Fullqubit alchemist: Quantum algorithm for alchemical free energy calculations

Po-Wei Huang,[1, 2, *] Gregory Boyd,[1] Gian-Luca R. Anselmetti,[3] Matthias Degroote,[3] Nikolaj Moll,[3] Raffaele Santagati,[3] Michael Streif,[3] Benjamin Ries,[4] Daniel Marti-Dafcik,[1, 5] Hamza Jnane,[1, 6] Sophia Simon,[7] Nathan Wiebe,[8, 9, 10] Thomas R. Bromley,[1, †] and Bálint Koczor[1, 2, ‡]

[1] *Quantum Motion, 9 Sterling Way, London N7 9HJ, United Kingdom*
[2] *Mathematical Institute, University of Oxford, Woodstock Road, Oxford OX2 6GG, United Kingdom*
[3] *Quantum Lab, Boehringer Ingelheim, 55218 Ingelheim am Rhein, Germany*
[4] *Medicinal Chemistry, Boehringer Ingelheim Pharma GmbH & Co KG, 88397 Biberach, Germany*
[5] *Physical and Theoretical Chemistry Laboratory, University of Oxford, South Parks Road, Oxford, OX1 3QZ, United Kingdom*
[6] *Department of Materials, University of Oxford, Parks Road, Oxford OX1 3PH, United Kingdom*
[7] *Department of Physics, University of Toronto, Canada*
[8] *Department of Computer Science, University of Toronto, Canada*
[9] *Pacific Northwest National Laboratory, Richland WA, USA*
[10] *Canadian Institute for Advanced Studies, Toronto, Canada*
(Dated: August 21, 2025)

Accurately computing the free energies of biological processes is a cornerstone of computer-aided drug design but it is a daunting task. The need to sample vast conformational spaces and account for entropic contributions makes the estimation of binding free energies very expensive. While classical methods, such as thermodynamic integration and alchemical free energy calculations, have significantly contributed to reducing computational costs, they still face limitations in terms of efficiency and scalability. A Liouvillian-simulation-based quantum algorithm for molecular dynamics was recently introduced, which achieves a computational advantage by simulating complex hybrid quantum-classical systems using quantum computers. In this work, we present a quantum algorithm for the estimation of free energy differences by adapting the existing Liouvillian approach and introducing several key algorithmic improvements. First, as opposed to simple Trotterization, we directly implement the Liouvillian operator and provide an efficient description of electronic forces acting on both nuclear and electronic particles on the quantum ground state potential energy surface. This leads to super-polynomial runtime scaling improvements in the precision of our Liouvillian simulation approach. Second, our algorithm calculates free energy differences via a fully quantum implementation of thermodynamic integration, thereby foregoing expensive entropy estimation subroutines used in prior works. Our results open new avenues towards the application of quantum computers in drug discovery.

I. INTRODUCTION

Biological processes occur in warm, aqueous environments where molecules are in constant thermal motion. Rather than adopting a single geometry, molecules in the human body continuously explore a large ensemble of configurations. As a result, thermodynamic properties such as entropy or free energy depend on the statistical distribution of a vast number of accessible conformations. Accurately estimating these properties *in silico* requires sampling this high-dimensional configuration space, which is very computationally intensive.

Modern computer-aided drug design pipelines rely heavily on the determination of thermodynamic quantities. In particular, the binding free energy between a ligand and its biological target is a key metric for ranking drug candidates [1]. Significant advancements in this field have been driven by two key factors: the exponential growth in computational power — consistent with

Moore's law and the rise of parallel computing architectures — and the development of sophisticated algorithms and various protocols for free energy estimation [2].

Free energy calculations focus primarily on calculating relative binding free energies, which quantify the change in free energy between two different molecular ensembles [3–5]. Thus, in such calculations, instead of evaluating two absolute binding free energies independently, one computes their difference directly. Alchemical free energy methods accomplish this by introducing a continuous, nonphysical transformation from one chemical species into another through a pathway of fictitious intermediary states [6], drawing parallels with transforming elements in the art of alchemy. The main benefit is that because free energy is a state function, the actual physical binding process does not need to be simulated. This flexibility avoids the need to model complex kinetic mechanisms and enables more efficient estimation of free energy differences. Furthermore, for condensed-phase systems such as liquids and solids — typical in drug design — the Gibbs free energy difference can be accurately approximated by the Helmholtz free energy, as the pressure-volume work is negligible under the nearly incompressible conditions of these phases [7, 8]. Among the most widely used techniques to estimate the Gibbs

* po-wei.huang@maths.ox.ac.uk
† thomas@quantummotion.tech
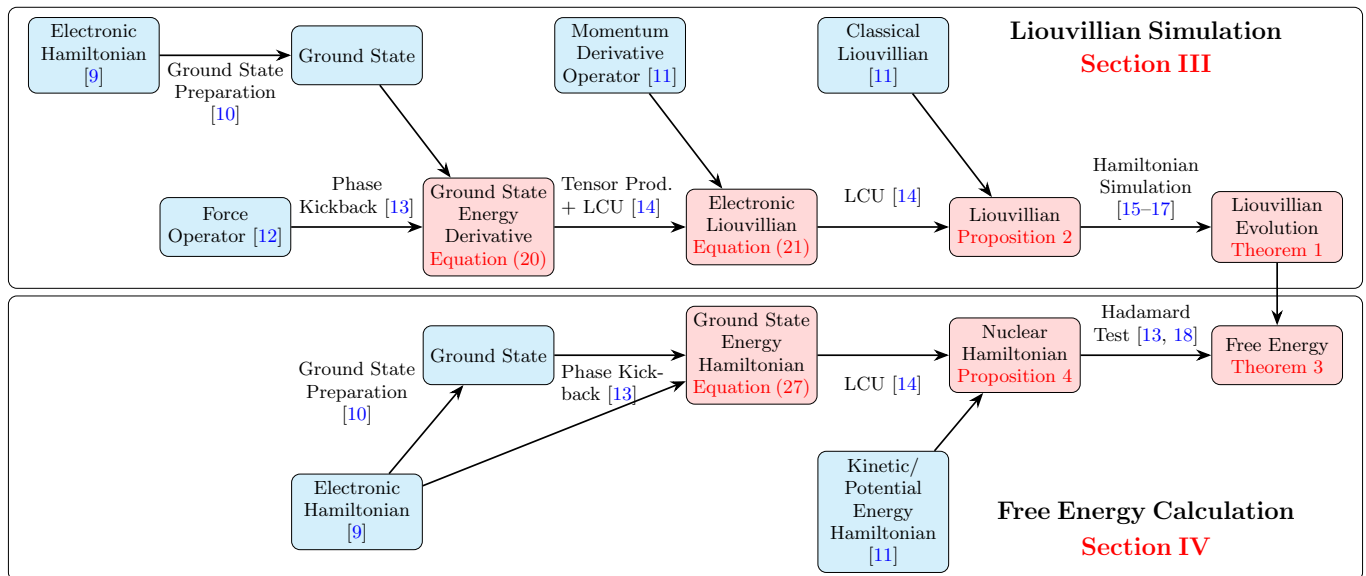‡ balint.koczor@maths.ox.ac.uk

FIG. 1. *Overview of our algorithm construction.* The results in the top panel of the figure regarding the efficient construction of a Liouvillian simulation quantum algorithm are featured in Section III. In addition, the results in the bottom panel of the figure for the design of a quantum algorithm for alchemical free energy calculation are featured in Section IV. In this figure, we color our technical contributions in red, while those from existing work are colored in cyan. For the Liouvillian simulation algorithm, we use the Hellmann-Feynman theorem [19, 20] to efficiently construct a diagonal block encoding of the ground state energy derivative for each point in phase-space, which is then used to construct the electronic and full Liouvillians. Subsequently, using Hamiltonian simulation [15–17], we can implement a Liouvillian evolution operator with logarithmic dependency on precision. Integrating these results with an efficient block encoding of the nuclear Hamiltonian, we can find the free energy difference between two systems through implementing the thermodynamic integration [3] via the Hadamard test [13, 18].

free energy differences via Helmholtz free energy differences are thermodynamic integration [21] and free energy perturbation [22, 23], both of which rely on sampling equilibrium ensembles on classical computers.

Among the available techniques for sampling the configuration space, molecular dynamics is one of the most widely used [3, 24]. It is routinely employed in academic and industrial settings to compute thermodynamic properties from first principles [25]. However, the efficiency of free energy calculations remains a considerable challenge due to the inherent complexity of accurately capturing entropic contributions to the free energy. Reliable sampling of conformational spaces, particularly in systems with high entropic barriers, requires substantial computational resources and robust statistical mechanical frameworks [1, 26].

To address some of these computational bottlenecks, the efficient simulation of molecular and quantum systems on quantum computers has emerged as a promising avenue [27–30]. Quantum simulation techniques have already shown significant improvements in runtime for calculating ground state energies of systems of correlated electrons, with potential applications spanning catalysis, materials science, and drug design [31–36].

Molecular dynamics requires simulating both electronic and nuclear degrees of freedom. In many practical cases, including most drug-like molecules, it is appropriate to treat the nuclei classically while modeling the electrons quantum mechanically, with the nuclei evolving under forces generated by the electrons [37]. Recognizing that the main bottleneck in classical computing lies in the electronic structure calculations, earlier work proposed a hybrid computational paradigm: use a quantum computer to compute electronic forces and then propagate the nuclei according to Newton's equations of motion on a classical computer [12, 38]. While this is a natural way of enhancing existing classical simulation pipelines with quantum computers, the approach is fundamentally limited by measurement overhead as each electronic force must be extracted from the quantum computer via repeated measurements, a process that can take hours to days and must be repeated at each time step [36].

Recent advances extend the runtime advantages of quantum algorithms to hybrid classical–quantum dynamics, enabling a fundamental shift from the paradigm described above [11, 39]. Notably, Simon *et al.* [11] introduced an approach based on the Liouvillian formalism [40] that allows the entire molecular dynamics simulation to be carried out on a quantum computer. By directly applying electronic forces to the classical nuclei via the phase kickback technique [13], it eliminates the need for repeated force measurements. When combined with quantum algorithms for estimating thermodynamic quantities [11], this framework enables the computation of free energies over exponentially large configuration spaces without relying on any classical-quantum feedback

TABLE I. *Overview of results on algorithmic complexity in terms of Toffoli gate count.* Here we consider the complexity only in terms of N, the number of nuclei, $\tilde{N}$, the number of electrons, $N_{\text{tot}} = N + \tilde{N}$, the total number of particles, t, the evolution time of the Liouvillian, ε, the precision of the state preparation/free energy estimation, δ, a lower bound for the overlap between the approximate ground state prepared by an oracle U_I and the true ground state, γ, a lower bound on the spectral gap of the electronic Hamiltonian, η, the number of grid points in phase-space, which is exponential in N, and $1-\xi$, the success probability for free energy estimation. We suppress the dependence on all other variables for simplicity. We use the notation $\mathcal{O}^*(\cdot)$ to hide both polylogarithmic dependencies and $o(1)$ terms in the exponent. Note, however, in terms of free energy calculation, we compute the difference of free energies between two systems while Simon *et al.* [11] computes the absolute free energy of a single system.

	Liouvillian simulation	Free energy calculation
Simon *et al.* [11]	$\mathcal{O}^* \left(\dfrac{N^2 \tilde{N}^2 N_{\text{tot}}^3 t}{\delta \gamma \varepsilon^{o(1)}} \right)$	$\mathcal{O}^* \left(\left(\dfrac{\eta^{o(1)} N^2 \tilde{N}^2 N_{\text{tot}}^3 t}{\delta \gamma \, \varepsilon} \left(N_{\text{tot}}^2 + \dfrac{\eta}{\sqrt{\varepsilon}} \right) + \dfrac{\tilde{N} N_{\text{tot}}^4}{\varepsilon^2} \right) \log \dfrac{1}{\xi} \right)$
Our work	$\tilde{\mathcal{O}} \left(\dfrac{N \tilde{N} N_{\text{tot}}^3 t}{\delta \gamma} \log^3 \dfrac{1}{\varepsilon} \right)$	$\tilde{\mathcal{O}} \left(\dfrac{N \tilde{N} N_{\text{tot}}^5 t}{\delta \gamma \varepsilon} \log \dfrac{1}{\xi} \right)$

loop at every time step.

Nonetheless, these quantum algorithms still face major bottlenecks in computational efficiency. In particular, the quantum simulation method presented in Ref. [11] relies on Trotterization [41, 42], which has a less favorable scaling with respect to error tolerance and evolution time than modern techniques based on block encodings [17, 43–45]. Furthermore, the technique lacks an efficient method for calculating the differences in free energies when comparing different compounds, which is critical for end-to-end applications such as drug discovery.

In this work, we introduce a new quantum algorithm for molecular dynamics that overcomes these limitations. The algorithm relies on an improved version of the original implementation of molecular dynamics in the Liouvillian formalism presented in Ref. [11]. The latter relied on the Suzuki-Trotter product formula [41, 46, 47] to implement the calculation of the electronic structure and to combine the electronic and classical Liouvillians. Here, instead, we introduce a block encoding of the electronic Liouvillian, preparing ground state energy forces on a separate electronic register, and then "kicking-back" the computed values onto the nuclear registers via a form of phase kickback [13]. This enables us to implement the quantum simulation with the full (electronic and classical) Liouvillian by Hamiltonian simulation algorithms [15–17]. In doing so, we achieve a super-polynomial improvement in runtime scaling with respect to precision and substantially decrease the dependency on the number of particles.

We then develop a new quantum algorithm for thermodynamic integration that runs solely on the quantum computer, building on these improvements. This algorithm avoids the need for a direct entropy estimation, which can scale exponentially with the number of atoms in the simulated system [11, 48]. It replaces it with internal energy estimation across interpolated molecular dynamic simulations, which can be implemented efficiently on a quantum computer and be further integrated with amplitude estimation [49] to provide additional speedups
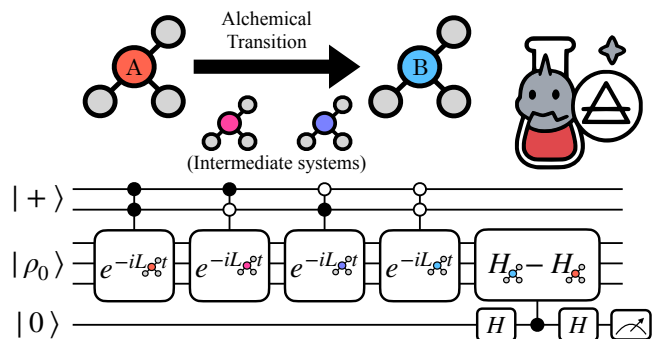

FIG. 2. *Alchemic free energy calculation on a quantum computer.* Given systems A and B, we perform Liouvillian simulations of the two systems and their interpolated intermediate systems in superposition. Note that while the operators are shown separately in this illustration, we implement them as one single operator via controlled weighting of LCU components without the multiplicative cost of executing separate simulations for each intermediate system. We then compute the internal energy difference and sum over the different results via the Hadamard test to provide the free energy difference.

over classical methods.

Figure 1 provides an overview of our quantum algorithms: the top two rows detail the Liouvillian simulation. In addition, the bottom two rows present the computation of free energy via alchemical methods. Using the Hellmann–Feynman theorem [19, 20], we construct diagonal block encodings of energy derivatives to build the Liouvillian operator, which is then used to implement its corresponding evolution operator. Combined with a block encoding of the nulcear Hamiltonian, this enables the computation of free energy differences by implementing thermodynamic integration via the Hadamard test [13, 18]. A sketch of the concept behind the implementation of thermodynamic integration on a quantum computer is provided in Figure 2. Note that the controlled Liouvillian is implemented as a single operator, rather than as separate operators. A comparison of algo-

rithmic complexities in terms of Toffoli gate counts between our work and that of Simon *et al.* [11] is presented in Table I. For the Liouvillian simulation algorithm, we provide a super-polynomial speedup in terms of the precision ε as well as a reduction in the particle numbers N and $\tilde{N}$ by a polynomial factor. On the other hand, for the free energy calculation, we remove the dependency on the number of grid points η, which can scale exponentially with the number of qubits, as well as reduce the dependency on precision ε by a polynomial factor.

The paper is structured as follows. After the Preliminaries in Section II, the improvement of the Liouvillian algorithm by implementing more efficient block encodings is presented in Section III. In Section IV, we devise a quantum algorithm for thermodynamic integration, evolving molecular dynamics through Liouvillian simulation in superposition. Finally, we conclude in Section V.

II. PRELIMINARIES

A. Molecular dynamics within the Liouvillian formalism

Simulating the time evolution of a physical system is a crucial task in physics, with numerous applications, such as molecular dynamics (MD) [3]. MD simulations are often used to evaluate ensemble properties of a physical system, which can be used to estimate macroscopic quantities such as free energies. Different ensembles can be defined. For example, the microcanonical ensemble [50], also known as NVE, is a statistical ensemble where the total number of particles (N), total volume (V), and the total energy (E) are held constant. However, for the calculation of chemically relevant properties, we require the canonical ensemble [50, 51], NVT, where the system is coupled to a heat bath to ensure a constant temperature (T). When coupled to the Nosé thermostat [51, 52], an extra degree of freedom s for the heat bath is introduced into the Hamiltonian. One then integrates over the heat bath variable to obtain the particle distribution after simulations. Furthermore, for many chemical systems and most biologically relevant systems, where nuclear dynamics is much slower than electronic dynamics, the Born-Oppenheimer approximation can achieve reliable predictions of the system's properties [37, 53, 54].

As in Ref. [11], our quantum algorithm operates within the Liouvillian formalism for simulating Koopman–von Neumann mechanics [55–57]. The Liouvillian formalism allows for the description of the time evolution of a physical system through the Liouville equation of motion:

$$\frac{\partial \rho}{\partial t} = -iL\rho , \qquad (1)$$

where $\rho(\boldsymbol{x}, \boldsymbol{p}, t)$ can be thought of as the probability density of a system of N classical particles with position

$\{\boldsymbol{x}_n\}_{n\in[N]}$ and momentum $\{\boldsymbol{p}_n\}_{n\in[N]}$, and where L is the Liouvillian operator, defined as:

$$L := -i\left((\nabla_{\boldsymbol{p}} H_{\text{nuc}}) \cdot \nabla_{\boldsymbol{x}} - (\nabla_{\boldsymbol{x}} H_{\text{nuc}}) \cdot \nabla_{\boldsymbol{p}}\right), \qquad (2)$$

where H_{nuc} is the nuclear Hamiltonian underlying the dynamics of the nuclear system (with additional ground state correction terms from the electronic system). The terms in the sum defining the Liouvillian operator depend directly on the derivatives of the nuclear Hamiltonian H_{nuc} with respect to the j-th components of position and momentum of the n-th particle. To map the classical probability density on a quantum computer, we exploit the Koopman-von Neumann description of classical mechanics, which directly connects the probability density to a quantum state [11, 58], by defining $\rho := \psi_{\text{KvN}}^* \psi_{\text{KvN}}$. Representing the KvN wavefunction ψ_{KvN} as a quantum state $|\rho\rangle$, the time evolution of the system can be computed as the evolution of the density under the Liouvillian:

$$|\rho_t\rangle = e^{-iLt} |\rho_0\rangle . \qquad (3)$$

This can then be implemented on a quantum computer using Hamiltonian simulation [14–17, 59].

