{\text{out}}, a_{\text{out}}, \epsilon_{\text{out}})$ state-preparation unitary that is used to combine all terms of the NVT Liouvillian, where

$$\alpha_{\text{out}} \in O\left(1\right), \tag{A108}$$

$$a_{\text{out}} \in O\left(1\right), \tag{A109}$$

$$\epsilon_{\text{out}} \in O\left(\frac{\epsilon}{\alpha_{NVT}}\right). \tag{A110}$$

The Toffoli cost of this state-preparation unitary is in

$$O\left(\log\left(\frac{\alpha_{NVT}}{\epsilon}\right)\right). \tag{A111}$$

As before, we utilize $O(1)$ SWAP registers to combine the individual block encodings efficiently. Controlled by the PREP$_m$ or PREP$_Z$ register, we swap the appropriate nuclear-position and -momentum variables into the SWAP registers. This allows us to apply the

nine basic block encodings only once to the SWAP registers holding the appropriate position and momentum variables rather than $O(N)$ or $O(N^2)$ times to each individual position or momentum variable. However, we do require a total of $O(N\log(g'))$ SWAP operations, where $g' = \max\{g_x, g_{p'}\} \leq g = \max\{g_x, g_{p'}, g_s, g_{p_s}\}$, implying $O(N\log g)$ Toffolis.

Going through the same analysis as for the classical *NVE* Liouvillian yields the desired complexity bounds. ■

APPENDIX B: EVOLUTION UNDER THE ELECTRONIC LIOUVILLIAN

As an analytic expression for the electronic ground-state energy E_{el} is unavailable, we cannot follow the same strategy as described in Appendix A for the nuclear part to implement $e^{-iL_{\mathrm{el}}t}$. In particular, we somehow need to approximate the derivative $\partial E_{\mathrm{el}}/\partial x_{n,j}$, which we do via a central-finite-difference formula of order $2d_e$. Recall from Eq. (20) that the resulting approximate operator is given by

$$D^{\mathrm{el}}_{n,j} = \frac{1}{h_x}\sum_{k=-d_e}^{d_e}\sum_{(n',j')\neq(n,j)}\sum_{\bar{x}_{n',j'}}\sum_{\bar{x}_{n,j}} c_{d_e,k} \times E_{\mathrm{el}}\left(\{x_{n',j'}\}, x_{n,j}+kh_x\right)|\bar{x}_{n',j'}\rangle\langle\bar{x}_{n',j'}|\otimes|\bar{x}_{n,j}\rangle\langle\bar{x}_{n,j}|, \tag{B1}$$

where the coefficients $\{c_{d_e,k}\}$ are as in Definition 1.

We prove the following lemma, which upper bounds the complexity of simulating $e^{-iL_{\mathrm{el}}t}$.

Lemma 11 (Complexity of simulating $e^{-iL_{\mathrm{el}}t}$). Assume the following:

(1) Let $\epsilon \in (0,1)$ and $t \in \mathbb{R}_{\geq 0}$.
(2) Let $U_{H_{\mathrm{el}}}$ be a Hermitian $(\lambda, a_{\mathrm{el}}, h_x h_p \epsilon/36Nd_e t)$ block encoding of the electronic Hamiltonian $H_{\mathrm{el}}(\{x_n\})$.
(3) Let γ be a lower bound on the spectral gap of the block-encoded operator $\widetilde{H}_{\mathrm{el}}(\{x_n\})$ over all phase-space grid points.
(4) For any $d_e \in \mathbb{N}_+$, it holds that $\max_{x^*\in[-x_{\max},x_{\max}]^{3N}} \left|(\partial^{(2d_e+1)}E_{\mathrm{el}}/\partial x_{n,j}^{(2d_e+1)})(x^*)\right| \leq \chi u^{2d_e+1}$ for some constant χ with units of energy and u with units of inverse length.
(5) Let U_I be the initial-state-preparation oracle from Definition 9 and let δ be a lower bound on the initial overlap with the true electronic ground state of $\widetilde{H}_{\mathrm{el}}$.

In order to implement an ϵ-precise Liouvillian simulation unitary $U_{L_{\mathrm{el}}}$ of $e^{-iL_{\mathrm{el}}t}$ with success probability at least $1-\xi$ it is sufficient to query $U_{H_{\mathrm{el}}}$ a total number of times

$$O\left(Nd_e\log\left(\frac{Nd_e}{\xi}\right)\left(\frac{\lambda t}{h_x h_p}+\log\left(\frac{N\lambda\ln(d_e)t}{h_x h_p\epsilon}\right)\times\log\left(\frac{Nd_e\log\left(\frac{N\lambda\ln(d_e)t}{h_x h_p\epsilon}\right)}{\epsilon}\right)+\frac{\lambda}{\gamma\,\delta}\log\left(\frac{Nd_e}{\delta\epsilon}\right)\right)\right), \tag{B2}$$

where

$$d_e \in O\left(\frac{\log\left(\frac{N\chi ut}{h_p\epsilon}\right)}{\log\left(\frac{1}{uh_x}\right)}\right). \tag{B3}$$

Furthermore, we require

$$O\left(\frac{Nd_e}{\delta}\log\left(\frac{Nd_e}{\xi}\right)\log\left(\frac{Nd_e}{\delta\epsilon}\right)\right) \tag{B4}$$

queries to the initial-state-preparation oracle U_I from Definition 9.

As mentioned in the main text, one important feature of the electronic Liouvillian L_{el} is that all summands commute with each other (see Definition 5). The evolution operator associated with L_{el} can thus be decomposed as follows:

$$e^{-iL_{\mathrm{el}}t} = e^{-i\left(i\sum_{n=1}^{N}\sum_{j=1}^{3}D^{\mathrm{el}}_{n,j}\otimes D^{1}_{p_{n,j}}\right)t} = \prod_{n,j} e^{D^{\mathrm{el}}_{n,j}\otimes D^{1}_{p_{n,j}}\,t}, \tag{B5}$$

where we use $p_{n,j}$ to denote either a real or virtual momentum variable.

Let us now explain how to implement a single exponential appearing in Eq. (B5). Note that $D^{\mathrm{el}}_{n,j}\otimes D^{1}_{p_{n,j}}$ acts nontrivially on the nuclear-momentum register. We deal with the discrete nuclear-momentum derivative $D^1_{p_{n,j}}$ via a quantum Fourier transform (QFT) the action of which on the nuclear-momentum register is defined as follows:

$$\mathrm{QFT}|\bar{p}\rangle := \frac{1}{\sqrt{g_p}}\sum_{l} e^{2\pi i\bar{p}l/g_p}|l\rangle. \tag{B6}$$

Here, we have dropped the n,j indices of the integer momentum variable $\bar{p}$ for ease of notation. The quantum Fourier transform diagonalizes finite-difference operators. Recall that D^1 is a first-order finite-difference operator of the form $D^1 := 1/2h_p\sum_{\bar{p}}(|\bar{p}-1\rangle\langle\bar{p}| - |\bar{p}\rangle\langle\bar{p}-1|)$. Thus,

$$
\begin{aligned}
D^1 &= \left(\mathrm{QFT}\ \mathrm{QFT}^{-1}\right) D_1 \left(\mathrm{QFT}\ \mathrm{QFT}^{-1}\right) = \mathrm{QFT}\left(\mathrm{QFT}^{-1} D_1 \mathrm{QFT}\right) \mathrm{QFT}^{-1} \\
&= \mathrm{QFT}\left(\frac{1}{\sqrt{g_p}} \sum_{\bar{p}} \sum_{l,k} \sum_{l',k'} e^{-2\pi i l k/g_p} e^{2\pi i l' k'/g_p} |l\rangle\langle k| \frac{|\bar{p}-1\rangle\langle\bar{p}| - |\bar{p}\rangle\langle\bar{p}-1|}{2h_p} |l'\rangle\langle k'|\right) \mathrm{QFT}^{-1} \\
&= \mathrm{QFT}\left(\frac{1}{\sqrt{g_p}} \sum_{\bar{p}} \sum_{l,k'} \frac{e^{-2\pi i l(\bar{p}-1)/g_p} e^{2\pi i \bar{p} k'/g_p} - e^{-2\pi i l \bar{p}/g_p} e^{2\pi i (\bar{p}-1) k'/g_p}}{2h_p} |l\rangle\langle k'|\right) \mathrm{QFT}^{-1} \\
&= \mathrm{QFT}\left(\sum_{l,k'} \frac{e^{2\pi i l/g_p} - e^{-2\pi i k'/g_p}}{2h_p} |l\rangle\langle k'|\right) \left(\sum_{\bar{p}} e^{2\pi i \bar{p}(k'-l)/g_p}\right) \mathrm{QFT}^{-1} \\
&= \mathrm{QFT}\left(i \sum_{l} \frac{\sin\left(2\pi l/g_p\right)}{h_p} |l\rangle\langle l|\right) \mathrm{QFT}^{-1}.
\end{aligned} \tag{B7}
$$

The above calculation shows that QFT does indeed diagonalize D^1. Higher-order finite-difference operators can also be diagonalized via QFT but will have different eigenvalues. For simplicity, we only consider a first-order finite-difference operator here. A single exponential of Eq. (B5) can then be expressed as follows:

$$
\begin{aligned}
e^{D^{\mathrm{el}}_{n,j} D^1_{p_{n,j}} t} &\equiv e^{D^{\mathrm{el}}_{n,j} \otimes D^1_{p_{n,j}} t} \\
&= (\mathbb{1} \otimes \mathrm{QFT})\, e^{iD^{\mathrm{el}}_{n,j} \otimes \sum_l \frac{\sin\left(2\pi l/g_p\right)}{h_p} |l\rangle\langle l| t} \\
&\quad \times \left(\mathbb{1} \otimes \mathrm{QFT}^{-1}\right) \\
&= (\mathbb{1} \otimes \mathrm{QFT}) \prod_l e^{iD^{\mathrm{el}}_{n,j} \otimes \frac{\sin\left(2\pi l/g_p\right)}{h_p} |l\rangle\langle l| t} \\
&\quad \times \left(\mathbb{1} \otimes \mathrm{QFT}^{-1}\right),
\end{aligned} \tag{B8}
$$

where we have used the fact that $Ue^A U^\dagger = e^{UAU^\dagger}$ for any square matrix A and unitary matrix U of the same dimension. The following lemma allows us to simplify the above expression.

Lemma 12 (Projector exponential). Let $\{A_l\}_{l=1}^M$ with $A_l \in \mathbb{C}^{M\times M}$ be a set of M matrices and let $\{P_l\}_{l=1}^M$ with $P_l \in \mathbb{C}^{M\times M}$ and $P_l^2 = P_l$ for all $l \in [M]$ be a set of M orthogonal projectors that satisfy $\sum_l P_l = \mathbb{1}$. Then, it holds that

$$
\prod_l e^{A_l \otimes P_l} = \sum_l e^{A_l} \otimes P_l. \tag{B9}
$$

Proof. Using the Taylor-series expansion of a matrix exponential and the fact that $P_l^2 = P_l$, $P_k P_l = 0$ if $k \neq l$ and $\sum_l P_l = \mathbb{1}$, we obtain

$$
\begin{aligned}
\prod_l e^{A_l \otimes P_l} &= \prod_l \sum_{k=0}^{\infty} \frac{(A_l \otimes P_l)^k}{k!} \\
&= \prod_l \left(\mathbb{1} \otimes \mathbb{1} + \left(e^{A_l} - \mathbb{1}\right) \otimes P_l\right) \\
&= \mathbb{1} \otimes \mathbb{1} + \sum_l \left(e^{A_l} - \mathbb{1}\right) \otimes P_l = \mathbb{1} \otimes \mathbb{1} \\
&\quad + \sum_l e^{A_l} \otimes P_l - \mathbb{1} \otimes \sum_l P_l \\
&= \sum_l e^{A_l} \otimes P_l.
\end{aligned} \tag{B10}
$$

■

Lemma 12 implies that

$$
\begin{aligned}
e^{D^{\mathrm{el}}_{n,j} D_{p_{n,j}} t} &= (\mathbb{1} \otimes \mathrm{QFT}) \sum_l e^{iD^{\mathrm{el}}_{n,j} \frac{\sin\left(2\pi l/g_p\right)}{h_p} t} \\
&\quad \otimes |l\rangle\langle l| \left(\mathbb{1} \otimes \mathrm{QFT}^{-1}\right).
\end{aligned} \tag{B11}
$$

Recalling that $D^{\mathrm{el}}_{n,j} = 1/h_x \sum_{k=-d_e}^{d_e} \sum_{\bar{x}_{n,j}} c_{d_e,k} E_{\mathrm{el}}(x_{n,j} + kh_x) |\bar{x}_{n,j}\rangle\langle\bar{x}_{n,j}|$, we obtain the following equality:

$$
\begin{aligned}
&e^{D^{\mathrm{el}}_{n,j} \otimes D^1_{p_{n,j}} t} \\
&= (\mathbb{1} \otimes \mathrm{QFT}) \sum_{\bar{x}_{n,j}} \sum_l \prod_k e^{ic_{d_e,k} \frac{E_{\mathrm{el}}(x_{n,j}+kh_x)}{h_x} \frac{\sin\left(2\pi l/g_p\right)}{h_p} t} \\
&\quad \times |\bar{x}_{n,j}\rangle\langle\bar{x}_{n,j}| \otimes |l\rangle\langle l| \left(\mathbb{1} \otimes \mathrm{QFT}^{-1}\right).
\end{aligned} \tag{B12}
$$

We implement the above expression via controlled Hamiltonian simulation.

Definition 11 (Controlled Hamiltonian simulation ([61], Definition 51)). Let $M = 2^J$ for some $J \in \mathbb{N}$, $\gamma \in \mathbb{R}$ and $\epsilon \geq 0$. We say that the unitary

$$W := \sum_{m=-M}^{M} |m\rangle\langle m| \otimes e^{im\gamma H} \tag{B13}$$

implements a controlled (M, γ) simulation of the Hamiltonian H, where $|m\rangle$ denotes a (signed) bit string $|b_J b_{J-1} \ldots b_0\rangle$ such that $m = -b_J 2^J + \sum_{j=0}^{J-1} b_j 2^j$.

Lemma 13 (Complexity of controlled Hamiltonian simulation ([61], Lemma 52)). Let $M = 2^J$ for some $J \in \mathbb{N}$, $\gamma \in \mathbb{R}$ and $\epsilon \geq 0$. Suppose that U is an $(\alpha, a, \epsilon/|8(J+1)^2 M\gamma|)$ block encoding of the Hamiltonian H. Then, we can implement a $(1, a+2, \epsilon)$ block encoding of a controlled (M, γ) simulation of the Hamiltonian H with

$$O\left(|\alpha M\gamma| + J \log\left(J/\epsilon\right)\right) \tag{B14}$$

uses of controlled-U or its inverse and with $O\left(a|\alpha M\gamma| + aJ \log\left(J/\epsilon\right)\right)$ two-qubit gates.

Before applying the above lemma, we first use coherent quantum arithmetic to compute an ϵ_{sin}-precise binary approximation of $\sin\left(2\pi l/g_p\right)$ in an ancilla register controlled by the $|l\rangle$ register. This can be done using a truncated Taylor-series expansion of the sine function and has Toffoli cost in $O\left(\log\left(1/\epsilon_{\text{sin}}\right)\right)$ since the error of the truncation vanishes exponentially quickly. The size of the ancilla register is also in $O\left(\log\left(1/\epsilon_{\text{sin}}\right)\right)$.

Controlled by this $\sin\left(2\pi l/g_p\right)$ ancilla register, we then simulate $\exp\left(-iH_{\text{el}}\left(\{x_n\}\right) t_{c_k,l}\right)$ using $U_{H_{\text{el}}}$, a Hermitian $(\lambda, a_{\text{el}}, \epsilon_{\text{be}})$ block encoding of H_{el}, where

$$t_{c_k,l} := \frac{c_{d_e,k}}{h_x} \frac{\sin\left(2\pi l/g_p\right)}{h_p} t \tag{B15}$$

is a rescaled time variable depending on the finite-difference coefficients $\{c_{d_e,k}\}$ of $D^{\text{el}}_{n,j}$. For convenience, let $\widetilde{H}_{\text{el}} := \lambda\left(\langle 0| \otimes \mathbb{1}\right) U_{H_{\text{el}}} \left(|0\rangle \otimes \mathbb{1}\right)$ denote the Hermitian matrix that $U_{H_{\text{el}}}$ block encodes. This implies that $\left\|H_{\text{el}} - \widetilde{H}_{\text{el}}\right\| \leq \epsilon_{\text{be}}$.

Note that Lemma 13 applies to integer values of m, which translates to integer values of $\sin\left(2\pi l/g_p\right)$ in our case. Hence, we need to "blow up" the values of $\sin\left(2\pi l/g_p\right)$ by a factor of $O\left(1/\epsilon_{\text{sin}}\right)$. This then entails a renormalization of the exponent by a factor of $O\left(\epsilon_{\text{sin}}\right)$, which can be done via a rescaling of the form $\gamma \to \epsilon'\gamma$ for an appropriate $\epsilon' \in O\left(\epsilon_{\text{sin}}\right)$.

The general strategy is first to shift the nuclear-position register of a single nuclear-position variable $\bar{x}_{n,j}$ according to the finite-difference scheme of order $2d_e$. This is done using a unitary adder. Then, we (approximately) prepare the ground state $|\widetilde{\psi}\{x_n\}\rangle$ of $\widetilde{H}_{\text{el}}$, controlled by the entire nuclear-position register $|\{\bar{x}_{n,j}\}\rangle$, in the electronic register. Next, controlled by the entire nuclear-position register $|\{\bar{x}_{n,j}\}\rangle$ and the Fourier-transformed momentum register associated with the one-dimensional (1D) momentum variable $\bar{p}_{n,j}$, $|l\rangle$, we apply $\exp\left(-iH_{\text{el}}\left(\{x_n\}\right) t_{c_k,l}\right)$ to the electronic register. This generates states of the form

$$e^{ic_{d_e,k} \frac{E_{\text{el}}(x_{n,j}+kh_x)}{h_x} \frac{\sin\left(2\pi l/g_p\right)}{h_p} t} |\bar{x}_{n,j} + k\rangle |l\rangle. \tag{B16}$$

Finally, we uncompute the electronic ground state. Now we simply repeat the above procedure for each stencil point of the finite-difference scheme. More precisely, we shift the nuclear-position register of the 1D nuclear-position variable of interest, $x_{n,j}$, to the next stencil point, prepare the electronic ground state for that nuclear configuration, and then apply $\exp\left(-iH_{\text{el}}\left(\{x_n\}\right) t_{c_k,l}\right)$ to the electronic register. In the last step, we shift the position register corresponding to $x_{n,j}$ back to the original state to obtain the desired phase factor,

$$\prod_k e^{ic_{d_e,k} \frac{E_{\text{el}}(x_{n,j}+kh_x)}{h_x} \frac{\sin\left(2\pi l/g_p\right)}{h_p} t} |\bar{x}_{n,j}\rangle |l\rangle. \tag{B17}$$

The overall procedure for simulating $e^{-iL_{\text{el}}t}$ is summarized in Algorithm 3 as well as Fig. 7.

One might consider using the gradient-computation algorithm developed in Ref. [62] to compute $D^{\text{el}}_{n,j}$ in the exponent. The hope is that $O\left(N^{1/2}\right)$ rather than $O\left(N\right)$ evaluations of the electronic ground-state energy are sufficient. However, a straightforward application fails in our case, since we have to compute the gradient in superposition over all nuclear positions. This is problematic

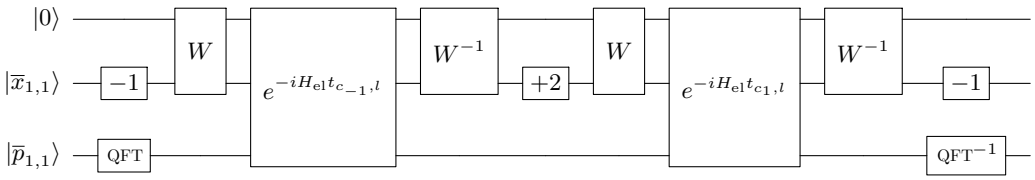

FIG. 7. The circuit for implementing the evolution under the electronic Liouvillian for a single nucleus in 1D. The top register corresponds to the electronic register. "-1" denotes a unitary adder of the form $\sum_{\bar{x}} |\bar{x}+1\rangle\langle\bar{x}|$ and, similarly, "$+2$" denotes a unitary adder of the form $\sum_{\bar{x}} |\bar{x}-2\rangle\langle\bar{x}|$.

because the gradient-computation algorithm produces different global phases for different nuclear positions, i.e., the global phases become local phases that cannot simply be ignored. Uncomputing these local phases is nontrivial and is left for future work.

Both the simulation of $\exp\left(-iH_{\mathrm{el}}\left(\{x_n\}\right)t_{c_k,l}\right)$ and the electronic ground-state preparation require access to $U_{H_{tel}}$, a block encoding of $H_{\mathrm{el}}\left(\{x_n\}\right)$ as given in Definition 11. For the ground-state preparation, it is important that $U_{H_{tel}}$ is a Hermitian block encoding, meaning that $\widetilde{H}_{\mathrm{el}}$ is Hermitian. This is discussed in more detail in the proof of Lemma 11. Note that the only way in which the nuclear positions $\{x_n\}$ enter the electronic Hamiltonian is via the phase factors of the electron-nucleus interaction terms. In Ref. [28], the nuclear positions are accessed via a quantum random access memory (QROM). The phase $k_{c-b}\cdot\bar{x}_n$ is computed in an ancilla register, which is then hit with a phase gradient to produce the phase factor $\exp\left(ik_{c-b}\cdot x_n\right)$. In our case, instead of using a QROM to access x_n, we swap the nuclear-position register $|\bar{x}_n\rangle$ into an ancilla register and compute $k_{c-b}\cdot\bar{x}_n$. The swap is controlled by the ancilla register preparing the state $\sum_{n=0}^{N-1}\sqrt{Z_n/Z}|n\rangle$ that is needed for block encoding the electron-nucleus interaction terms. The Toffoli cost associated with the controlled SWAP gates is in $O(N)$, which matches the complexity of the original QROM model. Apart from accessing the nuclear positions differently, we can employ exactly the same techniques as presented in Ref. [28] to block encode $H_{\mathrm{el}}\left(\{x_n\}\right)$, which leads to the complexity expressions of Lemma 3.

Let us now discuss the electronic ground-state preparation in more detail.

Definition 12 (Fidelity). Let $|x\rangle, |y\rangle \in \mathbb{C}^{2^n\times 2^n}$ be two quantum states. The fidelity $F(x,y)$ between $|x\rangle$ and $|y\rangle$ is given by

$$F(x,y) := |\langle x|y\rangle|. \tag{B18}$$

Lemma 14 (Ground-state preparation with a priori ground-state energy bound ([43], Theorem 6, reformulated)). Suppose that we have a Hamiltonian $\widetilde{H} = \sum_k \widetilde{E}_k|\widetilde{\psi}_k\rangle\langle\widetilde{\psi}_k| \in \mathbb{C}^{N\times N}$, where $\widetilde{E}_k \le \widetilde{E}_{k+1}$, which is given through its $(\lambda, m, 0)$ block encoding U_H. Also suppose that we have an initial state $|\phi_0\rangle$, prepared by a unitary U_I, together with a lower bound on the overlap $|\langle\widetilde{\psi}_0|\phi_0\rangle| \ge \delta$. Furthermore, we require the following bound on the ground-state energy and the spectral gap: $\widetilde{E}_0 \le \mu - \gamma/2 < \mu + \gamma/2 \le \widetilde{E}_1$, where μ is an upper bound on the ground-state energy and γ is a lower bound on the spectral gap of $\widetilde{H}$. Then, the ground state $|\widetilde{\psi}_0\rangle$ can be prepared with fidelity at least $1-\epsilon_{\mathrm{prep}}$ using

$$O\left(\frac{\lambda}{\gamma\delta}\log\left(\frac{1}{\delta\epsilon_{\mathrm{prep}}}\right)\right) \tag{B19}$$

queries to U_H and

$$O\left(\frac{1}{\delta}\right) \tag{B20}$$

queries to U_I.

We are now ready to prove Lemma 11, which provides an upper bound on the complexity of simulating the evolution under the electronic Liouvillian.

Proof of Lemma 11. The error in approximating $e^{-iL_{\mathrm{el}}t}$ consists of two parts. On the one hand, there is the simulation error ϵ_{sim}, which arises from approximately preparing the exact ground state $|\widetilde{\psi}_0\rangle$ of $\widetilde{H}_{\mathrm{el}}$ and approximately implementing $\exp\left(-iH_{\mathrm{el}}\left(\{x_n\}\right)t_{c_k,l}\right)$. On the other hand, we have the discretization error ϵ_{disc} associated with the finite-difference matrix $D^{\mathrm{el}}_{n,j}$ of the derivatives of the electronic ground-state energy and the finite binary representation of $\sin\left(2\pi l/g_p\right)$. Let $\widetilde{L}_{\mathrm{el}}$ denote the approximate discrete electronic Liouvillian. Then, the overall error associated with simulating $e^{-iL_{\mathrm{el}}t}$ is upper bounded as follows:

$$\begin{aligned}
&\left\|\left(\langle 0|\otimes\mathbb{1}\right)U_{L_{\mathrm{el}}}\left(|0\rangle\otimes\mathbb{1}\right) - e^{-iL_{\mathrm{el}}t}\right\| \\
&\quad\le \left\|\left(\langle 0|\otimes\mathbb{1}\right)U_{L_{\mathrm{el}}}\left(|0\rangle\otimes\mathbb{1}\right) - e^{-i\widetilde{L}_{\mathrm{el}}t}\right\| \\
&\qquad + \left\|e^{-i\widetilde{L}_{\mathrm{el}}t} - e^{-iL_{\mathrm{el}}t}\right\| \\
&\quad\le \epsilon_{\mathrm{sim}} + \left\|\widetilde{L}_{\mathrm{el}} - L_{\mathrm{el}}\right\| t \\
&\quad\le \epsilon_{\mathrm{sim}} + \epsilon_{\mathrm{disc}}t,
\end{aligned} \tag{B21}$$

where we have used Duhamel's formula in going from the second to the third line. The overall error is less or equal to ϵ if we ensure that $\epsilon_{\mathrm{sim}} \le \epsilon/2$ and $\epsilon_{\mathrm{disc}} \le \epsilon/(2t)$. Recall that L_{el} is a sum of $3N$ commuting terms and that each term involves a central-finite-difference formula of order $2d_e$. Thus, a total of $6Nd_e$ exponentials need to be implemented. We first discuss the simulation error $\epsilon_{\mathrm{sim},1}$ of a single exponential. By the triangle inequality, if $\epsilon_{\mathrm{sim},1} \le \epsilon/(12Nd_e)$, then $\epsilon_{\mathrm{sim}} \le \epsilon/2$. Let W denote the unitary that prepares an approximate ground state of $\widetilde{H}_{\mathrm{el}}$ for fixed nuclear positions according to Lemma 14, i.e.,

$$W|\{\bar{x}_{n,j}\}\rangle|0\rangle = |\{\bar{x}_{n,j}\}\rangle|\widetilde{\phi}_0\left(,\{x_{n,j}\}\right)\rangle \tag{B22}$$

with

$$|\langle\widetilde{\psi}_0\left(\{x_{n,j}\}\right)|\widetilde{\phi}_0\left(\{x_{n,j}\}\right)\rangle| \ge 1-\epsilon_{\mathrm{prep}}. \tag{B23}$$

Note that we can view $U_{H_{\mathrm{el}}}$ as an exact block encoding of $\widetilde{H}_{\mathrm{el}}$, which allows us to use Lemma 14 directly without further error propagation. In the following discussion, we will refrain from writing out the $\left(\{x_{n,j}\}\right)$ dependence explicitly.

Now, it holds that

$$|\widetilde{\phi}_0\rangle = e^{i\alpha}\left(1-\epsilon'_{\text{prep}}\right)|\widetilde{\psi}_0\rangle + \beta|\widetilde{\psi}_0^{\perp}\rangle \tag{B24}$$

for some angle $\alpha \in [0, 2\pi)$, $0 \leq \epsilon'_{\text{prep}} \leq \epsilon_{\text{prep}}$ and $|\beta|^2 = 2\epsilon'_{\text{prep}} - \left(\epsilon'_{\text{prep}}\right)^2 \leq 2\epsilon'_{\text{prep}}$. Letting

$$|\psi'_0\rangle := e^{i\alpha}|\widetilde{\psi}_0\rangle \tag{B25}$$

we thus have that

$$\left\||\widetilde{\phi}_0\rangle - |\psi'_0\rangle\right\| \leq \sqrt{2\epsilon_{\text{prep}}}. \tag{B26}$$

Let $U_{\text{e,el}}$ be an $\epsilon_{\text{e,el}}$-precise block encoding of

$$\sum_{\{\bar{x}_n\},l} |\{\bar{x}_n\}\rangle\langle\{\bar{x}_n\}| \otimes |\sin\left(2\pi l/g_p\right)\rangle\langle\sin\left(2\pi l/g_p\right)| \otimes e^{-iH_{\text{el}}t_{c_k,l}} \tag{B27}$$

and let $\widetilde{U}_{\text{e,el}} := (\langle 0| \otimes \mathbb{1})\, U_{\text{e,el}}\, (|0\rangle \otimes \mathbb{1})$ denote the block-encoded approximation to the above operator. Our goal is to bound the error of the phase factors obtained via phase kickback from the electronic register, i.e., we wish to bound

$$\epsilon_{\text{sim,1}} := \Bigg\| (\mathbb{1} \otimes \langle 0|)\, W^{-1}\widetilde{U}_{\text{e,el}} W\, (\mathbb{1} \otimes |0\rangle) - \sum_{\{\bar{x}_n\}} e^{-iE_{\text{el}}(\{x_{nj}\})t_{c_k,l}} |\{\bar{x}_n\}\rangle\langle\{\bar{x}_n\}| \Bigg\|, \tag{B28}$$

for fixed $t_{c_k,l}$. Note that the above definition implies that the electronic register is projected out to the $|0\rangle$ state at the end of the simulation. In other words, the error $\epsilon_{\text{sim,1}}$ is only measured within the Hilbert space of the nuclear-position and -momentum registers but not the electronic register. Importantly, the error matrix

$$\mathcal{E}_{\text{sim,1}} := (\langle 0| \otimes \mathbb{1})\, W^{-1}\widetilde{U}_{\text{e,el}} W\, (|0\rangle \otimes \mathbb{1}) - \sum_{\{\bar{x}_n\}} e^{-iE_{\text{el}}(\{x_{nj}\})t_{c_k,l}} |\{\bar{x}_n\}\rangle\langle\{\bar{x}_n\}| \tag{B29}$$

is diagonal in the nuclear-position and -momentum basis, since W and $\widetilde{U}_{\text{e,el}}$ act trivially on the nuclear-position and -momentum register. Hence, $\epsilon_{\text{sim,1}}$ is simply the largest value on the diagonal of $\mathcal{E}_{\text{sim,1}}$. This allows us to consider the phase error for each nuclear computational basis state separately. Let

$$\widetilde{U}_{\text{e,el}}\left(\{x_n\}, t_{c_k,l}\right) := \left(\langle\{\bar{x}_n\}|\langle\sin\left(2\pi l/g_p\right)| \otimes \mathbb{1}\right)\widetilde{U}_{\text{e,el}} \times \left(|\{\bar{x}_n\}\rangle|\sin\left(2\pi l/g_p\right)\rangle \otimes \mathbb{1}\right) \tag{B30}$$

denote a single exponential of $\widetilde{U}_{\text{e,el}}$ for fixed nuclear positions $\{x_n\}$ and Fourier parameter l. Similarly, let $W(\{x_n\})$ denote the electronic ground-state-preparation unitary for fixed nuclear positions $\{x_n\}$. Then, we have that

$$\begin{aligned}\epsilon_{\text{sim,1}} &= \max_{\{\bar{x}_n\},l} \Big| \langle 0|W^{-1}(\{x_n\})\, \widetilde{U}_{\text{e,el}}\left(\{x_n\}, t_{c_k,l}\right) W(\{x_n\})\, |0\rangle \\ &\quad - e^{-iE_{\text{el}}t_{c_k,l}(\{x_n\})} \Big| \\ &\leq \max_{\{\bar{x}_n\},l} \Big\| W^{-1}(\{x_n\})\, \widetilde{U}_{\text{e,el}}\left(\{x_n\}, t_{c_k,l}\right) W(\{x_n\})\, |0\rangle \\ &\quad - e^{-iE_{\text{el}}t_{c_k,l}(\{x_n\})}|0\rangle \Big\|. \end{aligned} \tag{B31}$$

In the following, we will not write out the $\{x_n\}$ and l dependence explicitly. Applying the triangle inequality repeatedly and using the submultiplicativity of the induced 2-norm, one finds the following upper bound on the approximation error for a single exponential:

$$\begin{aligned}\epsilon_{\text{sim,1}} &\leq \left\| W^{-1}\widetilde{U}_{\text{e,el}} W|0\rangle - e^{-iE_{\text{el}}t_{c_k}}|0\rangle \right\| \\ &\leq \left\| W^{-1}\widetilde{U}_{\text{e,el}} W|0\rangle - W^{-1}\widetilde{U}_{\text{e,el}}|\psi'_0\rangle \right\| \\ &\quad + \left\| W^{-1}\widetilde{U}_{\text{e,el}}|\psi'_0\rangle - e^{-iE_{\text{el}}t_{c_k}}|0\rangle \right\| \\ &\leq \left\| |\widetilde{\phi}_0\rangle - |\psi'_0\rangle \right\| \\ &\quad + \left\| W^{-1}\widetilde{U}_{\text{e,el}}|\psi'_0\rangle - W^{-1}e^{-i\widetilde{E}_{\text{el}}t_{c_k}}|\psi'_0\rangle \right\| \\ &\quad + \left\| W^{-1}e^{-i\widetilde{E}_{\text{el}}t_{c_k}}|\psi'_0\rangle - e^{-iE_{\text{el}}t_{c_k}}|0\rangle \right\| \\ &\leq \sqrt{2\epsilon_{\text{prep}}} + \left\| \widetilde{U}_{\text{e,el}}|\psi'_0\rangle - e^{-i\widetilde{E}_{\text{el}}t_{c_k}}|\psi'_0\rangle \right\| \\ &\quad + \left\| W^{-1}e^{-i\widetilde{E}_{\text{el}}t_{c_k}}|\psi'_0\rangle - W^{-1}e^{-iE_{\text{el}}t_{c_k}}|\psi'_0\rangle \right\| \\ &\quad + \left\| W^{-1}e^{-iE_{\text{el}}t_{c_k}}|\psi'_0\rangle - e^{-iE_{\text{el}}t_{c_k}}|0\rangle \right\|. \end{aligned} \tag{B32}$$

The second term is upper bounded by the block-encoding error $\epsilon_{\text{e,el}}$ of $U_{\text{e,el}}$. Duhamel's formula can be used to upper bound the third term:

$$\begin{aligned}&\left\| W^{-1}e^{-i\widetilde{E}_{\text{el}}t_{c_k}}|\psi'_0\rangle - W^{-1}e^{-iE_{\text{el}}t_{c_k}}|\psi'_0\rangle \right\| \\ &\quad \leq \left| e^{-i\widetilde{E}_{\text{el}}t_{c_k}} - e^{-iE_{\text{el}}t_{c_k}} \right| \leq \left| \widetilde{E}_{\text{el}}t_{c_k} - E_{\text{el}}t_{c_k} \right| \\ &\quad \leq \left| \widetilde{E}_{\text{el}} - E_{\text{el}} \right| \frac{t}{h_x h_p}. \end{aligned} \tag{B33}$$

Now recall that $\left\|\widetilde{H}_{\text{el}} - H_{\text{el}}\right\| \leq \epsilon_{\text{be}}$. Eigenvalue perturbation theory then tells us that [63]

$$\left|\widetilde{E}_{\text{el}} - E_{\text{el}}\right| \leq \epsilon_{\text{be}}. \tag{B34}$$

For the last term, note that

$$\begin{aligned}\left\| W^{-1}|\psi_0'\rangle - |0\rangle \right\| &= \left\| W^{-1}|\psi_0'\rangle - W^{-1}W|0\rangle \right\| \\ &\leq \left\| W^{-1} \right\| \left\| |\psi_0'\rangle - W|0\rangle \right\| \\ &= \left\| |\psi_0'\rangle - |\widetilde{\phi}_0\rangle \right\| \\ &\leq \sqrt{2\epsilon_{\text{prep}}}. \end{aligned} \tag{B35}$$

Putting it all together, we find that

$$\epsilon_{\text{sim},1} \leq 2\sqrt{2\epsilon_{\text{prep}}} + \epsilon_{\text{e,el}} + \frac{t}{h_x h_p}\epsilon_{\text{be}}. \tag{B36}$$

To achieve $\epsilon_{\text{sim},1} \leq \epsilon/12Nd_e$, it suffices to have

$$\epsilon_{\text{prep}} \leq \frac{1}{2}\left(\frac{\epsilon}{72Nd_e}\right)^2, \tag{B37}$$

$$\epsilon_{\text{e,el}} \leq \frac{\epsilon}{36Nd_e}, \tag{B38}$$

$$\epsilon_{\text{be}} \leq \frac{h_x h_p \epsilon}{36Nd_e t}. \tag{B39}$$

Next, let us discuss the discretization error $\epsilon_{\text{disc},1}$ of a single term of L_{el}. Let $\epsilon_{D^{\text{el}}_{n,j}}$ denote the error tolerance associated with approximating $\sum_{\bar{x}_{n,j}} \partial E_{\text{el}}/\partial x_{n,j} |\bar{x}_{n,j}\rangle\langle\bar{x}_{n,j}|$ with $D^{\text{el}}_{n,j}$, i.e.,

$$\left\| D^{\text{el}}_{n,j} - \sum_{\bar{x}_{n,j}} \frac{\partial E_{\text{el}}}{\partial x_{n,j}} |\bar{x}_{n,j}\rangle\langle\bar{x}_{n,j}| \right\| \leq \epsilon_{D^{\text{el}}_{n,j}}. \tag{B40}$$

Furthermore, let $\widetilde{\sin}\left(2\pi l/g_p\right)$ denote an approximation to $\sin\left(2\pi l/g_p\right)$ satisfying

$$|\widetilde{\sin}\left(2\pi l/g_p\right) - \sin\left(2\pi l/g_p\right)| \leq \epsilon_{\sin} \tag{B41}$$

for all $l \in [g_p]$. By the triangle inequality, we then have that

$$\begin{aligned}\epsilon_{\text{disc},1} &= \left\| D^{\text{el}}_{n,j} \frac{\widetilde{\sin}\left(2\pi l/g_p\right)}{h_p} \right. \\ &\quad \left. - \sum_{\bar{x}_{n,j}} \frac{\partial E_{\text{el}}}{\partial x_{n,j}} |\bar{x}_{n,j}\rangle\langle\bar{x}_{n,j}| \frac{\sin\left(2\pi l/g_p\right)}{h_p} \right\| \\ &\leq \left\| D^{\text{el}}_{n,j} \frac{\widetilde{\sin}\left(2\pi l/g_p\right)}{h_p} - D^{\text{el}}_{n,j} \frac{\sin\left(2\pi l/g_p\right)}{h_p} \right\| \\ &\quad + \left\| D^{\text{el}}_{n,j} \frac{\sin\left(2\pi l/g_p\right)}{h_p} \right. \\ &\quad \left. - \sum_{\bar{x}_{n,j}} \frac{\partial E_{\text{el}}}{\partial x_{n,j}} |\bar{x}_{n,j}\rangle\langle\bar{x}_{n,j}| \frac{\sin\left(2\pi l/g_p\right)}{h_p} \right\| \\ &\leq \frac{\left\| D^{\text{el}}_{n,j} \right\|}{h_p}\epsilon_{\sin} + \frac{1}{h_p}\epsilon_{D^{\text{el}}_{n,j}}. \end{aligned} \tag{B42}$$

We obtain $\epsilon_{\text{disc}} \leq \epsilon/2t$ if $\epsilon_{\text{disc},1} \leq \epsilon/6Nt$. This can be achieved by ensuring that

$$\epsilon_{\sin} \leq \frac{h_p \epsilon}{12N \left\| D^{\text{el}}_{n,j} \right\| t}, \tag{B43}$$

$$\epsilon_{D^{\text{el}}_{n,j}} \leq \frac{h_p \epsilon}{12Nt}. \tag{B44}$$

From Lemma 9, it follows that

$$\left\| D^{\text{el}}_{n,j} \right\| \leq \lambda \frac{2\ln\left(d_e+1\right)}{h_x}. \tag{B45}$$

The size of the ancilla register used for representing $\sin\left(2\pi l/g_p\right)$ is thus in

$$O\left(\log\left(\frac{1}{\epsilon_{\sin}}\right)\right) \subseteq O\left(\log\left(\frac{N\lambda\ln\left(d_e\right)t}{h_x h_p \epsilon}\right)\right). \tag{B46}$$

The order of the finite-difference approximation, d_e, is constrained by

$$\epsilon_{D^{\text{el}}_{n,j}} \leq \frac{h_p \epsilon}{12Nt}. \tag{B47}$$

Lemma 8 implies that

$$\epsilon_{D^{\text{el}}_{n,j}} \in O\left(\max_{x^* \in [-x_{\max}, x_{\max}]^{3N}} \left| \frac{\partial^{(2d_e+1)} E_{\text{el}}}{\partial x^{(2d_e+1)}_{n,j}}(x^*) \right| \left(\frac{eh_x}{2}\right)^{2d_e}\right). \tag{B48}$$

By assumption,

$$\max_{x^* \in [-x_{\max}, x_{\max}]^{3N}} \left| \frac{\partial^{(2d_e+1)} E_{\text{el}}}{\partial x^{(2d_e+1)}_{n,j}}(x^*) \right| \leq \chi u^{2d_e+1}. \tag{B49}$$

We can satisfy the constraint in Eq. (B47) by choosing

$$d_e \in O\left(\frac{\log\left(\frac{h_p\epsilon}{N\chi ut}\right)}{\log\left(uh_x\right)}\right) = O\left(\frac{\log\left(\frac{N\chi ut}{h_p\epsilon}\right)}{\log\left(\frac{1}{uh_x}\right)}\right). \tag{B50}$$

By Lemma 14, the $6Nd_e$ electronic ground-state preparations require

$$O\left(Nd_e\frac{\lambda}{\gamma\delta}\log\left(\frac{Nd_e}{\delta\epsilon}\right)\right) \tag{B51}$$

queries to $U_{H_{\rm el}}$ and

$$O\left(\frac{Nd_e}{\delta}\log\left(\frac{Nd_e}{\delta\epsilon}\right)\right) \tag{B52}$$

queries to the initial-state-preparation oracle from Definition 9. Furthermore, by Lemma 13, we need

$$O\left(Nd_e\left(\frac{\lambda t}{h_x h_p}+\log\left(\frac{N\lambda\ln(d_e)t}{h_x h_p\epsilon}\right)\times\log\left(\frac{Nd_e\log\left(\frac{N\lambda\ln(d_e)t}{h_x h_p\epsilon}\right)}{\epsilon}\right)\right)\right) \tag{B53}$$

queries to $U_{H_{\rm el}}$ for the $6Nd_e$ controlled simulations of $e^{-iH_{\rm el}t_{c_k,l}}$.

Lastly, note that the simulation of each of the $6Nd_e$ exponentials is associated with a certain failure probability due to the probabilistic nature of block encodings. By the union bound, we can ensure an overall success probability of at least $1-\xi$ if the failure probability of a single exponential is in $O(\xi/(Nd_e))$. This can be achieved via (fixed-point) amplitude amplification at the expense of a multiplicative factor of $\log(Nd_e/\xi)$ to the query complexities of $U_{H_{\rm el}}$ and U_I.

Combining all of the results yields the complexity expressions stated in Lemma 11. ■

APPENDIX C: IMPLEMENTATION OF THE OVERALL LIOUVILLIAN-EVOLUTION OPERATOR

The main goal of this appendix is to prove Theorem 1, which upper bounds the complexity of simulating e^{-iLt}. Let us first discuss some intermediate results. As explained previously, we implement the overall Liouvillian-evolution operator e^{-iLt} via a $(2k)$th-order Trotter product formula combining $e^{-iL_{\rm class}t}$ and $e^{-iL_{\rm el}t}$. The following lemma provides an upper bound on the query complexity of simulating Liouvillian dynamics in the *NVE* and *NVT* ensemble.

Lemma 15 (Query complexity of Born-Oppenheimer Liouvillian simulation). Let $L=L_{\rm class}+L_{\rm el}$ be the discrete Liouvillian operator either in the *NVE* ensemble (Definition 2) or the *NVT* ensemble (Definition 3). Let $k\in\mathbb{N}_+$. An ϵ-precise approximation to the evolution operator $U_L=e^{-iLt}$ can be implemented with success probability $\geq 1-\xi$ using

$$\widetilde{O}\left(5^k t\left(\alpha\log\left(\frac{\mu'}{\epsilon\xi}\right)+\mu'\left(\frac{\mu' t}{\epsilon}\right)^{1/2k}\log\left(\frac{1}{\xi}\right)\right)\right)$$

queries to an $(\alpha,-,\epsilon/5^k t)$ block encoding of the classical Liouvillian $L_{\rm class}$, where $\alpha\in\{\alpha_{NVE},\alpha_{NVT}\}$ and $\mu'\in\{\mu'_{NVE}(2k),\mu'_{NVT}(2k)\}$ is an upper bound on the spectral norm of the nested commutator of $L_{\rm class}$ and $L_{\rm el}$ as given in Definition 14. An additional

$$\widetilde{O}\left(5^k Nd_e\frac{\lambda}{\gamma\delta}\frac{(\mu' t)^{1+1/(2k)}}{\epsilon^{1/(2k)}}\log\left(\frac{1}{\xi}\right)\right)$$

queries to a Hermitian $(\lambda,-,h_x h_p\epsilon/5^k 36Nd_e t)$ block encoding of the electronic Hamiltonian $H_{\rm el}$ are needed. Lastly, we require

$$\widetilde{O}\left(\frac{5^k Nd_e}{\delta}\frac{(\mu' t)^{1+1/(2k)}}{\epsilon^{1/(2k)}}\log\left(\frac{1}{\xi}\right)\right)$$

queries to the initial electronic state-preparation oracle U_I from Definition 9.

To prove Lemma 15, we first need to discuss the complexity of quantum simulation via a higher-order Trotter product formula.

Lemma 16 (Trotter error with commutator scaling ([34], Theorem 6, Corollary 7)). Let $L=\sum_{\gamma=1}^{\Gamma}L_\gamma$ be an operator consisting of Γ Hermitian summands and $t\geq 0$. Let

$$\mathcal{S}_\ell(t)=\prod_{\upsilon=1}^{\Upsilon}\prod_{\gamma=1}^{\Gamma}e^{-ia_{(\upsilon,\gamma)}L_{\pi_\upsilon(\gamma)}t} \tag{C1}$$

be an ℓth-order product formula with $\ell\in\mathbb{N}_+$. Define

$$\tilde{\alpha}_c(\ell):=\sum_{\gamma_1,\gamma_2,\ldots,\gamma_{\ell+1}}^{\Gamma}\left\|[L_{\gamma_{\ell+1}},\ldots[L_{\gamma_2},L_{\gamma_1}]\ldots]\right\|. \tag{C2}$$

Then, the additive Trotter error, defined by $\mathcal{S}(t)=e^{-iLt}+\mathcal{A}(t)$, can be asymptotically bounded as

$$\|\mathcal{A}(t)\|\in O(\tilde{\alpha}_c t^{\ell+1}). \tag{C3}$$

We have $\left\|\mathcal{S}_\ell^r(t/r)-e^{-iLt}\right\|\in O(\epsilon)$ if

$$r\in O\left(\frac{\tilde{\alpha}_c^{1/\ell}t^{1+1/\ell}}{\epsilon^{1/\ell}}\right). \tag{C4}$$

Note that $\tilde{\alpha}_c(\ell)\leq\left(2\sum_{\gamma=1}^{\Gamma}\left\|L_\gamma\right\|\right)^{\ell+1}$. We can establish a tighter bound, as shown below.

The following definition will simplify the subsequent discussion of bounding $\tilde{\alpha}_c(\ell)$ for Liouvillian simulations in the Born-Oppenheimer approximation.

Definition 13 (Terms of the Liouvillian). Let L be the discrete Liouvillian operator either in the *NVE* ensemble (Definition 2) or the *NVT* ensemble (Definition 3). Then, we define

$$K_{n,j}^{(NVE)} := \frac{\partial H_{\text{class}}^{(NVE)}}{\partial p_{n,j}} D_{x_{n,j}} = \sum_{\bar{p}_{n,j}} D_{x_{n,j}} \otimes \frac{p_{n,j}}{m_n} |\bar{p}_{n,j}\rangle\langle\bar{p}_{n,j}|, \tag{C5}$$

$$K_{n,j}^{(NVT)} := \frac{\partial H_{\text{class}}^{(NVT)}}{\partial p_{n,j}} D_{x_{n,j}} = \sum_{\bar{p}_{n,j}} \sum_{\bar{s}} D_{x_{n,j}} \otimes \frac{p_{n,j}}{m_n (s + s_{\min})^2} |\bar{p}_{n,j}\rangle\langle\bar{p}_{n,j}| \otimes |\bar{s}\rangle\langle\bar{s}|, \tag{C6}$$

$$V_{n,n',j}^{\text{class}} := \left(\frac{\partial H_{\text{class}}^{(NVE/NVT)}}{\partial x_{n,j}} \right)_{n'} D_{p_{n,j}} = \sum_{\bar{x}_n} \sum_{\bar{x}_{n'}} \frac{Z_n Z_{n'} (x_{n,j} - x_{n',j})}{\left(\|x_n - x_{n'}\|^2 + \Delta^2 \right)^{3/2}} \times |\bar{x}_n\rangle\langle\bar{x}_n| \otimes |\bar{x}_{n'}\rangle\langle\bar{x}_{n'}| \otimes D_{p_{n,j}}, \tag{C7}$$

$$V_{n,j}^{\text{el}} := D_{n,j}^{\text{el}} \otimes D_{p_{n,j}}^{1}, \tag{C8}$$

$$K^{\text{bath}} := \frac{\partial H_{\text{class}}^{(NVT)}}{\partial_{p_s}} D_s = \sum_{\bar{p}_s} D_s \otimes \frac{p_s}{Q} |\bar{p}_s\rangle\langle\bar{p}_s|, \tag{C9}$$

$$V_{n,j}^{\text{bath}} := \left(\frac{\partial H_{\text{class}}^{(NVT)}}{\partial_s} D_{p_s} \right)_{n,j} = -\sum_{\bar{p}_{n,j}} \sum_{\bar{s}} \frac{2 p_{n,j}^2}{m_n (s + s_{\min})^3} |\bar{p}_{n,j}\rangle\langle\bar{p}_{n,j}| \otimes |\bar{s}\rangle\langle\bar{s}| \otimes D_{p_s}, \tag{C10}$$

$$V_T^{\text{bath}} := \left(\frac{\partial H_{\text{class}}^{(NVT)}}{\partial_s} D_{p_s} \right)_T = \sum_{\bar{s}} \frac{N_f k_B T}{s + s_{\min}} |\bar{s}\rangle\langle\bar{s}| \otimes D_{p_s}. \tag{C11}$$

With these terms defined, we can then compute the spectral norms of each of the Liouvillian terms. These norms are needed to compute the bounds on the Trotter errors, which dominate the scaling given in Theorem 1.

Lemma 17 (Spectral norm of Liouvillian terms). The spectral norm of the Liouvillian terms from Definition 13 can be upper bounded as follows:

$$\left\| K_{n,j}^{(NVE)} \right\| \leq \frac{p_{\max}}{m_{\min}} \frac{2 (\ln d_x + 1)}{h_x}, \tag{C12}$$

$$\left\| K_{n,j}^{(NVT)} \right\| \leq \frac{p_{\max}}{m_{\min} s_{\min}^2} \frac{2 (\ln d_x + 1)}{h_x}, \tag{C13}$$

$$\left\| V_{n,n',j}^{\text{class}} \right\| \leq \frac{2 Z_{\max}^2 x_{\max}}{\Delta^3} \frac{2 \left(\ln d_p + 1 \right)}{h_p}, \tag{C14}$$

$$\left\| V_{n,j}^{\text{el}} \right\| \leq \lambda \frac{2 (\ln d_e + 1)}{h_x h_p}, \tag{C15}$$

$$\left\| K^{\text{bath}} \right\| \leq \frac{p_{s,\max}}{Q} \frac{2 (\ln d_s + 1)}{h_s}, \tag{C16}$$

$$\left\| V_{n,j}^{\text{bath}} \right\| \leq \frac{2 p_{\max}^2}{m_{\min} s_{\min}^3} \frac{2 \left(\ln d_{p_s} + 1 \right)}{h_{p_s}}, \tag{C17}$$

$$\left\| V_T^{\text{bath}} \right\| \leq \frac{N_f k_B T}{s_{\min}} \frac{2 \left(\ln d_{p_s} + 1 \right)}{h_{p_s}}. \tag{C18}$$

Proof. First, recall that for any two matrices A and B, it holds that $\|A \otimes B\| = \|A\| \, \|B\|$. Next, note that the above terms are all of the form $A_{\text{diag}} \otimes D$, where A_{diag} is a diagonal matrix and D is a central-finite-difference matrix.