Here, we are interested in MD simulations within the Born-Oppenheimer approximation. The force on the nuclei consists of nuclear and electronic contributions, which we can obtain by considering the two components of the Liouvillian

$$L = L_{\text{class}} + L_{\text{el}} , \qquad (4)$$

where the classical part of the Liouvillian L_{class} is given by the kinetic term and nucleus-nucleus repulsion, and L_{el} adds the quantum interaction between the nuclei and electrons and amongst the electrons, which takes the form of the ground state energy. The electronic Liouvillian can thus be formulated as

$$L_{\text{el}} := i \sum_{n=1}^{N} \sum_{j=1}^{3} \left(\frac{\partial E_{\text{el}}(\boldsymbol{x})}{\partial x_{n,j}} \partial_{p_{n,j}}\right) , \qquad (5)$$

where E_{el} is the ground state energy. For simulations under the NVT canonical ensemble with the Nosé thermostat, an additional degree of freedom s and its momentum conjugate p_s are additionally considered in the Liouvillian simulation process, but integrated out to retrieve the final probability density $\rho(\boldsymbol{x}, \boldsymbol{p}, t)$. For implementation on a quantum computer, we prepare a set of registers $|\bar{x}_{n,j}\rangle$, $|\bar{p}'_{n,j}\rangle$, $|\bar{s}\rangle$, and $|\bar{p}_s\rangle$ such that operations are only carried out on the relevant registers, where the overbar stands for the discretized phase-space. The "virtual" momentum $\boldsymbol{p}'$ is a reparameterized version of the momentum dependent on both the true momentum $\boldsymbol{p}$ and the heat bath variable s. Details of the spatial discretization and NVT simulation are deferred to Appendix A.

Prior work by Simon *et al.* [11] has shown that a Liouvillian simulation algorithm can be achieved. They show

how to efficiently block-encode the classical Liouvillian and implement the evolution with Hamiltonian simulation via quantum singular value transform (QSVT) [15–17]. On the other hand, the direct implementation of electronic forces, and by extension, the electronic Liouvillian L_{el}, is circumvented by utilizing controlled Hamiltonian simulation [44], which allows for the implementation $e^{-iL_{\text{el}}t}$ in logarithmic dependency on error. However, since the evolution operator over the electronic Liouvillian L_{el} is implemented directly instead of the full Liouvillian, Trotterization is required to approximate the evolution over the full Liouvillian e^{-iLt}. Their results provide a Liouvillian simulation algorithm with a runtime of

$$\tilde{\mathcal{O}}\left(\frac{N^{2+o(1)}\tilde{N}^{2+o(1)}N_{\text{tot}}^{3+o(1)}t^{1+o(1)}}{\varepsilon^{o(1)}}\right). \quad (6)$$

To provide the relevant electronic forces in the ground state, we make use of an approximate ground state oracle U_I that prepares an initial electronic state $|\phi_{\text{init}}(\boldsymbol{x})\rangle$ that has nontrivial overlap with the ground state $|\psi_0(\boldsymbol{x})\rangle$ of the electronic Hamiltonian H_{el} describing the dynamics of the electronic system. Specifically, given $0 < \delta \leq 1$, we demand that for any $\bar{\boldsymbol{x}}$,

$$U_I |\bar{\boldsymbol{x}}\rangle |0\rangle = |\bar{\boldsymbol{x}}\rangle |\phi_{\text{init}}(\boldsymbol{x})\rangle, \quad (7)$$

where $|\langle\psi_0(\boldsymbol{x})|\phi_{\text{init}}(\boldsymbol{x})\rangle| \geq \delta$ for all nuclear configurations. δ is thus a lower bound on the overlap of the prepared approximate state with the true electronic ground state.

In Appendix A, we formalize the construction of the electronic Liouvillian L_{el} and further provide technical details required from implementation introduce in prior works, including the formulation and cost of implementing the electronic Hamiltonian H_{el} from Ref. [9] and the force operators from Ref. [12] on quantum computers.

B. Alchemical free energy calculation via thermodynamic integration

Thermodynamic quantities such as free energy and entropy can be determined through ensemble averaging over configurations sampled in molecular dynamics or Monte Carlo simulations [23] using appropriate Boltzmann weighting [22]. It is important to distinguish between the Helmholtz free energy, defined at constant volume V as $F = U - TS$, where U is the internal energy and S is entropy, and the Gibbs free energy, defined at constant pressure P as $G = U - TS + PV$ [3]. Since experiments are typically carried out at constant pressure, the Gibbs ensemble provides a more realistic representation of physical conditions. Nevertheless, in the context of molecular simulations relevant to drug discovery in aqueous systems, the difference between Gibbs and Helmholtz free energies is negligible, i.e., $\Delta G \approx \Delta F$, because $P\Delta V$ is nearly zero for drug-like molecules in water, as they are

effectively incompressible [7, 8]. As molecular dynamics often employ constant-volume ensembles, the Helmholtz free energy is often used in computational chemistry. In the following, we will therefore refer to the free energy as F throughout.

We now consider two closely related molecular systems, A and B, with microstate energies $E_{\mathcal{S}}(A)$ and $E_{\mathcal{S}}(B)$ where $\mathcal{S}$ labels the microstates, corresponding to different molecular geometries, in the NVT canonical ensemble. These molecular systems differ by only one atom or functional group, resulting in significant overlap in nuclear phase-space. Computing the absolute free energy of both systems requires extensive sampling for reliable convergence, making the process computationally demanding. As an alternative, we can employ a method that focuses solely on calculating the free energy difference ΔF of systems A and B, which reduces the required sampling and computational cost [2–5, 21].

Consider a new system with microstate energy functions $E_{\mathcal{S}}(\Lambda)$ which are differentiable functions of a new coupling parameter $\Lambda \in [0, 1]$ satisfying

$$E_{\mathcal{S}}(0) = E_{\mathcal{S}}(A) \quad (8)$$
$$E_{\mathcal{S}}(1) = E_{\mathcal{S}}(B). \quad (9)$$

To obtain the free energy in an NVT canonical ensemble, we need to consider the partition function of the system:

$$Z(N, V, T, \Lambda) = \sum_{\mathcal{S}} \exp(-E_{\mathcal{S}}(\Lambda)/k_B T). \quad (10)$$

The microstates $\mathcal{S}$ accessible to the system depend on the number of particles N, the volume V, and the temperature T. A sufficient number of states $\mathcal{S}$ must be chosen so that the partition function converges. Under Liouvillian simulation in the NVT canonical ensemble, we can directly obtain a phase-space density over such states. The Helmholtz free energy of this system is given by

$$F(N, V, T, \Lambda) = -k_B T \ln Z(N, V, T, \Lambda). \quad (11)$$

Helmholtz free energy differences are important because they approximate Gibbs free energy differences in aqueous solutions, and thus enable the calculation of reaction equilibrium constants using the relation $K = e^{-\Delta F/k_B T}$.

However, when solely calculating the free energy difference ΔF, the explicit calculation of the partition function can be avoided. The integral over the derivative of the free energy gives the free energy difference from A to B,

$$\begin{aligned}
\Delta F_{A \to B} &= \int_0^1 \frac{\partial F(\Lambda)}{\partial \Lambda} d\Lambda \\
&= -k_B T \int_0^1 \frac{1}{Z} \frac{\partial Z}{\partial \Lambda} d\Lambda \\
&= \int_0^1 \frac{1}{Z} \sum_{\mathcal{S}} \frac{\partial E_{\mathcal{S}}(\Lambda)}{\partial \Lambda} e^{-E_{\mathcal{S}}(\Lambda)/k_B T} d\Lambda \\
&= \int_0^1 \left\langle \frac{\partial E(\Lambda)}{\partial \Lambda} \right\rangle_\Lambda d\Lambda, \quad (12)
\end{aligned}$$

where the angular brackets denote an ensemble-averaged derivative of the microstate energies with respect to the coupling parameter Λ over the thermal distribution of the states $\mathcal{S}$ for a particular coupling parameter Λ. As noted, the distribution is provided by equilibration via the NVT canonical ensemble, so the partition function need not be further computed. Thus, under this scheme, rather than computing the derivative of the free energy directly, only the expectation value of the derivative of the microstate energy needs to be evaluated [3].

When choosing $E_{\mathcal{S}}(\Lambda)$ as a linear function between the microstate energies of system A and system B,

$$E_{\mathcal{S}}(\Lambda) = E_{\mathcal{S}}(A) + \Lambda(E_{\mathcal{S}}(B) - E_{\mathcal{S}}(A)), \qquad (13)$$

we obtain

$$\Delta F_{A \to B} = \int_0^1 \langle E(B) - E(A) \rangle_\Lambda \, d\Lambda. \qquad (14)$$

In practice, the integral is discretized by calculating the expectation values for a discrete number of values of Λ. Then, to determine the difference in free energy from A to B, the expectation values are summed over all values of Λ, yielding the following approximate free energy difference:

$$\Delta \tilde{F} := \frac{1}{N_\Lambda} \sum_\Lambda \langle E(B) - E(A) \rangle_\Lambda. \qquad (15)$$

Note that while we choose a linear path here for simplicity, selecting a non-linear path may be advantageous because the instantaneous difference in the block encoding constants for H_A and H_B may be chosen to be far smaller for such a thermodynamic integration rather than the differences, which can lead to smaller costs for the overall calculation as discussed in Ref. [60].

Under the Liouvillian formalism, the thermal distribution is obtained by evolving the phase-space density with the NVT canonical Liouvillian for some sufficient equilibration time t_{eq}, while thermal averages are obtained by taking the expectation value of the relevant operator over the thermal distribution. for each point in the phase-space. In our case, where a quantum state on registers represents the thermal distribution, the computational basis corresponds to a point in the discretized phase-space, and the square of the corresponding amplitude would be its probability. The microstate energies over the phase-space can then be encoded into a diagonal matrix to form the nuclear Hamiltonian, with each entry corresponding to the microstate energy to the relevant point in phase-space.

C. Block encodings and quantum linear algebra

We assume access to operators such as Hamiltonians and forces in the form of block encodings [17, 44] in this paper. A block encoding is an encoding scheme that embeds a general matrix into the top left corner of a unitary matrix, such that

$$|A - \alpha(\langle 0^a| \otimes \mathbb{I}_n)U_A(|0^a\rangle \otimes \mathbb{I}_n)| \leq \varepsilon. \qquad (16)$$

This embedding of A into a unitary U_A is then known as a (α, a, ε)-block-encoding of A. In the main text, we refer to α as the scaling factor of the block encoding when mentioned in isolation.

To manipulate such block encodings, we can use a framework known as quantum singular value transformation (QSVT) [17], which can be used to apply a polynomial transformation to the singular values of the matrix A such that

$$\text{QSVT}(U_A) = \begin{bmatrix} \text{Poly}^{(SV)}(A) & * \\ * & * \end{bmatrix}. \qquad (17)$$

In the case of Hermitian matrices, the transformations are applied to the eigenvalues of the matrix. We refer the reader to Appendix B for a detailed account of QSVT as a quantum linear algebra toolbox to perform ground state preparation [10] and Hamiltonian simulation [15–17], which are subroutines used in our main results. The reader will also find a brief review on quantum signal processing (QSP) without angle finding [61]. Furthermore, we deferred several new results to the Appendix that were derived as part of our proofs, such as algorithms for robust ground state preparation and robust Hamiltonian simulation via QSP without angle finding.

III. IMPROVED QUANTUM ALGORITHM FOR LIOUVILLIAN SIMULATION

We provide an algorithm that utilizes evolution under the Liouvillian directly by implementing the block encoding of the electronic Liouvillian from Ref. [11]. We then encode the ground state energy derivative on an electronic register and exploit a form of phase kickback [13] to report the results to the nuclear registers. This allows us to add together the classical and electronic Liouvillians, which we then exponentiate via QSVT (in the same way one would perform Hamiltonian simulation). The benefits of our approach are the following:

1. We achieve super-polynomial asymptotic reductions of the runtime in terms of precision ε from $\mathcal{O}(\varepsilon^{-o(1)})$ in the simulation relative to the quantum algorithm in Ref. [11] to $\mathcal{O}(\text{poly} \log \frac{1}{\varepsilon})$.

2. We decrease the dependency on the number of particles by order of $\tilde{\mathcal{O}}(N^{1+o(1)}\tilde{N}^{1+o(1)}N_{\text{tot}}^{o(1)})$ and time by an order of $\tilde{\mathcal{O}}(t^{o(1)})$ from the removal of Trotterization and from the direct implementation of energy derivatives when compared to Ref. [11].

These advantages are informally summarized in the following theorem, while the formal statement (Theorem C.6) and proof of this theorem are detailed in Appendix C.

Algorithm 1: Quantum algorithm for Liouvillian simulation

Input: Number of nuclei N, Evolution time t, Precision ε, Block encoding of the electronic Hamiltonian $H_{\text{el}}^{\text{ctrl}}$ [9], Block encodings of the nuclear-position-controlled electronic force operator $\partial_{x_{n,j}} H_{\text{el}}^{\text{ctrl}}$ for $n \in [N]$ and $j \in \{1,2,3\}$ [12], Block encoding of the discretized momentum derivative operator $D_{p'_{n,j}}$ for $n \in [N]$ and $j \in \{1,2,3\}$ [11], Block encoding of the classical Liouvillian L_{class} [11], Approximate ground state preparation oracle U_I, Quantum state $|\rho_0\rangle$ encoding initial phase-space density of nuclei and heat bath.

Output: ε-close approximation of $|\rho_t\rangle = e^{-iLt} |\rho_0\rangle$ in ℓ_2 distance

1 **for** *every query to the block encoding L in the QSVT-based Hamiltonian simulation algorithm [17]* **do**

2   Prepare state $\frac{1}{\sqrt{3N}} \sum_{n=1}^{N} |n\rangle \otimes \sum_{j=1}^{3} |j\rangle$ and additional electronic register.

3   Apply approximate ground state preparation oracle U_I to nuclear and electronic registers $U_I |\bar{\boldsymbol{x}}\rangle |0\rangle \to |\bar{\boldsymbol{x}}\rangle |\phi_{\text{init}}(\boldsymbol{x})\rangle$.

4   Apply ground state preparation algorithm with nuclear-position-controlled electronic Hamiltonian $H_{\text{el}}^{\text{ctrl}} = \sum_{\boldsymbol{x}} |\bar{\boldsymbol{x}}\rangle\langle\bar{\boldsymbol{x}}| \otimes H_{\text{el}}(\boldsymbol{x})$.

5   Controlled on $|n\rangle$ and $|j\rangle$, apply nuclear-position-controlled electronic force operator $\partial_{x_{n,j}} H_{\text{el}}^{\text{ctrl}} = \sum_{\boldsymbol{x}} |\bar{\boldsymbol{x}}\rangle\langle\bar{\boldsymbol{x}}| \otimes \frac{\partial H_{\text{el}}(\boldsymbol{x})}{\partial \boldsymbol{x}_{n,j}}$.

6   Uncompute the (approximate) electronic ground state to obtain block-encoding of $D_{n,j}^{\text{el}}$ controlled on $|n\rangle$ and $|j\rangle$.

7   Controlled on $|n\rangle$ and $|j\rangle$, tensor product $D_{n,j}^{\text{el}}$ with discretized momentum derivate operator $D_{p'_{n,j}}$.

8   Use LCU to sum over registers $|n\rangle$ and $|j\rangle$ to obtain block encoding of electronic Liouvillian L_{el}.

9   Use LCU to sum over classical Liouvillian L_{class} and electronic Liouvillian L_{el} to obtain block encoding of Liouvillian L.

10 Apply e^{-iLt} operator obtained from QSVT-based Hamiltonian simulation algorithm to $|\rho_0\rangle$.

11 **return** ε-close approximation of $|\rho_t\rangle = e^{-iLt} |\rho_0\rangle$.

Theorem 1 (Born-Oppenheimer Liouvillian simulation of coupled quantum-classical dynamics; informal). *Given an initial state $|\rho_0\rangle$ that encodes the initial discretized phase-space density and a discretized Liouvillian L under the NVT canonical ensemble, we can construct an explicit quantum algorithm that outputs a quantum state that is ε-close in ℓ_2 distance to $|\rho_t\rangle = e^{-iLt} |\rho_0\rangle$ using*

$$\tilde{\mathcal{O}}\left(\frac{N\tilde{N}N_{\text{tot}}^3 t}{\delta\gamma} \log^3 \frac{1}{\varepsilon} \right)$$

Toffoli gates,

$$\tilde{\mathcal{O}}\left(\frac{1}{\delta} \left(NN_{\text{tot}}t + \log\frac{1}{\gamma\varepsilon} \right) \right)$$

queries to the approximate ground state preparation oracle U_I, and $\tilde{\mathcal{O}}(N_{\text{tot}} + \log\frac{t}{\delta\gamma\varepsilon})$ qubits, where N is the number of nuclei, $\tilde{N}$ is the number of electrons, $N_{\text{tot}} = N + \tilde{N}$ is the total number of particles, δ is the lower bound for the overlap between the approximate and true ground state prepared by U_I and γ is a lower bound on the spectral gap of the electronic Hamiltonian over all nuclear phase-space grid points.

Let us now detail the steps in the construction of the algorithm, which is given above as Algorithm 1. What is required is to prepare the block encoding of the Liouvillian, which can be obtained by computing partial derivatives of the nuclear Hamiltonian (inclusive of nuclear kinetic, nuclear Coulombic, and ground state energy terms) of the system on both position and momentum as shown in Equation (2). Once we have the block encoding

of the Liouvillian, given that it remains self-adjoint, we can utilize a Hamiltonian simulation algorithm [15, 16] to obtain its corresponding evolution operator and apply it to evolve a prepared initial state $|\rho\rangle$ encoding the system's initial phase-space density.

We start with the block encoding of the electronic forces, which is found to be the derivative of the ground state energy function over nuclear positions:

$$D_{n,j}^{\text{el}} := \frac{\partial E_{\text{el}}}{\partial x_{n,j}}(\boldsymbol{x})|\bar{\boldsymbol{x}}\rangle\langle\bar{\boldsymbol{x}}|. \tag{18}$$

We make use of the Hellmann-Feynman theorem [19, 20] to implement this block encoding. This theorem states that the derivative of the total energy with respect to a parameter can be found by computing an expected value of the derivative of the Hamiltonian with respect to that same parameter, such that

$$\frac{\partial E(x)}{\partial x} = \left\langle \psi(x) \left| \frac{\partial H(x)}{\partial x} \right| \psi(x) \right\rangle, \tag{19}$$

where $H(x)$ is a Hamiltonian dependent on the continuous parameter x, $|\psi(x)\rangle$ is an eigenstate of $H(x)$ implicitly dependent on x, and $E(x)$ is the corresponding eigenenergy/eigenvalue.