Specifically, consider first the quantity $\|D_{x_{n,j}}\|$. We have from Definition 1 that

$$\|D_{x_{n,j}}\| = \frac{1}{h_x} \| \sum_{\bar{x}} \sum_{k=-d}^{d} c_{d,k} |\bar{x} - k\rangle\langle\bar{x}| \| \leq \frac{\sum_{k=-d}^{d} |c_{d,k}|}{h_x}. \tag{C19}$$

We then have from Lemma 9 that

$$\|D_{x_{n,j}}\| \leq \frac{2 (\ln d + 1)}{h_x}. \tag{C20}$$

As $m_n \geq m_{\min}$ and $|p_{n,j}| \leq p_{\max}$, we then have that

$$\left\| \sum_{\bar{p}_{n,j}} D_{x_{n,j}} \otimes \frac{p_{n,j}}{m_n} |\bar{p}_{n,j}\rangle\langle\bar{p}_{n,j}| \right\| \leq \frac{2 p_{\max} (\ln d_x + 1)}{m_{\min} h_x}. \tag{C21}$$

This validates the claim in Eq. (C11). The claim of Eq. (C12) immediately follows from the same reasoning and the fact that $s \geq s_{\min}$.

The result of Eq. (C13) also follows from the above bound on $\|D\|$ and the fact that

$$\left\| \sum_{\bar{x}_n} \sum_{\bar{x}_{n'}} \frac{Z_n Z_{n'} (x_{n,j} - x_{n',j})}{\left(\|x_n - x_{n'}\|^2 + \Delta^2 \right)^{3/2}} |\bar{x}_n\rangle\langle\bar{x}_n| \otimes |\bar{x}_{n'}\rangle\langle\bar{x}_{n'}| \right\| \leq \frac{Z_{\max}^2 \max |x_{n,j} - x_{n',j}|}{\min \left(\|x_n - x_{n'}\|^2 + \Delta^2 \right)^{3/2}} \leq \frac{2 Z_{\max}^2 x_{\max}}{\Delta^3}. \tag{C22}$$

The momentum derivative expression is exactly the same as previous, except that the grid spacing is h_p rather than

h_x. Putting this observation together with Eq. (C21) yields

$$\left\| V^{\text{class}}_{n,n',j} \right\| \leq \frac{2Z^2_{\max} x_{\max}}{\Delta^3} \|D_{p,n_j}\| \leq \frac{2Z^2_{\max} x_{\max}}{\Delta^3} \frac{2\left(\ln d_p + 1\right)}{h_p}. \tag{C23}$$

Next, note that D^1 is defined to be the centered-difference formula, which has a coefficient sum of 1. This observation then yields

$$\begin{aligned} \|D^{\text{el}}_{n,j}\| &= \frac{1}{h_x} \| \sum_{k=-d_e}^{d_e} \sum_{(n',j')\neq(n,j)} \sum_{\bar{x}_{n',j'}} \sum_{\bar{x}_{n,j}} c_{d_e,k} \\ &\quad \times E_{\text{el}}\left(\{x_{n',j'}\}, x_{n,j} + k h_x\right) |\bar{x}_{n',j'}\rangle\langle\bar{x}_{n',j'}| \\ &\quad \otimes |\bar{x}_{n,j}\rangle\langle\bar{x}_{n,j}| \| \\ &\leq \frac{\max(E_{\text{el}}) \sum_k |c_{d_e,k}|}{h_x} \leq \frac{\lambda}{h_x}. \end{aligned} \tag{C24}$$

Next, using the bound of Eq. (C19) with the substitution of $x \to p$, we find that

$$\left\| V^{\text{el}}_{n,j} \right\| \leq \lambda \frac{2(\ln d_e + 1)}{h_x h_p}. \tag{C25}$$

The remaining bounds then follow precisely from the above bound techniques for D and noting the minimum values of s and maximum values of p. ■

Definition 14 (Commutator spectral norm of the Liouvillian). Let L be the discretized Liouvillian in the *NVE* or *NVT* ensemble and let $\ell \in \mathbb{N}_+$. Then, we define

$$\mu'_{NVE}(\ell) := 3N \left\| K^{(NVE)}_{n,j} \right\| + 6N\ell \left\| V^{\text{class}}_{n,n',j} \right\| + 3N \left\| V^{\text{el}}_{n,j} \right\| \tag{C26}$$

and

$$\begin{aligned} \mu'_{NVT}(\ell) := 3N \left\| K^{(NVT)}_{n,j} \right\| + 6N\ell \left\| V^{\text{class}}_{n,n',j} \right\| + 3N \left\| V^{\text{el}}_{n,j} \right\| \\ + \left\| K^{\text{bath}} \right\| + \ell \left\| V^{\text{bath}}_{n,j} \right\| + \left\| V^{\text{bath}}_{T} \right\|. \end{aligned} \tag{C27}$$

Lemma 18 (Upper bound on $\tilde{\alpha}_c(\ell)$ for Liouvillian simulations in the Born-Oppenheimer approximation). Let $\tilde{\alpha}_c(\ell)$ be defined as in Lemma 16. Let $L = L_{\text{class}} + L_{\text{el}}$ be the discrete Liouvillian operator for the *NVE* ensemble (Definition 2). Then, $\tilde{\alpha}_c(\ell)$ associated with approximating e^{-iLt} with an ℓth-order product formula involving $e^{-iL_{\text{class}}t}$ and $e^{-iL_{\text{el}}t}$ is upper bounded as follows:

$$\tilde{\alpha}^{(NVE)}_c(\ell) \leq 2^\ell \left(\mu'_{NVE}\right)^{\ell+1}. \tag{C28}$$

For Liouvillian simulations in the *NVT* ensemble (Definition 3), we have that

$$\tilde{\alpha}^{(NVE)}_c(\ell) \leq 2^\ell \left(\mu'_{NVT}\right)^{\ell+1}. \tag{C29}$$

Proof. We prove Lemma 18 via induction on ℓ. The first-order formula with $\ell = 1$ constitutes the base case. The only nonzero commutator at this level is $[L_{\text{class}}, L_{\text{el}}]$, since $[L_{\text{class}}, L_{\text{class}}] = [L_{\text{el}}, L_{\text{el}}] = 0$. Note that all $V^{\text{class}}_{n,n',j}$ terms, K^{bath} and V^{bath}_T commute with all terms $V^{\text{el}}_{n,j}$ of L_{el}. Hence, we only need to upper bound commutators of the following types: $\left[K^{(NVE/NVT)}_{n,j}, V^{\text{el}}_{n',j'}\right]$ and $\left[V^{\text{bath}}_{n,j}, V^{\text{el}}_{n',j'}\right]$. Generally, $\left[K^{(NVE/NVT)}_{n,j}, V^{\text{el}}_{n',j'}\right] \neq 0$, since $D^{\text{el}}_{n,j}$ depends on all nuclear-position variables $\{x_n\}$. In particular, $\left[D_{x_{n,j}}, D^{\text{el}}_{n',j'}\right] \neq 0$ in general. We upper bound these commutators by the product of the norms of the individual operators, resulting in

$$\left\| \left[K^{(NVE/NVT)}_{n,j}, V^{\text{el}}_{n',j'}\right] \right\| \leq 2 \left\| K^{(NVE/NVT)}_{n,j} \right\| \left\| V^{\text{el}}_{n',j'} \right\|. \tag{C30}$$

There are a total of $3N \times 3N = 9N^2$ commutators of the above form.

On the other hand, $\left[V^{\text{bath}}_{n,j}, V^{\text{el}}_{n',j'}\right]$ can only be nonzero if $n = n'$ and $j = j'$. In that case, we obtain

$$\left\| \left[V^{\text{bath}}_{n,j}, V^{\text{el}}_{n,j}\right] \right\| \leq 2 \left\| V^{\text{bath}}_{n,j} \right\| \left\| V^{\text{el}}_{n,j} \right\|. \tag{C31}$$

There are a total of $3N$ commutators of the above form. For the *NVE* ensemble, we therefore have that, for $\ell = 1$,

$$\begin{aligned} \tilde{\alpha}^{(NVE)}_c(1) &\leq 18N^2 \left\| K^{(NVE)}_{n,j} \right\| \left\| V^{\text{el}}_{n,j} \right\| \\ &\leq 2\left(3N \left\| K^{(NVE)}_{n,j} \right\| + 6N\ell \left\| V^{\text{class}}_{n,n',j} \right\| \right. \\ &\quad \left. + 3N \left\| V^{\text{el}}_{n,j} \right\| \right)^2. \end{aligned} \tag{C32}$$

Similarly, we find, for $\ell = 1$, the following for the *NVT* ensemble:

$$\begin{aligned} &\tilde{\alpha}^{(NVT)}_c(1) \\ &\quad \leq 18N^2 \left\| K^{(NVT)}_{n,j} \right\| \left\| V^{\text{el}}_{n,j} \right\| + 6N \left\| V^{\text{bath}}_{n,j} \right\| \left\| V^{\text{el}}_{n,j} \right\| \\ &\quad \leq 2\left(3N \left\| K^{(NVT)}_{n,j} \right\| + 6N\ell \left\| V^{\text{class}}_{n,n',j} \right\| + 3N \left\| V^{\text{el}}_{n,j} \right\| \right. \\ &\qquad \left. + \left\| K^{\text{bath}} \right\| + \ell \left\| V^{\text{bath}}_{n,j} \right\| + \left\| V^{\text{bath}}_{T} \right\| \right)^2. \end{aligned} \tag{C33}$$

This establishes the $\ell = 1$ base case.

Let us now discuss the induction step for the *NVE* ensemble. By assumption, assume that there exists a value

of $\ell \geq 1$ such that

$$\tilde{\alpha}_c^{(NVE)}(\ell) \leq 2^\ell \left(3N \left\| K_{n,j}^{(NVE)} \right\| + 6N\ell \left\| V_{n,n',j}^{\text{class}} \right\| + 3N \left\| V_{n,j}^{\text{el}} \right\|\right)^{\ell+1}. \tag{C34}$$

A single summand of the nested commutator of $\tilde{\alpha}_c^{(NVE)}(\ell)$ is then a string $S_{\ell+1}$ of $\ell+1$ operators $O_a \in \left\{K_{n,j}^{(NVE)}, V_{n,n',j}^{\text{class}}, V_{n,j}^{\text{el}}\right\}$, where $a \in \{1,2,\ldots,\ell+1\}$. Importantly, only certain combinations of operators yield "nontrivial" strings, i.e., strings that have a nonzero contribution to the nested commutator. An example of a trivial string would be a string containing the same operator $\ell+1$ times. An alternating sequence of two noncommuting operators, such as $K_{n,j}^{(NVE)}$ and $V_{n,j}^{\text{el}}$, would be an example of a nontrivial string.

For the induction step, we now add an $(\ell+2)$th operator $O_{\ell+2}$ to $S_{\ell+1}$ to construct nontrivial strings of length $\ell+2$ ($S_{\ell+2}$) as needed for an $(\ell+1)$th-order product formula. First, let us try adding a $K_{n,j}^{(NVE)}$ term to some fixed nontrivial string $S_{\ell+1}$. If $S_{\ell+1}$ contains at least one $V_{n,j}^{\text{el}}$ term, then the resulting string $S_{\ell+2}$ will be nontrivial. Given some fixed $V_{n,j}^{\text{el}}$, there are $3N$ choices for $K_{n,j}^{(NVE)}$ to create a nontrivial string $S_{\ell+2}$ from $S_{\ell+1}$.

Next, let us try adding a $V_{n,n',j}^{\text{class}}$ term to some fixed nontrivial string $S_{\ell+1}$. In the worst case, $S_{\ell+1}$ contains up to $\ell+1$ different $K_{n,j}^{(NVE)}$ terms. Given some fixed $K_{n,j}^{(NVE)}$, there are $6(N-1)$ different $V_{n,n',j}^{\text{class}}$ terms that would yield a nonzero commutator. Hence, for any given string $S_{\ell+1}$, there are at most $6N(\ell+1)$ possibilities to create a nontrivial string $S_{\ell+2}$ via addition of a $V_{n,n',j}^{\text{class}}$ term.

Lastly, let us try adding a $V_{n,j}^{\text{el}}$ term to some fixed nontrivial string $S_{\ell+1}$. If $S_{\ell+1}$ contain at least one $K_{n,j}^{(NVE)}$ term, then the resulting string $S_{\ell+1}$ will be nontrivial. Hence, there are $3N$ choices for $V_{n,j}^{\text{el}}$ to create a nontrivial string $S_{\ell+2}$ from $S_{\ell+1}$.

Putting everything together, we therefore obtain the following recursion:

$$\tilde{\alpha}_c^{(NVE)}(\ell+1) \leq 2\left(3N \left\| K_{n,j}^{(NVE)} \right\| + 6N(\ell+1) \left\| V_{n,n',j}^{\text{class}} \right\| + 3N \left\| V_{n,j}^{\text{el}} \right\|\right) \tilde{\alpha}_c^{(NVE)}(\ell). \tag{C35}$$

Using the hypothesis [Eq. (C33)], we arrive at

$$\tilde{\alpha}_c^{(NVE)}(\ell+1) \leq 2^{\ell+1}\left(3N \left\| K_{n,j}^{(NVE)} \right\| + 6N(\ell+1) \left\| V_{n,n',j}^{\text{class}} \right\| + 3N \left\| V_{n,j}^{\text{el}} \right\|\right)^{\ell+2}, \tag{C36}$$

as desired. This demonstrates the inductive step and our proof then follows trivially by induction, using the fact that the base case of $\ell = 1$ has already been demonstrated in Eq. (C31).

Let us now turn to the induction step for the *NVT* ensemble. We use the same strategy as for the *NVE* ensemble. As an induction hypothesis, assume that there exists $\ell \geq 1$ such that

$$\tilde{\alpha}_c^{(NVT)}(\ell) \leq 2^\ell \left(3N \left\| K_{n,j}^{(NVT)} \right\| + 6N\ell \left\| V_{n,n',j}^{\text{class}} \right\| + 3N \left\| V_{n,j}^{\text{el}} \right\| + \left\| K^{\text{bath}} \right\| + \ell \left\| V_{n,j}^{\text{bath}} \right\| + \left\| V_T^{\text{bath}} \right\|\right)^{\ell+1}. \tag{C37}$$

A single summand of the nested commutator of $\tilde{\alpha}_c^{(NVT)}(\ell)$ is now a string $S_{\ell+1}$ of $\ell+1$ operators $O_a \in \left\{K_{n,j}^{(NVT)}, V_{n,n',j}^{\text{class}}, V_{n,j}^{\text{el}}, K^{\text{bath}}, V_{n,j}^{\text{bath}}, V_T^{\text{bath}}\right\}$, where $a \in \{1,2,\ldots,\ell+1\}$.

For the induction step, we now add an $(\ell+2)$th operator $O_{\ell+2}$ to $S_{\ell+1}$ to construct nontrivial strings of length $\ell+2$ ($S_{\ell+2}$) as needed for an $(\ell+1)$th-order product formula. First, let us try adding a $K_{n,j}^{(NVT)}$ term to some fixed nontrivial string $S_{\ell+1}$. If $S_{\ell+1}$ contains at least one $V_{n,j}^{\text{el}}$ term, then the resulting string $S_{\ell+2}$ will be nontrivial. Given some fixed $V_{n,j}^{\text{el}}$, there are $3N$ choices for $K_{n,j}^{(NVT)}$ to create a nontrivial string $S_{\ell+2}$ from $S_{\ell+1}$.

Next, let us try adding a $V_{n,n',j}^{\text{class}}$ term to some fixed nontrivial string $S_{\ell+1}$. In the worst case, $S_{\ell+1}$ contains up to $\ell+1$ different $K_{n,j}^{(NVT)}$ terms. Given some fixed $K_{n,j}^{(NVT)}$ there are $6(N-1)$ different $V_{n,n',j}^{\text{class}}$ terms that would yield a nonzero commutator. Hence, for any given string $S_{\ell+1}$, there are at most $6N(\ell+1)$ possibilities to create a nontrivial string $S_{\ell+2}$ via addition of a $V_{n,n',j}^{\text{class}}$ term.

Let us now try adding a $V_{n,j}^{\text{el}}$ term to some fixed nontrivial string $S_{\ell+1}$. If $S_{\ell+1}$ contain at least one $K_{n,j}^{(NVT)}$ term, then the resulting string $S_{\ell+1}$ will be nontrivial. Hence, there are $3N$ choices for $V_{n,j}^{\text{el}}$ to create a nontrivial string $S_{\ell+2}$ from $S_{\ell+1}$.

The K^{bath} term does not commute with $K_{n,j}^{(NVT)}$, $V_{n,j}^{\text{bath}}$ or V_T^{bath}. Hence, we can create a nontrivial string $S_{\ell+2}$ by adding K^{bath} to $S_{\ell+1}$ if $S_{\ell+1}$ contains $K_{n,j}^{(NVT)}$, $V_{n,j}^{\text{bath}}$ or V_T^{bath}.

Next, let us try adding a $V_{n,j}^{\text{bath}}$ term to some fixed nontrivial string $S_{\ell+1}$. In the worst case, $S_{\ell+1}$ contains up to $\ell+1$ different $K_{n,j}^{(NVT)}$ terms. Hence, for any given string $S_{\ell+1}$ there are at most $\ell+1$ possibilities to create a nontrivial string $S_{\ell+2}$ via addition of a $V_{n,j}^{\text{bath}}$ term.

Lastly, let us try adding the V_T^{bath} term to some fixed nontrivial string $S_{\ell+1}$. We can create a nontrivial string $S_{\ell+2}$ if $S_{\ell+1}$ contains the K^{bath} term.

Putting everything together, we therefore obtain the following recursion:

$$\tilde{\alpha}_c^{(NVT)}(\ell+1) \leq 2\Big(3N\left\|K_{n,j}^{(NVT)}\right\| + 6N(\ell+1)\left\|V_{n,n',j}^{\text{class}}\right\| + 3N\left\|V_{n,j}^{\text{el}}\right\| + \left\|K^{\text{bath}}\right\| + (\ell+1)\left\|V_{n,j}^{\text{bath}}\right\| + \left\|V_T^{\text{bath}}\right\|\Big)\tilde{\alpha}_c^{(NVT)}(\ell). \tag{C38}$$

Using the hypothesis [Eq. (C36)], we arrive at

$$\tilde{\alpha}_c^{(NVT)}(\ell+1) \leq 2^{(\ell+1)}\Big(3N\left\|K_{n,j}^{(NVT)}\right\| + 6N(\ell+1)\left\|V_{n,n',j}^{\text{class}}\right\| + 3N\left\|V_{n,j}^{\text{el}}\right\| + \left\|K^{\text{bath}}\right\| + (\ell+1)\left\|V_{n,j}^{\text{bath}}\right\| + \left\|V_T^{\text{bath}}\right\|\Big)^{\ell+2}, \tag{C39}$$

as desired. The bound then immediately follows by induction, given that the $\ell = 1$ base case has already been demonstrated in Eq. (C32) ■

The bounds in Lemma 18 exploit the commutator structure of $L = L_{\text{class}} + L_{\text{el}}$. Not taking the commutator structure into account, one obtains the following bounds on $\tilde{\alpha}_c^{(NVE/NVT)}(\ell)$ (Lemma 1 of Ref. [34]):

$$\tilde{\alpha}_c^{(NVE)}(\ell) \leq 2^{\ell}\left(3N\left\|K_{n,j}^{(NVE)}\right\| + 6N^2\left\|V_{n,n',j}^{\text{class}}\right\| + 3N\left\|V_{n,j}^{\text{el}}\right\|\right)^{\ell+1},$$

$$\tilde{\alpha}_c^{(NVT)}(\ell) \leq 2^{\ell}\left(3N\left\|K_{n,j}^{(NVT)}\right\| + 6N^2\left\|V_{n,n',j}^{\text{class}}\right\| + 3N\left\|V_{n,j}^{\text{el}}\right\| + \left\|K^{\text{bath}}\right\| + 3N\left\|V_{n,j}^{\text{bath}}\right\| + \left\|V_T^{\text{bath}}\right\|\right)^{\ell+1}.$$

The main improvement of Lemma 18 over these bounds lies in the reduction of the coefficients of $\left\|V_{n,n',j}^{\text{class}}\right\|$ and $\left\|V_{n,j}^{\text{bath}}\right\|$ from $6N^2$ to $6N\ell$ and from $3N$ to ℓ, respectively.

While the above results apply to general product formulas, we will only be using $2k$th-order Trotter-Suzuki product formulas for our simulation. Hence, we have $\ell = 2k$ with $k \in \mathbb{N}_+$ in the following discussion.

Before proving Theorem 1, it will also be useful to bound the total evolution time associated with a $2k$th-order product formula. The total evolution time is the sum of the absolute values of the evolution time of each segment for a fixed operator.

Lemma 19 (Total evolution time of a higher-order product formula). Let $t \geq 0$ be the desired evolution time of the simulation. The total evolution time of a $2k$th-order product formula is

$$T_{2k} \leq 5^{k-1}t \in O\left(5^k t\right). \tag{C40}$$

Proof. Recall the following recursive definition of the $2k$th-order product formula $\mathcal{S}_{2k}(t)$ from Definition 7:

$$\mathcal{S}_2(t) := e^{L_1\frac{t}{2}}\cdots e^{L_\Gamma\frac{t}{2}}e^{L_\Gamma\frac{t}{2}}\cdots e^{L_1\frac{t}{2}}, \tag{C41}$$

$$\mathcal{S}_{2k}(t) := \mathcal{S}_{2k-2}^2(u_k t)\mathcal{S}_{2k-2}((1-4u_k)t)\mathcal{S}_{2k-2}^2(u_k t), \tag{C42}$$

where

$$\frac{1}{3} \leq u_k := \frac{1}{\left(4 - 4^{\frac{1}{2k-1}}\right)} \leq \frac{1}{2} \quad \forall k \in \mathbb{N}, k \geq 2. \tag{C43}$$

Hence,

$$T_{2k}(t) = 4T_{2k-2}(u_k t) + |T_{2k-2}((1-4u_k)t)| \leq 5T_{2k-2}(t). \tag{C44}$$

Together with the base case, $T_2 = t$, this implies that

$$T_{2k}(t) \leq 5^{k-1}t. \tag{C45}$$

■

The number of exponentials for a $2k$th-order product formula with Γ summands is given by [42]

$$N_{\text{exp}} = 2(\Gamma-1)\,5^{k-1} + 1. \tag{C46}$$

In our case, $\Gamma = 2$ (L_{class} and L_{el}), so $N_{\text{exp}} = 2 \times 5^{k-1} + 1 \in O(5^k)$.

We are now ready to prove Lemma 15.

Proof of Lemma 15. As explained earlier, the discretized Liouvillian L is split into a classical part, L_{class} (Definition 6), and an electronic part, L_{el} (Definition 5). We then use a $2k$th-order product formula to recombine the two parts. The time evolution is divided into r time steps, resulting in a total number of $O\left(r5^k\right)$ exponentials, with r chosen according to Lemma 16. Each exponential is then implemented using qubitization [56]. By the triangle inequality, we can achieve overall simulation error $\leq \epsilon$ if the error of a single exponential is in $O\left(\epsilon/(r5^k)\right)$. Furthermore, recall that each exponential is simulated using a QSVT-based method, meaning that each exponential comes with a certain failure probability. Invoking the union bound, we can ensure an overall success probability of at least $1-\xi$ if the failure probability of a single exponential is in $O\left(\xi/(r5^k)\right)$. This can be achieved via amplitude amplification at the expense of a multiplicative factor of $\log\left(r5^k/\xi\right)$ to the query complexities. The evolution under the classical Liouvillian L_{class} is simulated using qubitization and the

QSVT (Lemma 4) with a total evolution time as in Lemma 19, resulting in

$$O\left(\log\left(\frac{r5^k}{\xi}\right)\left(\alpha 5^k t + r5^k \log\left(\frac{r5^k}{\epsilon}\right)\right)\right) \tag{C47}$$

queries to the block encoding of L_{class}, where $\alpha = \alpha_{NVE}$ for simulations in the *NVE* ensemble (Lemma 1) and $\alpha = \alpha_{NVT}$ for simulations in the *NVT* ensemble (Lemma 2).

From Lemma 16, we have that $r \in O\left(\tilde{\alpha}_c^{1/2k} t^{1+1/2k}/\epsilon^{1/2k}\right)$ and from Lemma 18, we have that $\tilde{\alpha}_c(2k) \leq 2^{2k}(\mu')^{2k+1}$, where $\mu' \in \{\mu'_{NVE}, \mu'_{NVT}\}$. The number of queries to a block encoding of L_{class} is then in

$$\widetilde{O}\left(5^k t\left(\alpha \log\left(\frac{\mu'}{\epsilon\xi}\right) + \mu'\left(\frac{\mu' t}{\epsilon}\right)^{1/2k} \log\left(\frac{1}{\xi}\right)\right)\right). \tag{C48}$$

The evolution under the electronic Liouvillian L_{el} is simulated according to Lemma 11 with a total evolution time as in Lemma 19, resulting in

$$\begin{aligned} &O\left(5^k N d_e \log\left(\frac{r5^k N d_e}{\xi}\right)\left(\frac{\lambda t}{h_x h_p} + r\log\left(\frac{r5^k N\lambda \ln(d_e) t}{h_x h_p \epsilon}\right)\right.\right. \\ &\quad \left.\left. \times \log\left(\frac{r5^k N d_e \log\left(\frac{r5^k N\lambda \ln(d_e) t}{h_x h_p \epsilon}\right)}{\epsilon}\right)\right) \right. \\ &\quad \left. + r5^k N d_e \frac{\lambda}{\gamma\,\delta} \log\left(\frac{r5^k N d_e}{\delta\epsilon}\right)\log\left(\frac{r5^k N d_e}{\xi}\right)\right) \end{aligned} \tag{C49}$$

queries to a block encoding of H_{el}, where $d_e \in O\left(\log\left(Nr5^k \chi t/h_p\epsilon\right)/\log(1/h_x)\right)$. Furthermore,

$$O\left(\frac{r5^k N d_e}{\delta}\log\left(\frac{r5^k N d_e}{\xi}\right)\right) \tag{C50}$$

queries to the state-preparation oracle U_I are needed.