When combined with the ground state preparation algorithm by Lin and Tong [10], one can utilize the Hellmann-Feynman theorem to encode electronic forces directly into the amplitude, eliminating the need for central finite difference schemes to approximate these values. Usage of the Hellmann-Feynman theorem was previously explored in Ref. [12] under the context of estimating the molecular forces themselves, using a quantum

phase estimation [18] of the Szegedy walk unitary [62] to implement the overlap estimation algorithm [63]. These forces can then be integrated into classical MD methods by replacing subsystems that compute quantum mechanical effects. In this paper, we develop a diagonal block encoding method that encodes the molecular forces for each distinct nuclear position configuration. Rather than leveraging phase estimation to extract the forces, we utilize the Hellmann-Feynman theorem directly as a series of matrix multiplication operations, block-encoding the forces by first computing the ground state, applying the force operator, and then directly uncomputing the ground state. This block-encodes the scalar value of the forces in the top left element.

When implemented in superposition, by applying the controlled operators above, and subsequently projecting the electronic registers into the zero state, we can encode the force values onto the nuclear registers for all different nuclear positions by phase kickback [13]. Therefore, we can implement

$$D_{n,j}^{\text{el}} = \left\langle \psi_0(\boldsymbol{x}) \left| \frac{\partial H(\boldsymbol{x})}{\partial x_{n,j}} \right| \psi_0(\boldsymbol{x}) \right\rangle |\bar{\boldsymbol{x}}\rangle\langle\bar{\boldsymbol{x}}|, \qquad (20)$$

for the discretized grid space of position on the quantum computer. This approach results in the construction of the energy derivative operator with a logarithmic dependency on precision as shown in Lemma C.2. Combining the results with an implementation of the discretized momentum derivative operator $D_{p'_{n,j}}$ shown in Ref. [11], this provides a block encoding of the electronic Liouvillian via linear combination of unitaries [14]:

$$\tilde{L}_{\text{el}} = i \sum_{n=1}^{N} \sum_{j=1}^{3} D_{n,j}^{\text{el}} \otimes D_{p'_{n,j}}. \qquad (21)$$

The details of this block encoding are shown in Proposition C.3. This further provides a logarithmic dependency on the error of the total Liouvillian simulation algorithm, as the Liouvillian is directly and analytically block-encoded.

In addition to decreasing the runtime dependency on precision, by using the Hellmann-Feynman theorem to block-encode the force operator instead of a central finite difference approximation [11], we reduce the scaling factor of the block encoding of the electronic Liouvillian by a factor of $\mathcal{O}(N_{\text{tot}})$ [12]. This is due to the reduced number of terms to implement in the Hellmann-Feynman approach. When combined with the classical Liouvillian, this produces a reduction in the total scaling of $\mathcal{O}(\tilde{N})$ in the full Liouvillian, with the implementation cost and details shown in the proposition below:

Proposition 2 (Block encoding of the Liouvillian; informal). *We can construct an explicit $(\alpha_L, a_L, \varepsilon_L)$-block-encoding of the Liouvillian L where*

$$\alpha_L \in \mathcal{O}\left(NN_{\text{tot}}\right)$$

$$a_L \in \mathcal{O}\left(\tilde{N} + \log \frac{N_{\text{tot}}}{\varepsilon_L}\right).$$

This block encoding can be prepared with $\mathcal{O}(\frac{1}{\delta})$ queries to U_I, and

$$\tilde{\mathcal{O}}\left(\frac{\tilde{N}N_{\text{tot}}^2}{\delta\gamma} \log^2 \frac{1}{\varepsilon_L}\right)$$

Toffoli gates.

The formal statement (Proposition C.5) and proof of this proposition can be found as part of the proof of Theorem 1 in Appendix C. In the context of the Liouvillian simulation algorithm, this results in a reduction in runtime by $\mathcal{O}(\tilde{N})$ in the Hamiltonian simulation step, due to the reduced scaling factor of central finite difference methods.

In addition to the runtime improvements resulting from the direct implementation of the Liouvillian and the use of the Hellmann-Feynman theorem, further speedups can be achieved. Namely, all sublinear dependencies are removed solely by using direct evolution as opposed to Trotterization. Furthermore, by directly implementing the electronic Liouvillian in superposition via LCU [14] instead of taking the product of multiple commuting controlled Hamiltonian simulations, we can effectively reduce the number of queries to the ground state preparation algorithm. This results in a further reduction of the runtime by a factor of $\mathcal{O}(N)$.

Lastly, while our results are primarily focused on the NVT canonical ensemble, these improvements can be easily translated to the NVE microcanonical ensemble by replacing the block encoding of the classical Liouvillian with its NVE counterpart in Ref. [11]. These results are shown in Appendix C 4.

IV. QUANTUM ALGORITHM FOR ALCHEMICAL FREE ENERGY CALCULATION

The results we developed in the previous section give us access to a block encoding of the full Liouvillian as per Proposition 2. This allows us to implement algorithms that require access to weighted sums of Liouvillian terms. With simulations under the NVT ensemble, this enables the efficient preparation of thermal states that can be further used to compute thermal averages such as the free energy.

Prior work in Ref. [11] computes the free energy by first estimating both the internal energy and Gibbs entropy separately, which are then subtracted. While the internal energy can be efficiently estimated by the Hadamard test with the nuclear Hamiltonian, the estimation of Gibbs entropy scales with the dimension η of the density matrix $|\rho_t\rangle\langle\rho_t|$ [48]. Given that $|\rho_t\rangle$ is prepared on a collection of registers $|\bar{x}_{n,j}\rangle$, $|\bar{p}'_{n,j}\rangle$, $|\bar{s}\rangle$, and $|\bar{p}_s\rangle$, the dimension η corresponds to the dimension of the classical nuclear phase-space. If an individual register has dimension g, then

---

Algorithm 2: Quantum algorithm for thermodynamic integration

Input: Number of interpolation points N_Λ, Equilibration time t_{eq}, Precision ε, Failure rate ξ, Block encoding of Liouvillian of systems A/B over nuclear phase-space U_{L_A}/U_{L_B} and scaling factors $\alpha_{L_A}/\alpha_{L_B}$ from Proposition 2, Block encoding of the difference of the nuclear Hamiltonian between systems A and B over nuclear phase-space $U_{\Delta H}$ and scaling factor α_Δ, Initial nuclear phase-space density $|\rho_0\rangle = \sum_{\boldsymbol{x},\boldsymbol{p}',s,p_s} c_{\boldsymbol{x},\boldsymbol{p}',s,p_s} |\bar{\boldsymbol{x}}\rangle |\bar{\boldsymbol{p}}'\rangle |\bar{s}\rangle |\bar{p}_s\rangle$

Output: Helmholtz free energy difference $\Delta F_{A\to B}$ within ε additive error with success rate $1 - \xi$

1 Prepare uniform superposition over N_Λ such that $\frac{1}{\sqrt{N_\Lambda}} \sum_{\bar{\Lambda}=0}^{N_\Lambda-1} |\Lambda\rangle$ where $\bar{\Lambda} = N_\lambda \cdot \Lambda$.

2 Use quantum arithmetic circuits to compute a fixed point representation of $\alpha_\Lambda = \alpha_{L_A} + \Lambda(\alpha_{L_B} - \alpha_{L_A})$ such that we have $\frac{1}{\sqrt{N_\Lambda}} \sum_{\bar{\Lambda}=0}^{N_\Lambda-1} |\Lambda\rangle |\alpha_\Lambda\rangle$.

3 **if** *QSVT-based Hamiltonian simulation [17]* **then**

4   Controlled on $|\alpha_\Lambda\rangle$, compute, on a separate register, the fixed-point representation of phase angles $\{\theta_j^\Lambda\}_{j\in[\mathcal{D}]}$ for polynomial approximation of function $f_\Lambda(x) = \exp(-i\alpha_\Lambda x t_{\text{eq}})$ for degree $\mathcal{D} \in \mathcal{O}(\max(\alpha_{L_A}, \alpha_{L_B})t_{\text{eq}} + \log\frac{1}{\varepsilon})$.

5 **else if** *QSP-without-angle-finding-based Hamiltonian simulation [61]* **then**

6   Controlled on $|\alpha_\Lambda\rangle$, using quantum arithmetic circuits to compute $\exp(-it\alpha_\Lambda \cos\frac{2\pi k}{4\mathcal{D}})$ for $k \in [4\mathcal{D}]$ for degree $\mathcal{D} \in \mathcal{O}(\max(\alpha_{L_A}, \alpha_{L_B})t_{\text{eq}} + \log\frac{1}{\varepsilon})$.

7   Use controlled rotations to create a diagonal block encoding of the values $\exp(-it\alpha_\Lambda \cos\frac{2\pi k}{4\mathcal{D}})$ for $\Lambda \in [N_\Lambda]$, $k \in [4\mathcal{D}]$.

8 **for** *every query to the block encoding L in the Hamiltonian simulation algorithm* **do**

9   Use quantum arithmetic circuits to compute a fixed point representation of $\frac{(1-\Lambda)\alpha_{L_A}}{\alpha_{L_\Lambda}}$ such that we have

        $\frac{1}{\sqrt{N_\Lambda}} \sum_{\bar{\Lambda}=0}^{N_\Lambda-1} |\Lambda\rangle |\alpha_\Lambda\rangle \left|\frac{(1-\Lambda)\alpha_{L_A}}{\alpha_{L_\Lambda}}\right\rangle$

10   Controlled on $\left|\frac{(1-\Lambda)\alpha_{L_A}}{\alpha_{L_\Lambda}}\right\rangle$, take weighted LCU to prepare block encoding of $L_\Lambda = L_A + \Lambda(L_B - L_A)$.

11   Uncompute $\left|\frac{(1-\Lambda)\alpha_{L_A}}{\alpha_{L_\Lambda}}\right\rangle$.

12 Controlled on $|\Lambda\rangle$, obtain $e^{-iL_\Lambda t_{\text{eq}}}$ from Hamiltonian simulation algorithm such that we have $\sum_{\bar{\Lambda}=0}^{N_\Lambda-1} |\Lambda\rangle\langle\Lambda| \otimes e^{-iL_\Lambda t_{\text{eq}}}$.

13 Apply $\sum_{\bar{\Lambda}=0}^{N_\Lambda-1} |\Lambda\rangle\langle\Lambda| \otimes e^{-iL_\Lambda t_{\text{eq}}}$ to $\frac{1}{\sqrt{N_\Lambda}} \sum_{\bar{\Lambda}=0}^{N_\Lambda-1} |\Lambda\rangle \otimes |\rho_0\rangle$.

14 Duplicate register $|\bar{s}\rangle$ using CNOTs to retain information of the bath register.

15 Apply the Hadamard test [13] to the block encoding of ΔH on the registers $|\bar{\boldsymbol{x}}\rangle |\bar{\boldsymbol{p}}'\rangle |\bar{s}\rangle$ to obtain amplitude $\sqrt{\frac{1}{2}\left(\frac{\Delta\tilde{F}}{\alpha_\Delta} + 1\right)}$ on $|0\rangle$ of the measurement qubit.

16 Use amplitude estimation [49] to obtain $\frac{1}{2}\left(\frac{\Delta\tilde{F}}{\alpha_\Delta} + 1\right)$ to $\frac{\varepsilon}{2\alpha_\Delta}$ precision with $1 - \xi$ success rate.

17 **return** ε-close estimate of Helmholtz free energy difference ΔF

$\eta \in \mathcal{O}(g^{6N+2})$ and is hence exponential to the number of atoms. However, this exponential scaling of entropy calculation may not be due to the algorithm itself, as the problem of computing absolute free energies is found to be NP-hard by reducing it to the problem of partition function computation, which is NP-complete [64].

Contrary to the computation of absolute free energies, relative free energies are known to be easier to compute, with various classical algorithms dedicated to this purpose [21, 22]. Aside from the reduced computational complexity, relative free energies are more relevant to applications because they directly determine observable quantities like binding affinities and reaction equilibria [7]. Hence, in our work, we compute relative free energies instead of absolute free energies. Borrowing techniques from classical computational chemistry literature and leveraging alchemical methods described in Section II B enables us to calculate the free energy differences between two closely related systems. With this strategy, we provide the following improvements to calculating free energy differences.

1. Compared to Ref. [11], by implementing thermodynamic integration, we bypass the use of entropy estimation algorithms [48], which scale with the total size of the nuclear phase-space η used in the simulation (unless entropy estimation is evaluated over a coarsened grid) and therefore exponentially with the number of qubits $\Theta(N)$ that are used to represent the nuclear degrees of freedom. Our algorithm is only dependent on calculating the internal energy, which can be achieved efficiently via the Hadamard test and is not dependent on η.

2. We achieve $\tilde{\mathcal{O}}\left(\frac{1}{\varepsilon}\right)$ scaling for Helmholtz free energy estimation by using the Liouvillian simulation algorithm in the previous section, as well as a more efficient block-encoding of the nuclear Hamiltonian that depend only logarithmically on the overall precision — we achieve this by replacing quantum phase estimation [18] for ground state energy computation with phase kickback [13] techniques.

Below, we state our main result:

Theorem 3 (Quantum complexity for thermodynamic integration; informal)**.** *Given an initial state $|\rho_0\rangle$ that encodes the initial discretized phase-space density and two discretized Liouvillians, L_A, and L_B, under the NVT canonical ensemble encoding the dynamics for systems A and B, as well as an equilibration time t_{eq} for Liouvillian simulation, there exists a quantum algorithm that can compute the free energy difference between the two systems up to additive ε-precision with $1 - \xi$ success probability using*

$$\tilde{\mathcal{O}}\left(\frac{N\tilde{N}N_{\text{tot}}^5 t_{\text{eq}}}{\delta\gamma\varepsilon}\log\frac{1}{\xi}\right)$$

Toffoli gates,

$$\tilde{\mathcal{O}}\left(\frac{NN_{\text{tot}}^3 t_{\text{eq}}}{\delta\varepsilon}\log^2\left(\frac{1}{\gamma}\right)\log\left(\frac{1}{\xi}\right)\right)$$

queries to the approximate ground state preparation oracle U_I, and $\tilde{\mathcal{O}}(N_{\text{tot}} + \log\frac{t_{\text{eq}}}{\delta\gamma\varepsilon})$ qubits, where N is the number of nuclei, $\tilde{N}$ is the number of electrons, $N_{\text{tot}} = N + \tilde{N}$ is total number of particles, δ is the lower bound for the overlap on the approximate and true ground state prepared by U_I and γ is a lower bound on the spectral gap of the electronic Hamiltonian over all nuclear phase-space grid points.

The formal statement (Theorem D.6) and proof of this proposition have been deferred to Appendix D.

We are now ready to describe the main algorithm (Algorithm 2). Suppose we want to compute the free energy differences between two systems A and B. Then, to perform thermodynamic integration over Λ in superposition, we need to apply an interpolated NVT Liouvillian L_Λ, which can be easily constructed from the Λ-dependent Hamiltonian $H_\Lambda = (1 - \Lambda)H_A + \Lambda H_B$ as the Liouvillian is linearly dependent on the Hamiltonian as seen in Equation (2). Given this linearity, the corresponding Liouvillian is $L_\Lambda = (1 - \Lambda)L_A + \Lambda L_B$, where L_A and L_B may act on different sets of nuclear registers. Registers describing nuclear degrees of freedom that are shared between A and B can be acted on by both L_A and L_B, or only one of them (if there is an increase or decrease of nuclei).

To compute the free energy, starting with a common initial state $|\rho_0\rangle$, for each Λ, we evolve the state by the interpolated NVT Liouvillian L_Λ over a predetermined and sufficient equilibration time t_{eq} such that we obtain the thermal state $|\rho_\Lambda\rangle = e^{-iL_\Lambda t_{\text{eq}}}|\rho_0\rangle$. Given that the state includes registers corresponding to the heat bath, we then trace out the bath registers in a quantum analog of integrating over them to obtain $\rho_{\text{sys},\Lambda} = \text{Tr}_{\text{bath}}(e^{-iL_\Lambda t_{\text{eq}}}|\rho_0\rangle\langle\rho_0|e^{iL_\Lambda t_{\text{eq}}})$. By the thermodynamic integration algorithm, we can then compute the free energy difference between the systems A and B by computing

$$\Delta\tilde{F} = \frac{1}{N_\Lambda}\sum_\Lambda \text{Tr}(\rho_{\text{sys},\Lambda}(H_B - H_A))$$

$$= \frac{1}{N_\Lambda}\sum_\Lambda \langle\rho_\Lambda|(H_B - H_A)\otimes\mathbb{I}|\rho_\Lambda\rangle \qquad (22)$$

This quantity can be computed on a quantum computer via the Hadamard test [13, 18], where we first create an uniform superposition over N_Λ points with the unitary PREP_Λ, apply time evolution with L_Λ, and a final application of a controlled block encoding of the nuclear Hamiltonian difference $H_B - H_A$. We show an illustration of the quantum circuit in Figure 3.

Delving further into the implementation details, we first discuss the implementation of the evolution of Λ-controlled Liouvillians. From the results of Proposition 2, we can produce block encodings of the Liouvillians L_A and L_B, but with different scaling factors α_{L_A} and α_{L_B}. We block-encode the interpolated Liouvillian L_Λ by a weighted LCU whose weights are dependent on α_{L_A}, α_{L_B}, and Λ, which has a scaling factor of

$$\alpha_\Lambda = (1 - \Lambda)\alpha_{L_A} + \Lambda\alpha_{L_B} \qquad (23)$$

To perform the Hamiltonian simulation by QSVT, the polynomial approximations of $e^{-iL_\Lambda t_{\text{eq}}}$ will also depend on Λ as the number of queries to U_{L_Λ} is dependent on α_Λ. We provide two strategies to perform this transformation of L_Λ, first using standard QSVT, and then an approach using QSVT without angle finding [61] that provides the $\mathcal{O}(\frac{1}{\varepsilon})$ scaling of our algorithm.

Firstly, under conventional QSVT schemes, for each Λ in superposition, we require calculation of the phase angles $\{\theta_\Lambda\}$ with either analytical methods, or with optimization methods [65–67], where the scaling of the number of Λ-s required, or N_Λ, can be shown to be

$$N_\Lambda \in \mathcal{O}\left(\frac{\|L_A - L_B\|\|H_A - H_B\|t_{\text{eq}}}{\varepsilon}\right), \qquad (24)$$

by bounding left Riemann sum errors and where a proof is presented in Appendix D 1. The QSVT sequence must have a length sufficient for the largest polynomial degree required for any Λ. For other values of Λ, the final angles can be set to 0 such that the remaining applications of U_{L_Λ} and its inverse create the identity. In the case of optimization methods, one can set the degree of optimization $\mathcal{D}$ directly to the maximum degree. If the angles are classically loaded, then we need an extra $\mathcal{O}(N_\Lambda)$ arbitrary rotation gates to load all required phase angles per query of U_{L_Λ}, as we need a different phase gate for each Λ.