Using

$$r \in O\left(\frac{\tilde{\alpha}_c^{1/2k} t^{1+1/2k}}{\epsilon^{1/2k}}\right), \tag{C51}$$

$$\tilde{\alpha}_c(2k) \leq 2^{2k}(\mu')^{2k+1}, \tag{C52}$$

$$\frac{\lambda}{h_x h_p} \leq \mu', \tag{C53}$$

we find that

$$\begin{aligned} &\widetilde{O}\left(5^k N d_e t\left(\frac{\lambda}{h_x h_p}\log\left(\frac{\tilde{\alpha}_c}{\epsilon\xi}\right) + \left(\frac{\tilde{\alpha}_c t}{\epsilon}\right)^{1/(2k)}\right.\right. \\ &\qquad \left.\left. \times \left(\log\left(\frac{\lambda}{h_x h_p}\right) + \frac{\lambda}{\gamma\delta}\right)\log\left(\frac{1}{\xi}\right)\right)\right) \\ &\quad \subseteq \widetilde{O}\left(5^k N d_e \frac{\lambda}{\gamma\delta}\frac{(\mu' t)^{1+1/(2k)}}{\epsilon^{1/(2k)}}\log\left(\frac{1}{\xi}\right)\right) \end{aligned} \tag{C54}$$

queries to the block encoding of H_{el} and

$$\widetilde{O}\left(\frac{5^k N d_e}{\delta}\frac{(\mu' t)^{1+1/(2k)}}{\epsilon^{1/(2k)}}\log\left(\frac{1}{\xi}\right)\right) \tag{C55}$$

queries to the state-preparation oracle U_I are sufficient. ■

1. Proof of Theorem 1

The previous results now give us the tools that we need to prove Theorem 1, which provides upper bounds on the Toffoli complexity of simulating Liouvillian dynamics. We restate it here for convenience.

Theorem 1 (Complexity of Born-Oppenheimer Liouvillian simulation). There exists a quantum algorithm that solves Problem 1 with success probability $\geq 1 - \xi$ using

$$\widetilde{O}\left(\frac{N_{\text{tot}}\, d\mu^{2+o(1)} t^{1+o(1)}}{\widetilde{\gamma}\,\widetilde{\delta}\,\epsilon^{o(1)}}\log\left(\frac{1}{\xi}\right)\right)$$

Toffoli gates, where d is the maximum order of the finite-difference schemes used, $\mu \in \{\mu_{NVE}, \mu_{NVT}\}$ is an upper bound on the spectral norm of the discretized Liouvillian $L \in \{L_{NVE}, L_{NVT}\}$, and $\widetilde{\gamma}$ is a lower bound on the spectral gap of the discretized electronic Hamiltonian over all phase-space grid points that are associated with a nonzero amplitude at some point during the simulation. Additionally,

$$\widetilde{O}\left(\frac{N d\mu^{1+o(1)} t^{1+o(1)}}{\widetilde{\delta}\,\epsilon^{o(1)}}\log\left(\frac{1}{\xi}\right)\right)$$

queries to the initial electronic state-preparation oracle $\widetilde{U}_I$ are needed.

Proof. The upper bound on the number of queries to $\widetilde{U}_I$ follows from Lemma 15. Note that Lemma 15 deals with errors between operators, i.e., the complexity bounds hold for the worst-case input state. Problem 1, on the other hand, is formulated in terms of the simulation error for a fixed input state. This means that here we only need a good initial electronic state for grid points associated with a nonzero amplitude at some point during the simulation, because any simulation errors that occur on grid points that

are associated with zero amplitude throughout the simulation do not contribute to the error of the final state. Hence, we can use $\widetilde{U}_I$ instead of U_I.

Realizing that $\mu' \leq \mu$ and choosing

$$k = \text{round}\left[\sqrt{\frac{1}{2}\log_5(\mu t/\epsilon) + 1}\right], \tag{C56}$$

we then find that the query complexity for $\widetilde{U}_I$ is in

$$\widetilde{O}\left(5^{\sqrt{2\log_5(\mu t/\epsilon)}} \frac{N d_e \mu t}{\widetilde{\delta}} \log\left(\frac{1}{\xi}\right)\right) \subseteq \widetilde{O}\left(\frac{N d_e \mu^{1+o(1)} t^{1+o(1)}}{\widetilde{\delta}\,\epsilon^{o(1)}} \log\left(\frac{1}{\xi}\right)\right), \tag{C57}$$

which is subpolynomial in $1/\epsilon$ and almost linear in μ and t. The above heuristic choice for k can be obtained by balancing the factor 5^k with $(\mu t/\epsilon)^{1/2k}$ to minimize the overall complexity with respect to k [42].

Next, let us discuss the overall Toffoli complexity, which follows from multiplying the query complexities of the block encodings with their respective Toffoli complexity. More specifically, from Lemma 15, we have that

$$\widetilde{O}\left(5^k t\left(\alpha \log\left(\frac{\mu'}{\epsilon\xi}\right) + \mu'\left(\frac{\mu' t}{\epsilon}\right)^{1/2k} \log\left(\frac{1}{\xi}\right)\right)\right) \subseteq \widetilde{O}\left(5^k \frac{\mu^{1+1/2k} t^{1+1/2k}}{\epsilon^{1/2k}} \log\left(\frac{1}{\xi}\right)\right) \tag{C58}$$

queries to an $\epsilon/(5^k t)$-precise block encoding of the classical Liouvillian L_{class} are sufficient. Lemmas 1 and 2 imply that such a block encoding requires

$$\widetilde{O}\left(N \log\left(\frac{g\alpha 5^k t}{\epsilon}\right) + \log^{\log 3}\left(\frac{\alpha 5^k t}{\epsilon}\right) + d\log(g)\right) \subseteq \widetilde{O}\left((N+d)\log^{\log 3}\left(\frac{\mu 5^k t}{\epsilon}\right)\right) \tag{C59}$$

Toffoli gates, where d is the maximum order of the finite-difference schemes. Choosing k as in Eq. (C55), we find that the Toffoli complexity associated with simulating the classical Liouvillian is in

$$\widetilde{O}\left((N+d)\frac{\mu^{1+o(1)} t^{1+o(1)}}{\epsilon^{o(1)}} \log\left(\frac{1}{\xi}\right)\right). \tag{C60}$$

According to Lemma 15, we also need

$$\widetilde{O}\left(5^k N d_e \frac{\lambda}{\gamma\delta} \frac{(\mu' t)^{1+1/(2k)}}{\epsilon^{1/(2k)}} \log\left(\frac{1}{\xi}\right)\right) \subseteq \widetilde{O}\left(5^k d_e \frac{\mu^{2+1/2k} t^{1+1/2k}}{\gamma\,\delta\,\epsilon^{1/2k}} \log\left(\frac{1}{\xi}\right)\right) \subseteq \widetilde{O}\left(d_e \frac{\mu^{2+o(1)} t^{1+o(1)}}{\gamma\,\delta\,\epsilon^{o(1)}} \log\left(\frac{1}{\xi}\right)\right) \tag{C61}$$

queries to an $h_x h_p \epsilon/5^k 36 N d_e t$-precise block encoding of the electronic Hamiltonian H_{el}. Lemma 3 implies that such a block encoding has Toffoli cost in

$$O\left(N + \tilde{N} + \log\left(\frac{B 5^k 36 N d_e t}{h_x h_p \epsilon}\right)\right). \tag{C62}$$

The Toffoli cost associated with simulating the electronic Liouvillian is then in

$$\widetilde{O}\left(N_{\text{tot}} d_e \frac{\mu^{2+o(1)} t^{1+o(1)}}{\gamma\,\delta\,\epsilon^{o(1)}} \log\left(\frac{1}{\xi}\right)\right). \tag{C63}$$

Combining all of the results, we find that the overall Toffoli complexity of simulating e^{-iLt} is in

$$\widetilde{O}\left(\frac{N_{\text{tot}} d \mu^{2+o(1)} t^{1+o(1)}}{\gamma\,\delta\,\epsilon^{o(1)}} \log\left(\frac{1}{\xi}\right)\right). \tag{C64}$$

Note that the above statements are independent of the initial quantum state encoding the initial phase-space density. In particular, γ is a lower bound on the spectral gap of the block-encoded operator $\widetilde{H}_{\text{el}}(\{x_n\})$ over all phase-space grid points. Similarly, δ is a lower bound on the overlap of the initial electronic state with the true electronic ground state over all phase-space grid points. However, if we are dealing with a fixed initial state as in Problem 1, we only need to consider the spectral gap and the electronic ground state of the grid points associated with a nonzero amplitude at some point during the simulation, because any simulation errors that occur on grid points that are associated with zero amplitude throughout the simulation do not contribute to the error of the final state. Let $\widetilde{\gamma} \geq \gamma$ be a lower bound on the spectral gap of $\widetilde{H}_{\text{el}}(\{x_n\})$ over that subset of grid points. Likewise, let $\widetilde{\delta} \geq \delta$ be a lower bound on the overlap of the initial electronic state with the true electronic ground state over the same subset of grid points. Then, Problem 1 can be solved using only $O\left(1/\widetilde{\gamma}\widetilde{\delta}\right)$ rather than $O\left(1/\gamma\delta\right)$ Toffoli gates. ■

APPENDIX D: DETAILS ON ESTIMATING THE FREE ENERGY

Recall from Definition 4 that the free energy is given by

$$\mathcal{F} = \mathcal{U} - T\mathcal{S}_G, \tag{D1}$$

where $\mathcal{U}$ is the internal energy of the system and $\mathcal{S}_G$ is the Gibbs entropy of the system. The concept of Gibbs entropy can be extended to the quantum world in the form of the von Neumann entropy, as follows.

Definition 15 (Von Neumann entropy). Let $\rho \in \mathbb{C}^{\eta\times\eta}$ be a density matrix. Then,

$$S_N := -\mathrm{Tr}\left(\rho \ln \rho\right) \tag{D2}$$

is the von Neumann entropy associated with ρ.

Note that $S_N = 0$ for a pure state. The idea is to estimate the Gibbs entropy and the internal energy of our system separately and add the results to estimate the free energy. This means that our algorithm requires at least two separate simulations. Before we show how to obtain the Gibbs entropy from a quantum algorithm for estimation of the von Neumann entropy, let us explain the usage of the Nosé thermostat within the Liouvillian framework in a little more detail. The main difference compared to plain Liouvillian dynamics in the *NVE* ensemble is the new variable for the heat bath, s, and its associated momentum variable, p_s. Furthermore, the momenta $\{p'_n\}$ appearing in the *NVT* Hamiltonian H_{NVT} from Definition 23 are virtual momenta of the extended system. They are related to the real momenta $\{p_n\}$ of the physical system via the relation $p_n = p'_n/s$. In the discretized setting, we introduce a cutoff $s_{\min}$ to avoid infinities in the simulation.

In the following, we will drop the particle index of the position and momentum variables for ease of notation. As mentioned in the main text, for continuous variables it can be shown that the microcanonical partition function $\mathcal{Z}$ of the extended system gives rise to the canonical partition function when restricted to the real system [37,39]. More specifically, it can be proven, via a change of variables, that

$$\begin{aligned}\mathcal{Z} &\propto \int d\{x\} \int d\{p'\} \int ds \\ &\quad\times \int dp_s\, \delta\left(H_{NVT}\left(\{x\},\{p'\},s,p_s\right) - E_{\mathrm{ext}}\right) \\ &\propto \int d\{x\} \int d\{p\} e^{-H_{NVE}(\{x\},\{p\})/(k_B T)},\end{aligned} \tag{D3}$$

where E_{ext} is the conserved energy of the extended system and H_{NVE} is the *NVE* Hamiltonian from Eq. (9).

In terms of our quantum algorithm, we now have three types of quantum registers representing classical variables: the nuclear-positions register $|\{\overline{x}\}\rangle$, the nuclear (virtual) momentum register $|\{\overline{p}'\}\rangle$, and the bath register

$$|\overline{S}\rangle := |\overline{s}\rangle|\overline{p}_s\rangle. \tag{D4}$$

Let

$$|\psi_0\rangle := \sum_{\overline{x},\overline{p}',\overline{S}} c_{x,p',S}(0)|\{\overline{x}\}\rangle|\{\overline{p}'\}\rangle|\overline{S}\rangle \tag{D5}$$

be a quantum state encoding the initial KvN wave function of the system plus bath, where the $\{c_{x,p',S}(0)\}$ are complex amplitudes. We time evolve $|\psi_0\rangle$ according to the *NVT* Liouvillian from Definition 3, resulting in

$$|\psi_t\rangle := U_{L_{NVT}}|\psi_0\rangle = \sum_{\overline{x},\overline{p}',\overline{S}} c_{x,p',S}(t)|\{\overline{x}\}\rangle|\{\overline{p}'\}\rangle|\overline{S}\rangle, \tag{D6}$$

where

$$U_{L_{NVT}} := e^{-iL_{NVT}t} \tag{D7}$$

is the unitary that implements the Liouvillian time evolution of the system plus bath. The discrete analogue of integrating out the bath variables as done in Eq. (26) would be to trace out the bath register $|\overline{S}\rangle$. However, at this stage, we cannot simply trace out $|\overline{S}\rangle$ since we would lose all information of $|\overline{s}\rangle$, which is needed to compute the real momenta $\{p\}$. In other words, we first need to perform a discrete analog of the change of variables $p' \to p'/s = p$. We do so by duplicating the $|\overline{s}\rangle$ register via a unitary U_{dup} to obtain

$$\begin{aligned}|\Psi_t\rangle &:= U_{\mathrm{dup}}|\psi_t\rangle|0\rangle = \sum_{\overline{x},\overline{p}',\overline{S}} c_{x,p',S}(t)|\{\overline{x}\}\rangle|\{\overline{p}'\}\rangle|\overline{S}\rangle|\overline{s}\rangle \\ &= \sum_{\overline{x},\overline{p}',\overline{s},\overline{p}_s} c_{x,p',s,p_s}(t)|\{\overline{x}\}\rangle|\{\overline{p}'\}\rangle|\overline{s}\rangle|\overline{p}_s\rangle|\overline{s}\rangle.\end{aligned} \tag{D8}$$

Note that U_{dup} can be implemented using $O(\log g_s)$ CNOT gates. More specifically, we apply a single CNOT to each qubit of the $|\overline{s}\rangle$ register, where each CNOT has a different target qubit in the duplication ancilla register.

The above quantum state $|\Psi_t\rangle$ can be regarded as a purification of the following density matrix, which describes the dynamics of the nuclei under the influence of the heat bath:

$$\begin{aligned}\rho_{\mathrm{sys}}(t) &:= \mathrm{Tr}_S\left(|\Psi_t\rangle\langle\Psi_t|\right) \\ &= \sum_{\substack{\overline{x},\overline{p}',\overline{S}\\ \overline{x}',\overline{p}''}} c_{x,p',S}(t)c^*_{x',p'',S}(t)|\{\overline{x}\}\rangle|\{\overline{p}'\},s\rangle\langle\{\overline{x}'\}|\langle\{\overline{p}''\},\overline{s}| \\ &= \sum_{\substack{\overline{x},\overline{p}',\overline{s},\overline{p}_s\\ \overline{x}',\overline{p}''}} c_{x,p',s,p_s}(t)c^*_{x',p'',s,p_s}(t)|\{\overline{x}\}\rangle|\{\overline{p}'\},\overline{s}\rangle\langle\{\overline{x}'\}| \\ &\qquad\times\langle\{\overline{p}''\},\overline{s}|.\end{aligned} \tag{D9}$$

Note that the combined register $|\{\overline{p}'\},\overline{s}\rangle$ can be regarded as the real momentum register. It is effectively just a different representation of $|\{\overline{p}'/\overline{s}\}\rangle$. The dimension of $\rho_{\rm sys}(t)$ is

$$\eta = g_x^{3N} g_{p'}^{3N} g_s \tag{D10}$$

and the probability of finding the nuclei in a particular configuration $|\{\overline{x}^*,\overline{p}'^*,\overline{s}^*\}\rangle$ is given by

$$\begin{aligned}&\langle\{\overline{x}^*,\overline{p}'^*,\overline{s}^*\}|\rho_{\rm sys}(t)|\{\overline{x}^*,\overline{p}'^*,\overline{s}^*\}\rangle\\ &\quad=\sum_{\overline{S}}|c_{x^*,p'^*,S}(t)|^2=:b_{x^*,p'^*,s^*}(t).\end{aligned} \tag{D11}$$

Using the above ideas, we can reduce the problem of estimating the Gibbs entropy associated with $\rho_{\rm sys}(t)$ to the problem of estimating the von Neumann entropy of a modified density matrix. The reason for requiring a modified density matrix is that in contrast to the von Neumann entropy, the Gibbs entropy associated with $\rho_{\rm sys}(t)$ depends only on the diagonal elements of $\rho_{\rm sys}(t)$, since these represent the classical probabilities of the different microstates [see Eq. (D11)]. We can eliminate the off-diagonal elements by applying controlled phase gradients to the purification $|\Psi_t\rangle$, as we will now explain in more detail.

Let $|j\,\rangle$ denote a computational basis state of a $\log(\eta_v)$-qubit ancilla register, where

$$\eta_v := g_x^{3N} g_{p'}^{3N}. \tag{D12}$$

Furthermore, let $|n\rangle$ denote a η_v-dimensional computational basis state obtained by considering all the nuclear-position and virtual momentum variables as a single register, i.e.,

$$|n\rangle \equiv |\{\overline{x}\}\rangle|\{\overline{p}'\}\rangle. \tag{D13}$$

This change in perspective simplifies the implementation of the controlled phase gradients [64]. Preparation of a uniform superposition over the $|j\,\rangle$ register and applying controlled phase gradients then yields

$$\begin{aligned}\frac{1}{\sqrt{\eta_v}}\sum_j|\Psi_t\rangle|j\,\rangle&=\frac{1}{\sqrt{\eta_v}}\sum_{n,\overline{S},j}c_{n,S}(t)|n\rangle|\overline{S}\rangle|\overline{s}\rangle|j\,\rangle\\ &\xrightarrow{U_{\rm pg}}\frac{1}{\sqrt{\eta_v}}\sum_{n,\overline{s},j}c_{n,S}(t)e^{-2\pi i n\frac{j}{\eta_v}}|n\rangle|\overline{S}\rangle|\overline{s}\rangle|j\,\rangle\\ &=:|\Psi_t'\rangle=:U'_{L_{NVT}}|\psi_0\rangle|0\rangle|0\rangle,\end{aligned} \tag{D14}$$

where $U_{\rm pg}$ denotes the controlled phase-gradient unitary and

$$U'_{L_{NVT}}:=U_{\rm pg}\cdot\left(U_{\rm dup}\otimes\mathbb{1}\right)\cdot\left(U_{L_{NVT}}\otimes\mathbb{1}\otimes\mathbb{1}\right)\in\mathbb{C}^{\eta_{\rm pur}\times\eta_{\rm pur}}, \tag{D15}$$

with

$$\eta_{\rm pur}:=g_x^{6N}g_{p'}^{6N}g_s^2g_{p_s}. \tag{D16}$$

Let us now check that this gives the correct density matrix after tracing out the $|\overline{S}\rangle$ and $|j\,\rangle$ registers. First, note that the purification state $|\Psi_t'\rangle$ can be written in density-matrix notation as

$$\begin{aligned}|\Psi_t'\rangle\langle\Psi_t'|=\frac{1}{\eta_v}\sum_{\substack{n,\overline{S},j\\ n',\overline{S}',j'}}&c_{n,S}(t)c^*_{n',S'}(t)e^{-2\pi i\frac{nj-n'j'}{\eta_v}}\\ &\times|n\rangle|\overline{S}\rangle|\overline{s}\rangle|j\,\rangle\langle n'|\langle\overline{S}'|\langle\overline{s}'|\langle j'|.\end{aligned} \tag{D17}$$

Tracing out the $|j\,\rangle$ register results in

$$\begin{aligned}\mathrm{Tr}_j\left(|\Psi_t'\rangle\langle\Psi_t'|\right)&=\sum_{\substack{n,\overline{S}\\ n',\overline{S}'}}c_{n,S}(t)c^*_{n',S'}(t)\left(\frac{1}{\eta_v}\sum_j e^{-2\pi ij\frac{n-n'}{\eta_v}}\right)\\ &\quad\times|n\rangle|\overline{S}\rangle|\overline{s}\rangle\langle n'|\langle\overline{S}'|\langle\overline{s}'|\\ &=\sum_{\substack{n,\overline{S}\\ n',\overline{S}'}}c_{n,S}(t)c^*_{n',S'}(t)\delta_{n,n'}|n\rangle|\overline{S}\rangle|\overline{s}\rangle\langle n'|\langle\overline{S}'|\langle\overline{s}'|\\ &=\sum_{n,\overline{S},\overline{S}'}c_{n,S}(t)c^*_{n,S'}(t)|n\rangle|\overline{S}\rangle|\overline{s}\rangle\langle n|\langle\overline{S}'|\langle\overline{s}'|\end{aligned} \tag{D18}$$

and tracing out the bath register $|\overline{S}\rangle$ yields

$$\begin{aligned}\rho'_{\rm sys}(t)&:=\sum_{n,\overline{S}}|c_{n,S}(t)|^2|n\rangle|\overline{s}\rangle\langle n|\langle\overline{s}|\\ &=\sum_{\overline{x},\overline{p}',\overline{s},\overline{p}_s}|c_{x,p',s,p_s}(t)|^2|\{\overline{x}\}\rangle|\{\overline{p}'\},\overline{s}\rangle\langle\{\overline{x}\}|\langle\{\overline{p}'\},\overline{s}|,\end{aligned} \tag{D19}$$

which consists only of the diagonal elements of $\rho_{\rm sys}(t)$, as desired. Note that an alternative method of eliminating the off-diagonal elements of $\rho_{\rm sys}(t)$ consists of duplicating the entire nuclear register $|\{\overline{x}\}\rangle|\{\overline{p}'\}\rangle$ and then tracing out the duplicated register, similarly to what was done with the $|\overline{s}\rangle$ register earlier on. In terms of complexity, this duplication approach is essentially equivalent to the phase-gradient approach. Algorithm 4 summarizes the key steps of our free-energy-estimation protocol.

1. Estimation of the Gibbs entropy

Let us now explain how to estimate the Gibbs entropy of our system. First, note that, in general, we cannot implement the Liouvillian-evolution operator $U_{L_{NVT}}$ exactly.

ALGORITHM 4. Free-energy estimation.

Input: Quantum state $|\psi_0\rangle = \sum_{\overline{x},\overline{p}',\overline{S}} c_{x,p',S}(0)|\{\overline{x}\}\rangle|\{\overline{p}'\}\rangle|\overline{S}\rangle$ encoding initial KvN wavefunction of the system together with the heat bath.
Further input parameters:
$t, \epsilon, \xi, k, N, \tilde{N}, \{m_n\}_{n=1}^N, \{Z_n\}_{n=1}^N, x_{\max}, p'_{\max}, h_x, h_{p'}, d_x, d_p, d_e, \Delta, B, h_{\mathrm{el}}, \delta, \widetilde{\gamma}, \chi, N_f, T, Q, s_{\min}, h_s, h_{p_s}, d_s, d_{p_s}$.

Output: With success probability $\geq 1-\xi$ an ϵ-precise estimate of the free energy associated with the phase space density after time t.

1. Apply the NVT Liouvillian simulation algorithm as summarized in Algorithm 1 to $|\psi_0\rangle$ to obtain
$$|\psi_t\rangle = U_{L_{NVT}}|\psi_0\rangle = \sum_{\overline{x},\overline{p}',\overline{S}} c_{x,p',S}(t)|\{\overline{x}\}\rangle|\{\overline{p}'\}\rangle|\overline{S}\rangle.$$
2. Duplicate the $|s\rangle$ register of the bath using U_{dup} to retain the information of s. This yields $|\Psi_t\rangle$ which is a purification of the density matrix $\rho_{\mathrm{sys}}(t)$ for which we want to estimate the von Neumann entropy;
3. Eliminate the off-diagonal elements of $\rho_{\mathrm{sys}}(t)$ via controlled phase gradients between an ancillary register $|j\rangle$ and $|\Psi_t\rangle$;
4. Tracing out the ancillary register $|j\rangle$ and the bath register $|S\rangle$ but not the duplicated $|s\rangle$ register, we obtain a reduced phase space density over the nuclear position and momentum registers $|\{\overline{x}_n\}\rangle|\{\overline{p}_n\}\rangle$;
5. Use the algorithm associated with Theorem 13 of [47] to estimate the Gibbs entropy of $\rho_{sys}(t)$ within error $\epsilon/(2k_BT)$;
6. Apply the Hadamard test to each of the three internal energy components H_{kin}, H_{pot} and $H_{E_{\mathrm{el}}}$ to estimate their expectation values within error $\epsilon/6$;
7. Classically add the estimates of the Gibbs entropy and the internal energy components to obtain an ϵ-precise estimate of the free energy $\mathcal{F}$;

However, Theorem 1 shows that we can efficiently construct an approximation $\widetilde{U}_{L_{NVT}}$ such that

$$\left\|\widetilde{U}_{L_{NVT}} - U_{L_{NVT}}\right\| \leq \epsilon \tag{D20}$$

for any $\epsilon \in (0,1)$. Let us assume that U_{pg} and U_{dup} can be implemented with negligible error. Then, the resulting approximation $\widetilde{U}'_{L_{NVT}}$ of $U'_{L_{NVT}}$ satisfies

$$\left\|\widetilde{U}'_{L_{NVT}} - U'_{L_{NVT}}\right\| \leq \epsilon. \tag{D21}$$

We denote the corresponding approximate density matrix of the system by $\widetilde{\rho}'_{\mathrm{sys}}(t)$. The following inequality will be useful for upper bounding the difference in the von Neumann entropy associated with $\rho'_{\mathrm{sys}}(t)$ and $\widetilde{\rho}'_{\mathrm{sys}}(t)$ in terms of their trace distance.

Definition 16 (Trace distance). Let $\rho, \sigma \in \mathbb{C}^{\eta\times\eta}$ be density matrices. Then, their trace distance is given by

$$\begin{aligned}\mathcal{T}(\rho,\sigma) &:= \frac{1}{2}\|\rho-\sigma\|_1 = \frac{1}{2}\mathrm{Tr}\left(\sqrt{(\rho-\sigma)^\dagger(\rho-\sigma)}\right)\\ &= \frac{1}{2}\sum_{j=1}^{\eta}\left|\lambda_j\right|,\end{aligned} \tag{D22}$$

where $\lambda_j \in \mathbb{R}$ is the jth eigenvalue of $\rho-\sigma$.

In the following, we will use $\|\cdot\|_1$ to refer to the trace norm (i.e., the Schatten 1-norm). As before, $\|\cdot\|$ denotes the (induced) 2-norm.

Lemma 20 (Fannes inequality [65,66]). Let ρ and σ be η-dimensional density matrices. If $\mathcal{T}(\rho,\sigma) \leq 1/(2e)$, then

$$|S_N(\rho) - S_N(\sigma)| \leq 2\mathcal{T}\log_2(\eta) - 2\mathcal{T}\log_2(2\mathcal{T}). \tag{D23}$$

We now show how to estimate the Gibbs entropy $\mathcal{S}_G$ of our system.

Lemma 21 (Estimation of the Gibbs entropy). Let $\epsilon \in (0,1)$ and let $\rho'_{\mathrm{sys}}(t)$ be the η-dimensional diagonal density matrix of the system as defined in Eq. (D19), where $\eta \geq 6$. Let η_{pur} be the dimension of $U'_{L_{NVT}}$ as defined in Eq. (D15). Furthermore, let $\widetilde{U}'_{L_{NVT}}$ be an $\left(\epsilon/4\eta_{\mathrm{pur}}\log(\eta/\nu)\right)$-precise approximation to $U'_{L_{NVT}}$, where $\nu \in (0,1)$ is a lower bound on $2\mathcal{T}\left(\widetilde{\rho}'_{\mathrm{sys}}(t), \rho'_{\mathrm{sys}}(t)\right)$ and

$$\epsilon \leq \frac{2\log(\eta/\nu)}{e}. \tag{D24}$$

There exists a quantum algorithm that outputs an estimate of the Gibbs entropy associated with $\rho_{\mathrm{sys}}(t)$ within error ϵ with success probability $\geq 1-\xi$ using

$$\widetilde{O}\left(\frac{\eta}{\epsilon^{1.5}}\log\left(\frac{1}{\xi}\right)\right) \tag{D25}$$

queries to $\widetilde{U}'_{L_{NVT}}$.

Proof. By construction, estimation of the Gibbs entropy of $\rho_{\text{sys}}(t)$ is equivalent to estimation of the von Neumann entropy of $\rho'_{\text{sys}}(t)$, since $\rho'_{\text{sys}}(t)$ consists only of the diagonal elements of $\rho_{\text{sys}}(t)$. However, we only have access to $\widetilde{U}'_{L_{NVT}}$, an ϵ_U-precise approximation of $U'_{L_{NVT}}$ in ℓ^2-distance. We will show that ϵ_U needs to be upper bounded by

$$\epsilon_U \leq \frac{\epsilon}{4\eta_{\text{pur}} \log(\eta/\nu)}. \tag{D26}$$

This gives rise to an approximation $\widetilde{\rho}'_{\text{sys}}(t)$ of $\rho'_{\text{sys}}(t)$. The idea is then to use Theorem 13 of Ref. [47] to obtain an ϵ_{est}-precise estimate $\widetilde{S}\left(\widetilde{\rho}'_{\text{sys}}(t)\right)$ of the von Neumann entropy of $\widetilde{\rho}'_{\text{sys}}(t)$, where $\epsilon_{\text{est}} \in (0,1)$. The algorithm of Ref. [47] requires access to a purification of the density matrix, which in our case is simply $|\Psi'_t\rangle$. The work of Ref. [35] shows that a polynomial of degree $\widetilde{O}\left(\sqrt{\eta/\epsilon_{\text{est}}}\right)$ to approximate $\log\left(1/\widetilde{b}_{x,p',s}(t)\right)$ within error ϵ_{est}, where $\{\widetilde{b}_{x,p',s}(t)\}$ are the (diagonal) elements of $\widetilde{\rho}'_{\text{sys}}(t)$. This implies that quantum amplitude estimation [67] can be used to learn $\widetilde{S}\left(\widetilde{\rho}'_{\text{sys}}(t)\right)$ with constant success probability within error ϵ_{est} using $\widetilde{O}\left(\eta/\epsilon_{\text{est}}^{1.5}\right)$ queries to $\widetilde{U}'_{L_{NVT}}$. The Chernoff bound implies that we can achieve a success probability $\geq 1-\xi$ with $\log(1/\xi)$ repetitions of the algorithm. Next, let us discuss the required block-encoding precision of $U'_{L_{NVT}}$. By the triangle inequality, we have that

$$\begin{aligned}
&\left|\widetilde{S}\left(\widetilde{\rho}'_{\text{sys}}(t)\right) - S\left(\rho'_{\text{sys}}(t)\right)\right| \\
&\quad\leq \left|\widetilde{S}\left(\widetilde{\rho}'_{\text{sys}}(t)\right) - S\left(\widetilde{\rho}'_{\text{sys}}(t)\right)\right| \\
&\quad\quad + \left|S\left(\widetilde{\rho}'_{\text{sys}}(t)\right) - S\left(\rho'_{\text{sys}}(t)\right)\right| \\
&\quad\leq \epsilon_{\text{est}} + \epsilon_{\text{Fan}},
\end{aligned} \tag{D27}$$

where ϵ_{Fan} is determined by the Fannes inequality (Lemma 20). To achieve overall error $\leq \epsilon$, it suffices to ensure that $\epsilon_{\text{est}} \leq \epsilon/2$ and $\epsilon_{\text{Fan}} \leq \epsilon/2$. Let us now bound ϵ_{Fan} in terms of ϵ_U and ν. For simplicity, let

$$|\Psi_0\rangle := |\psi_0\rangle|0\rangle|0\rangle. \tag{D28}$$

Then, we have that

$$\begin{aligned}
&\left\|\widetilde{U}'_{L_{NVT}}|\Psi_0\rangle\langle\Psi_0|\widetilde{U}'^{\dagger}_{L_{NVT}} - U'_{L_{NVT}}|\Psi_0\rangle\langle\Psi_0|U'^{\dagger}_{L_{NVT}}\right\| \\
&\quad\leq \left\|\widetilde{U}'_{L_{NVT}}|\Psi_0\rangle\langle\Psi_0|\widetilde{U}'^{\dagger}_{L_{NVT}} - \widetilde{U}'_{L_{NVT}}|\Psi_0\rangle\langle\Psi_0|U'^{\dagger}_{L_{NVT}}\right\| \\
&\quad\quad + \left\|\widetilde{U}'_{L_{NVT}}|\Psi_0\rangle\langle\Psi_0|U'^{\dagger}_{L_{NVT}} - U'_{L_{NVT}}|\Psi_0\rangle\langle\Psi_0|U'^{\dagger}_{L_{NVT}}\right\| \\
&\quad\leq \left\|\widetilde{U}'^{\dagger}_{L_{NVT}} - U'^{\dagger}_{L_{NVT}}\right\| + \left\|\widetilde{U}'_{L_{NVT}} - U'_{L_{NVT}}\right\| \\
&\quad\leq 2\epsilon_U.
\end{aligned} \tag{D29}$$

It follows from Definition 16 that

$$\begin{aligned}
&\frac{1}{2}\left\|\widetilde{U}'_{L_{NVT}}|\Psi_0\rangle\langle\Psi_0|\widetilde{U}'^{\dagger}_{L_{NVT}} - U'_{L_{NVT}}|\Psi_0\rangle\langle\Psi_0|U'^{\dagger}_{L_{NVT}}\right\|_1 \\
&\quad\leq \eta_{\text{pur}}\epsilon_U,
\end{aligned} \tag{D30}$$

where η_{pur} is the dimension of $U'_{L_{NVT}}$ (or, equivalently, of $|\Psi_0\rangle$). Since the trace distance is contractive under the partial trace, we obtain the following bound:

$$\mathcal{T}\left(\widetilde{\rho}'_{\text{sys}}(t), \rho'_{\text{sys}}(t)\right) \leq \eta_{\text{pur}}\epsilon_U. \tag{D31}$$

By Lemma 20, we then have that

$$\begin{aligned}
\epsilon_{\text{Fan}} &= \left|S\left(\widetilde{\rho}'_{\text{sys}}(t)\right) - S\left(\rho'_{\text{sys}}(t)\right)\right| \\
&\leq 2\eta_{\text{pur}}\epsilon_U\left(\log(\eta) - \log(\nu)\right)
\end{aligned} \tag{D32}$$

as long as $\mathcal{T}\left(\widetilde{\rho}'_{\text{sys}}(t), \rho'_{\text{sys}}(t)\right) \in [\nu/2, 1/2e]$. If

$$\epsilon_U \leq \frac{\epsilon}{4\eta_{\text{pur}}\log(\eta/\nu)} \leq \frac{1}{2\eta_{\text{pur}}e}, \tag{D33}$$

then $\epsilon_{\text{Fan}} \leq \epsilon/2$, as desired. Note that this requires $\epsilon \leq 2\log(\eta/\nu)/e$. In our case, we always have $\eta \geq 6$, since the phase space is at least six dimensional. This implies that $2\log(\eta/\nu)/e \geq 1$ for any $\nu \in (0,1)$. Demanding $\epsilon \in (0,1)$ is thus a sufficiently restrictive criterion. ■

A challenge facing this algorithm arises from its scaling with the dimension of the space. In general, it scales exponentially with the number of qubits and hence we cannot compute the entropy directly. An alternative approach is to coarse grain the position and momentum variables of the nuclei, e.g., by tracing out the l least significant qubits associated with each position or momentum variable. This effectively reduces the dimension of the density matrices $\rho_{\text{sys}}(t)$ and $\rho'_{\text{sys}}(t)$ from

$$\eta = 2^{3N\left(\log g_x + \log g_{p'}\right) + \log(g_s)} \tag{D34}$$

to

$$\eta' = 2^{3N\left(\log g_x + \log g_{p'} - 2l\right) + \log(g_s)} \tag{D35}$$

and accordingly only $\widetilde{O}\left(\eta'/\epsilon^{1.5}\log(1/\xi)\right)$ queries to $\widetilde{U}'_{L_{NVT}}$ are required. The exact entropy can then be estimated by extrapolating the entropy in the limit where the coarse graining tends to zero.

2. Estimation of the internal energy

Next, let us discuss how to estimate the internal energy $\mathcal{U}$ of our system. First, note that a classical system can

be described by a density matrix ρ and a Hamiltonian H, both of which are diagonal in the computational basis. The internal energy of a classical system can thus be computed as follows:

$$\mathcal{U} = \mathrm{Tr}\,(\rho H)\,. \tag{D36}$$

In our case, we can identify $\rho \equiv \rho'_{\mathrm{sys}}(t)$ and $H \equiv H_{\mathrm{nuc}}$. Recall from Sec. II that

$$H_{\mathrm{nuc}} = H_{\mathrm{kin}} + H_{\mathrm{pot}} + H_{E_{\mathrm{el}}}, \tag{D37}$$

where

$$\begin{aligned} H_{\mathrm{kin}} &= \sum_{n,j}\sum_{\overline{p}'_{n,j}}\sum_{\overline{s}} \frac{p'^{2}_{n,j}}{m_n(s+s_{\mathrm{min}})^2}|\overline{p}'_{n,j}\rangle\langle\overline{p}'_{n,j}| \otimes |\overline{s}\rangle\langle\overline{s}|, \\ H_{\mathrm{pot}} &= \sum_{n'\neq n}\sum_{\overline{x}_n}\sum_{\overline{x}_{n'}} \frac{Z_n Z_{n'}}{\left(\|x_n - x_{n'}\|^2 + \Delta^2\right)^{1/2}}|\overline{x}_n\rangle\langle\overline{x}_n| \\ &\quad \otimes |\overline{x}_{n'}\rangle\langle\overline{x}_{n'}|, \\ H_{E_{\mathrm{el}}} &= \sum_{\{\overline{x}_n\}} E_{\mathrm{el}}\left(\{x_n\}\right)|\{\overline{x}_n\}\rangle\langle\{\overline{x}_n\}|. \end{aligned}$$

The idea is then to block encode each of the three terms of H_{nuc}, use the Hadamard test to estimate the expectation value of each term individually, and then add the results classically.

Note that H_{nuc} is diagonal in the nuclear-position and -momentum basis. Since the block encoding of H_{nuc} will also be diagonal in the nuclear-position and -momentum basis, we technically do not need to worry about getting rid of the off-diagonal elements of $\rho_{\mathrm{sys}}(t)$, since

$$\mathrm{Tr}\left(\rho_{\mathrm{sys}}(t)H_{\mathrm{nuc}}\right) = \mathrm{Tr}\left(\rho'_{\mathrm{sys}}(t)H_{\mathrm{nuc}}\right). \tag{D38}$$

However, we will use $\rho'_{\mathrm{sys}}(t)$ to be consistent with the previous discussion on estimation of the Gibbs entropy.

The following three lemmas provide upper bounds on the cost of block encoding the three terms of H_{nuc}.

Lemma 22 (Block encoding of H_{kin}). There exists an $(\alpha_{\mathrm{kin}}, a_{\mathrm{kin}}, \epsilon)$ block encoding of H_{kin} with normalization constant

$$\alpha_{\mathrm{kin}} \in O\left(N\frac{p'^{2}_{\mathrm{max}}}{m_{\mathrm{min}}s^2_{\mathrm{min}}}\right) \tag{D39}$$

and a number of ancilla qubits

$$a_{\mathrm{kin}} \in O\left(\log\left(\frac{\alpha_{\mathrm{kin}}}{\epsilon}\right)\right) \tag{D40}$$

that can be implemented using

$$O\left(N\log\left(\frac{g_{p'}\alpha_{\mathrm{kin}}}{\epsilon}\right) + \log^{\log 3}\left(\frac{\alpha_{\mathrm{kin}}}{\epsilon}\right)\right) \tag{D41}$$

Toffoli gates.

Proof. The proof of Lemma 22 proceeds along the same lines as the proof of Lemma 1. In particular, we use the alternating-sign trick [28,54] to block encode a single summand of H_{kin} and then use Lemma 7 to combine the block encodings of all $3N$ terms. More specifically, we use the alternating-sign trick to construct $U_{p_{n,j}}$, which provides an $\left(\alpha_p, a_p, \epsilon_p\right)$ block encoding of

$$\sum_{\overline{p}'_{n,j}}\sum_{\overline{s}} \frac{p'^{2}_{n,j}}{(s+s_{\mathrm{min}})^2}|\overline{p}'_{n,j}\rangle\langle\overline{p}'_{n,j}| \otimes |\overline{s}\rangle\langle\overline{s}|. \tag{D42}$$

We then use the following to prepare the distribution of coefficients for the masses of each particle in the system under the assumption of three-dimensional dynamics:

$$\mathrm{PREP}_m|0\rangle := \sum_{n=1}^{N}\sqrt{\frac{1/m_n}{\alpha_m}}|n\rangle \otimes \frac{1}{\sqrt{3}}\sum_{j=1}^{3}|j\rangle, \tag{D43}$$

where

$$\alpha_m = \sum_{n,j}\frac{1}{m_n} \leq \frac{3N}{m_{\mathrm{min}}}. \tag{D44}$$

The above definition implies that $a_m = \lceil \log N \rceil + \lceil \log 3 \rceil$.

Let us now explain how to use the alternating-sign trick to implement $U_{p_{n,j}}$. The idea is to prepare an ancilla register consisting of a_p qubits in the state

$$\mathrm{PREP}_p|0\rangle := \sum_{l=0}^{2^{a_p}-1}\frac{1}{\sqrt{2^{a_p}}}|l\rangle \tag{D45}$$

and then use the inequality-testing circuit from Fig. 3 to test the following inequality:

$$l\,(\overline{s}+\overline{s}_{\mathrm{min}})^2 \leq \overline{p}'^{\,2}_{n,j}, \tag{D46}$$

where $\overline{s}_{\mathrm{min}} \in \mathbb{N}$ such that $s_{\mathrm{min}} = \overline{s}_{\mathrm{min}}h_s$. As long as l satisfies the above inequality, the coefficient of $|l\rangle$ is set to +1, as is done in existing work involving LCU or qubitization for general-purpose simulations [28,54]. For larger l, the coefficient of $|l\rangle$ is set to alternate between ± 1. To test the inequality, we first use $O(1)$ quantum Karatsuba multiplications [59] to compute the left- and right-hand sides of Eq. (D46). This can be done using $O\left(a_p^{\log 3}\right)$ Toffoli gates, whereas the inequality test itself requires only $O\left(a_p\right)$ Toffolis (see Lemma 10). We then have that $U_{p_{n,j}} = \mathrm{PREP}_p^{\dagger}\cdot \mathrm{SEL}_p \cdot \mathrm{PREP}_p$, where SEL_p includes the quantum Karatsuba multiplications, the inequality testing, and a CZ gate to obtain the desired alternating sequence of ± 1. Figure 5 shows a circuit diagram of the alternating-sign trick for the slightly simpler case of block encoding

$\sum_{\bar{p}_{n,j}=0}^{g_p-1} p_{n,j}\,|\bar{p}_{n,j}\rangle\langle\bar{p}_{n,j}|$. The number of ancilla qubits, a_p, determines the precision ϵ_p of $U_{p_{n,j}}$. In particular,

$$\left\| (\langle 0|\otimes\mathbb{1})U_{p_{n,j}}(|0\rangle\otimes\mathbb{1}) - \frac{1}{\alpha_p}\sum_{\bar{p}'_{n,j}}\sum_{\bar{s}}\frac{p'^2_{n,j}}{(s+s_{\min})^2}|\bar{p}'_{n,j}\rangle\langle\bar{p}'_{n,j}|\otimes|\bar{s}\rangle\langle\bar{s}| \right\| \leq \frac{1}{2^{a_p}}. \tag{D47}$$

Note that $\alpha_p \in O\left(p'^2_{\max}/s^2_{\min}\right)$. We can ensure that $(\langle 0|\otimes I)U_{p_{n,j}}(|0\rangle\otimes I)$ is an ϵ_p-precise approximation to $\sum_{\bar{p}'_{n,j}}\sum_{\bar{s}} p'^2_{n,j}/(s+s_{\min})^2\,|\bar{p}'_{n,j}\rangle\langle\bar{p}'_{n,j}|\otimes|\bar{s}\rangle\langle\bar{s}|$ by choosing $a_p \in \Theta\left(\log\left(\alpha_p/\epsilon_p\right)\right)$.

Instead of constructing $3N$ different block encoding for each of the $3N$ terms, we use an additional ancilla register that we call a "SWAP register" [28]. The SEL operation can then be modified to swap the appropriate (virtual) momentum variable into the SWAP register controlled by the PREP_m register. This allows us to apply the block encoding $U_{p_{n,j}}$ only once (to the SWAP register holding the appropriate momentum variable) rather than $3N$ times (to each individual momentum variable). However, we do require a total of $O\left(N\log\left(g_{p'}\right)\right)$ SWAP operations, implying $O\left(N\log\left(g_{p'}\right)\right)$ Toffolis.

Application of Lemma 7 to PREP_m and $\left\{U_{p_{n,j}}\right\}$ yields an $\left(\alpha_m\alpha_p, a_m+a_p, \alpha_p\epsilon_m+\alpha_m\epsilon_p\right)$ block encoding of H_{kin}. This implies that

$$\alpha_{\text{kin}} = \alpha_m\alpha_p \in O\left(\frac{Np'^2_{\max}}{m_{\min}s^2_{\min}}\right). \tag{D48}$$

To achieve overall block-encoding error $\leq\epsilon$, it suffices to ensure that $\epsilon_m \leq \epsilon/2\alpha_p$ and $\epsilon_p \leq \epsilon/2\alpha_m$. Thus,

$$a_{\text{kin}} = a_m + a_p \in O\left(\log\left(\frac{\alpha_{\text{kin}}}{\epsilon}\right)\right). \tag{D49}$$

It follows from Lemma 5 that we need to prepare the state $\text{PREP}_m|0\rangle$ within error $\epsilon_m/\alpha_m\sqrt{N}$. Such a general quantum state preparation has Toffoli cost in

$$O\left(N\log\left(\frac{\alpha_m\sqrt{N}}{\epsilon_m}\right)\right) \subseteq O\left(N\log\left(\frac{\alpha_{\text{kin}}}{\epsilon}\right)\right), \tag{D50}$$

where we have used the assumption that we choose the uncertainty to saturate $\epsilon_m = \epsilon/2\alpha_p$. We require another

$$O\left(a_p^{\log 3}\right) \subseteq O\left(a_{\text{kin}}^{\log 3}\right) \subseteq O\left(\log^{\log 3}\left(\frac{\alpha_{\text{kin}}}{\epsilon}\right)\right) \tag{D51}$$

Toffolis for the quantum Karatsuba multiplications [59] used in the comparison test given in Eq. (D46). Addition can be performed in linear time and thus the cost of performing the entire comparison test is given by the cost of multiplication. This cost is additive to the cost of the state preparation given in Eq. (D50). Combining all of the results yields the desired complexity expressions. ■

Lemma 23 (Block encoding of H_{pot}). There exists an $(\alpha_{\text{pot}}, a_{\text{pot}}, \epsilon)$ block encoding of H_{pot} with normalization constant

$$\alpha_{\text{pot}} \in O\left(N^2\frac{Z^2_{\max}}{\Delta}\right) \tag{D52}$$

and a number of ancilla qubits

$$a_{\text{pot}} \in O\left(\log\left(\frac{\alpha_{\text{pot}}}{\epsilon}\right)\right). \tag{D53}$$

This block encoding can be implemented using

$$O\left(N\log\left(\frac{g_x\alpha_{\text{pot}}}{\epsilon}\right) + \log^{\log 3}\left(\frac{\alpha_{\text{pot}}}{\epsilon}\right)\right) \tag{D54}$$

Toffoli gates, where g_x is the number of discrete positions considered in the classical part of the Liouvillian.

Proof. We use the same strategy as in the proof of Lemma 22. In particular, we use the alternating-sign trick [54] to block encode a single summand of H_{pot} and then use Lemma 7 to combine the block encodings of all $O\left(N^2\right)$ terms. More specifically, we use the alternating-sign trick to construct $U_{V_{n,n',j}}$, an $(\alpha_V, a_V, \epsilon_V)$ block encoding of

$$\sum_{\bar{x}_n}\sum_{\bar{x}_{n'}}\frac{1}{\left(\|x_n-x_{n'}\|^2+\Delta^2\right)^{1/2}}|\bar{x}_n\rangle\langle\bar{x}_n|\otimes|\bar{x}_{n'}\rangle\langle\bar{x}_{n'}|. \tag{D55}$$

We then use the following PREP to attach the atomic numbers $\{Z_n\}$:

$$\text{PREP}_Z|0\rangle := \frac{1}{\sqrt{\alpha_Z}}\sum_{n=1}^{N}\sqrt{Z_N}|n\rangle\otimes\sum_{n'=1}^{N}\sqrt{Z_{n'}}\otimes\sum_{j=1}^{3}|j\rangle, \tag{D56}$$

where $\alpha_Z = \sum_{n,n',j} Z_nZ_{n'} \leq 3N^2Z^2_{\max}$. The above definition implies that

$$a_Z = 2\lceil\log N\rceil + \lceil\log 3\rceil. \tag{D57}$$

Importantly, the resultant state is a product state, meaning that $\sum_{n=1}^{N}\sqrt{Z_n}|n\rangle$, $\sum_{n'=1}^{N}\sqrt{Z_{n'}}|n'\rangle$, and $\sum_{j=1}^{3}|j\rangle$ can be prepared individually.

Let us now explain the construction of $U_{V_{n,n',j}}$. Using $\text{PREP}_V|0\rangle := \sum_{l=0}^{2^{a_V}-1} 1/\sqrt{2^{a_V}}|l\rangle$, we test the following

inequality:

$$l^2\left(\|\bar{x}_n-\bar{x}_{n'}\|^2+\overline{\Delta}^2\right)\leq 1, \tag{D58}$$

where $\overline{\Delta}\in\mathbb{N}$ such that $\Delta=\overline{\Delta}h_x$. As long as l satisfies the above inequality, the coefficient of $|l\rangle$ is set to +1. For larger l, the coefficient of $|l\rangle$ is set to alternate between ± 1. To determine the correct sign, we also need to test $x_{n,j}\leq x_{n',j}$, which has Toffoli complexity in $O(\log(g_x))$. The advantage of testing Eq. (A43) rather than

$$l\leq\frac{1}{\left(\|\bar{x}_n-\bar{x}_{n'}\|^2+\overline{\Delta}^2\right)^{1/2}} \tag{D59}$$

directly is that we do not have to calculate fractions containing square roots. However, the inequality test in Eq. (D58) does require us first to compute the left- and right-hand sides of the inequality using $O(1)$ quantum Karatsuba multiplications. This can be done using $O\left((a_V)^{\log 3}\right)$ Toffoli gates [59], whereas the inequality test itself requires only $O(a_V)$ Toffolis [59].

The number of ancilla qubits a_V determines the precision ϵ_V of $U_{V_{n,n',j}}$. In particular,

$$\left\|(\langle 0|\otimes\mathbb{1})U_{V_{n,n',j}}(|0\rangle\otimes\mathbb{1})-\frac{1}{\alpha_V}\sum_{\bar{x}_n}\sum_{\bar{x}_{n'}}\frac{1}{\left(\|x_n-x_{n'}\|^2+\Delta^2\right)^{1/2}}|\bar{x}_n\rangle\langle\bar{x}_n|\otimes|\bar{x}_{n'}\rangle\langle\bar{x}_{n'}|\right\|\leq\frac{1}{2^{a_V}}. \tag{D60}$$

Note that $\alpha_V\in O(1/\Delta)$. We can ensure that $U_{V_{n,n',j}}$ is an ϵ_V-precise approximation by choosing

$$a_V\in\Theta\left(\log\left(\frac{\alpha_V}{\epsilon_V}\right)\right). \tag{D61}$$

Instead of constructing $O\left(N^2\right)$ different block encodings for each of the $O\left(N^2\right)$ terms of H_{pot}, we use six SWAP registers for the six nuclear-position variables appearing in $1/\left(\|x_n-x_{n'}\|^2+\Delta^2\right)^{1/2}$. Here, the factor of 6 occurs because we assume that we are interested in dynamics in three spatial dimensions. Controlled by the PREP_Z register, we swap the appropriate position variables into the SWAP registers. This allows us to apply the block encoding $U_{V_{n,n',j}}$ only once (to the SWAP registers holding the appropriate position variables) rather than $O\left(N^2\right)$ times (for each individual term). However, we do require a total of $O(N\log(g_x))$ SWAP operations, resulting in $O(N\log(g_x))$ Toffolis.