However, one can bypass this by implementing the classical computations on a separate quantum register in superposition and then loading the results directly into the phase angles, which would then cost at most $\mathcal{O}(\text{poly}\,\mathcal{D}\,\text{poly}\log N_\Lambda)$ additional gates to load the phase angles. Furthermore, suppose we replace the QSVT procedure with generalized quantum signal processing (GQSP) [68]. In that case, it is known that the classical algorithm for computing the phase angles scales as $\mathcal{O}(\mathcal{D}\log\mathcal{D})$. Thus, when implemented on a quantum

FIG. 3. *Circuit for obtaining the free energy difference.* The dashed block indicates the repeated linear combination of block encodings to produce a block encoding of L_Λ, which is then transformed to $e^{-iL_\Lambda t_{eq}}$ via QSVT using Λ-dependent phase angles. Methods for this are discussed in the main text and circuits given in Appendix D 2. Here we illustrate the use of the standard QSVT scheme in Ref. [17], where we would load the phase angles from a separate quantum register and where an angle finding algorithm would be implemented. We ignore the LCU required to add cos and sin together, as well as amplitude amplification for simplicity. Note the CNOT gate that copied the contents of the bath register $|\bar{s}\rangle$ before it is traced out. This is due to the extended Hamiltonian with the Nosé thermostat being dependent on s and is thus required for a description of the thermal state, while still being needed to be traced out in an analog of being integrated over in classical molecular dynamics simulations. The small $|0\rangle$ on the wires indicates that the ancilla qubits are returned to the zero state after robust oblivious amplitude amplification [43] such that they can be reused as the ancilla qubits for the Hadamard test.

computer, we expect the number of gates to be at most $\tilde{\mathcal{O}}(\mathcal{D} \operatorname{poly} \log N_\Lambda)$. However, the precise scaling of such algorithms when implemented on quantum computers, as well as the quantum circuit for such implementations, is not obvious.

An alternative method sidesteps the requirement of angle finding by using recent results on QSVT without classical angle finding [61] by instead encoding a functional representation in a diagonal block encoding, in this case with entries $\exp(-it\alpha_\Lambda \cos \frac{2\pi k}{4\mathcal{D}})$ for $k \in [4\mathcal{D}]$, where $\mathcal{D} \in \mathcal{O}(\max(\alpha_{L_A}, \alpha_{L_B})t_{eq} + \log \frac{1}{\varepsilon})$ is the polynomial approximation degree. Such function entries are computed in superposition via quantum arithmetic circuits while controlled on Λ. This can then be encoded into the amplitude of the diagonal block encoding via controlled rotations or the alternating-sign trick [59]. The reason that the additional cosine in the function is needed is due to the use of qubitized operators, whose eigenvalues are arccos of the original block encoded matrix. This method allows for a gate count scaling $\mathcal{O}(\log^2 N_\Lambda)$ to implement the function representation. Adding the additional $\mathcal{O}(\log^2 N_\Lambda)$ cost to produce the weighted Liouvillian per query to U_{L_Λ}, this requires an additional $\mathcal{O}(\mathcal{D} \log^2 N_\Lambda)$ gates for implementation. On the other hand, this method requires an additional constant factor calls to the block encoding of the interpolated Liouvillians U_{L_Λ} as well as more ancilla qubits when com-

pared to standard QSVT/GQSP. However, it bypasses the requirement of implementing phase angle finding algorithms on a quantum computer, as needed for standard QSVT/GQSP implementations. A more detailed discussion of an exact construction is provided in Appendix D 2.

We note that while the dependency on N_Λ is logarithmic in the runtime, this does not indicate an exponential speedup when compared to classical methods. While conventional thermodynamic integration implementations would evaluate the different N_Λ interpolated systems individually and obtain the free energy via numerical integration methods such as Simpson's rule or specifically designed physics-based fitting functions [69] for higher accuracy, Monte Carlo integration [70] can be applied to obtain faster runtime asymptotics. The number of samples in Monte Carlo integration would then depend on the square of the spectral norm of the nuclear Hamiltonian difference ΔH as opposed to N_Λ, for which the runtime is at most logarithmically dependent on the Monte Carlo integration. This logarithmic dependency stems from the precision of floating point representation on classical computers, it being $\mathcal{O}(N_\Lambda^{-1})$.

Moving on from the implementation of interpolated Liouvillians, one then time evolves the system for a duration t_{eq} that is sufficient to equilibrate all N_Λ Liouvillians $\{L_\Lambda\}$ such that their corresponding phase-space densities ρ_Λ serve as classical thermal distributions. We then ob-

tain the state

$$\frac{1}{\sqrt{N_\Lambda}} \sum_\Lambda |\Lambda\rangle |\rho_\Lambda\rangle \,, \qquad (25)$$

where $|\rho_\Lambda\rangle$ is the state of the nuclear and environmental degrees of freedom produced by the evolution of the Liouvillian L_Λ. Under NVT Liouvillian equilibration, the state obtained by tracing out the bath registers $|\bar{s}\rangle$ and $|\bar{p}_s\rangle$ corresponds to a thermal state of the nuclear registers. Note that the heat bath variable s should still be part of the description of the thermal state as the extended Hamiltonian with the Nosé thermostat is dependent on s. Therefore, s is required for a description of the thermal state, and will have corresponding terms in Hamiltonians H_A and H_B to be interacted upon in the Hadamard test. Thus, we duplicate the contents of $|\bar{s}\rangle$ to a separate register by CNOT gates, which can be traced out while the original is part of the registers that the Hadamard test is applied to [11]. We refer the reader to Appendix A for a more in-depth discussion as to why $|s\rangle$ is needed. Note that this is shown in Figure 3 as the additional CNOT gate from the $|\bar{s}\rangle$ to the $|0\rangle_s$ register.

To obtain the nuclear Hamiltonian differences $\Delta H = H_B - H_A$, we take block encodings of the nuclear Hamiltonian for both systems A and B, where for each system, we have

$$H_{\rm nuc} = H_{\rm kin} + H_{\rm pot} + H_{\rm gse} \,, \qquad (26)$$

where the three terms correspond to diagonal block encodings of the values of the kinetic, potential, and ground state Hamiltonian of the relevant points in phase-space, respectively. While the kinetic and potential energy terms are previously efficiently implemented in Ref. [11], we draw attention to the implementation of $H_{\rm gse}$, which we achieve by preparing the electronic ground state for each point in the discretized grid space for position and applying a block encoding of the electronic Hamiltonian $H_{\rm el}$ before uncomputing the ground states such that

$$\begin{aligned} H_{\rm gse} &= \sum_{\bar{\boldsymbol{x}}} E_{\rm el}(\boldsymbol{x}) |\bar{\boldsymbol{x}}\rangle\langle\bar{\boldsymbol{x}}| \\ &= \sum_{\bar{\boldsymbol{x}}} \langle\psi_0(\boldsymbol{x})|H_{\rm el}(\boldsymbol{x})|\psi_0(\boldsymbol{x})\rangle |\bar{\boldsymbol{x}}\rangle\langle\bar{\boldsymbol{x}}| \,. \end{aligned} \qquad (27)$$

This encodes the ground state energy in the main nuclear registers by phase kickback in a similar fashion to our implementation of encoding electronic forces in Section III. This approach to implementing $H_{\rm gse}$ eliminates the use of phase estimation, as shown in previous approaches [11], and reduces the runtime dependency on precision to a logarithmic scaling.

The difference in energies is then implemented by taking the difference between the block encoding of H_A and H_B via the LCU approach. Here, thermodynamic integration requires the assumption that systems A and B have a relatively high overlap in their nuclear phase-spaces, and thus, most kinetic and potential Hamiltonian

terms can be expected to cancel each other out. However, this is not the case for the ground state energy of the electronic Hamiltonians due to the separate ground state preparations for systems A and B, and would still have a scaling factor of $\mathcal{O}(\tilde{N}N_{\rm tot})$.

In the following result, we construct the difference via LCU for simplicity, while noting that in practice, the block encoding implementation can be made more efficient and with a lower scaling factor.

Proposition 4 (Block encoding of the nuclear Hamiltonian; informal). *We can explicit construct a $(\alpha_H, a_H, \varepsilon_H)$-block-encoding of the nuclear Hamiltonian $H_{\rm nuc}$ where*

$$\alpha_H \in \mathcal{O}\left(N_{\rm tot}^2\right)$$
$$a_H \in \mathcal{O}\left(\tilde{N} + \log\frac{N_{\rm tot}}{\varepsilon_{nuc}}\right)$$

can be prepared using $\mathcal{O}(\frac{1}{\delta})$ queries to U_I, and

$$\tilde{\mathcal{O}}\left(\frac{\tilde{N}N_{\rm tot}^2}{\delta\gamma} \log^2\left(\frac{1}{\varepsilon_{\rm nuc}}\right)\right)$$

Toffoli gates.

The formal statement (Proposition D.5) and proof of this proposition can be found in Appendix D 3. Using our results above, we can block-encode $\Delta H = H_B - H_A$ via LCU with a scaling factor of $\alpha_{H_A} + \alpha_{H_B}$.

Given this block encoding, we extract the free energy value using the Hadamard test. The probability of measuring the $|0\rangle$ state on the Hadamard test ancilla register is

$$\begin{aligned} \mathbb{P}_{\Delta\tilde{F}} &= \frac{1}{2N_\Lambda} \sum_\Lambda \left({\rm Re}\,\langle\rho_\Lambda|U_\Delta|\rho_\Lambda\rangle + 1\right) \\ &= \frac{1}{2N_\Lambda} \sum_\Lambda \left(\frac{\langle H_B - H_A\rangle_\Lambda}{\alpha_\Delta} + 1\right) \\ &= \frac{1}{2}\left(\frac{\Delta\tilde{F}}{\alpha_\Delta} + 1\right) \,, \qquad (28) \end{aligned}$$

giving us the value of the free energy difference. Then, instead of directly measuring the $|0\rangle$ state, we can use amplitude estimation [49] on the result of the Hadamard test. We can obtain the estimation of $\Delta\tilde{F}$ up to ε-precision by $\mathcal{O}(\frac{\alpha_{H_A} + \alpha_{H_B}}{\varepsilon})$ queries to the full Hadamard test circuit. We can treat the superposed Liouvillian simulations as quantum sampling access over the Λ-dependent classical thermal states. Thus, our approach achieves a scaling in terms of the precision ε that is quadratically better than a classical thermodynamic integration modified to utilize Monte Carlo integration [71] without considering potential improvements from implementing MD simulations on quantum computers as mentioned in Section III.

V. CONCLUSION

In this work, we present a quantum algorithm that achieves super-polynomial improvements over prior quantum algorithms for simulating molecular dynamics using a Liouvillian formulation. Building on our approach, we then develop an efficient quantum algorithm for computing free energy differences via alchemical methods.

While previous work [11] focused on block encodings of the classical Liouvillian, we introduce efficient block encodings of the full (electronic and classical) Liouvillian. Our block encoding relies on phase kickback techniques to directly transfer forces and energies computed on electronic registers to the nuclear registers and, therefore, removes the need for phase estimation [18] to obtain electronic forces [12], or the need for Trotterization to interleave separate classical and electronic Liouvillian evolutions [11]. This block encoding can then be used in a Hamiltonian simulation algorithm to provide a super-polynomial improvement in the scaling of the runtime of hybrid quantum-classical molecular dynamics with respect to the precision. Furthermore, by utilizing the Hellmann-Feynman theorem [19, 20] and direct block encodings of electronic force operators [12], we significantly decrease the dependency of the runtime on the number of particles compared to central finite difference methods.

Our fundamentally improved Liouvillian simulation then enables us to develop a quantum algorithm for alchemical free energy calculation that achieves a $\tilde{\mathcal{O}}(\frac{1}{\varepsilon})$ scaling with respect to precision. Our alchemical approach is a key enabler as it forgoes fundamentally more expensive entropy estimation, whose runtime on a quantum computer scales exponentially with the number of qubits representing the nuclear registers — and hence with the number of atoms — due to the exponential growth of the phase-space with the number of particles [11, 48]. Specifically, we use thermodynamic integration [3, 21] to compute the free energy difference between two molecular ensembles directly. This is achieved by defining a thermodynamic path connecting two ensembles and integrating over the change of internal energy along the path.

Our approach performs this numerical integration coherently in a quantum computer: we simulate molecular dynamics at each interpolation point along the path by coherently evolving a superposition of phase-space densities, each under a distinct Liouvillian. A key algorithmic development here is that we construct methods to apply different polynomial transformations in superposition using both standard QSVT/GQSP as well as QSP without angle finding [61], enabling efficient application of different transformations tailored to each block-encoded Liouvillian on the thermodynamic path. Finally, we extract the free energy difference via the Hadamard test followed by amplitude estimation. The gate overhead for this superposition of Liouvillian dynamics is only polylogarithmic in the number of interpolation points that discretize the thermodynamic path.

We now briefly discuss potential advantages of our methods with classical algorithms. Regarding to our improved Liouvillian simulation algorithm, we compare it to classical MD methods, which at best have a scaling of $\mathcal{O}(\varepsilon^{-o(1)})$. Classical MD methods operate by iteratively solving Newton's equations of motion such that the position and momentum of classical particles are updated each time step τ [3, 24]. To obtain an estimation of the phase-space density over time, either a large series of such runs is executed with varying initial conditions to provide a spatial average, or a single run is executed for a significantly long amount of time and provide a time average to replace the spatial average under the assumption that a single long trajectory can explore the phase-space sufficiently. One common method to compute these trajectories is the Verlet integration algorithm [72], a second-order symplectic integrator that yields a global error in the computed trajectory up to $\mathcal{O}(\tau^2)$ for each position and momentum value. Upper bounding the final error by ε, we note that the algorithm runtime would then be $\mathcal{O}(\tau^{-1})$, or $\mathcal{O}(1/\sqrt{\varepsilon})$. While this error is over the position and momentum coordinates, the errors from symplectic integrators can be found to have the same order on the phase-space density by viewing dynamics under the Liouvillian formalism and the Suzuki-Trotter formula [73], paralleling the methods of Ref. [11]. To improve this bound, higher-order symplectic integrators can be used [74], such that for a q-th order symplectic integrator, a $\mathcal{O}(\tau^q)$ error is incurred on the phase-space density, and a small polynomial runtime of $\mathcal{O}(\varepsilon^{-1/q})$ is needed, though these integrators are rarely used in practice. Our quantum algorithm has a super-polynomial speedup with respect to precision over those classical methods, given that our asymptotic complexity depends on the logarithm of the precision. We note that we do not compare our method to classical Monte Carlo methods for generating the thermal state [70] such as the Metropolis-Hastings algorithm [75, 76], whose runtime depends on a mixing time t_{mix} that may be entirely of a different scale compared to the equilibration time t_{eq} for MD simulations, and the underlying mechanism of which is very different from MD methods. We further note that we do not take into account the errors from the discretized phase-space into runtime comparisons and there may be some dependencies on these factors. We provide a brief qualitative discussion of such factors in Appendix E.

Furthermore, the quantum implementation of obtaining electronic ground state energies/forces inherit from the previously discussed advantages of modeling electronic structures on quantum computers [77] as compared to classical methods such as density functional theory (DFT) [78–80] and *ab initio* quantum chemistry methods [81] that are used in hybrid QM/MM (quantum mechanics/molecular mechanics) methods [82, 83] for integrating electronic structure calculations into MD. Advantages of moving such simulations to quantum computers

include the ability to efficiently simulate a large basis set of plane-waves as well as efficient representations of quantum operators with a logarithmic dependency on precision [9, 12], which serves as the basis of the logarithmic dependency of our runtime reductions by phase kickback. Our work thus provides a method for these advantages of modeling electronic structures to be efficiently integrated with the modeling of nuclear interactions.

Thus, the quantum algorithms presented here contain three potential sources of asymptotic speedups over classical methods: (1) the replacement of iterative updates for MD simulation with a single evolution operator, (2) efficient coherent simulation of a large number of molecular dynamics trajectories for thermodynamic integration, and (3) efficient computation of the quantum dynamics of the electronic ground state and associated forces that can potentially be hard to simulate on a classical computer to guaranteed precision, where the last point is of particular value.

While we can achieve $\tilde{\mathcal{O}}(\frac{1}{\varepsilon})$ scaling in terms of precision, we note that so far we have not addressed our algorithm's dependence on the equilibration time t_{eq}. With completely integrable systems, the equilibration time can scale with $\mathcal{O}(\log \frac{1}{\varepsilon})$ with additional weighting over time averages [84]. However, while the Coulomb energy forces give rise to a multidimensional harmonic oscillator, an integrable system, the potential terms provided by the Born-Oppenheimer potential energy surface are generally not believed to be completely integrable (although empirical results show that this bound can be observed for some non-integrable systems [85]). It is therefore generally a challenge to bound this parameter for general systems, and such bounds may depend on the eigenstate thermalization hypothesis (ETH) when applied to open systems [86–89]. We therefore treat t_{eq} as a potentially unbounded parameter in our scaling results.

An alternative quantum computational approach would involve representing both the nuclei and electrons as quantum particles and leveraging algorithms for the preparation of quantum thermal states [90, 91], and is more similar in nature to Monte Carlo methods. It would be an interesting future research direction to compare our quantum algorithm, which uses a hybrid quantum-classical representation of molecules, with such fully-quantum approaches. Currently, it is not obvious which of the algorithms would have the most favorable scaling with respect to the number of particles and whether the progress on bounds on the mixing times of Lindbladians for specific cases [92–95] would yield an asymptotic advantage over the classical equilibration times in the Liouvillian framework we use here.

Looking forward, it will be important to identify concrete scenarios where the alchemical quantum algorithm offers an absolute runtime advantage over classical methods for free energy estimation. This requires obtaining fine-grained resource estimates on systems representative of practical applications to determine constant-level overheads. Performing such an analysis will require the construction of suitable block encodings for the molecular systems as well as the implemenation of an efficient oracle for approximate ground state preparation at all points in phase-space.

Appendix A: Simulating molecular dynamics on a quantum computer

In this section, we provide further details on simulating molecular dynamics on quantum computers, including definitions and implementation costs of relevant operators.

1. Nuclear interactions under the NVT Liouvillian

Under NVT canonical simulations with the Nosé thermostat, the nuclear Hamiltonian has additional terms

concerning an extra degree of freedom s that represents the heat bath that equilibrates the system to a fixed temperature T. The nuclear Hamiltonian governing the dynamics can thus be formulated as follows [11, 51]:

$$H_{\text{nuc}}^{(\text{NVT})} = \sum_{n=1}^{N} \sum_{j=1}^{3} \frac{p_{n,j}'^2}{2m_n s^2} + \sum_{k=1}^{N-1} \sum_{n=k+1}^{N} \frac{Z_n Z_k}{\|x_n - x_k\|}$$
$$+ \frac{p_s^2}{2Q} + N_f k_B T \ln(s) + E_{\text{el}}(\boldsymbol{x}), \quad (A.1)$$

where m_n is the mass of the n-th nucleus, Z_n is the atomic number of the n-th nucleus, p_s is the momentum variable conjugate to s, Q is an effective mass of s that controls the coupling of the system to the heat bath. k_B is the Boltzmann constant, and N_f is the number of degrees of freedom of the system. In this extended system, $p_{n,j}'$ is the conjugate momentum variable to $x_{n,j}$ that is known as a "virtual" momentum variable where the real momentum $p_{n,j}$ can be obtained by

$$p_{n,j} = \frac{p_{n,j}'}{s}. \quad (A.2)$$

The last term E_{el} is the adjustment of the Hamiltonian that adds the ground state electronic energy at a nuclear configuration $\boldsymbol{x}$, which is responsible for the electronic forces.