Application of Lemma 7 to PREP_Z and $\left\{U_{V_{n,n',j}}\right\}$ yields an $(\alpha_Z\alpha_V, a_Z+a_V, \alpha_V\epsilon_Z+\alpha_Z\epsilon_V)$ block encoding of H_{pot}. This implies that

$$\alpha_{\text{pot}}=\alpha_Z\alpha_V\in O\left(\frac{N^2Z_{\max}^2}{\Delta}\right). \tag{D62}$$

To achieve overall block-encoding error $\leq\epsilon$, it suffices to ensure that $\epsilon_Z\leq\epsilon/2\alpha_V$ and $\epsilon_V\leq\epsilon/2\alpha_Z$. Thus the total number of qubits required for the block encoding of the potential operator is

$$a_{\text{pot}}=a_Z+a_V\in O\left(\log\left(\frac{\alpha_{\text{pot}}}{\epsilon}\right)\right), \tag{D63}$$

where the latter asymptotic bound follows from substituting into Eq. (D61). It follows from Lemma 5 that we need to prepare the state $\text{PREP}_Z|0\rangle$ within error $\epsilon_Z/N\alpha_Z$. Preparation of such a product state has Toffoli cost in

$$O\left(N\log\left(\frac{\alpha_ZN}{\epsilon_Z}\right)\right)\subseteq O\left(N\log\left(\frac{\alpha_{\text{pot}}}{\epsilon}\right)\right). \tag{D64}$$

We require another

$$O\left(a_Z^{\log 3}\right)\subseteq O\left(a_{\text{pot}}^{\log 3}\right)\subseteq O\left(\log^{\log 3}\left(\frac{\alpha_{\text{pot}}}{\epsilon}\right)\right) \tag{D65}$$

Toffolis for the quantum Karatsuba multiplications. Combining all of the results yields the desired complexity expressions. ■

Lemma 24 (Block encoding of $H_{E_{\text{el}}}$). There exists an $\left(\alpha_{E_{\text{el}}}, a_{E_{\text{el}}}, \epsilon\right)$ block encoding of $H_{E_{\text{el}}}$ with normalization constant

$$\alpha_{E_{\text{el}}}\in O(\lambda) \tag{D66}$$

and a number of ancilla qubits

$$a_{E_{\text{el}}}\in O\left(\log\left(\frac{\lambda}{\epsilon}\right)\right). \tag{D67}$$

This block encoding can be implemented using

$$\widetilde{O}\left(\lambda\left(\frac{N+\tilde{N}+\log(B)}{\epsilon}+\frac{N+\tilde{N}+\log(B)+\log^2(1/\epsilon)}{\gamma\delta}\right)\right) \tag{D68}$$

Toffoli gates and

$$O\left(\frac{1}{\delta}\right) \tag{D69}$$

queries to the initial electronic state-preparation oracle U_I from Definition 9.

Proof. The main strategy is to prepare the electronic ground states in superposition over all nuclear configurations and then use qubitization together with quantum phase estimation (QPE) to obtain estimates of the ground-state energies of H_{el} in superposition over all nuclear configurations. We then use the alternating-sign trick to construct a block encoding of $H_{E_{\mathrm{el}}}$. Let us now discuss the different subroutines and their errors in more detail.

First, we require access to a block encoding of the electronic Hamiltonian H_{el}. Reference [28] provides explicit $\mathrm{PREP}_{\mathrm{el}}$ and $\mathrm{SEL}_{\mathrm{el}}$ subroutines for obtaining $U_{H_{\mathrm{el}}}$, a $(\lambda, a_{\mathrm{el}}, \epsilon_{\mathrm{el}})$ block encoding of H_{el}. Let $\widetilde{H}_{\mathrm{el}}$ denote the block-encoded operator satisfying

$$\left\|\widetilde{H}_{\mathrm{el}} - H_{\mathrm{el}}\right\| \leq \epsilon_{\mathrm{el}}. \tag{D70}$$

Since H_{el} and $\widetilde{H}_{\mathrm{el}}$ are both Hermitian, we can use eigenvalue perturbation theory [63] to conclude that $\left|\widetilde{E}_{\mathrm{el}} - E_{\mathrm{el}}\right| \leq \epsilon_{\mathrm{el}}$ for all $\{x_n\}$, where $\widetilde{E}_{\mathrm{el}}$ is the ground-state energy of the block-encoded operator $\widetilde{H}_{\mathrm{el}}$. It then holds that

$$\left\|\widetilde{H}_{E_{\mathrm{el}}} - H_{E_{\mathrm{el}}}\right\| \leq \epsilon_{\mathrm{el}}, \tag{D71}$$

where

$$\widetilde{H}_{E_{\mathrm{el}}} := \sum_{\{\overline{x}_n\}} \widetilde{E}_{\mathrm{el}}\left(\{x_n\}\right) |\{\overline{x}_n\}\rangle\langle\{\overline{x}_n\}|. \tag{D72}$$

Next, we explain the electronic ground-state preparation. Let W denote the unitary that prepares an approximate ground state of $\widetilde{H}_{\mathrm{el}}$ for fixed nuclear positions according to Lemma 14, i.e.,

$$W|\{\overline{x}_n\}\rangle|0\rangle = |\{\overline{x}_n\}\rangle|\widetilde{\phi}_0\left(\{x_{n,j}\}\right)\rangle, \tag{D73}$$

with

$$|\langle\widetilde{\psi}_0\left(\{x_{n,j}\}\right)|\widetilde{\phi}_0\left(\{x_{n,j}\}\right)\rangle| \geq 1 - \epsilon_{\mathrm{prep}}. \tag{D74}$$

Note that we can view $U_{H_{\mathrm{el}}}$ as an exact block encoding of $\widetilde{H}_{\mathrm{el}}$, which allows us to use Lemma 14 directly without further error propagation. This means that we can prepare $|\widetilde{\phi}_0\left(\{x_{n,j}\}\right)\rangle$ using

$$O\left(\frac{\lambda}{\gamma\delta}\log\left(\frac{1}{\delta\epsilon_{\mathrm{prep}}}\right)\right) \tag{D75}$$

queries to $U_{H_{\mathrm{el}}}$ and

$$O\left(\frac{1}{\delta}\right) \tag{D76}$$

queries to U_I. In the following discussion, we will mostly refrain from writing out the $\left(\{x_{n,j}\}\right)$ dependence explicitly unless needed for clarity. Now, it holds that

$$|\widetilde{\phi}_0\rangle = e^{i\alpha}\left(1 - \epsilon'_{\mathrm{prep}}\right)|\widetilde{\psi}_0\rangle + \beta|\widetilde{\psi}_0^{\perp}\rangle \tag{D77}$$

for some angle $\alpha \in [0, 2\pi)$, $0 \leq \epsilon'_{\mathrm{prep}} \leq \epsilon_{\mathrm{prep}}$ and $|\beta|^2 = 2\epsilon'_{\mathrm{prep}} - \left(\epsilon'_{\mathrm{prep}}\right)^2 \leq 2\epsilon'_{\mathrm{prep}}$. Letting

$$|\psi'_0\rangle := e^{i\alpha}|\widetilde{\psi}_0\rangle \tag{D78}$$

we thus have that

$$\begin{aligned}\left\|W|\{\overline{x}_n\}\rangle|0\rangle - |\{\overline{x}_n\}\rangle|\psi'_0\rangle\right\| &= \left\||\{\overline{x}_n\}\rangle|\widetilde{\phi}_0\rangle - |\{\overline{x}_n\}\rangle|\psi'_0\rangle\right\| \\ &\leq \sqrt{2\epsilon_{\mathrm{prep}}}.\end{aligned} \tag{D79}$$

Next, we apply QPE with the following qubitization operator to the electronic register holding the electronic ground states:

$$Q := (2|0\rangle\langle 0| - \mathbb{1}) \cdot \mathrm{PREP}_{\mathrm{el}}^{\dagger} \cdot \mathrm{SEL}_{\mathrm{el}} \cdot \mathrm{PREP}_{\mathrm{el}}. \tag{D80}$$

The work of Ref. [27] shows that Q has eigenvalues $e^{\pm i\cos^{-1}\left(\widetilde{E}_k/\lambda\right)}$. QPE allows us to obtain the state $|\psi_{\widetilde{E}'_{\mathrm{el}}}\rangle$, which encodes an ϵ_{QPE}-precise estimate $\widetilde{E}'_{\mathrm{el}}$ of the ground-state energy $\widetilde{E}_{\mathrm{el}}$ of $\widetilde{H}_{\mathrm{el}}$ for fixed nuclear positions, with success probability $\geq 1 - \xi_{QPE}$ using

$$O\left(\frac{\lambda}{\epsilon_{QPE}}\log\left(\frac{1}{\xi_{QPE}}\right)\right) \tag{D81}$$

queries to Q. In other words,

$$|\psi_{\widetilde{E}'_{\mathrm{el}}}\rangle = \sqrt{1 - \xi'_{QPE}}|\widetilde{E}'_{\mathrm{el}}\rangle + \sqrt{\xi'_{QPE}}|\widetilde{E}^{\perp}_{\mathrm{el}}\rangle \tag{D82}$$

for some $0 \leq \xi'_{QPE} \leq \xi_{QPE}$. We refer to the corresponding unitary that prepares $|\psi_{\widetilde{E}'_{\mathrm{el}}}\rangle$ as U_Q, i.e.,

$$\begin{aligned}U_Q|\{\overline{x}_n\}\rangle|\psi'_0\rangle|0\rangle = &\sqrt{1 - \xi'_{QPE}}|\{\overline{x}_n\}\rangle|\psi'_0\rangle|\widetilde{E}'_{\mathrm{el}}\rangle \\ &+ \sqrt{\xi'_{QPE}}|\{\overline{x}_n\}\rangle|\psi'_0\rangle|\widetilde{E}^{\perp}_{\mathrm{el}}\rangle.\end{aligned} \tag{D83}$$

Lastly, we apply the alternating-sign trick, which is explained in detail in the proof of Lemma 1, to the $|\psi_{\widetilde{E}'_{\mathrm{el}}}\rangle$

register to obtain U_{alt}, an ϵ_{alt}-precise block encoding of

$$H_{\widetilde{E}'_{\mathrm{el}}} := \sum_{\{\bar{x}_n\}} \widetilde{E}'_{\mathrm{el}}\left(\{x_n\}\right) |\{\bar{x}_n\}\rangle\langle\{\bar{x}_n\}|. \tag{D84}$$

The overall block-encoding error $\epsilon_{E_{\mathrm{el}}}$ of $H_{E_{\mathrm{el}}}$ is then given by

$$\epsilon_{E_{\mathrm{el}}} := \Big\| \alpha_{E_{\mathrm{el}}} \left(\langle 0| \otimes \mathbb{1} \otimes \langle 0| \otimes \langle 0|\right) W^{-1} U_Q^{-1} U_{\mathrm{alt}} \times U_Q W \left(|0\rangle \otimes \mathbb{1} \otimes |0\rangle \otimes |0\rangle\right) - H_{E_{\mathrm{el}}} \Big\|, \tag{D85}$$

where the first register in the expression $(|0\rangle \otimes \mathbb{1} \otimes |0\rangle \otimes |0\rangle)$ consists of the block-encoding ancilla qubits for the alternating-sign trick, the second register is the nuclear-position and -momentum register, the third register is the electronic register, and the last register is the phase-estimation register. Note that the above definition implies that the electronic register as well as the phase-estimation register are uncomputed and projected out to the $|0\rangle$ state at the end of the simulation. In other words, the error $\epsilon_{E_{\mathrm{el}}}$ is only measured within the Hilbert space of the nuclear-position and -momentum registers. Importantly, the error matrix

$$\alpha_{E_{\mathrm{el}}} \left(\langle 0| \otimes \mathbb{1} \otimes \langle 0| \otimes \langle 0|\right) W^{-1} U_Q^{-1} U_{\mathrm{alt}} U_Q W \times \left(|0\rangle \otimes \mathbb{1} \otimes |0\rangle \otimes |0\rangle\right) - H_{E_{\mathrm{el}}} \tag{D86}$$

is diagonal in the nuclear-position and -momentum basis, since W, $\widetilde{U}_Q$, U_{alt}, and $H_{E_{\mathrm{el}}}$ are all diagonal in the nuclear-position and -momentum basis. Hence, $\epsilon_{E_{\mathrm{el}}}$ is simply the largest value on the diagonal of $\mathcal{E}_{E_{\mathrm{el}}}$. This allows us to consider the block-encoding error for each nuclear computational basis state separately. It also implies that $\alpha_{E_{\mathrm{el}}} \in O(\lambda)$. Suppressing the nuclear-momentum register, we then obtain the following:

$$\begin{aligned}
\epsilon_{E_{\mathrm{el}}} &= \max_{\{\bar{x}_n\}} \|\alpha_{E_{\mathrm{el}}} \left(\langle 0| \otimes \mathbb{1} \otimes \langle 0| \otimes \langle 0|\right) W^{-1} U_Q^{-1} U_{\mathrm{alt}} U_Q W \left(|0\rangle \otimes \mathbb{1} \otimes |0\rangle \otimes |0\rangle\right) |\{\bar{x}_n\}\rangle - H_{E_{\mathrm{el}}} |\{\bar{x}_n\}\rangle\| \\
&\leq \max_{\{\bar{x}_n\}} \|\alpha_{E_{\mathrm{el}}} \left(\langle 0| \otimes \mathbb{1} \otimes \mathbb{1} \otimes \mathbb{1}\right) W^{-1} U_Q^{-1} U_{\mathrm{alt}} U_Q W \left(|0\rangle \otimes \mathbb{1} \otimes \mathbb{1} \otimes \mathbb{1}\right) |\{\bar{x}_n\}\rangle|0\rangle|0\rangle - H_{E_{\mathrm{el}}} \otimes \mathbb{1} \otimes \mathbb{1} |\{\bar{x}_n\}\rangle|0\rangle|0\rangle\| \\
&\leq \max_{\{\bar{x}_n\}} \|\alpha_{E_{\mathrm{el}}} W^{-1} U_Q^{-1} \left(\langle 0| \otimes \mathbb{1} \otimes \mathbb{1} \otimes \mathbb{1}\right) U_{\mathrm{alt}} \left(|0\rangle \otimes \mathbb{1} \otimes \mathbb{1} \otimes \mathbb{1}\right) U_Q W |\{\bar{x}_n\}\rangle|0\rangle|0\rangle - H_{E_{\mathrm{el}}} \otimes \mathbb{1} \otimes \mathbb{1} |\{\bar{x}_n\}\rangle|0\rangle|0\rangle\|,
\end{aligned} \tag{D87}$$

where we have used the fact that $|0\rangle\langle 0| \otimes \mathbb{1} \otimes \mathbb{1} \otimes \mathbb{1}$ commutes with W and U_Q, since W and U_Q act trivially on the first register. In the following discussion, we will drop the "$\otimes \mathbb{1}$" for ease of notation. The general strategy now is to apply the triangle inequality repeatedly to "peel off" the errors stemming from different subroutines layer by layer. First, we will isolate the error associated with W:

$$\begin{aligned}
\epsilon_{E_{\mathrm{el}}} \leq \max_{\{\bar{x}_n\}} &\|\alpha_{E_{\mathrm{el}}} W^{-1} U_Q^{-1} \left(\langle 0|U_{\mathrm{alt}}|0\rangle\right) U_Q W |\{\bar{x}_n\}\rangle|0\rangle|0\rangle - \alpha_{E_{\mathrm{el}}} W^{-1} U_Q^{-1} \left(\langle 0|U_{\mathrm{alt}}|0\rangle\right) U_Q |\{\bar{x}_n\}\rangle|\psi'_0\rangle|0\rangle\| \\
&+ \max_{\{\bar{x}_n\}} \left\|\alpha_{E_{\mathrm{el}}} W^{-1} U_Q^{-1} \left(\langle 0|U_{\mathrm{alt}}|0\rangle\right) U_Q |\{\bar{x}_n\}\rangle|\psi'_0\rangle|0\rangle - H_{E_{\mathrm{el}}} |\{\bar{x}_n\}\rangle|0\rangle|0\rangle\right\|.
\end{aligned} \tag{D88}$$

By definition, the spectral norm is subordinate to the ℓ^2-norm. Furthermore, we have that $\|U\| = 1$ for any unitary U. With this in mind, we can bound the first term on the right-hand side of the above inequality as follows:

$$\begin{aligned}
&\max_{\{\bar{x}_n\}} \left\|\alpha_{E_{\mathrm{el}}} W^{-1} U_Q^{-1} \left(\langle 0|U_{\mathrm{alt}}|0\rangle\right) U_Q W |\{\bar{x}_n\}\rangle|0\rangle|0\rangle - \alpha_{E_{\mathrm{el}}} W^{-1} U_Q^{-1} \left(\langle 0|U_{\mathrm{alt}}|0\rangle\right) U_Q |\{\bar{x}_n\}\rangle|\psi'_0\rangle|0\rangle\right\| \\
&\leq \max_{\{\bar{x}_n\}} \alpha_{E_{\mathrm{el}}} \left\|W^{-1}\right\| \left\|U_Q^{-1}\right\| \|\left(\langle 0|U_{\mathrm{alt}}|0\rangle\right)\| \left\|U_Q\right\| \left\|W|\{\bar{x}_n\}\rangle|0\rangle|0\rangle - |\{\bar{x}_n\}\rangle|\psi'_0\rangle|0\rangle\right\| \leq \alpha_{E_{\mathrm{el}}} \sqrt{2\epsilon_{\mathrm{prep}}}.
\end{aligned} \tag{D89}$$

To bound the second term on the right-hand side of Eq. (D88), we use the triangle inequality again:

$$\begin{aligned}
&\max_{\{\bar{x}_n\}} \left\|\alpha_{E_{\mathrm{el}}} W^{-1} U_Q^{-1} \left(\langle 0|U_{\mathrm{alt}}|0\rangle\right) U_Q |\{\bar{x}_n\}\rangle|\psi'_0\rangle|0\rangle - H_{E_{\mathrm{el}}} |\{\bar{x}_n\}\rangle|0\rangle|0\rangle\right\| \\
&\leq \max_{\{\bar{x}_n\}} \|\alpha_{E_{\mathrm{el}}} W^{-1} U_Q^{-1} \left(\langle 0|U_{\mathrm{alt}}|0\rangle\right) U_Q |\{\bar{x}_n\}\rangle|\psi'_0\rangle|0\rangle - \alpha_{E_{\mathrm{el}}} W^{-1} U_Q^{-1} \left(\langle 0|U_{\mathrm{alt}}|0\rangle\right) |\{\bar{x}_n\}\rangle|\psi'_0\rangle|\widetilde{E}'_{\mathrm{el}}\rangle\| \\
&\quad + \max_{\{\bar{x}_n\}} \left\|\alpha_{E_{\mathrm{el}}} W^{-1} U_Q^{-1} \left(\langle 0|U_{\mathrm{alt}}|0\rangle\right) |\{\bar{x}_n\}\rangle|\psi'_0\rangle|\widetilde{E}'_{\mathrm{el}}\rangle - H_{E_{\mathrm{el}}} |\{\bar{x}_n\}\rangle|0\rangle|0\rangle\right\|.
\end{aligned} \tag{D90}$$

The first term on the right-hand side of the above inequality can be bounded as follows:

$$\begin{aligned}&\max_{\{\bar{x}_n\}} \left\| \alpha_{E_{\mathrm{el}}} W^{-1} U_Q^{-1} \left(\langle 0|U_{\mathrm{alt}}|0\rangle\right) U_Q |\{\bar{x}_n\}\rangle|\psi_0'\rangle|0\rangle - \alpha_{E_{\mathrm{el}}} W^{-1} U_Q^{-1} \left(\langle 0|U_{\mathrm{alt}}|0\rangle\right) |\{\bar{x}_n\}\rangle|\psi_0'\rangle|\widetilde{E}_{\mathrm{el}}'\rangle \right\| \\ &\quad \leq \max_{\{\bar{x}_n\}} \alpha_{E_{\mathrm{el}}} \left\| W^{-1} \right\| \left\| U_Q^{-1} \right\| \left\| \left(\langle 0|U_{\mathrm{alt}}|0\rangle\right) \right\| \left\| U_Q |\{\bar{x}_n\}\rangle|\psi_0'\rangle|0\rangle - |\{\bar{x}_n\}\rangle|\psi_0'\rangle|\widetilde{E}_{\mathrm{el}}'\rangle \right\| \leq \alpha_{E_{\mathrm{el}}} \sqrt{\xi_{QPE}}. \end{aligned} \tag{D91}$$

Note that the failure probability ξ_{QPE} of the phase-estimation step is now part of the block-encoding error of $H_{E_{\mathrm{el}}}$ in addition to the actual phase-estimation error ϵ_{QPE}. Before explaining how ϵ_{QPE} contributes to the block-encoding error $\epsilon_{E_{\mathrm{el}}}$, we first isolate the error ϵ_{alt} associated with the alternating-sign trick. This can be done by applying the triangle inequality to the second term on the right-hand side of Eq. (D90):

$$\begin{aligned}&\max_{\{\bar{x}_n\}} \left\| \alpha_{E_{\mathrm{el}}} W^{-1} U_Q^{-1} \left(\langle 0|U_{\mathrm{alt}}|0\rangle\right) |\{\bar{x}_n\}\rangle|\psi_0'\rangle|\widetilde{E}_{\mathrm{el}}'\rangle - H_{E_{\mathrm{el}}} |\{\bar{x}_n\}\rangle|0\rangle|0\rangle \right\| \\ &\quad \leq \max_{\{\bar{x}_n\}} \left\| \alpha_{E_{\mathrm{el}}} W^{-1} U_Q^{-1} \left(\langle 0|U_{\mathrm{alt}}|0\rangle\right) |\{\bar{x}_n\}\rangle|\psi_0'\rangle|\widetilde{E}_{\mathrm{el}}'\rangle - W^{-1} U_Q^{-1} H_{\widetilde{E}_{\mathrm{el}}'} |\{\bar{x}_n\}\rangle|\psi_0'\rangle|\widetilde{E}_{\mathrm{el}}'\rangle \right\| \\ &\qquad + \max_{\{\bar{x}_n\}} \left\| W^{-1} U_Q^{-1} H_{\widetilde{E}_{\mathrm{el}}'} |\{\bar{x}_n\}\rangle|\psi_0'\rangle|\widetilde{E}_{\mathrm{el}}'\rangle - H_{E_{\mathrm{el}}} |\{\bar{x}_n\}\rangle|0\rangle|0\rangle \right\|. \end{aligned} \tag{D92}$$

The first term on the right-hand side of the above inequality can then be bounded as follows:

$$\begin{aligned}&\max_{\{\bar{x}_n\}} \left\| \alpha_{E_{\mathrm{el}}} W^{-1} U_Q^{-1} \left(\langle 0|U_{\mathrm{alt}}|0\rangle\right) |\{\bar{x}_n\}\rangle|\psi_0'\rangle|\widetilde{E}_{\mathrm{el}}'\rangle - W^{-1} U_Q^{-1} H_{\widetilde{E}_{\mathrm{el}}'} |\{\bar{x}_n\}\rangle|\psi_0'\rangle|\widetilde{E}_{\mathrm{el}}'\rangle \right\| \\ &\quad \leq \max_{\{\bar{x}_n\}} \left\| W^{-1} \right\| \left\| U_Q^{-1} \right\| \left\| \alpha_{E_{\mathrm{el}}} \left(\langle 0|U_{\mathrm{alt}}|0\rangle\right) |\{\bar{x}_n\}\rangle|\psi_0'\rangle|\widetilde{E}_{\mathrm{el}}'\rangle - H_{\widetilde{E}_{\mathrm{el}}'} |\{\bar{x}_n\}\rangle|\psi_0'\rangle|\widetilde{E}_{\mathrm{el}}'\rangle \right\| \leq \epsilon_{\mathrm{alt}}. \end{aligned} \tag{D93}$$

Application of the triangle inequality to the second term on the right-hand side of Eq. (D92) now allows us to isolate the phase-estimation error ϵ_{QPE}:

$$\begin{aligned}&\max_{\{\bar{x}_n\}} \left\| W^{-1} U_Q^{-1} H_{\widetilde{E}_{\mathrm{el}}'} |\{\bar{x}_n\}\rangle|\psi_0'\rangle|\widetilde{E}_{\mathrm{el}}'\rangle - H_{E_{\mathrm{el}}} |\{\bar{x}_n\}\rangle|0\rangle|0\rangle \right\| \\ &\quad \leq \max_{\{\bar{x}_n\}} \left\| W^{-1} U_Q^{-1} H_{\widetilde{E}_{\mathrm{el}}'} |\{\bar{x}_n\}\rangle|\psi_0'\rangle|\widetilde{E}_{\mathrm{el}}'\rangle - W^{-1} U_Q^{-1} H_{\widetilde{E}_{\mathrm{el}}} |\{\bar{x}_n\}\rangle|\psi_0'\rangle|\widetilde{E}_{\mathrm{el}}'\rangle \right\| \\ &\qquad + \max_{\{\bar{x}_n\}} \left\| W^{-1} U_Q^{-1} H_{\widetilde{E}_{\mathrm{el}}} |\{\bar{x}_n\}\rangle|\psi_0'\rangle|\widetilde{E}_{\mathrm{el}}'\rangle - H_{E_{\mathrm{el}}} |\{\bar{x}_n\}\rangle|0\rangle|0\rangle \right\|. \end{aligned} \tag{D94}$$

The first term on the right-hand side of the above inequality can be bounded as follows:

$$\begin{aligned}&\max_{\{\bar{x}_n\}} \left\| W^{-1} U_Q^{-1} H_{\widetilde{E}_{\mathrm{el}}'} |\{\bar{x}_n\}\rangle|\psi_0'\rangle|\widetilde{E}_{\mathrm{el}}'\rangle - W^{-1} U_Q^{-1} H_{\widetilde{E}_{\mathrm{el}}} |\{\bar{x}_n\}\rangle|\psi_0'\rangle|\widetilde{E}_{\mathrm{el}}'\rangle \right\| \\ &\quad \leq \max_{\{\bar{x}_n\}} \left\| W^{-1} \right\| \left\| U_Q^{-1} \right\| \left\| H_{\widetilde{E}_{\mathrm{el}}'} |\{\bar{x}_n\}\rangle|\psi_0'\rangle|\widetilde{E}_{\mathrm{el}}'\rangle - H_{\widetilde{E}_{\mathrm{el}}} |\{\bar{x}_n\}\rangle|\psi_0'\rangle|\widetilde{E}_{\mathrm{el}}'\rangle \right\| \leq \epsilon_{QPE}. \end{aligned} \tag{D95}$$

Next, we isolate the block-encoding error ϵ_{el} of the electronic Hamiltonian by applying the triangle inequality to the second term on the right-hand side of Eq. (D94):

$$\begin{aligned}&\max_{\{\bar{x}_n\}} \left\| W^{-1} U_Q^{-1} H_{\widetilde{E}_{\mathrm{el}}} |\{\bar{x}_n\}\rangle|\psi_0'\rangle|\widetilde{E}_{\mathrm{el}}'\rangle - H_{E_{\mathrm{el}}} |\{\bar{x}_n\}\rangle|0\rangle|0\rangle \right\| \\ &\quad \leq \max_{\{\bar{x}_n\}} \left\| W^{-1} U_Q^{-1} H_{\widetilde{E}_{\mathrm{el}}} |\{\bar{x}_n\}\rangle|\psi_0'\rangle|\widetilde{E}_{\mathrm{el}}'\rangle - W^{-1} U_Q^{-1} H_{E_{\mathrm{el}}} |\{\bar{x}_n\}\rangle|0\rangle|0\rangle \right\| \\ &\qquad + \max_{\{\bar{x}_n\}} \left\| W^{-1} U_Q^{-1} H_{E_{\mathrm{el}}} |\{\bar{x}_n\}\rangle|\psi_0'\rangle|\widetilde{E}_{\mathrm{el}}'\rangle - H_{E_{\mathrm{el}}} |\{\bar{x}_n\}\rangle|0\rangle|0\rangle \right\|. \end{aligned} \tag{D96}$$

The first term on the right-hand side of the above inequality can then be bounded as follows:

$$\max_{\{\bar{x}_n\}} \left\| W^{-1} U_Q^{-1} H_{\widetilde{E}_{\rm el}} |\{\bar{x}_n\}\rangle |\psi_0'\rangle |\widetilde{E}_{\rm el}'\rangle - W^{-1} U_Q^{-1} H_{E_{\rm el}} |\{\bar{x}_n\}\rangle |0\rangle |0\rangle \right\| \leq \max_{\{\bar{x}_n\}} \left\| W^{-1} \right\| \left\| U_Q^{-1} \right\| \left\| H_{\widetilde{E}_{\rm el}} |\{\bar{x}_n\}\rangle |\psi_0'\rangle |\widetilde{E}_{\rm el}'\rangle - H_{E_{\rm el}} |\{\bar{x}_n\}\rangle |\psi_0'\rangle |\widetilde{E}_{\rm el}'\rangle \right\| \leq \epsilon_{\rm el}. \tag{D97}$$

To bound the second term on the right-hand side of Eq. (D96) recall that $H_{E_{\rm el}}$ is diagonal in the $|\{\bar{x}_n\}\rangle$ basis, i.e., $H_{\widetilde{E}_{\rm el}} |\{\bar{x}_n\}\rangle = E_{\rm el}\left(\left(\{x_n\}\right)\right) |\{\bar{x}_n\}\rangle$. Thus, we have that

$$\begin{aligned} &\max_{\{\bar{x}_n\}} \left\| W^{-1} U_Q^{-1} H_{E_{\rm el}} |\{\bar{x}_n\}\rangle |\psi_0'\rangle |\widetilde{E}_{\rm el}'\rangle - H_{E_{\rm el}} |\{\bar{x}_n\}\rangle |0\rangle |0\rangle \right\| \\ &\leq \max_{\{\bar{x}_n\}} \left| E_{\rm el}\left(\{x_n\}\right) \right| \left\| W^{-1} U_Q^{-1} |\{\bar{x}_n\}\rangle |\psi_0'\rangle |\widetilde{E}_{\rm el}'\rangle - |\{\bar{x}_n\}\rangle |0\rangle |0\rangle \right\| \\ &\leq \alpha_{E_{\rm el}} \left\| W^{-1} U_Q^{-1} |\{\bar{x}_n\}\rangle |\psi_0'\rangle |\widetilde{E}_{\rm el}'\rangle - W^{-1} |\{\bar{x}_n\}\rangle |\psi_0'\rangle |0\rangle \right\| + \alpha_{E_{\rm el}} \left\| W^{-1} |\{\bar{x}_n\}\rangle |\psi_0'\rangle |0\rangle - |\{\bar{x}_n\}\rangle |0\rangle |0\rangle \right\|. \end{aligned} \tag{D98}$$

The first term on the right-hand side of the above inequality can then be bounded as follows:

$$\begin{aligned} &\alpha_{E_{\rm el}} \left\| W^{-1} U_Q^{-1} |\{\bar{x}_n\}\rangle |\psi_0'\rangle |\widetilde{E}_{\rm el}'\rangle - W^{-1} |\{\bar{x}_n\}\rangle |\psi_0'\rangle |0\rangle \right\| \\ &\leq \alpha_{E_{\rm el}} \left\| W^{-1} \right\| \left\| U_Q^{-1} |\{\bar{x}_n\}\rangle |\psi_0'\rangle |\widetilde{E}_{\rm el}'\rangle - |\{\bar{x}_n\}\rangle |\psi_0'\rangle |0\rangle \right\| \\ &\leq \alpha_{E_{\rm el}} \sqrt{\xi_{QPE}}. \end{aligned} \tag{D99}$$

Lastly, the second term on the right-hand side of Eq. (D98) can be bounded in terms of $\epsilon_{\rm prep}$:

$$\alpha_{E_{\rm el}} \left\| W^{-1} |\{\bar{x}_n\}\rangle |\psi_0'\rangle |0\rangle - |\{\bar{x}_n\}\rangle |0\rangle |0\rangle \right\| \leq \alpha_{E_{\rm el}} \sqrt{2\epsilon_{\rm prep}}. \tag{D100}$$

Putting everything together, we find that

$$\epsilon_{E_{\rm el}} \leq \alpha_{E_{\rm el}} \left(2\sqrt{2\epsilon_{\rm prep}} + 2\sqrt{\xi_{QPE}} \right) + \epsilon_{\rm alt} + \epsilon_{QPE} + \epsilon_{\rm el}. \tag{D101}$$

We can ensure $\epsilon_{E_{\rm el}} \leq \epsilon$ by having

$$\epsilon_{\rm prep} \leq \frac{1}{2} \left(\frac{\epsilon}{10\alpha_{E_{\rm el}}} \right)^2, \tag{D102}$$

$$\xi_{QPE} \leq \left(\frac{\epsilon}{10\alpha_{E_{\rm el}}} \right)^2, \tag{D103}$$

$$\epsilon_{\rm alt} \leq \frac{\epsilon}{5}, \tag{D104}$$

$$\epsilon_{QPE} \leq \frac{\epsilon}{5}, \tag{D105}$$

$$\epsilon_{\rm el} \leq \frac{\epsilon}{5}. \tag{D106}$$

The condition $\epsilon_{\rm prep} \leq \frac{1}{2} \left(\epsilon / 10\alpha_{E_{\rm el}} \right)^2$ can be satisfied by using

$$O\left(\frac{\lambda}{\gamma\delta} \log\left(\frac{\lambda}{\delta\epsilon} \right) \right) \tag{D107}$$

queries to $U_{H_{\rm el}}$ and

$$O\left(\frac{1}{\delta} \right) \tag{D108}$$

queries to U_I. The conditions $\xi_{QPE} \leq \left(\epsilon / 10\alpha_{E_{\rm el}} \right)^2$ and $\epsilon_{QPE} \leq \epsilon/5$ can be satisfied by using

$$O\left(\frac{\lambda}{\epsilon} \log\left(\frac{\lambda}{\epsilon} \right) \right) \tag{D109}$$

queries to Q (or, equivalently, $U_{H_{\rm el}}$). Next, the condition $\epsilon_{\rm alt} \leq \epsilon/5$ can be satisfied by using

$$a_{E_{\rm el}} \in O\left(\log\left(\frac{\max_{\{x_n\}} \widetilde{E}_{\rm el}'\left(\{x_n\}\right)}{\epsilon} \right) \right) \in O\left(\log\left(\frac{\lambda}{\epsilon} \right) \right) \tag{D110}$$

ancilla qubits. The associated Toffoli cost is in $O\left(a_{E_{\rm el}}\right)$ due to the inequality testing required for the alternating-sign trick. Lastly, by Lemma 3, we need

$$O\left(N + \tilde{N} + \log\left(\frac{B}{\epsilon} \right) \right) \tag{D111}$$

Toffoli gates to ensure that the block-encoding error $\epsilon_{\rm el}$ of $H_{\rm el}$ is at most $\epsilon/5$.

The overall Toffoli complexity of block encoding $H_{E_{\rm el}}$ is dominated by the number of Toffolis needed for all

queries to the walk operator Q and the number of Toffolis needed for the electronic ground-state preparation. In either case, we multiply the respective query complexity with the Toffoli cost of block encoding H_{el} to obtain the desired complexity expression. ■

Let us now prove the following lemma on the query complexity of estimating the internal energy.

Lemma 25 (Query complexity of estimating the internal energy). Let $U_{H_{\mathrm{kin}}}$ be an $(\alpha_{\mathrm{kin}}, a_{\mathrm{kin}}, \epsilon/9)$ block encoding of H_{kin}, let $U_{H_{\mathrm{pot}}}$ be an $(\alpha_{\mathrm{pot}}, a_{\mathrm{pot}}, \epsilon/9)$ block encoding of H_{pot}, and let $U_{E_{\mathrm{el}}}$ be an $(\alpha_{E_{\mathrm{el}}}, a_{E_{\mathrm{el}}}, \epsilon/9)$ block encoding of $H_{E_{\mathrm{el}}}$. Furthermore, let $\widetilde{U}'_{L_{NVT}} \in \mathbb{C}^{\eta_{\mathrm{pur}} \times \eta_{\mathrm{pur}}}$ be an $\epsilon/\left(18\eta_{\mathrm{pur}}\alpha_{\mathrm{nuc}}\right)$-precise approximation to $U'_{L_{NVT}}$ as defined in Eq. (D15), where $\alpha_{\mathrm{nuc}} := \alpha_{\mathrm{kin}} + \alpha_{\mathrm{pot}} + \alpha_{E_{\mathrm{el}}}$. There exists a quantum algorithm for estimating the internal energy $\mathcal{U}$ associated with $\rho'_{\mathrm{sys}}(t)$ within error ϵ with probability at least $1 - \xi$ using

$$O\left(\frac{\alpha_{\mathrm{nuc}}}{\epsilon}\log\left(\frac{1}{\xi}\right)\right) \tag{D112}$$

queries to $\widetilde{U}'_{L_{NVT}}$, $U_{H_{\mathrm{kin}}}$, $U_{H_{\mathrm{pot}}}$ and $U_{E_{\mathrm{el}}}$.

Proof. First, note that the internal energy $\mathcal{U}$ of the system can be computed as follows:

$$\begin{aligned}\mathcal{U} = \mathrm{Tr}\left(\rho'_{\mathrm{sys}}(t)H_{\mathrm{nuc}}\right) &= \mathrm{Tr}\left(\rho'_{\mathrm{sys}}(t)H_{\mathrm{kin}}\right) \\ &+ \mathrm{Tr}\left(\rho'_{\mathrm{sys}}(t)H_{\mathrm{pot}}\right) + \mathrm{Tr}\left(\rho'_{\mathrm{sys}}(t)H_{E_{\mathrm{el}}}\right).\end{aligned} \tag{D113}$$

The idea is then to estimate each term individually within error $\epsilon/3$ using the Hadamard test as shown in Fig. 8. Then, we add the results classically, which yields an ϵ-precise estimate of $\mathcal{U}$.

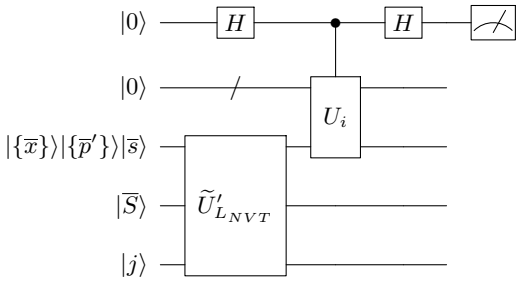

FIG. 8. The circuit for implementing the Hadamard test to estimate the internal energy $\mathcal{U}$ of the nuclei, where $\widetilde{U}'_{L_{NVT}}$ is an approximation to the evolution operator from Eq. (D15) and $U_i \in \{U_{H_{\mathrm{kin}}}, U_{H_{\mathrm{pot}}}, U_{E_{\mathrm{el}}}\}$. The second register from the top is the ancilla register needed for block encoding H_{kin}, H_{pot} or $H_{E_{\mathrm{el}}}$. The fact that U_i does not act on the bath register $|\overline{S}\rangle$ or the phase-gradient ancilla register $|j\rangle$ can be understood as taking the partial trace over those registers when computing the probability of measuring the top qubit as 0.

Let

$$\widetilde{H}_i := \alpha_i \left(\langle 0| \otimes \mathbb{1}\right) U_i \left(|0\rangle \otimes \mathbb{1}\right) \tag{D114}$$

be the Hamiltonian term block encoded in U_i, where $U_i \in \{U_{H_{\mathrm{kin}}}, U_{H_{\mathrm{pot}}}, U_{E_{\mathrm{el}}}\}$, and $\alpha_i \in \{\alpha_{\mathrm{kin}}, \alpha_{\mathrm{pot}}, \alpha_{E_{\mathrm{el}}}\}$. The probability of measuring the top qubit in Fig. 8 as 0 is given by

$$P''_i(0) := \frac{1}{2}\left(1 + \frac{\mathrm{Tr}\left(\widetilde{\rho}'_{\mathrm{sys}}(t)\widetilde{H}_i\right)}{\alpha_i}\right), \tag{D115}$$

where $\widetilde{\rho}'_{\mathrm{sys}}(t)$ is the reduced density matrix of our system obtained from $\widetilde{U}'_{L_{NVT}}$. Let $\hat{P}''_i(0)$ denote our estimate of $P''_i(0)$ based on the outcome of the Hadamard test. Furthermore, let

$$P'_i(0) := \frac{1}{2}\left(1 + \frac{\mathrm{Tr}\left(\rho'_{\mathrm{sys}}(t)\widetilde{H}_i\right)}{\alpha_i}\right), \tag{D116}$$

denote the success probability of the Hadamard test when used with an error-free block encoding of $U'_{L_{NVT}}$. Lastly, let

$$P_i(0) := \frac{1}{2}\left(1 + \frac{\mathrm{Tr}\left(\rho'_{\mathrm{sys}}(t)H_i\right)}{\alpha_i}\right) \tag{D117}$$

denote the success probability of the Hadamard test when applied to error-free block encodings of $U'_{L_{NVT}}$ and H_i, where $H_i \in \{H_{\mathrm{kin}}, H_{\mathrm{pot}}, H_{E_{\mathrm{el}}}\}$. Estimation of $P_i(0)$ within error $\epsilon/6\alpha_i$ allows us to obtain an $\epsilon/3$-precise estimate of $\mathrm{Tr}\left(\rho'_{\mathrm{sys}}(t)H_i\right)$.

By the triangle inequality, it holds that

$$\begin{aligned}|\hat{P}''_i(0) - P_i(0)| \le |\hat{P}''_i(0) - P''_i(0)| + |P''_i(0) - P'_i(0)| \\ + |P'_i(0) - P_i(0)|.\end{aligned} \tag{D118}$$

We now show that each term on the right-hand side of Eq. (D118) is upper bounded by $\epsilon/18\alpha_i$, which implies that

$$|\hat{P}''_i(0) - P_i(0)| \le \frac{\epsilon}{6\alpha_i}, \tag{D119}$$

as desired.

The error associated with the last term stems from the block-encoding error of H_i. Let ρ'_j denote the jth eigenvalue of $\rho'_{\mathrm{sys}}(t)$ and let $H^-_{i,j}$ denote the jth eigenvalue of $\widetilde{H}_i - H_i$. Using von Neumann's trace inequality, we obtain

the following bound:

$$\begin{aligned}|P_i'(0)-P_i(0)| &= \frac{1}{2\alpha_i}\left|\mathrm{Tr}\left(\rho'_{\mathrm{sys}}(t)\widetilde{H}_i\right)-\mathrm{Tr}\left(\rho'_{\mathrm{sys}}(t)H_i\right)\right| \\ &= \frac{1}{2\alpha_i}\left|\mathrm{Tr}\left(\rho'_{\mathrm{sys}}(t)\left(\widetilde{H}_i-H_i\right)\right)\right| \\ &\le \frac{1}{2\alpha_i}\sum_j\left|\rho'_j\right|\left|H^-_{i,j}\right| \le \frac{\epsilon}{18\alpha_i}\sum_j\left|\rho'_j\right| \\ &\le \frac{\epsilon}{18\alpha_i}, \end{aligned} \tag{D120}$$

where we have used the fact that $\left|H^-_{i,j}\right| \le \left\|\widetilde{H}_i - H_i\right\| \le \epsilon/9$ by choice of the block-encoding precision.

The error associated with the second term on the right-hand side of Eq. (D118) stems from the simulation error of $U'_{L_{NVT}}$. By assumption,

$$\left\|\widetilde{U}'_{L_{NVT}} - U'_{L_{NVT}}\right\| \le \frac{\epsilon}{18\eta_{\mathrm{pur}}\alpha_{\mathrm{nuc}}}. \tag{D121}$$

This implies that

$$\begin{aligned}&\left\|\widetilde{U}'_{L_{NVT}}|\Psi_0\rangle\langle\Psi_0|\widetilde{U}'^{\dagger}_{L_{NVT}} - U'_{L_{NVT}}|\Psi_0\rangle\langle\Psi_0|U'^{\dagger}_{L_{NVT}}\right\| \\ &\le \left\|\widetilde{U}'_{L_{NVT}}|\Psi_0\rangle\langle\Psi_0|\widetilde{U}'^{\dagger}_{L_{NVT}} - \widetilde{U}'_{L_{NVT}}|\Psi_0\rangle\langle\Psi_0|U'^{\dagger}_{L_{NVT}}\right\| \\ &\quad + \left\|\widetilde{U}'_{L_{NVT}}|\Psi_0\rangle\langle\Psi_0|U'^{\dagger}_{L_{NVT}} - U'_{L_{NVT}}|\Psi_0\rangle\langle\Psi_0|U'^{\dagger}_{L_{NVT}}\right\| \\ &\le \left\|\widetilde{U}'^{\dagger}_{L_{NVT}} - U'^{\dagger}_{L_{NVT}}\right\| + \left\|\widetilde{U}'_{L_{NVT}} - U'_{L_{NVT}}\right\| \\ &\le \frac{2\epsilon}{18\eta_{\mathrm{pur}}\alpha_{\mathrm{nuc}}} = \frac{\epsilon}{9\eta_{\mathrm{pur}}\alpha_{\mathrm{nuc}}}, \end{aligned} \tag{D122}$$

where, as before, $|\Psi_0\rangle$ is the initial state of the purification of $\rho'_{\mathrm{sys}}(t)$. It then follows from Definition 16 that

$$\begin{aligned}&\frac{1}{2}\left\|\widetilde{U}'_{L_{NVT}}|\Psi_0\rangle\langle\Psi_0|\widetilde{U}'^{\dagger}_{L_{NVT}} - U'_{L_{NVT}}|\Psi_0\rangle\langle\Psi_0|U'^{\dagger}_{L_{NVT}}\right\|_1 \\ &\quad \le \frac{\epsilon}{18\alpha_{\mathrm{nuc}}}. \end{aligned} \tag{D123}$$

Since the trace distance is contractive under the partial trace, we obtain the following bound:

$$\mathcal{T}\left(\widetilde{\rho}'_{\mathrm{sys}}(t), \rho'_{\mathrm{sys}}(t)\right) \le \frac{\epsilon}{18\alpha_{\mathrm{nuc}}}. \tag{D124}$$

Let ρ_j^- denote the jth eigenvalue of $\widetilde{\rho}'_{\mathrm{sys}}(t) - \rho'_{\mathrm{sys}}(t)$ and let $\widetilde{H}_{i,j}$ denote the jth eigenvalue of $\widetilde{H}_i$. Using von Neumann's trace inequality, we then have that

$$\begin{aligned}|P_i''(0)-P_i'(0)| &= \frac{1}{2\alpha_i}\left|\mathrm{Tr}\left(\widetilde{\rho}'_{\mathrm{sys}}(t)\widetilde{H}_i\right)-\mathrm{Tr}\left(\rho'_{\mathrm{sys}}(t)\widetilde{H}_i\right)\right| \\ &= \frac{1}{2\alpha_i}\left|\mathrm{Tr}\left(\left(\widetilde{\rho}'_{\mathrm{sys}}(t)-\rho'_{\mathrm{sys}}(t)\right)\widetilde{H}_i\right)\right| \\ &\le \frac{1}{2\alpha_i}\sum_j\left|\rho_j^-\right|\left|\widetilde{H}_{i,j}\right| \\ &\le \frac{1}{\alpha_i}\mathcal{T}\left(\widetilde{\rho}'_{\mathrm{sys}}(t), \rho'_{\mathrm{sys}}(t)\right)\alpha_i \\ &\le \frac{\epsilon}{18\alpha_{\mathrm{nuc}}} \le \frac{\epsilon}{18\alpha_i}. \end{aligned} \tag{D125}$$

Lastly, we need to ensure that $|\hat{P}_i''(0) - P_i''(0)| \le \epsilon/18\alpha_i$. The idea is to use amplitude estimation to obtain an $\epsilon/18\alpha_i$-precise estimate $\hat{P}_i''(0)$ of $P_i''(0)$ with constant success probability. This requires

$$O\left(\frac{\alpha_i}{\epsilon}\right) \subseteq O\left(\frac{\alpha_{\mathrm{nuc}}}{\epsilon}\right) \tag{D126}$$

applications of the Hadamard test, which is equivalent to $O\left(\frac{\alpha_{\mathrm{nuc}}}{\epsilon}\right)$ (controlled) applications of $\widetilde{U}'_{L_{NVT}}$ and U_i.

By the union bound, we can obtain an ϵ-precise estimate of $\mathcal{U}$ with success probability $\ge 1-\xi$ by ensuring that the failure probability associated with the estimation of each term $\mathrm{Tr}\left(\rho'_{\mathrm{sys}}(t)H_i\right)$ is $\le \xi/3$. This can be achieved via (fixed-point) amplitude amplification at the expense of an additional multiplicative factor of $\log(1/\xi)$ to the query complexities of $\widetilde{U}'_{L_{NVT}}$ and U_i. ■

3. Proof of Theorem 2

For convenience, let us restate Theorem 2 here.

Theorem 2 (Estimation of the free energy). Let $\eta := g_x^{3N} g_{p'}^{3N} g_s g_{p_s}$ be the number of grid points of the discretized phase space and assume that $\log\left(\eta^2/\epsilon\right) \le \eta$. Then there exists a quantum algorithm that solves Problem 2 with success probability at least $1-\xi$ using

$$\widetilde{O}\left(\left(\frac{\eta^{o(1)} N_{\mathrm{tot}} d \mu_{NVT}^{2+o(1)} t^{1+o(1)}}{\widetilde{\gamma}\,\widetilde{\delta}\,\epsilon^{1+o(1)}}\left(\alpha_{\mathrm{nuc}} + \frac{\eta (k_b T)^{1.5+o(1)}}{\sqrt{\epsilon}}\right) + \frac{N_{\mathrm{tot}}\alpha_{\mathrm{nuc}}\lambda}{\epsilon^2}\right)\log\left(\frac{1}{\xi}\right)\right)$$

Toffoli gates. Additionally,

$$\widetilde{O}\left(\frac{\eta^{o(1)} N d \mu_{NVT}^{1+o(1)} t^{1+o(1)}}{\widetilde{\delta}\,\epsilon^{1+o(1)}}\log\left(\frac{1}{\xi}\right) \times \left(\alpha_{\mathrm{nuc}} + \frac{\eta (k_b T)^{1.5+o(1)}}{\sqrt{\epsilon}}\right)\right)$$

queries to the initial electronic state-preparation oracle $\widetilde{U}_I$ are needed.

Proof. We use Lemma 21 to estimate the Gibbs entropy associated with $\rho_{\mathrm{sys}}(t)$ within error $\epsilon/2k_BT$ with failure probability at most $\xi/2$. This requires

$$O\left(\frac{\eta(k_BT)^{1.5}}{\epsilon^{1.5}}\log(1/\xi)\right) \tag{D127}$$

queries to an $\left(\epsilon/8\eta_{\mathrm{pur}}k_bT\log(\eta/\nu)\right)$-precise approximation $\widetilde{U}'_{L_{NVT}}$ to $U'_{L_{NVT}}$, where $\nu\in(0,1)$ is again a lower bound on $2\mathcal{T}\left(\widetilde{\rho}'_{\mathrm{sys}}(t),\rho'_{\mathrm{sys}}(t)\right)$. For simplicity, we assume that $\nu\in\Theta(\epsilon/\eta)$, which should be fairly easy to achieve. The promise $\log\left(\eta^2/\epsilon\right)\leq\eta$ ensures that this choice of ν does not exceed the upper bound on $2\mathcal{T}\left(\widetilde{\rho}'_{\mathrm{sys}}(t),\rho'_{\mathrm{sys}}(t)\right)$.

It then follows from Theorem 1 that estimation of the entropy requires a total of

$$\widetilde{O}\left(\frac{\eta^{1+o(1)}N_{\mathrm{tot}}d(k_BT)^{1.5+o(1)}\mu_{NVT}^{2+o(1)}t^{1+o(1)}}{\widetilde{\gamma}\,\widetilde{\delta}\,\epsilon^{1.5+o(1)}}\log\left(\frac{1}{\xi}\right)\right) \tag{D128}$$

Toffoli gates and

$$\widetilde{O}\left(\frac{\eta^{1+o(1)}Nd(k_BT)^{1.5+o(1)}\mu_{NVT}^{1+o(1)}t^{1+o(1)}}{\widetilde{\delta}\,\epsilon^{1.5+o(1)}}\log\left(\frac{1}{\xi}\right)\right) \tag{D129}$$

queries to the initial electronic state-preparation oracle $\widetilde{U}_I$.