For operations involving the phase-space density on a quantum computer, we need to consider a finite, discretized phase-space. Under the Born-Oppenheimer approximation, we separate the nuclear and electronic motion, and model them in separate phase-spaces characterized by different sets of quantum registers. For the nuclear phase-space, we discretize the position and momentum and cast the values onto a grid such that

$$g_x := \frac{x_{\max}}{h_x} \in \mathbb{N} \quad (A.3)$$

$$g_{p'} := \frac{p_{\max}'}{h_{p'}} \in \mathbb{N} \quad (A.4)$$

$$g_s := \frac{s_{\max}}{h_s} \in \mathbb{N} \quad (A.5)$$

$$g_{p_s} := \frac{p_{s,\max}}{h_{p_s}} \in \mathbb{N} \quad (A.6)$$

where $x_{\max}/p_{\max}/s_{\max}/p_{s,\max}$ is the maximum value of any $x_{n,j}/p_{n,j}/s_{n,j}/p_{s_n,j}$ and $h_x/h_p/h_s/h_{p_s}$ is the size of the grid spacing. Boundary conditions are assumed to be periodic in our system for simplicity.

When implemented on a quantum computer, each grid point of the discretized phase-space corresponds to a computational basis state in the form of

$$|\bar{\boldsymbol{x}}, \bar{\boldsymbol{p}}', \bar{s}, \bar{p}_s\rangle := \bigotimes_{n,j} \left(|\bar{x}_{n,j}\rangle \otimes |\bar{p}_{n,j}'\rangle \right) \otimes |\bar{s}\rangle \otimes |\bar{p}_s\rangle , \quad (A.7)$$

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

To obtain the NVT Liouvillian, we note that the full formulation is as follows:

$$L^{(\text{NVT})} := -i \sum_{n=1}^{N} \sum_{j=1}^{3} \left(\frac{\partial H_{\text{nuc}}^{(\text{NVT})}}{\partial p_{n,j}'} \partial_{x_{n,j}} - \frac{\partial H_{\text{nuc}}^{(\text{NVT})}}{\partial x_{n,j}} \partial_{p_{n,j}'} \right)$$
$$- i \left(\frac{\partial H_{\text{nuc}}^{(\text{NVT})}}{\partial p_s} \partial_s - \frac{\partial H_{\text{nuc}}^{(\text{NVT})}}{\partial s} \partial_{p_s} \right). \quad (A.8)$$

Dropping the NVT notation, under the discretized phase-space, we obtain

$$L_{\text{disc}} := -i \sum_{n=1}^{N} \sum_{j=1}^{3} \left(\frac{\partial H_{\text{nuc}}}{\partial p_{n,j}'} \otimes D_{x_{n,j}} - \frac{\partial H_{\text{nuc}}}{\partial x_{n,j}} \otimes D_{p_{n,j}'} \right)$$
$$- i \left(\frac{\partial H_{\text{nuc}}}{\partial p_s} \otimes D_s - \frac{\partial H_{\text{nuc}}}{\partial s} \otimes D_{p_s} \right), \quad (A.9)$$

where $D_{x_{n,j}}/D_{p_{n,j}'}/D_s/D_{p_s}$ are discrete derivative approximations obtained by central finite differences, and the partial derivatives on H are analytically obtained from Equation (A.1) and applied to the corresponding registers. The implementation of the NVT Liouvillian, barring the terms related to the ground state energy $E_{\text{el}}(\boldsymbol{x})$, is termed the classical Liouvillian and block-encoded in Ref. [11]. Given the classical Hamiltonian $H_{\text{cl}} = H_{\text{nuc}} - E_{\text{el}}$, we write the discretized classical Liouvillian

$$L_{\text{cl,disc}} := -i \sum_{n=1}^{N} \sum_{j=1}^{3} \left(D_{x_{n,j}} \otimes \frac{\partial H_{\text{cl}}}{\partial p_{n,j}'} - \frac{\partial H_{\text{cl}}}{\partial x_{n,j}} \otimes D_{p_{n,j}'} \right)$$
$$- i \left(D_s \otimes \frac{\partial H_{\text{cl}}}{\partial p_s} - \frac{\partial H_{\text{cl}}}{\partial s} \otimes D_{p_s} \right), \quad (A.10)$$

where

$$\frac{\partial H_{\text{cl}}}{\partial p_{n,j}'} = \sum_{\bar{p}_{n,j}'} \sum_{\bar{s}} \frac{p_{n,j}'}{m_n (s + s_{\min})^2} |\bar{p}_{n,j}'\rangle\langle\bar{p}_{n,j}'| \otimes |\bar{s}\rangle\langle\bar{s}|$$

$$(A.11)$$

$$\frac{\partial H_{\text{cl}}}{\partial x_{n,j}} = \sum_{n' \neq n} \sum_{\bar{x}_n} \sum_{\bar{x}_{n'}} \frac{-Z_n Z_{n'}}{(\|x_n - x_{n'}\|^2 + \Delta^2)^{3/2}} \quad (A.12)$$
$$\times (x_{n,j} - x_{n',j}) |\bar{x}_n\rangle\langle\bar{x}_n| \otimes |\bar{x}_{n'}\rangle\langle\bar{x}_{n'}|$$

$$\frac{\partial H_{\text{cl}}}{\partial p_s} = \sum_{\bar{p}_s} \frac{p_s}{Q} |\bar{p}_s\rangle\langle\bar{p}_s| \quad (A.13)$$

$$\frac{\partial H_{\text{cl}}}{\partial s} = \sum_{\bar{p}_{n,j}} \sum_{\bar{s}} \frac{-2p_{n,j}'^2}{m_n (s + s_{\min})^3} |\bar{p}_{n,j}\rangle\langle\bar{p}_{n,j}| \otimes |\bar{s}\rangle\langle\bar{s}|$$
$$+ \sum_{\bar{s}} \frac{N_f k_B T}{s + s_{\min}} |\bar{s}\rangle\langle\bar{s}|$$

$$(A.14)$$

Thus, the electronic Liouvillian can then be found to be

$$L_{\text{el}} = L - L_{\text{cl}} := i \sum_{n=1}^{N} \sum_{j=1}^{3} \frac{\partial E_{\text{el}}(\boldsymbol{x})}{\partial x_{n,j}} \partial_{p_{n,j}'} \quad (A.15)$$

We discuss this further in the next section.

We briefly note that, given that "virtual" rescaled momentum is used in this simulation, a distortion of dynamics in the aspect of time occurs. Across subsystems with different s values, the length of the time steps is different, and is dependent on s [51]. However, as long as the entire system is under the "virtual" basis, this does not affect the classical thermal state preparation. It is only upon obtaining thermal averages of time-dependent operators, such as the evolution time or average momentum, with the thermal state, that this time distortion requires attention, and a rescaling by s is required on the operator. For our algorithm, we obtain thermal averages on the nuclear Hamiltonian, which is time-independent. Hence, it suffices to describe everything under the "virtual" momentum and time picture for our algorithm.

To obtain thermal averages over the thermal state prepared by the simulation of the NVT Liouvillian, we integrate the degrees of freedom s and p_s, which in the discretized quantum implementation corresponds to taking the sum of their values, or taking a partial trace over the corresponding registers. However, given that the thermal state is dependent on the true momentum $\boldsymbol{p}$ (or alternatively $\boldsymbol{p}'$ and s in the virtual basis) the thermal state must have registers for both $\boldsymbol{p}'$ and s in the virtual basis to constitute a true description. To both be able to trace out the $|s\rangle$ register and retain it for a thermal state description, we duplicate the contents of the $|s\rangle$ register into an ancilla register $|0\rangle_s$ such that we have $|s\rangle \otimes |s\rangle$. We then trace out one of the registers and retain the other one for the thermal state distribution. Note that for our alchemical free energy calculation algorithm, the nuclear Hamiltonian, discretized from the Nosé Hamiltonian, does indeed have s-related terms, hence would require a corresponding $|s\rangle$ register to be applied to.

2. Quantum electronic interactions on Born-Oppenheimer potential energy surface

We now discuss the quantum electronic interactions on the ground state potential energy surface under the Born-Oppenheimer approximation. We provide the formal definition of the electronic Liouvillian for an NVT ensemble under the discretized grid system:

Definition A.1 (Electronic Liouvillian). Let $D_{n,j}^{\mathrm{el}}$ be as follows:

$$D_{n,j}^{\mathrm{el}} = \sum_{\bar{\boldsymbol{x}}} \frac{\partial E_{\mathrm{el}}(\boldsymbol{x})}{\partial x_{n,j}} |\bar{\boldsymbol{x}}\rangle\langle\bar{\boldsymbol{x}}|.$$

and $D_{p'_{n,j}}$ be a discrete derivative approximation to $\partial_{p'_{n,j}}$ obtained by central finite difference as follows:

$$D_{p'_{n,j}} = \frac{1}{h_{p'}} \sum_{k=-d_{p'}}^{d_{p'}} c_{d_{p'},k} |\vec{p}'_{n,j} - k\rangle\langle\vec{p}'_{n,j}|.$$

In the NVT ensemble, the discretized electronic Liouvillian acting on the nuclei is given by

$$L_{\mathrm{el, disc}} := i \sum_{n=1}^{N} \sum_{j=1}^{3} D_{n,j}^{\mathrm{el}} \otimes D_{p'_{n,j}}.$$

Here $x_{n,j}$ and $p'_{n,j}$ are the j-th components of position and associated virtual momentum of the n-th nucleus, respectively. Errors from this discretization scheme are discussed in Appendix E.

In the electronic Liouvillian, we require the operator $D_{n,j}^{\mathrm{el}}$, which encodes the electronic forces on the ground state energy surface in Definition A.1. To compute such forces, we require an extra set of electronic registers and describe the electronic system by evaluating the electronic Hamiltonian acting on electronic orbitals at different points of the nuclear grid space, which is then kicked-back onto the nuclear registers.

For simulations of the electronic Hilbert space, instead of utilizing a grid discretization, we use a finite set of B plane waves as the basis, which takes the following form:

$$\phi_{\boldsymbol{b}}(\boldsymbol{r}) := \frac{1}{\sqrt{\Omega}} e^{-i\boldsymbol{k}_{\boldsymbol{b}} \cdot \boldsymbol{r}} \tag{A.16}$$

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

Under first quantization, the electronic basis states can then be denoted as $|b_0\rangle|b_1\rangle\cdots|b_{\tilde{N}-1}\rangle$, where each $|b_j\rangle$ is a quantum register of size $\lceil \log B \rceil$ specifying the index $b \in [B]$ of the plane wave basis that is occupied by electron j. Under this representation, the electronic Hamiltonian can be efficiently computed from the nuclear position registers and the plane wave momenta through quantum arithmetic circuits [9].

Given $G_0 := [-B^{1/3}, B^{1/3}]^3 \subset \mathbb{Z}^3 \backslash \{(0,0,0)\}$, the block encoding of the electronic Hamiltonian is provided as follows:

Lemma A.1 (Block encoding of the electronic Hamiltonian – Lemma 1, [9]; Formulation from Lemma 3, revised, [11]). *There exists a Hermitian $(\lambda, a_{\mathrm{el}}, \varepsilon)$-block-encoding of the discretized electronic Hamiltonian H_{el} in a plane wave basis*

$$H_{\mathrm{el}}(\boldsymbol{x}) := \sum_{p=1}^{\tilde{N}} \sum_{\boldsymbol{b} \in G} \frac{\|\boldsymbol{k}_{\boldsymbol{b}}\|^2}{2} |\boldsymbol{b}\rangle\langle\boldsymbol{b}|_p$$

$$- \frac{4\pi}{\Omega} \sum_{n=1}^{N} \sum_{p=1}^{\tilde{N}} \sum_{\substack{\boldsymbol{b},\boldsymbol{c} \in G \\ \boldsymbol{b} \neq \boldsymbol{c}}} \left(Z_n \frac{e^{i\boldsymbol{k}_{\boldsymbol{c}-\boldsymbol{b}} \cdot \boldsymbol{x}_n}}{\|\boldsymbol{k}_{\boldsymbol{b}-\boldsymbol{c}}\|^2} \right) |\boldsymbol{b}\rangle\langle\boldsymbol{c}|_p$$

TABLE II. Parameters that determine the complexity of our quantum algorithm for simulating NVT Liouvillian dynamics in the Born-Oppenheimer approximation.

Description	
Evolution time	t
Desired precision	ε
Failure probability	ξ
Number of nuclei and electrons	$N, \tilde{N}$
Mass of the lightest nucleus	$m_{\min}$
Maximum atomic number over all nuclei	$Z_{\max}$
Maximum value of a component of the nuclear position vectors	$x_{\max}$
Maximum value of a component of the (virtual) momentum vectors	$p'_{\max}$
Grid spacing for a component of the discretized variables	$h_x, h_{p'}, h_s, h_{p_s}$
Order of the finite difference scheme used for approximating derivatives	$d_x, d_{p'}, d_s, d_{p_s}$
Gap parameter to regularize the Coulomb potential	Δ
Number of plane wave basis functions in the electronic Hamiltonian	B
Inverse grid spacing for a component of the electronic wave number	h_{el}
Lower bound on the overlap of the initial and true electronic ground state	δ
Lower bound on the spectral gap of H_{el}	γ
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	$p_{s,\max}$
Coupling parameter between systems for thermodynamic integration	Λ
Number of interpolation points in thermodynamic integration	N_Λ

$$+ \frac{2\pi}{\Omega} \sum_{p \neq q=1}^{\tilde{N}} \sum_{\boldsymbol{b},\boldsymbol{c} \in G} \sum_{\substack{\boldsymbol{\nu} \in G_0 \\ (\boldsymbol{b}+\boldsymbol{\nu}) \in G \\ (\boldsymbol{c}-\boldsymbol{\nu}) \in G}} \frac{1}{\|\boldsymbol{k}_{\boldsymbol{\nu}}\|^2}$$
$$\times |\boldsymbol{b}+\boldsymbol{\nu}\rangle\langle\boldsymbol{b}|_p |\boldsymbol{c}-\boldsymbol{\nu}\rangle\langle\boldsymbol{c}|_q,$$

where

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

and

$$a_{\mathrm{el}} \in \mathcal{O}\left(\log \frac{N\tilde{N}B}{\varepsilon} \right).$$

This block encoding can be implemented using

$$\mathcal{O}\left(N + \log B \left(\tilde{N} + \log \frac{Bg_x}{\varepsilon} \right) \right)$$

Toffoli gates.

Note that $\boldsymbol{x}_n$ is the nuclear position in grid space. By combining the above results with a ground state preparation algorithm [10], one can obtain the ground state corresponding to this electronic Hamiltonian and further use this to characterize operations on the ground state surface.

On the other hand, we use the block encoding of the force operator in the Hellmann-Feynman theorem to obtain the derivative of the ground state surface and implement $D_{n,j}^{\mathrm{el}}$. The block encoding of such force operators

in first quantization under the plane wave basis is shown in Section IV C of Ref. [12], and repeated as follows:

Lemma A.2 (Block encoding of $\frac{\partial H_{\mathrm{el}}}{\partial x_{n,j}}$ – [12]). *The electronic Hamiltonian derivative/force operator*

$$\frac{\partial H_{\mathrm{el}}}{\partial x_{n,j}} := \sum_{\boldsymbol{\nu} \in G_0} \frac{2\pi Z_n k_{\nu,j}}{\Omega \|\boldsymbol{k}_{\boldsymbol{\nu}}\|^2} \sum_{p=1}^{\tilde{N}} \sum_{\ell \in \{0,1\}} \left(i e^{-i\boldsymbol{k}_{\boldsymbol{\nu}} \cdot \boldsymbol{x}_n} \right.$$
$$\left. \times \sum_{\boldsymbol{b} \in G} (-1)^{\ell[(\boldsymbol{b}-\boldsymbol{\nu}) \notin G]} |\boldsymbol{b}-\boldsymbol{\nu}\rangle\langle\boldsymbol{b}|_p \right)$$

can be implemented as a $(\tau_n, a_{\mathrm{force}}, \varepsilon_{\mathrm{force}})$-block-encoding, where

$$\tau_n \in \mathcal{O}\left(\frac{\tilde{N}Z_n}{h_{\mathrm{el}}^2} \right)$$

$$a_{\mathrm{force}} \in \mathcal{O}\left(\log \frac{\tilde{N}B}{\varepsilon_{\mathrm{force}}} \right).$$

This block encoding can be implemented using $\mathcal{O}\left(\log B \left(\tilde{N} + \log \frac{Bg_x}{\varepsilon_{\mathrm{force}}} \right) \right)$ *Toffoli gates.*

These results are used as basic building blocks that construct the block encoding of the Liouvillian as shown in Appendix C.

Lastly, we establish an index of variable names that correspond to various physical quantities that are used to implement the Liouvillian operator. The parameters are summarized in Table II.

Appendix B: The quantum linear algebra toolbox

In this section, we provide definitions and useful theorems from the quantum linear algebra toolbox, composing of details on block encodings [16, 44], the quantum singular value transformation (QSVT) [17], its applications in Hamiltonian simulation [16, 17, 96] and ground state preparation [10], as well as quantum signal processing (QSP) without angle finding [61]. For the numbering of definitions and theorems regarding Ref. [17], we refer to the full version on arXiv instead of the shorter published version due to some results we use being omitted in the published version.

1. Block encodings

We first review the constructions of block encodings and their operations.

Definition B.1 (Block encoding – Definition 43, [17], see also [16, 44]). Let $A \in \mathcal{M}_{2^n \times 2^n}(\mathbb{C})$, $\alpha, \varepsilon \in \mathbb{R}_+$ and $a \in \mathbb{N}$. We say that the $(n + a)$-qubit unitary U is a (α, a, ε)-block-encoding of A if

$$\|A - \alpha(\langle 0|_a \otimes \mathbb{I}_n)U(|0\rangle_a \otimes \mathbb{I}_n)\| \leq \varepsilon.$$

Given block encodings of operators A_i, we can construct a block encoding of their linear combination $A = \sum_i y_i A_i$ by using the linear combination of unitaries (LCU) [14] framework as well as a construction known as a "state-preparation pair". Recall that $\|\cdot\|_1$ is the ℓ_1 norm.

Definition B.2 (State preparation pair – Definition 51, [17]). Let $\boldsymbol{y} \in \mathbb{C}^m$ and $\|\boldsymbol{y}\|_1 \leq \beta$. The pair of unitaries (P_L, P_R) is called a $(\beta, b, \varepsilon_{\text{SP}})$-state-preparation-pair for $\boldsymbol{y}$ if

$$P_L |0^b\rangle = \sum_{j=1}^{2^b} c_j |j\rangle , \quad P_R |0^b\rangle = \sum_{j=1}^{2^b} d_j |j\rangle ,$$

such that $\sum_{j=1}^m |y_j - \beta c_j^* d_j| \leq \varepsilon_{\text{SP}}$ and $c_j^* d_j = 0$ for $j = m + 1, \ldots, 2^b$.

In essence, a state preparation pair encodes a set of weights $\boldsymbol{y}$ in the first m elements of a length-2^b column vector whose elements are $c_j^* d_j$, up to an error of ε_{SP}, where β is a normalization factor. To realize the LCU implementation, one would apply P_R on a set of ancilla qubits, then control upon the ancillae, apply controlled versions of the block encoding of A_i, and then apply P_L on the ancillae to produce a block encoding of $A = \sum_i y_i A_i$.