Next, we use Lemma 25 to estimate the internal energy $\mathcal{U}$ of the nuclei within error $\epsilon/2$ with failure probability at most $\xi/2$. This requires

$$O\left(\frac{\alpha_{\mathrm{nuc}}}{\epsilon}\log\left(\frac{1}{\xi}\right)\right) \tag{D130}$$

queries to an $\left(\epsilon/36\eta_{\mathrm{pur}}\alpha_{\mathrm{nuc}}\right)$-precise approximation of $U'_{L_{NVT}}$. By Theorem 1, the associated Toffoli complexity is then in

$$\widetilde{O}\left(\frac{\eta^{o(1)}N_{\mathrm{tot}}d\,\alpha_{\mathrm{nuc}}\,\mu_{NVT}^{2+o(1)}t^{1+o(1)}}{\widetilde{\gamma}\,\widetilde{\delta}\,\epsilon^{1+o(1)}}\log\left(\frac{1}{\xi}\right)\right). \tag{D131}$$

Furthermore, we need

$$\widetilde{O}\left(\frac{\eta^{o(1)}Nd\,\alpha_{\mathrm{nuc}}\,\mu_{NVT}^{1+o(1)}t^{1+o(1)}}{\widetilde{\delta}\,\epsilon^{1+o(1)}}\log\left(\frac{1}{\xi}\right)\right) \tag{D132}$$

queries to the initial electronic state-preparation oracle $\widetilde{U}_I$. According to Lemma 25, we also require

$$O\left(\frac{\alpha_{\mathrm{nuc}}}{\epsilon}\log\left(\frac{1}{\xi}\right)\right) \tag{D133}$$

queries to $\epsilon/18$-precise block encodings of H_{kin}, H_{pot} and $H_{E_{\mathrm{el}}}$. Lemmas 22–24 imply that the combined Toffoli complexity of all these queries is in

$$\begin{aligned}&\widetilde{O}\left(\frac{\alpha_{\mathrm{nuc}}}{\epsilon}\log\left(\frac{1}{\xi}\right)\left(N\log\left(\frac{g\alpha_{\mathrm{nuc}}}{\epsilon}\right)+\log^{\log 3}\left(\frac{\alpha_{\mathrm{nuc}}}{\epsilon}\right)\right.\right.\\&\qquad\left.\left.+\;N_{\mathrm{tot}}\lambda\left(\frac{1}{\epsilon}+\frac{1}{\gamma\delta}\right)\right)\right)\\&\quad\subseteq\widetilde{O}\left(N_{\mathrm{tot}}\alpha_{\mathrm{nuc}}\lambda\log\left(\frac{1}{\xi}\right)\left(\frac{1}{\epsilon^2}+\frac{1}{\gamma\delta\epsilon}\right)\right).\end{aligned} \tag{D134}$$

Similarly to the spectral-gap argument used in the proof of Theorem 1, we only need to provide a lower bound $\widetilde{\gamma}$ on the spectral gap of the electronic Hamiltonian over those phase-space grid points that are associated with a nonzero amplitude at some point during the simulation. Likewise, we only need a lower bound $\widetilde{\delta}$ on the overlap of the initial electronic state with the true electronic ground state over phase-space grid points that are visited at some point during the simulation. The reason for this is that any simulation errors that occur on grid points that are associated with zero amplitude throughout the simulation do not contribute to the error of the final estimate. This means that Problem 2 can be solved using only $O\left(1/\widetilde{\gamma}\widetilde{\delta}\right)$ rather than $O(1/\gamma\delta)$ Toffoli gates.

Combining the previous results, we find that the overall Toffoli complexity associated with estimating $\mathcal{F}$ is in

$$\begin{aligned}&\widetilde{O}\left(\left(\frac{\eta^{o(1)}N_{\mathrm{tot}}d\,\alpha_{\mathrm{nuc}}\,\mu_{NVT}^{2+o(1)}t^{1+o(1)}}{\widetilde{\gamma}\,\widetilde{\delta}\,\epsilon^{1+o(1)}}\right.\right.\\&\qquad+\;\frac{\eta^{1+o(1)}N_{\mathrm{tot}}d(k_BT)^{1.5+o(1)}\mu_{NVT}^{2+o(1)}t^{1+o(1)}}{\widetilde{\gamma}\,\widetilde{\delta}\,\epsilon^{1.5+o(1)}}\\&\qquad\left.\left.+\frac{N_{\mathrm{tot}}\alpha_{\mathrm{nuc}}\lambda}{\epsilon^2}\right)\log\left(\frac{1}{\xi}\right)\right),\end{aligned} \tag{D135}$$

which can be simplified as follows:

$$\begin{aligned}&\widetilde{O}\left(\left(\frac{\eta^{o(1)}N_{\mathrm{tot}}d\mu_{NVT}^{2+o(1)}t^{1+o(1)}}{\widetilde{\gamma}\,\widetilde{\delta}\,\epsilon^{1+o(1)}}\left(\alpha_{\mathrm{nuc}}+\frac{\eta(k_bT)^{1.5+o(1)}}{\sqrt{\epsilon}}\right)\right.\right.\\&\qquad\left.\left.+\;\frac{N_{\mathrm{tot}}\alpha_{\mathrm{nuc}}\lambda}{\epsilon^2}\right)\log\left(\frac{1}{\xi}\right)\right).\end{aligned} \tag{D136}$$

Furthermore, the overall number of queries to $\widetilde{U}_I$ is in

$$\begin{aligned}&\widetilde{O}\left(\frac{\eta^{o(1)}Nd\,\alpha_{\mathrm{nuc}}\,\mu_{NVT}^{1+o(1)}t^{1+o(1)}}{\widetilde{\delta}\,\epsilon^{1+o(1)}}\log\left(\frac{1}{\xi}\right)\right.\\&\qquad\left.+\frac{\eta^{1+o(1)}Nd(k_BT)^{1.5+o(1)}\mu_{NVT}^{1+o(1)}t^{1+o(1)}}{\widetilde{\delta}\,\epsilon^{1.5+o(1)}}\log\left(\frac{1}{\xi}\right)\right),\end{aligned} \tag{D137}$$

which can be rewritten as follows:

$$\widetilde{O}\left(\frac{\eta^{o(1)}Nd\mu_{NVT}^{1+o(1)}t^{1+o(1)}}{\widetilde{\delta}\,\epsilon^{1+o(1)}}\log\left(\frac{1}{\xi}\right)\times\left(\alpha_{\text{nuc}}+\frac{\eta(k_bT)^{1.5+o(1)}}{\sqrt{\epsilon}}\right)\right). \tag{D138}$$

■

APPENDIX E: COMPUTATIONAL COST SCALING OF FORWARD EULER INTEGRATION

The computational costs for the calculation of molecular forces on a fault-tolerant quantum computer are given in Ref. [20]; however, the propagation of errors from an individual gradient estimate to the updated particle positions is not bounded. In the following, we extend that result by giving an upper bound on the run time of an MD simulation algorithm where the quantum computer is used to compute the forces on the particles that are then used to update the nuclear positions on a classical computer with forward Euler's method.

The goal of Euler's forward method is to update the value of a variable $y(T)$ (e.g., the position of a nucleus) at a time $T+h$ using the derivative $y'(T)$ and the step h:

$$y(T+h)=y(T)+hy'(T). \tag{E1}$$

We define $y(j\,|y(j-1))$ as the value of the position after j steps in the case of a perfect update (i.e., with the exact derivative) given the same perfectly updated position at the previous iteration. In contrast, we have $\tilde{y}(j\,|\tilde{y}(j-1))$ and $\tilde{y}(1)=\tilde{y}(1|y(0))$ in the case of updates with an approximate derivative (i.e., with some error δ in the derivative calculation). Unless otherwise stated, we assume that

$$\begin{aligned} y_{j-1}&=y(j-1|y_{j-2}),\\ y_{j-2}&=y(j-2|y_{j-3}),\\ &\dots \end{aligned}$$

We now want to determine an upper bound on the error performed in the update after N steps of updates with approximate derivatives, defined as the difference between this and the same updates computed with perfect derivatives: $|y(N|y_{N-1})-\tilde{y}(N|\tilde{y}_{N-1})|$. Additionally, we consider the solutions to Newton's equation of motion for the updated variable y and the approximate variable $\tilde{y}$: $y=e^{At}y_0$ and $\tilde{y}=e^{\tilde{A}t}y_0$, given the initial condition y_0. We call $y(j)=y_j$ the value of our variable after j time steps. Furthermore, we are assuming that the error on the derivatives is such that $\left\|A-\tilde{A}\right\|\le K_{\text{Lips}}$, where K_{Lips} is the Lipschitz constant, which is directly related to the norm of the differential operator.

For a single integration step of size h, if starting from the same previous value, we accumulate an error $|y(j\,|y_{j-1})-\tilde{y}(j\,|y_{j-1})|$ that is upper bounded by the error on the derivative estimation multiplied by the step:

$$\begin{aligned} |y(j\,|y_{j-1})-\tilde{y}(j\,|y_{j-1})|&\le\delta h,\\ \tilde{y}(j\,|y_{j-1})&\le e^{K_{\text{Lips}}h}y_{j-1},\\ \tilde{y}(j\,|\tilde{y}_{j-1})&\le e^{K_{\text{Lips}}h}\tilde{y}_{j-1}. \end{aligned} \tag{E2}$$

Therefore, we have

$$\begin{aligned} &|y(j\,|y_{j-1})-\tilde{y}(j\,|\tilde{y}_{j-1})|\\ &\quad\le\ |y(j\,|y_{j-1})-\tilde{y}(j\,|y_{j-1})|+|\tilde{y}(j\,|y_{j-1})-\tilde{y}(j\,|\tilde{y}_{j-1})|\\ &\quad\le\delta h+|\tilde{y}(j\,|y_{j-1})-\tilde{y}(j\,|\tilde{y}_{j-1})|\\ &\quad\le\delta h+e^{K_{\text{Lips}}h}|y_{j-1}-\tilde{y}_{j-1}|\\ &\quad\le\delta h\left(1+\sum_{n=1}^{j}e^{nK_{\text{Lips}}h}\right). \end{aligned} \tag{E3}$$

After N steps with $N=T/h$ (where T is the total time) of Euler's forward method, we obtain

$$\begin{aligned} &|y(N|y_{N-1})-\tilde{y}(N|\tilde{y}_{N-1})|\\ &\quad\le\delta h\left(1+\sum_{n=1}^{N}e^{nK_{\text{Lips}}h}\right)\le\delta h\left(1+\frac{e^{K_{\text{Lips}}h(N+1)}-1}{e^{K_{\text{Lips}}h}-1}\right)\\ &\quad\le\delta h\left(1+\frac{e^{K_{\text{Lips}}h}e^{K_{\text{Lips}}T}-1}{e^{K_{\text{Lips}}h}-1}\right)\\ &\quad=\delta h\left(1+\frac{e^{K_{\text{Lips}}T}-e^{-K_{\text{Lips}}h}}{1-e^{-K_{\text{Lips}}h}}\right)\\ &\quad\le h\delta(1+2e^{K_{\text{Lips}}T})\le 3h\delta e^{K_{\text{Lips}}T}, \end{aligned} \tag{E4}$$

where we have chosen $K_{\text{Lips}}h\le\ln(2)$ and $N=T/h$.

We want to make sure that the error is at most ϵ_{MD}:

$$h\delta(1+2e^{K_{\text{Lips}}T})\le 3h\delta e^{K_{\text{Lips}}T}\le\epsilon_{\text{MD}}. \tag{E5}$$

We choose the step size to be $h=\ln(2)/K_{\text{Lips}}$, so we have that our error on the single gradient estimation needs to be

$$\delta\le\frac{\epsilon_{\text{MD}}K_{\text{Lips}}}{3e^{K_{\text{Lips}}T}\ln(2)}. \tag{E6}$$

We want to impose this error δ as the target error in the single gradient estimation necessary to achieve an overall simulation error ϵ_{MD}. From Table IV of Ref. [20] in the case of first-quantized plane-waves, the time complexity is $T_{\text{grad}}=N_a^{7/2}\delta^{-1}$ (with the number of atoms N_a considered proportional to the number of orbitals). Therefore, the time

complexity of a single gradient estimation to achieve the target error is

$$\mathrm{ToffCount}_{\mathrm{grad}} \in O\left(\frac{N_a^{\frac{7}{2}} e^{K_{\mathrm{Lips}}T}}{K_{\mathrm{Lips}}\epsilon_{\mathrm{MD}}}\right). \tag{E7}$$

Since we need to perform $N = T/h = K_{\mathrm{Lips}}T/\ln(2)$ steps, the total time is given by

$$\mathrm{ToffCount}_{\mathrm{MD}} \in O\left(N\frac{N_a^{\frac{7}{2}} e^{K_{\mathrm{Lips}}T}}{K_{\mathrm{Lips}}\epsilon_{\mathrm{MD}}}\right) = O\left(T\frac{N_a^{\frac{7}{2}} e^{K_{\mathrm{Lips}}T}}{\epsilon_{\mathrm{MD}}}\right). \tag{E8}$$

This means that the time for simulating a system with Euler's integration method scales exponentially with the simulation time T, while still scaling polynomially with the other parameters.

It is worth noting that while the underlying trajectories are potentially unstable, the overall probability density formed by an ensemble of such trajectories generically is not. In particular, if we instead were to focus on the error in phase-space density, then this scaling would become polynomial if the shadowing-theorem conditions hold [68]. This suggests that the relative cost between this approach and our own may be somewhat deceptive; however, it is fair regardless to say, without assuming that we are interested in estimating a single-particle trajectory, that the number of Toffoli gates needed for an accurate simulation may scale exponentially with the evolution time in the worst-case scenario.

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

[2] J. Kempe, A. Kitaev, and O. Regev, The complexity of the local Hamiltonian problem, SIAM J. Computing **35**, 1070 (2006).

[3] B. P. Lanyon, J. D. Whitfield, G. G. Gillett, M. E. Goggin, M. P. Almeida, I. Kassal, J. D. Biamonte, M. Mohseni, B. J. Powell, M. Barbieri, A. Aspuru-Guzik, and A. G. White, Towards quantum chemistry on a quantum computer, Nat. Chem. **2**, 106 (2010).

[4] S. Aaronson and A. Arkhipov, in *Proceedings of the Forty-Third Annual ACM Symposium on Theory of Computing*, STOC '11 (Association for Computing Machinery, New York, 2011), p. 333.

[5] R. Babbush, D. W. Berry, R. Kothari, R. D. Somma, and N. Wiebe, Exponential quantum speedup in simulating coupled classical oscillators, ArXiv:2303.13012 (2023).

[6] M. Tuckerman, B. J. Berne, and G. J. Martyna, Reversible multiple time scale molecular dynamics, J. Chem. Phys. **97**, 1990 (1992).

[7] G. J. Martyna, M. E. Tuckerman, D. J. Tobias, and M. L. Klein, Explicit reversible integrators for extended systems dynamics, Mol. Phys. **87**, 1117 (1996).

[8] H. Xu, The slow but steady rise of binding free energy calculations in drug discovery, J. Comput. Aided Mol. Des. **37**, 67 (2023).

[9] B. Ries, K. Normak, R. G. Weiß, S. Rieder, E. P. Barros, C. Champion, G. König, and S. Riniker, Relative free-energy calculations for scaffold hopping-type transformations with an automated RE-EDS sampling procedure, J. Comput. Aided Mol. Des. **36**, 117 (2022).

[10] R. Santagati, A. Aspuru-Guzik, R. Babbush, M. Degroote, L. Gonzalez, E. Kyoseva, N. Moll, M. Oppel, R. M. Parrish, N. C. Rubin, M. Streif, C. S. Tautermann, H. Weiss, N. Wiebe, and C. Utschig-Utschig, Drug design on quantum computers, ArXiv:2301.04114 (2023).

[11] M. E. Tuckerman and G. J. Martyna, Understanding modern molecular dynamics: Techniques and applications, J. Phys. Chem. B **104**, 159 (2000).

[12] S. Nosé, A molecular dynamics method for simulations in the canonical ensemble, Mol. Phys. **52**, 255 (1984).

[13] S. D. Bond, B. J. Leimkuhler, and B. B. Laird, The Nosé–Poincaré method for constant temperature molecular dynamics, J. Comput. Phys. **151**, 114 (1999).

[14] I. Kassal, J. D. Whitfield, A. Perdomo-Ortiz, M.-H. Yung, and A. Aspuru-Guzik, Simulating chemistry using quantum computers, Annu. Rev. Phys. Chem. **62**, 185 (2011).

[15] P. J. Ollitrault, G. Mazzola, and I. Tavernelli, Nonadiabatic molecular quantum dynamics with quantum computers, Phys. Rev. Lett. **125**, 260511 (2020).

[16] D. A. Fedorov, M. J. Otten, S. K. Gray, and Y. Alexeev, Ab initio molecular dynamics on quantum computers, J. Chem. Phys. **154**, 164103 (2021).

[17] K. Kuroiwa, T. Ohkuma, H. Sato, and R. Imai, Quantum Car-Parrinello molecular dynamics: A cost-efficient molecular simulation method on near-term quantum computers, ArXiv:10.48550/arxiv.2212.11921 (2022).

[18] I. O. Sokolov, P. K. Barkoutsos, L. Moeller, P. Suchsland, G. Mazzola, and I. Tavernelli, Microcanonical and finite-temperature *ab initio* molecular dynamics simulations on quantum computers, Phys. Rev. Res. **3**, 013125 (2021).

[19] T. E. O'Brien, B. Senjean, R. Sagastizabal, X. Bonet-Monroig, A. Dutkiewicz, F. Buda, L. DiCarlo, and L. Visscher, Calculating energy derivatives for quantum chemistry on a quantum computer, npj Quantum Inf. **5**, 113 (2019).

[20] T. E. O'Brien, M. Streif, N. C. Rubin, R. Santagati, Y. Su, W. J. Huggins, J. J. Goings, N. Moll, E. Kyoseva, M. Degroote, C. S. Tautermann, J. Lee, D. W. Berry, N. Wiebe, and R. Babbush, Efficient quantum computation of molecular forces and other energy gradients, Phys. Rev. Res. **4**, 043210 (2022).

[21] M. Steudtner, S. Morley-Short, W. Pol, S. Sim, C. L. Cortes, M. Loipersberger, R. M. Parrish, M. Degroote, N. Moll, R. Santagati, and M. Streif, Fault-tolerant quantum computation of molecular observables, ArXiv:2303.14118 (2023).

[22] M. Manathunga, A. W. Götz, and K. M. Merz, Computer-aided drug design, quantum-mechanical methods for biological problems, Curr. Opin. Struct. Biol. **75**, 102417 (2022).

[23] B. O. Koopman, Hamiltonian systems and transformation in Hilbert space, Proc. Natl. Acad. Sci. **17**, 315 (1931).

[24] D. Mauro, On Koopman–von Neumann waves, Int. J. Mod. Phys. A **17**, 1301 (2002).

[25] I. Joseph, Koopman–von Neumann approach to quantum simulation of nonlinear classical dynamics, Phys. Rev. Res. **2**, 043102 (2020).
[26] S. Jin, N. Liu, and Y. Yu, Time complexity analysis of quantum algorithms via linear representations for nonlinear ordinary and partial differential equations, J. Comput. Phys. **487**, 112149 (2023).
[27] R. Babbush, D. W. Berry, J. R. McClean, and H. Neven, Quantum simulation of chemistry with sublinear scaling in basis size, npj Quantum Inf. **5**, 92 (2019).
[28] Y. Su, D. W. Berry, N. Wiebe, N. Rubin, and R. Babbush, Fault-tolerant quantum simulations of chemistry in first quantization, PRX Quantum **2**, 040332 (2021).
[29] I. Kivlichan, N. Wiebe, R. Babbush, and A. Aspuru-Guzik, Bounding the costs of quantum simulation of many-body physics in real space, J. Phys. A: Math. Theor. **50**, 305301 (2017).
[30] L. Grover and T. Rudolph, Creating superpositions that correspond to efficiently integrable probability distributions, ArXiv:quant-ph/0208112 (2002).
[31] D. Ramacciotti, A.-I. Lefterovici, and A. F. Rotundo, A simple quantum algorithm to efficiently prepare sparse states, ArXiv:2310.19309 (2023).
[32] S. McArdle, A. Gilyén, and M. Berta, Quantum state preparation without coherent arithmetic, ArXiv:2210.14892 (2022).
[33] J. Li, General explicit difference formulas for numerical differentiation, J. Comput. Appl. Math. **183**, 29 (2005).
[34] A. M. Childs, Y. Su, M. C. Tran, N. Wiebe, and S. Zhu, Theory of Trotter error with commutator scaling, Phys. Rev. X **11**, 011020 (2021).
[35] A. Gilyén, Y. Su, G. H. Low, and N. Wiebe, in *Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing STOC* (ACM, New York, NY, USA, 2019), p. 193.
[36] W. Yu and A. D. MacKerell, in *Antibiotics: Methods and Protocols*, edited by P. Sass (Springer New York, New York, NY, 2017), p. 85.
[37] P. H. Hünenberger, in *Advanced Computer Simulation: Approaches for Soft Matter Sciences I*, edited by C. Holm and K. Kremer (Springer Berlin Heidelberg, Berlin, Heidelberg, 2005), p. 105.
[38] D. Frenkel and B. Smit, in *Understanding Molecular Simulation*, 2nd ed., edited by D. Frenkel and B. Smit (Academic Press, San Diego, 2002), p. 1 (Chapter 1).
[39] S. Nosé, A unified formulation of the constant temperature molecular dynamics methods, J. Chem. Phys. **81**, 511 (1984).
[40] S. Istrail, in *Proceedings of the Thirty-Second Annual ACM Symposium on Theory of Computing*, STOC '00 (Association for Computing Machinery, New York, 2000), p. 87.
[41] M. Suzuki, General theory of fractal path integrals with applications to many-body theories and statistical physics, J. Math. Phys. **32**, 400 (1991).
[42] D. W. Berry, G. Ahokas, R. Cleve, and B. C. Sanders, Efficient quantum algorithms for simulating sparse Hamiltonians, Commun. Math. Phys. **270**, 359 (2006).
[43] L. Lin and Y. Tong, Near-optimal ground state preparation, Quantum **4**, 372 (2020).
[44] M. Reiher, N. Wiebe, K. M. Svore, D. Wecker, and M. Troyer, Elucidating reaction mechanisms on quantum computers, Proc. Natl. Acad. Sci. **114**, 7555 (2017).
[45] N. M. Tubman, C. Mejuto-Zaera, J. M. Epstein, D. Hait, D. S. Levine, W. Huggins, Z. Jiang, J. R. McClean, R. Babbush, M. Head-Gordon, and K. B. Whaley, Postponing the orthogonality catastrophe: Efficient state preparation for electronic structure simulations on quantum devices, ArXiv:1809.05523 (2018).
[46] S. Fomichev, K. Hejazi, M. S. Zini, M. Kiser, J. F. Morales, P. A. M. Casares, A. Delgado, J. Huh, A.-C. Voigt, J. E. Mueller, and J. M. Arrazola, Initial state preparation for quantum chemistry on quantum computers, ArXiv:2310.18410 (2023).
[47] A. Gilyén and T. Li, Distributional property testing in a quantum world, ArXiv:1902.00814 (2019).
[48] A. J. Short and T. C. Farrelly, Quantum equilibration in finite time, New J. Phys. **14**, 013063 (2012).
[49] F. Baras and O. Politano, Molecular dynamics simulations of nanometric metallic multilayers: Reactivity of the Ni-Al system, Phys. Rev. B **84**, 024113 (2011).
[50] G. Lamoureux and B. Roux, Modeling induced polarization with classical drude oscillators: Theory and molecular dynamics simulation algorithm, J. Chem. Phys. **119**, 3025 (2003).
[51] G. Bussi, D. Donadio, and M. Parrinello, Canonical sampling through velocity rescaling, J. Chem. Phys. **126**, 014101 (2007).
[52] S. C. Harvey, R. K.-Z. Tan, and T. E. Cheatham III, The flying ice cube: Velocity rescaling in molecular dynamics leads to violation of energy equipartition, J. Comput. Chem. **19**, 726 (1998).
[53] D. Marx and J. Hutter, in *Modern Methods and Algorithms of Quantum Chemistry* (2000), Vol. 1, p. 301, https://scholar.google.ca/scholarq=Ab+initio+molecular+dynamics+Theory+and+Implementation&hl=en&as_sdt=0&as_vis=1&oi=scholart.
[54] D. W. Berry, A. M. Childs, R. Cleve, R. Kothari, and R. D. Somma, in *Proceedings of the Forty-Sixth Annual ACM Symposium on Theory of Computing*, STOC '14 (Association for Computing Machinery, New York, 2014), p. 283.
[55] S. Jin and N. Liu, Quantum algorithms for computing observables of nonlinear partial differential equations, ArXiv:2202.07834 (2022).
[56] G. H. Low and I. L. Chuang, Hamiltonian simulation by qubitization, Quantum **3**, 163 (2019).
[57] A. M. Childs and N. Wiebe, Hamiltonian simulation using linear combinations of unitary operations, Quantum Inf. Comput. **12**, 901 (2012).
[58] J. M. Martyn, Y. Liu, Z. E. Chin, and I. L. Chuang, Efficient fully-coherent quantum signal processing algorithms for real-time dynamics simulation, J. Chem. Phys. **158**, 024106 (2023).
[59] C. Gidney, Asymptotically efficient quantum Karatsuba multiplication, ArXiv:1904.07356 (2019).
[60] M. A. Nielsen and I. L. Chuang, *Quantum Computation and Quantum Information: 10th Anniversary Edition* (Cambridge University Press, Cambridge, England, 2010).
[61] S. Chakraborty, A. Gilyén, and S. Jeffery, in *46th International Colloquium on Automata, Languages, and*

Programming (ICALP 2019), Leibniz International Proceedings in Informatics (LIPIcs), edited by C. Baier, I. Chatzigiannakis, P. Flocchini, and S. Leonardi (Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, Dagstuhl, Germany, 2019), Vol. 132, p. 33:1.

[62] A. Gilyén, S. Arunachalam, and N. Wiebe, in *Proceedings of the 2019 Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)* (SIAM, San Diego, California, 2019), p. 1425.

[63] R. A. Horn and C. R. Johnson, in *Matrix Analysis* (Cambridge University Press, Cambridge, England, 1985), p. 343.

[64] C. Gidney, Halving the cost of quantum addition, Quantum **2**, 74 (2018).

[65] M. Fannes, A continuity property of the entropy density for spin lattice systems, Commun. Math. Phys. **31**, 291 (1973).

[66] K. M. R. Audenaert, A sharp continuity estimate for the von Neumann entropy, J. Phys. A: Math. Theor. **40**, 8127 (2007).

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

[68] E. Ott, *Chaos in Dynamical Systems* (Cambridge University Press, Cambridge, England, 2002), 2nd ed.