For simplification purposes, in the case where $\boldsymbol{y} \in \mathbb{R}^m$, we assume that $P = P_L = P_R$ and term P as the *state preparation unitary*. The following lemma improves upon the results of Lemma 5 of Ref. [11].

Lemma B.1 (Error and cost of the state preparation unitary). *Let $\boldsymbol{y} \in \mathbb{R}_+^m$ and $\|\boldsymbol{y}\|_1 \leq \beta$. Given state preparation unitary P such that $\left\| P |0\rangle - \sum_j \sqrt{\frac{y_j}{\beta}} |j\rangle \right\| \leq \frac{\varepsilon_{\text{SP}}}{2\beta}$, then (P, P) is $(\beta, b, \varepsilon_{\text{SP}})$-state-preparation-pair for $\boldsymbol{y}$. P can be constructed with $\mathcal{O}(m \log \frac{\beta}{\varepsilon_{\text{SP}}})$ Toffoli gates.*

Proof. From Definition B.2, we know the error of the state preparation pair is found by upper bounding $\sum_{j=1}^m |y_j - \beta c_j^2|$. Note that $\sum_{j=1}^m c_j^2 = 1$. Then

$$\sum_{j=1}^m |y_j - \beta c_j^2| = \beta \sum_{j=1}^m \left| \left(\sqrt{\frac{y_j}{\beta}} - c_j \right) \left(\sqrt{\frac{y_j}{\beta}} + c_j \right) \right|$$
$$= \beta \sum_{j=1}^m \left| \sqrt{\frac{y_j}{\beta}} - c_j \right| \left| \sqrt{\frac{y_j}{\beta}} + c_j \right|$$
$$\leq \beta \sum_{j=1}^m \left| \sqrt{\frac{y_j}{\beta}} - c_j \right| \left(\left| \sqrt{\frac{y_j}{\beta}} \right| + |c_j| \right) .$$
(B.1)

Let $\boldsymbol{v}, \boldsymbol{w}, \boldsymbol{u} \in \mathbb{R}_+^m$ such that $v_j = \left| \sqrt{\frac{y_j}{\beta}} - c_j \right|$, $w_j = \left| \sqrt{\frac{y_j}{\beta}} \right|$, and $u_j = |c_j|$. Then we can write the above as

$$\sum_{j=1}^m |y_j - \beta c_j^2| \leq \beta \boldsymbol{v} \cdot (\boldsymbol{w} + \boldsymbol{u}) \leq \beta \|\boldsymbol{v}\|(\|\boldsymbol{w} + \boldsymbol{u}\|)$$
$$\leq \beta \|\boldsymbol{v}\|(\|\boldsymbol{w}\| + \|\boldsymbol{u}\|)$$
$$= \beta \|\boldsymbol{v}\| \left(\sqrt{\frac{\|\boldsymbol{y}\|_1}{\beta}} + \sqrt{\sum_{j=1}^m c_j^2} \right) \leq 2\beta \|\boldsymbol{v}\| .$$
(B.2)

Now writing out $\|\boldsymbol{v}\|$ we get the following equality:

$$\|\boldsymbol{v}\| = \left\| \sum_{j=1}^m \left| \sqrt{\frac{y_j}{\beta}} - c_j \right| |j\rangle \right\|$$
$$= \left\| \sum_{j=1}^m \sqrt{\frac{y_j}{\beta}} |j\rangle - \sum_{j=1}^m c_j |j\rangle \right\|$$
$$= \left\| \sum_{j=1}^m \sqrt{\frac{y_j}{\beta}} |j\rangle - P |0\rangle \right\| .$$
(B.3)

Thus, setting $\left\| P |0\rangle - \sum_j \sqrt{\frac{y_j}{\beta}} |j\rangle \right\| \leq \frac{\varepsilon_{\text{SP}}}{2\beta}$, we find that $\sum_{j=1}^m |y_j - \beta c_j^2| \leq \varepsilon_{\text{SP}}$, making (P, P) a $(\beta, b, \varepsilon_{\text{SP}})$-state-preparation-pair.

Lastly, given a general quantum state preparation process, we know that preparing a ε-close state to an m-length quantum state requires $\mathcal{O}(m \log \frac{1}{\varepsilon})$ Toffolis [97]. Plugging in relevant error bounds we see that $\mathcal{O}(m \log \frac{\beta}{\varepsilon_{\text{SP}}})$ Toffolis are required. $\square$

The consequences of this improved lemma are that all complexity results in Ref. [11] can be slightly reduced by

a logarithmic factor. Still, under runtime analyses that hide the poly-logarithmic factors, this has no impact.

The following lemma, originally found in Ref. [17], contains a minor typo in the error term, which we have corrected.

Lemma B.2 (Linear combination of block encodings – Lemma 52, revised, [17])**.** *Let $A = \sum_{j=1}^{m} y_j A_j$ be an n-qubit operator and $\varepsilon \in \mathbb{R}^+$. Suppose that (P_L, P_R) is a $(\beta, b, \varepsilon_1)$-state-preparation-pair for $\boldsymbol{y}$ and*

$$W = \sum_{j=1}^{m} |j\rangle\langle j| \otimes U_j + \left(\left(\mathbb{I} - \sum_{j=1}^{m} |j\rangle\langle j| \right) \otimes \mathbb{I}_a \otimes \mathbb{I}_n \right)$$

is an $n+a+b$ qubit unitary such that for all $j \in 1, \dots, m$ we have that U_j is a $(\alpha, a, \varepsilon_2)$-block-encoding of A_j. Then we can implement a $(\alpha\beta, a+b, \alpha\varepsilon_1 + \beta\varepsilon_2)$-block-encoding of A, with a single use of W, P_R and $P_L^\dagger$.

We can also construct a block encoding of a product of two block-encoded matrices.

Lemma B.3 (Product of block encodings – Lemma 53, [17])**.** *Let U_A be a $(\alpha, a, \varepsilon_A)$-block-encoding of A and U_B be a $(\beta, b, \varepsilon_B)$-block-encoding of B, where A, B are n-qubit operators. Then, $(\mathbb{I}_b \otimes U_A)(\mathbb{I}_a \otimes U_B)$ is a $(\alpha\beta, a+b, \alpha\varepsilon_B + \beta\varepsilon_A)$-block-encoding of AB.*

2. Quantum singular value transformation and its applications

Given a block encoding U_A of a Hermitian matrix A, one can use the QSVT [17] to construct a block encoding of a polynomial transformation f of the matrix A, by applying the polynomial to its eigenvalues. More generally, in the case where A is not Hermitian, the polynomial transformation is applied to the singular values of the matrix A.

Lemma B.4 (Quantum polynomial eigenvalue transformation – Theorem 56, [17])**.** *Suppose that U_A is a (α, a, ε)-encoding of a Hermitian matrix A. Given $\xi \geq 0$ and a real $\mathcal{D}$-degree polynomial $f \in \mathbb{R}[x]$ satisfying that*

$$\forall x \in [-1, 1] : |f(x)| \leq \frac{1}{2},$$

one can prepare a $(1, a+2, 4\mathcal{D}\sqrt{\varepsilon/\alpha} + \xi)$-block-encoding of $f(A/\alpha)$ by using $\mathcal{D}$ queries to U_A or its inverse, a single query to controlled-U_A and an additional $\mathcal{O}((a+1)\mathcal{D})$ other one- and two-qubit gates. Further, one can compute a description of such a circuit classically in time $\mathcal{O}(\mathrm{poly}(\mathcal{D}, \log(1/\xi)))$.

Remark B.4.1. For polynomials of definite parity, a $(1, a+1, 4\mathcal{D}\sqrt{\varepsilon/\alpha} + \xi)$-block-encoding of $f(A/\alpha)$ can be constructed. This result can be seen from Corollary 18 of Ref. [17], or indeed, Theorem 2 of its published version.

Applications of the quantum eigenvalue transformation by QSVT lead to an optimal implementation of the Hamiltonian simulation task by producing a polynomial approximation to the function $\exp(-iHt)$ and applying it to a Hamiltonian H. The robust version of the algorithm with regard to block encoding errors is as follows:

Lemma B.5 (Robust Hamiltonian simulation – Corollary 62, [17])**.** *Let $t \in \mathbb{R}$, $\varepsilon \in (0, 1)$ and let U_H be a $(\alpha, a, \frac{\varepsilon}{|2t|})$-block-encoding of the Hamiltonian H. One can then implement a $(1, a+2, \varepsilon)$-block-encoding of e^{itH} with*

$$6\alpha|t| + 9\ln\frac{12}{\varepsilon}$$

queries of U_H or its inverse, 3 queries of controlled-U_H or its inverse, and an additional $\mathcal{O}\left(a\left(\alpha|t| + \log\frac{1}{\varepsilon}\right)\right)$ two-qubit gates.

Corollary B.5.1. *Alternatively, one can use an extra $\mathcal{O}\left(\left(\alpha|t| + \log\frac{1}{\varepsilon}\right)\log\frac{\alpha|t|}{\varepsilon}\right)$ Toffoli gates in place of the arbitrary two-qubit gates.*

Proof. From the ancilla-controlled phase shift operators, we know that a continuous rotation gate is applied per query of U_H, in addition to $\mathcal{O}(1)$ m-ancilla controlled NOT gates.

We first prepare a unitary using Lemma B.5 to prepare a Hamiltonian simulation up to $\frac{\varepsilon}{2}$-precision. Then, using $\mathcal{O}(\log\frac{1}{\varepsilon_{\mathrm{gate}}})$ Toffolis to prepare a continuous rotation gate up to $\varepsilon_{\mathrm{gate}}$-precision in the spectral norm difference of the resulting unitary, we note that by triangular inequality, replacing all continuous rotation gates with Toffolis will result in an error in spectral norm difference of $\mathcal{O}\left((\alpha|t| + \log\frac{1}{\varepsilon})\varepsilon_{\mathrm{gate}}\right)$. Capping the upper bound by $\frac{\varepsilon}{2}$, we note that $\frac{1}{\varepsilon_{\mathrm{gate}}} \in \mathcal{O}\left(\frac{\alpha|t|\log\frac{1}{\varepsilon}}{\varepsilon}\right)$.

The number of Toffolis required is then found to be $\mathcal{O}\left(\left(\alpha|t| + \log\frac{1}{\varepsilon}\right)\log\frac{\alpha|t|}{\varepsilon}\right)$. $\qquad\square$

Further, QSVT can be used for ground state preparation [10] by preparing a reflection operator that flips the eigenstates greater than the ground state upper bound μ and uses amplitude amplification [49] to amplify the ground state. The reflector is constructed by applying the following threshold function to the Hamiltonian's eigenvalues:

$$f_{\mathrm{thresh}}(x) = \begin{cases} -1 & x > \mu \\ 1 & x \leq \mu. \end{cases} \tag{B.4}$$

This produces the block encoding

$$R_\mu(H) = \sum_{k:E_k \leq \mu} |\psi_k\rangle\langle\psi_k| - \sum_{k:E_k > \mu} |\psi_k\rangle\langle\psi_k| \tag{B.5}$$

of the phase oracle used for amplitude amplification. When μ is set to be between the ground state energy and the first excited state energy, $R(H)$ reflects all other

states apart from the ground state and can be used to amplify the latter. We prove and show a robust version of Theorem 6 in Ref. [10] where we do not assume perfect block encoding of the Hamiltonian and do not consider the implementation of perfect continuous rotation gates. If the error of the block encoding is much smaller than that of the Hamiltonian's spectral gap, then Equation (B.4) can still be used to successfully filter out the ground state energy despite its perturbations. To start, we require the following lemma:

Lemma B.6 (Polynomial approximations of the sign function – Lemma 10, [98]; Formulation from Lemma 3, [10]). *For all $0 < \gamma < 1$, $0 < \xi < 1$, there exists an efficiently computable odd polynomial $S(x; \gamma, \xi) \in \mathbb{R}[x]$ of degree $\mathcal{D} = \mathcal{O}(\frac{1}{\gamma} \log \frac{1}{\xi})$, such that*

1. for all $x \in [-1, 1]$, $|S(x; \gamma, \xi)| \leq 1$, and

2. for all $x \in [-1, -\gamma] \cup [\gamma, 1]$, $|S(x; \gamma, \xi) - \text{sign}(x)| \leq \xi$.

Proposition B.7 (Robust ground state preparation with *a priori* ground state energy bound). *Given a Hamiltonian $H = \sum_k E_k |\psi_k\rangle\langle\psi_k| \in \mathbb{C}^{N \times N}$, where $E_k \leq E_{k+1}$ and whose ground state energy and the spectral gap obeys the following inequality: $E_0 \leq \mu - \gamma/2 < \mu + \gamma/2 \leq E_1$, where μ is an upper bound on the ground state energy and γ is a lower bound on the spectral gap of H. Suppose the Hamiltonian is given through its $(\lambda, m, \varepsilon_H)$-block-encoding U_H, where*

$$\varepsilon_H \in \tilde{\mathcal{O}}\left(\frac{\delta^2 \gamma^2 \varepsilon_{\text{prep}}^2}{\lambda}\right),$$

and $\varepsilon_H \leq \frac{\gamma}{4}$. Also suppose we have an initial state $|\phi_{\text{init}}\rangle$, prepared by a unitary U_I, together with a lower bound on the overlap $|\langle\psi_0|\phi_{\text{init}}\rangle| \geq \delta$. Then the ground state $|\psi_0\rangle$ can be prepared with fidelity at least $1 - \varepsilon_{\text{prep}}$ using

$$\mathcal{O}\left(\frac{\lambda}{\delta\gamma} \log \frac{1}{\delta\varepsilon_{\text{prep}}}\right)$$

queries to U_H,

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

queries to U_I, and

$$\mathcal{O}\left(\frac{m\lambda}{\delta\gamma} \log \frac{1}{\delta\varepsilon_{\text{prep}}}\right)$$

one- or two-qubit gates.

Proof. The main algorithm and correctness of results follow from Theorem 6 of Ref. [10]. We now reiterate parts of the proof and make relevant adjustments so that the error in the block encoding is considered. The first step is the shift the block encoding by a factor of μ and from Lemma B.2, we note that we can obtain a $(1, m+1, \frac{\varepsilon_H}{\lambda})$-block-encoding of $\frac{H - \mu I}{\lambda + |\mu|}$. The next step is then to apply

a sign function to $\frac{H - \mu I}{\lambda + |\mu|}$, but before doing so, there are a few things to note. First, due to the error in the Hamiltonian block encoding, all eigenenergies of the Hamiltonian H, and by extension, $H - \mu I$, may be shifted. Let $\tilde{H} = \sum_k \tilde{E}_k |\tilde{\psi}_k\rangle\langle\tilde{\psi}_k|$ be the actual Hamiltonian that is block-encoded. By Weyl's inequality [99], we note the difference in eigenenergies can be upper bounded as follows:

$$|\tilde{E}_k - E_k| \leq \|\tilde{H} - H\| \leq \varepsilon_H. \tag{B.6}$$

Let $\varepsilon_H \leq \frac{\gamma}{4}$. Then by reverse triangle inequality, we can find that the spectral gap for $\frac{H - \mu I}{\lambda + |\mu|}$ is lower bounded by $\frac{\gamma}{2(\lambda + |\mu|)}$, which can be further lower bounded by $\frac{\gamma}{4\lambda}$.

Then, applying the polynomial approximation of the sign function $S(x; \frac{\gamma}{8\lambda}, \xi)$ where for all $x \in [-1, -\frac{\gamma}{8\lambda}] \cup [\frac{\gamma}{8\lambda}, 1]$, $|S(x; \frac{\gamma}{8\lambda}, \xi) - \text{sign}(x)| \leq \xi$, we can still produce a reflection operator that reflects the ground state of $\tilde{H}$ up to error ξ. We now quantify the error on the actual Hamiltonian H with the assurance that the error in the block encoding is small enough not to affect the results of applying the sign function. By Lemma B.6, we note that the polynomial approximation should be taken to degree $\mathcal{D} \in \mathcal{O}(\frac{\lambda}{\gamma} \log \frac{1}{\xi})$. Then by Lemma B.4, we obtain a $(1, m+2, \mathcal{D}\sqrt{\frac{\varepsilon_H}{\lambda}} + \xi)$-block-encoding of $R_\mu(H)$ as defined in Equation (B.5).

Given the reflection operator $R_\mu(H)$ we have now produced, we can use this operator and the approximate ground state oracle U_I to produce the ground state via $\mathcal{O}(\frac{1}{\delta})$ rounds of amplitude amplification [49]. To produce a ground state with at least fidelity $1 - \varepsilon_{\text{prep}}$, we require the error of the block encoding of $R_\mu(H)$ to be within $\delta\varepsilon_{\text{prep}}$. Hence, the equation $\mathcal{D}\sqrt{\frac{\varepsilon_H}{\lambda}} + \xi \leq \delta\varepsilon_{\text{prep}}$ has to be satisfied. Setting $\xi = \frac{\delta\varepsilon_{\text{prep}}}{2}$, we note that $\mathcal{D} \in \mathcal{O}(\frac{\lambda}{\gamma} \log \frac{1}{\delta\varepsilon_{\text{prep}}})$ and

$$\varepsilon_H \in \mathcal{O}\left(\frac{\delta^2 \gamma^2 \varepsilon_{\text{prep}}^2}{\lambda \log^2(1/\delta\varepsilon_{\text{prep}})}\right) \tag{B.7}$$

has to be satisfied, or

$$\varepsilon_H \in \tilde{\mathcal{O}}\left(\frac{\delta^2 \gamma^2 \varepsilon_{\text{prep}}^2}{\lambda}\right). \tag{B.8}$$

Noting that $\mathcal{D}$ is of the same order as when we have a perfect block encoding. The number of queries to U_H, U_I, and the number of additional gates is the same as that of Theorem 6 of Ref. [10]. $\square$

Corollary B.7.1. *Alternatively, the ground state can be prepared to at least $1 - \varepsilon_{\text{prep}}$ using the same number of queries to U_H and U_I, and an extra*

$$\mathcal{O}\left(\frac{\lambda}{\delta\gamma} \log \frac{1}{\delta\varepsilon_{\text{prep}}} \left(m + \log\left(\frac{\lambda}{\delta\gamma\varepsilon_{\text{prep}}}\right)\right)\right)$$

Toffoli gates.

Proof. The extra gates of Proposition B.7 originate from three sources: the weighted LCU to shift the eigenvalues of H by μ, the ancilla-controlled phase shift operators, and the amplitude amplification step. To calculate the number of Toffoli gates required, we first determine how many continuous rotation gates are used. From the weighted LCU, we first construct a block encoding of $H - \mu I$, which requires two continuous rotation gates per query of U_H. From the ancilla-controlled phase shift operators, we know that a continuous rotation gate is applied per query of U_H, in addition to $\mathcal{O}(1)$ m-ancilla controlled NOT gates per query. Lastly, a set of reflectors implemented by $\mathcal{O}(1)$ m-ancilla controlled NOT gates are used per query of U_I in the amplitude amplification step. In total, there are $\mathcal{O}\left(\frac{\lambda}{\delta\gamma}\log\frac{1}{\delta\varepsilon_{\text{prep}}}\right)$ continuous rotation gates and $\mathcal{O}\left(\frac{\lambda}{\delta\gamma}\log\frac{1}{\delta\varepsilon_{\text{prep}}}\right)$ m-ancilla controlled NOT gates.

We first prepare a unitary using Proposition B.7 to prepare a ground state up to $\frac{\varepsilon_{\text{prep}}}{2}$-precision. Then, using $\mathcal{O}(\log\frac{1}{\varepsilon_{\text{gate}}})$ Toffolis to prepare a continuous rotation gate up to $\varepsilon_{\text{gate}}$-precision in the spectral norm difference of the resulting unitary, we note that by triangular inequality, replacing all continuous rotation gates with Toffolis will result in an error in spectral norm difference of $\mathcal{O}\left(\frac{\lambda\varepsilon_{\text{gate}}}{\delta\gamma}\log\frac{1}{\delta\varepsilon_{\text{prep}}}\right)$. Capping the upper bound by $\frac{\varepsilon_{\text{prep}}}{2}$, we note that $\frac{1}{\varepsilon_{\text{gate}}}\in\mathcal{O}\left(\frac{\lambda}{\delta\gamma\varepsilon_{\text{prep}}}\log\frac{1}{\delta\varepsilon_{\text{prep}}}\right)$.

The number of Toffolis required is then found to be $\mathcal{O}\left(\frac{\lambda}{\delta\gamma}\log\frac{1}{\delta\varepsilon_{\text{prep}}}\log\left(\frac{\lambda}{\delta\gamma\varepsilon_{\text{prep}}}\log\frac{1}{\delta\varepsilon_{\text{prep}}}\right)\right)$ from the continuous rotations, and $\mathcal{O}\left(\frac{m\lambda}{\delta\gamma}\log\frac{1}{\delta\varepsilon_{\text{prep}}}\right)$ from the m-controlled NOT gates. $\square$

Remark B.7.1. For a set of Hamiltonians $\{H_i\}$ prepared in superposition such that $H = \sum_i |i\rangle\langle i| \otimes H_i$, and given the approximate ground state oracle U_I such that $U_I |i\rangle |0\rangle \to |i\rangle |\phi_{\text{init,i}}\rangle$, one can use Proposition B.7 to prepare the ground state $|\psi_{0,i}\rangle$ with fidelity at least $1 - \varepsilon_{\text{prep}}$.

Note that the controlled Hamiltonian $H = \sum_i |i\rangle\langle i| \otimes H_i$ can be reexpressed as a direct sum of the terms H_i, or $\bigoplus_i H_i$. The singular value decomposition of a direct sum of matrices is equivalent to taking the direct sum of the decomposed components of the original matrices. Hence, the diagonalized controlled Hamiltonian H is equal to the direct sum of the diagonalized individual Hamiltonians H_i. Hence, we can prepare the reflector in superposition, or $\sum_i |i\rangle\langle i| \otimes R(H_i)$.

To construct the other reflection, given that the approximate ground state oracle can produce $U_I |i\rangle |0\rangle \to |i\rangle |\phi_{\text{init,i}}\rangle$, we apply $U_I (\mathbb{I} \otimes (2|0\rangle\langle 0| - \mathbb{I})) U_I^\dagger$, which is equal to $\sum_i |i\rangle\langle i| \otimes (2|\phi_{\text{init,i}}\rangle\langle\phi_{\text{init,i}}| - \mathbb{I})$.

Thus, the Grover diffusion operator takes the form of

$$-\sum_i |i\rangle\langle i| \otimes \left(R(H_i) \times (2|\phi_{\text{init,i}}\rangle\langle\phi_{\text{init,i}}| - \mathbb{I}) \right), \quad \text{(B.9)}$$

which allows us to prepare the ground state of different Hamiltonians in superposition via amplitude amplification.

3. Quantum signal processing without angle finding

Lastly, we review an alternate version of QSP proposed by Alase [61] which doesn't require the use of phase angle computation on classical computers. Here, we only discuss the specific application of quantum eigenvalue transformation. These methods enable the computation of functional representations in superposition on quantum computers, allowing for the execution of QSP with respect to different functions in superposition.

We provide the following definition of a function modified for the QET setting.

Definition B.3 (Diagonal block encoding of functions – Definition 1, modified, [61]). Let $f : [-1, 1] \to \mathbb{C}$, with $|f(x)| \leq 1$. For $\mathcal{D} = 2^m$, where $m \in \mathbb{N}$, we say a $(1, a, \varepsilon)$-block-encoding encodes function f if

$$\|D_{f,4\mathcal{D}} - (\mathbb{I} \otimes \langle 0^a|)U_{f,4\mathcal{D}}(\mathbb{I} \otimes |0^a\rangle)\| \leq \varepsilon$$

and

$$\langle j'| D_{f,4\mathcal{D}} |j\rangle = \delta_{j,j'} f\left(\cos\frac{2\pi j}{4\mathcal{D}}\right)$$

for $j, j' \in [4\mathcal{D}]$.

Given a $(n+a)$ qubit unitary U_H that block-encodes a n-qubit Hermitian H, we define the following operators:

$$U = (2|0^a\rangle\langle 0^a| - \mathbb{I}_a)U_H \quad \text{(B.10)}$$

$$\mathcal{W}_U = \sum_{j=0}^{4\mathcal{D}-1} U^j \otimes |j\rangle\langle j| \quad \text{(B.11)}$$

$$|\varphi_k\rangle = \sum_{j=0}^{4\mathcal{D}-1} \frac{1}{\sqrt{4\mathcal{D}}} e^{\frac{2i\pi jk}{4\mathcal{D}}} |j\rangle$$

$$= \text{QFT}_{4\mathcal{D}} |k\rangle \quad \text{(B.12)}$$

$$F_{4\mathcal{D}} = \sum_{k=0}^{4\mathcal{D}-1} f\left(\cos\frac{2\pi k}{4\mathcal{D}}\right) |\varphi_k\rangle\langle\varphi_k|$$

$$= \text{QFT}_{4\mathcal{D}} D_{f,4\mathcal{D}} \text{QFT}_{4\mathcal{D}}^\dagger \quad \text{(B.13)}$$

$$|+_{2\mathcal{D}}\rangle = \frac{1}{\sqrt{2\mathcal{D}}} \sum_{j=\mathcal{D}}^{3\mathcal{D}-1} |j\rangle \quad \text{(B.14)}$$

$$|+_{4\mathcal{D}}\rangle = \frac{1}{\sqrt{4\mathcal{D}}} \sum_{j=0}^{4\mathcal{D}-1} |j\rangle . \quad \text{(B.15)}$$

With these operations, one can construct a degree $\mathcal{D}$ Laurent polynomial approximation $f_\mathcal{D}(H)$ such that

$$f_\mathcal{D}(H) = \sqrt{2} \langle +_{2\mathcal{D}}| W_U^\dagger \text{QFT}_{4\mathcal{D}} D_{f,4\mathcal{D}} \text{QFT}_{4\mathcal{D}}^\dagger W_U |+_{4\mathcal{D}}\rangle .$$
$$\text{(B.16)}$$

One would then obtain the following proposition,

Proposition B.8 (Quantum polynomial eigenvalue transform by angleless QSP – Theorem 5, modified, [61]). *Let $f : [-1, 1] \to \mathbb{C}$, with $|f(x)| \leq 1$. Given a $(n + a)$ qubit unitary U_H that block-encodes a n-qubit Hermitian H, and a unitary $U_{f,4\mathcal{D}}$ for $\mathcal{D} = 2^m$ that is a $(1, b, \delta)$-block-encoding, then we can construct a $(\sqrt{2}, m + a + b + 2, (1 + \sqrt{2})E_{\mathcal{D}}(f \circ \cos \circ \arg) + \sqrt{2}\delta)$-block-encoding of $f(H)$ with $4\mathcal{D} - 1$ queries of $c - U$ and $c - U^\dagger$ each, one query to $U_{f,4\mathcal{D}}$, and $\mathcal{O}(\mathcal{D}a)$ Toffoli gates, where $E_{\mathcal{D}}(g)$ is the approximation error of $g(x)$ and its best degree $\mathcal{D}$ Laurent polynomial approximation $g_{\mathcal{D},*}(x)$.*

We note in the conventional QSP, the eigenvalues λ of the Hamiltonian are transformed into $e^{i \arccos \lambda}$ from the qubitization approach [16] and the error stems from the Chebyshev polynomial approximation error $\varepsilon_{\mathcal{D}}(f)$. Hence, the corresponding polynomial error in this approach is that of $f \circ \cos \circ \arg$. Given that $E_{\mathcal{D}}(f \circ \cos \circ \arg)$ is the best Laurent approximation of $f \circ \cos \circ \arg$, we note that $E_{\mathcal{D}}(f \circ \cos \circ \arg) \leq \varepsilon_{\mathcal{D}}(f)$.

We now apply these results to robust Hamiltonian simulation.

Proposition B.9 (Robust Hamiltonian simulation by angleless QSP). *Let $t \in \mathbb{R}$, $\varepsilon \in (0, 1)$ and let U_H be a $(\alpha, a, \frac{\varepsilon}{2|t|})$-block-encoding of the Hamiltonian H. Further let $U_{f,4\mathcal{D}}$ be a $(1, b, \frac{\varepsilon}{24\sqrt{2}})$-block-encoding of the function $f(x) = e^{-i\alpha t x}$. One can then implement a $(1, a+b+3, \varepsilon)$-block-encoding of e^{-itH} with*

$$48\alpha|t| + 72 \ln \frac{48(1 + \sqrt{2})}{\varepsilon} - 6$$

queries of U_H, 3 queries to $U_{f,4\mathcal{D}}$, and an additional $\mathcal{O}\left(a\left(\alpha|t| + \log\frac{1}{\varepsilon}\right)\right)$ Toffoli gates.

Proof. Let $H' = \alpha(\langle 0|_a \otimes I_n)U_H(|0\rangle_a \otimes I_n)$. Then by Lemma 50 of Ref. [44], we have $\|e^{itH} - e^{itH'}\| \leq \|H - H'\|t$ which we use to cap the error from Hamiltonian block encoding to $\frac{\varepsilon}{2|t|}$, and producing a total error of $\frac{\varepsilon}{2}$.

We aim to create a $(\sqrt{2}, a+b+2, \frac{\varepsilon}{6})$-block-encoding of e^{-itH}. Then by Proposition B.8, we see that we require a $(1, b, \frac{\varepsilon}{24\sqrt{2}})$-block-encoding of the function $f(x) = e^{-i\alpha t x}$, or $U_{f,4\mathcal{D}}$, where the error stems for first halving $\frac{\varepsilon}{12}$, then factoring the $\sqrt{2}$ from angleless QSP. From polynomial approximations of $e^{-i\alpha t x}$ as shown in Lemma 57 and 59 of Ref. [17], we require

$$\mathcal{D} = r\left(\frac{e\alpha t}{2}, \frac{\varepsilon}{48(1 + \sqrt{2})}\right) \leq 2\alpha|t| + 3 \ln \frac{48(1 + \sqrt{2})}{\varepsilon} \quad \text{(B.17)}$$

by Proposition B.8 to produce a $\frac{\varepsilon}{48(1+\sqrt{2})}$ approximation for cos and sin separately, resulting in a total $\frac{\varepsilon}{24}$ error when factoring in the $1 + \sqrt{2}$ from angleless QSP.

Given a $(\sqrt{2}, a + b + 2, \frac{\varepsilon}{12})$-block-encoding of $e^{-itH'}$, we then tensor product another Hadamard gate, which would result in a $(2, a+b+3, \frac{\varepsilon}{12})$-block-encoding of e^{-itH}.

Then by one round of robust oblivious amplitude amplification [43], we obtain a $(1, a + b + 3, \frac{\varepsilon}{2})$-block-encoding of e^{-itH}, or a $(1, a+b+3, \varepsilon)$-block-encoding of e^{-itH}. In total, we then need $12\mathcal{D} - 3$ queries of $c - U_H$ and $12\mathcal{D} - 3$ queries of $c - U_H^\dagger$ each from the results of Proposition B.8 and the amplitude amplification, or

$$48\alpha|t| + 72 \ln \frac{48(1 + \sqrt{2})}{\varepsilon} - 6 \quad \text{(B.18)}$$

queries in total.

The Toffoli gates stem from the projection operators in the qubitized Hamiltonian, which are multi-controlled-NOT gates, which can be decomposed into $\mathcal{O}\left(a\left(\alpha|t| + \log\frac{1}{\varepsilon}\right)\right)$ Toffoli gates accordingly. The other source is the decomposition of QFT into Toffolis, which requires a trivial $\mathcal{O}(\log^2 \mathcal{D})$ Toffolis. $\square$

Appendix C: Construction and proof of the Liouvillian simulation algorithm

We detail in this appendix the construction of the Liouvillian simulation algorithm. As mentioned in the main text, the simulation algorithm is realized by a block encoding of the Liouvillian L. With the implementation of the classical Liouvillian shown in Ref. [11], we show the implementation of the electronic Liouvillian L_{el} as defined in Definition A.1.

In the first subsection, we detail the implementation of $D_{n,j}^{\text{el}}$ by the Hellmann-Feynman theorem and ground state preparation subroutines. We then use this to build L_{el} and L in the subsequent subsection, and combine everything to construct the NVT Liouvillian evolution operator e^{-iLt} in the third subsection. Lastly, we provide results on the NVE Liouvillian in the fourth subsection.

1. Implementation of the force diagonal block encoding

We first note that while the nuclear positions required for the implementation of the block encoding of the electronic Hamiltonian are loaded via QROM in Ref. [9], the nuclear positions are input into the electronic Hamiltonian via SWAP gates in Ref. [11]. Thus, this construction provides the following operator:

$$H_{\text{el}}^{\text{ctrl}} = \sum_{\bar{\boldsymbol{x}}} |\bar{\boldsymbol{x}}\rangle\langle\bar{\boldsymbol{x}}| \otimes H_{\text{el}}(\boldsymbol{x}). \quad \text{(C.1)}$$

From Remark B.7.1, we know that Proposition B.7 can be used to prepare all ground states $|\psi_0(\boldsymbol{x})\rangle$ of $H_{\text{el}}^{\text{ctrl}}$ in superposition given the following assumption:

Assumption C.1. *We are given a $(\lambda, a_{\text{el}}, \varepsilon_{\text{el}})$-block-encoding U_H of the nuclear-position-controlled electronic Hamiltonian*

$$H_{\text{el}}^{\text{ctrl}} = \sum_{\bar{\boldsymbol{x}}} |\bar{\boldsymbol{x}}\rangle\langle\bar{\boldsymbol{x}}| \otimes H_{\text{el}}(\boldsymbol{x}).$$

We are given μ and γ such that $E_0(\boldsymbol{x}) \leq \mu - \gamma/2 < \mu + \gamma/2 \leq E_1(\boldsymbol{x})$ for all $\boldsymbol{x}$ and a state preparation oracle U_I that prepares an approximate ground state such that $U_I |\bar{\boldsymbol{x}}\rangle |0\rangle \rightarrow |\bar{\boldsymbol{x}}\rangle |\phi_{\text{init}}(\boldsymbol{x})\rangle$ where for all $\boldsymbol{x}$, $|\langle \psi_0(\boldsymbol{x})|\phi_{\text{init}}(\boldsymbol{x})\rangle| \geq \delta$ is satisfied.

Recall that

$$D_{n,j}^{\text{el}} = \sum_{\bar{\boldsymbol{x}}} \frac{\partial E_{\text{el}}(\boldsymbol{x})}{\partial x_{n,j}} |\bar{\boldsymbol{x}}\rangle\langle\bar{\boldsymbol{x}}| . \qquad (C.2)$$

For each (n, j) pair, we aim to prepare the energy derivative operator $D_{n,j}^{\text{el}}$, which one can do by leveraging the Hellmann-Feynman theorem [19, 20]. From the first quantization formulation of the electronic Hamiltonian, one can analytically derive and then implement the operators $\frac{\partial H_{\text{el}}}{\partial x_{n,j}}$ for each (n, j), the results of which have been shown in Lemma A.2. Given the smaller scaling factor compared to the electronic Hamiltonian, the total runtime can be lowered in the Hamiltonian simulation step when compared to using central finite difference methods to implement the energy derivative.

Similar to the electronic Hamiltonian, when the values of $\boldsymbol{x}$ are input via SWAP operations rather than calling from QROM, the operator that we obtain is a nuclear-position-controlled force operator

$$\sum_{\bar{\boldsymbol{x}}} |\bar{\boldsymbol{x}}\rangle\langle\bar{\boldsymbol{x}}| \otimes \frac{\partial H_{\text{el}}(\boldsymbol{x})}{\partial x_{n,j}} . \qquad (C.3)$$

We now use this operator to encode the derivative of the ground state energy for different nuclear positions, or the operator $D_{n,j}^{\text{el}}$. The circuit detailing this implementation is shown in Figure 4.

Lemma C.2 (Block encoding of $D_{n,j}^{\text{el}}$). *Suppose we have a $(\lambda, a_{\text{el}}, \varepsilon_{\text{el}})$-block-encoding U_H of the nuclear-position-controlled electronic Hamiltonian satisfying Assumption C.1 in regards to spectral gap γ, ground state upper bound μ, and approximate ground state preparation unitary U_I with up to an overlap $1 - \delta$. Further, we are given a $(\tau_n, a_{\text{force}}, \varepsilon_{\text{force}})$-block-encoding of the nuclear-position-controlled electronic force operator as shown in Equation (C.3) by Lemma A.2. Let $\tau = \frac{\tau_n}{Z_n}$. If*

$$\varepsilon_{\text{el}} \in \tilde{\mathcal{O}}\left(\frac{\delta^2 \gamma^2 \varepsilon_{D_E}^4}{\lambda \tau^4 Z_{\max}^4}\right), \quad \varepsilon_{\text{force}} = \frac{\varepsilon_{D_E}}{2}$$

then then exists a $(\tau_n, a_{D_E}, \varepsilon_{D_E})$-block-encoding of $D_{n,j}^{\text{el}}$ where

$$a_{D_E} \in \max(a_{\text{el}}, a_{\text{force}}) + \mathcal{O}(\tilde{N} \log B).$$

This block encoding can be prepared with $\mathcal{O}(\frac{\lambda}{\delta\gamma} \log \frac{\tau Z_{\max}}{\delta \varepsilon_{D_E}})$ queries to U_H, $\mathcal{O}(\frac{1}{\delta})$ queries to U_I, one query to U_{force}, and an additional $\mathcal{O}\left(\frac{\lambda}{\delta\gamma} \log \frac{\tau Z_{\max}}{\delta \varepsilon_{D_E}} \left(\max(a_{\text{el}}, a_{\text{force}}) + \log \frac{\lambda \tau Z_{\max}}{\delta\gamma \varepsilon_{D_E}}\right)\right)$ Toffoli gates.

Proof. We first show the correctness of the block encoding construction. Let W be the ground state preparation unitary such that $W(\boldsymbol{x}) |0\rangle = |\psi_0(\boldsymbol{x})\rangle$. We denote the unitary as $W_{\boldsymbol{x}}$ for simplicity and readability. Assuming perfect block encoding and ground state preparation, we note that preparing the ground state in superposition gives us

$$\sum_{\bar{\boldsymbol{x}}} |\bar{\boldsymbol{x}}\rangle |0\rangle_{\text{anc}} \rightarrow \sum_{\bar{\boldsymbol{x}}} |\bar{\boldsymbol{x}}\rangle (W_{\boldsymbol{x}} |0\rangle) |0\rangle_{\text{anc}} . \qquad (C.4)$$

Then, applying the controlled Hamiltonian derivative operator, we obtain

$$\sum_{\bar{\boldsymbol{x}}} |\bar{\boldsymbol{x}}\rangle \left(\frac{1}{\tau_n} \left(\frac{\partial H_{\text{el}}}{\partial x_{n,j}} W_{\boldsymbol{x}} |0\rangle\right) |0\rangle_{\text{anc}} + \sum_j \lambda_j' |\text{bad}_j\rangle |j\rangle_{\text{anc}}\right). \qquad (C.5)$$

where λ_j' is a normalization factor for some bad state $|\text{bad}_j\rangle$. Then, uncomputing the ground state, we obtain

$$\sum_{\bar{\boldsymbol{x}}} |\bar{\boldsymbol{x}}\rangle \left(\frac{1}{\tau_n} \left(W_{\boldsymbol{x}}^\dagger \frac{\partial H_{\text{el}}}{\partial x_{n,j}} W_{\boldsymbol{x}} |0\rangle\right) |0\rangle_{\text{anc}} + \sum_j \lambda_j' W_{\boldsymbol{x}}^\dagger |\text{bad}_j\rangle |j\rangle_{\text{anc}}\right). \qquad (C.6)$$

By the Hellmann-Feynman theorem, we note the following:

$$W_{\boldsymbol{x}}^\dagger \frac{\partial H_{\text{el}}}{\partial x_{n,j}} W_{\boldsymbol{x}} |0\rangle = \frac{\partial E_{\text{el}}}{\partial x_{n,j}} |0\rangle + \sum_k \lambda_k' |k\rangle , \qquad (C.7)$$

where E_{el} is the ground state energy of H_{el}. Thus, Equation (C.6) can be written as

$$\sum_{\bar{\boldsymbol{x}}} |\bar{\boldsymbol{x}}\rangle \left(\frac{1}{\tau_n} \frac{\partial E_{\text{el}}}{\partial x_{n,j}} |0\rangle_{\text{anc}} + \sum_k \frac{\lambda_k'}{\tau_n} |k\rangle |0\rangle_{\text{anc}} + \sum_j \lambda_j' W_{\boldsymbol{x}}^\dagger |\text{bad}_j\rangle |j\rangle_{\text{anc}}\right). \qquad (C.8)$$

Therefore, when post-selecting on $|0\rangle_{\text{anc}}$, we find that the encoded amplitude on $|\bar{\boldsymbol{x}}\rangle$ is then $\frac{1}{\tau_n} \langle 0| W_{\boldsymbol{x}}^\dagger \frac{\partial H_{\text{el}}}{\partial x_{n,j}} W_{\boldsymbol{x}} |0\rangle = \frac{1}{\tau_n} \frac{\partial E_{\text{el}}}{\partial x_{n,j}}$. Hence, this process block-encodes the electronic forces on the nuclear registers by phase kickback.

Next, we proceed to the factors of the block encoding. The scaling factor follows from the Hamiltonian derivative operator in Lemma A.2, which is $\tau_n \in \mathcal{O}\left(\frac{\tilde{N} Z_n}{h_{\text{el}}^2}\right)$. Given that the electronic register is computed and then uncomputed, when encoding the ground state energy onto the nuclear register, the ancilla qubits are the max of the ancilla qubits between H_{el} and H_{force}, or a_{el} and a_{force}, plus the size of the electronic register, or $\mathcal{O}(\tilde{N} \log B)$.

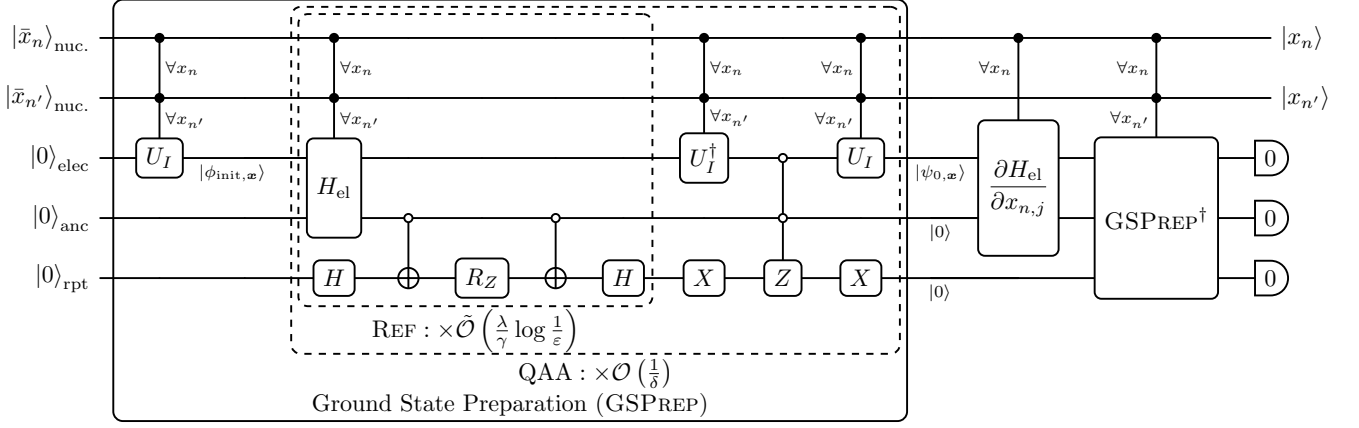FIG. 4. *Block encoding implementation of the $D_{n,j}^{el}$ operator.* This diagonal block encoding is implemented using the Hellmann-Feynman theorem [19, 20], where we first prepare the ground state from the electronic Hamiltonian [10], apply the Hamiltonian derivative operator, and then uncompute the ground state. In this figure, we draw the approximate ground state preparation oracle U_I, electronic Hamiltonian H_{el} and force operator $\frac{\partial H_{el}}{\partial x_{n,j}}$ with controls on the nuclear registers to symbolize that they are nuclear-register controlled. However, these operators are implemented as one single operator and need not be reimplemented for each different set of $\boldsymbol{x}$. In this figure and the figures after this, we symbolize the ancilla qubits of the block encoding by appending a $\boxed{0}$ symbol at the end of the qubit wire.

Regarding the error, we first prepare the ground state of H_{el}^{ctrl} up to fidelity $1 - \varepsilon_{prep}$ with Proposition B.7. This produces a state $|\phi_0\rangle$ such that $|(\langle\psi_0| \otimes \langle 0|)|\phi_0\rangle| \geq 1 - \varepsilon_{prep}$. For cleanliness, we denote $|\tilde{\psi}_0\rangle = |\psi_0\rangle \otimes |0\rangle$. Then the ℓ_2 distance between the two states can be found to be

$$\left\| |\phi_0\rangle - |\tilde{\psi}_0\rangle \right\| = \sqrt{(\langle\phi_0| - \langle\tilde{\psi}_0|)(|\phi_0\rangle - |\tilde{\psi}_0\rangle)}$$
$$= \sqrt{2 - 2|\langle\phi_0|\tilde{\psi}_0\rangle|} \leq \sqrt{2\varepsilon_{prep}}. \quad (C.9)$$

Given that the final block encoding is that of a diagonal matrix on different nuclear positions, the error term of the block encoding (spectral norm difference) is less than that of the maximum difference on a single entry. We derive the difference as follows via a series of triangular inequalities and drop the notation on nuclear positions, as the derivation is for a single position:

$$\left| \tau_n \langle 0| W^\dagger U_{force} W |0\rangle - \frac{\partial E_{el}}{\partial x_{n,j}} \right|$$
$$\leq \tau_n \left| \langle 0| W^\dagger U_{force} W |0\rangle - \langle 0| W^\dagger U_{force} |\tilde{\psi}_0\rangle \right|$$
$$+ \tau_n \left| \langle 0| W^\dagger U_{force} |\tilde{\psi}_0\rangle - \langle\tilde{\psi}_0| U_{force} |\tilde{\psi}_0\rangle \right|$$
$$+ \left| \tau_n \langle\tilde{\psi}_0| U_{force} |\tilde{\psi}_0\rangle - \langle\tilde{\psi}_0| \frac{\partial H_{el}}{\partial x_{n,j}} |\tilde{\psi}_0\rangle \right|. \quad (C.10)$$

We now derive the upper bounds of the three absolute differences. The first term can be bounded by the Cauchy–Schwarz inequality such that

$$\tau_n \left| \langle 0| W^\dagger U_{force} W |0\rangle - \langle 0| W^\dagger U_{force} |\tilde{\psi}_0\rangle \right|$$
$$= \tau_n \left| \langle 0| W^\dagger U_{force} |\phi_0\rangle - \langle 0| W^\dagger U_{force} |\tilde{\psi}_0\rangle \right|$$

$$= \tau_n \left| \langle 0| W^\dagger U_{force} \left(|\phi_0\rangle - |\tilde{\psi}_0\rangle \right) \right|$$
$$\leq \tau_n \left\| \langle 0| W^\dagger U_{force} \right\| \left\| |\phi_0\rangle - |\tilde{\psi}_0\rangle \right\|$$
$$= \tau_n \left\| |\phi_0\rangle - |\tilde{\psi}_0\rangle \right\|, \quad (C.11)$$

where the last equality holds because W is unitary.

The second term can be similarly bounded such that

$$\tau_n \left| \langle 0| W^\dagger U_{force} |\tilde{\psi}_0\rangle - \langle\tilde{\psi}_0| U_{force} |\tilde{\psi}_0\rangle \right|$$
$$= \tau_n \left\| \langle\phi_0| - \langle\tilde{\psi}_0| \right\| \left\| U_{force} |\tilde{\phi}_0\rangle \right\|$$
$$= \tau_n \left\| |\phi_0\rangle - |\tilde{\psi}_0\rangle \right\|, \quad (C.12)$$

The third term can be bounded as follows:

$$\left| \langle\tilde{\psi}_0| \tau_n W^\dagger U_{force} |\tilde{\psi}_0\rangle - \langle\tilde{\psi}_0| \frac{\partial H_{el}}{\partial x_{n,j}} |\tilde{\psi}_0\rangle \right|$$
$$= \left| \langle\psi_0| \left(\tau_n(\mathbb{I} \otimes \langle 0|)U_{force}(\mathbb{I} \otimes |0\rangle) - \frac{\partial H_{el}}{\partial x_{n,j}} \right) |\psi_0\rangle \right|$$
$$\leq \|\langle\psi_0|\| \left\| \tau_n(\mathbb{I} \otimes \langle 0|)U_{force}(\mathbb{I} \otimes |0\rangle) - \frac{\partial H_{el}}{\partial x_{n,j}} \right\| \||\psi_0\rangle\|$$
$$\leq \left\| \left(\tau_n(\mathbb{I} \otimes \langle 0|)U_{force}(\mathbb{I} \otimes |0\rangle) - \frac{\partial H_{el}}{\partial x_{n,j}} \right) \right\| \leq \varepsilon_{force}. \quad (C.13)$$

where the second-to-last inequality is by Cauchy–Schwarz inequality and the last inequality is by the definition of the spectral norm.

Putting the results together, we get:

$$\left| \tau_n \langle 0| W^\dagger U_{force} W |0\rangle - \frac{\partial E_{el}}{\partial x_{n,j}} \right|$$

$$\leq 2\tau_n \|\,|\phi_0\rangle - |\tilde{\psi}_0\rangle\| + \varepsilon_{\text{force}}$$
$$\leq 2\tau_n \sqrt{2\varepsilon_{\text{prep}}} + \varepsilon_{\text{force}}\,. \qquad (C.14)$$

Set $\varepsilon_{\text{force}} = \frac{\varepsilon_{D_E}}{2}$ and $2\tau_n\sqrt{2\varepsilon_{\text{prep}}} = \frac{\varepsilon_{D_E}}{2}$. Replacing the bounds of $\varepsilon_{\text{prep}}$ of that obtained in Proposition B.7, where we see that

$$\varepsilon_{\text{el}} \in \tilde{\mathcal{O}}\left(\frac{\delta^2\gamma^2\varepsilon_{\text{prep}}^2}{\lambda}\right) \subseteq \tilde{\mathcal{O}}\left(\frac{\delta^2\gamma^2\varepsilon_{D_E}^4}{\lambda\tau_n^4}\right)\,. \qquad (C.15)$$

For consistency across different n, we set ε_{el} to a tighter bound. Let $\tau = \frac{\tau_n}{Z_n}$. We then let

$$\varepsilon_{\text{prep}} \in \mathcal{O}\left(\frac{\varepsilon_{D_E}^2}{\tau^2 Z_{\max}^2}\right) \qquad (C.16)$$

$$\varepsilon_{\text{el}} \in \tilde{\mathcal{O}}\left(\frac{\delta^2\gamma^2\varepsilon_{\text{prep}}^2}{\lambda}\right) \subseteq \tilde{\mathcal{O}}\left(\frac{\delta^2\gamma^2\varepsilon_{D_E}^4}{\lambda\tau^4 Z_{\max}^4}\right)\,. \qquad (C.17)$$

which still satisfies the former bound. The runtime is largely dominated by ground state preparation plus one query to the force operator. Replacing $\varepsilon_{\text{prep}}$ by its bounds of ε_{D_E}, we note that we require $\mathcal{O}(\frac{\lambda}{\delta\gamma}\log\frac{\tau Z_{\max}}{\delta\varepsilon_{D_E}})$ queries to U_H. The number of Toffoli gates needed can be found by Corollary B.7.1 to be

$$\mathcal{O}\left(\frac{\lambda}{\delta\gamma}\log\frac{\tau Z_{\max}}{\delta\varepsilon_{D_E}}\left(\max(a_{\text{el}}, a_{\text{force}}) + \log\left(\frac{\lambda\tau Z_{\max}}{\delta\gamma\varepsilon_{D_E}}\right)\right)\right)\,. \qquad (C.18)$$

$\square$

2. Implementation of the Liouvillian

Now that we have the implementation of $D_{n,j}^{\text{el}}$ at hand, we can implement the electronic Liouvillian, which we state again as follows:

$$L_{\text{el}}^{(\text{NVT})} := i\sum_{n=1}^{N}\sum_{j=1}^{3} D_{n,j}^{\text{el}} \otimes D_{p'_{n,j}}\,, \qquad (C.19)$$

where the central finite difference operator reads,

$$D_{p'_{n,j}} = \frac{1}{h_{p'}}\sum_{k=-d_{p'}}^{d_{p'}} c_{d_{p'},k}|\bar{p}'_{n,j} - k\rangle\langle\bar{p}'_{n,j}|\,. \qquad (C.20)$$

From the results of Equations (A79) to (A82) in Ref. [11], we know that a $(\alpha_{D_{p'}}, a_{D_{p'}}, \varepsilon_{D_{p'}})$-block-encoding of $D_{p'_{n,j}}$, where

$$\alpha_{D_{p'}} \in \mathcal{O}\left(\frac{\log d_{p'}}{h_{p'}}\right)\,, \qquad (C.21)$$

$$a_{D_{p'}} \in \mathcal{O}(\log d_{p'})\,, \qquad (C.22)$$

can be prepared with $\mathcal{O}\left(d_{p'}\log\frac{g_{p'}\log d_{p'}}{h_{p'}\varepsilon_{D_{p'}}}\right)$ Toffoli gates using a weighted combination of unitary adders [100]

with the adjustments made in Lemma B.1. Then, to implement L_{el}, we simply take a weighted sum over $D_{n,j}^{\text{el}} \otimes D_{p'_{n,j}}$ via LCU, or Lemma B.2. We again implement this operator by PREP†–SEL–PREP formalism. Given that the SEL operators would have different weightings dependent on n, we separate n and j into two registers. Note that given for each $D_{n,j}^{\text{el}}$, the scaling factor τ_n differs by a factor of Z_n, we need to prepare the state

$$\sum_{n=1}^{N}\sqrt{\frac{Z_n}{\sum_{n'=1}^{N} Z_{n'}}}\,|n\rangle \otimes \frac{1}{\sqrt{3}}(|00\rangle + |01\rangle + |10\rangle) \qquad (C.23)$$

via the PREP$_Z$ operator. On the other hand, the SEL operator is formulated as follows:

$$\sum_{n=1}^{N}\sum_{j=1}^{3}|n\rangle\langle n| \otimes |j\rangle\langle j| \otimes D_{n,j}^{\text{el}} \otimes D_{p'_{n,j}}\,. \qquad (C.24)$$

Note that, since the $D_{n,j}^{\text{el}}$ operator utilizes ground state preparation on the electronic registers, we can leverage the ground state preparation algorithm across all LCU components by querying it twice, thereby reducing the runtime. Further, the $D_{p'}$ only needs to be queried once if we use an ancilla register that swaps in the momentum states and applies the operator in superposition. A complete circuit diagram can be shown in Figure 5. We formalize the results as follows:

Proposition C.3 (Block encoding of the electronic Liouvillian). *Suppose we have a nuclear-position-controlled electronic Hamiltonian that can be block-encoded in the first quantization scheme with scaling factor λ satisfying Assumption C.1 in regards to spectral gap γ, ground state upper bound μ. We are given an approximate ground state preparation oracle U_I that can prepare ground states up to an overlap of $1 - \delta$. Then there exists a $(\alpha_{L_{\text{el}}}, a_{L_{\text{el}}}, \varepsilon_{L_{\text{el}}})$-block-encoding of L_{el} where*

$$\alpha_{L_{\text{el}}} \in \mathcal{O}\left(\frac{N\tilde{N}Z_{\max}\log d_{p'}}{h_{\text{el}}^2 h_{p'}}\right)$$

$$a_{L_{\text{el}}} \in \mathcal{O}\left(\tilde{N}\log B + \log\frac{d_{p'}}{\delta\gamma} + \log\frac{\alpha_{L_{\text{el}}}}{\varepsilon_{L_{\text{el}}}}\right)\,.$$

This block encoding can be prepared with $\mathcal{O}(\frac{1}{\delta})$ queries to U_I, and

$$\tilde{\mathcal{O}}\left(\frac{\lambda}{\delta\gamma}\log\frac{\alpha_{L_{\text{el}}}}{\varepsilon_{L_{\text{el}}}}\left(N + \tilde{N}\log B + \log B\log\frac{Bg_x\alpha_{L_{\text{el}}}}{\varepsilon_{L_{\text{el}}}}\right)\right.$$

$$\left. + N\log g_{p'} + d_{p'}\log\frac{\alpha_{L_{\text{el}}}}{\varepsilon_{L_{\text{el}}}}\right)$$

Toffoli gates.

Proof. We first calculate the scaling factors for the block encoding. We are given a $(\tau_n, a_{D_E}, \varepsilon_{D_E})$-block-encoding of $D_{n,j}^{\text{el}}$ and a $(\alpha_{D_{p'}}, a_{D_{p'}}, \varepsilon_{D_{p'}})$-block-encoding of $D_{p'_{n,j}}$.
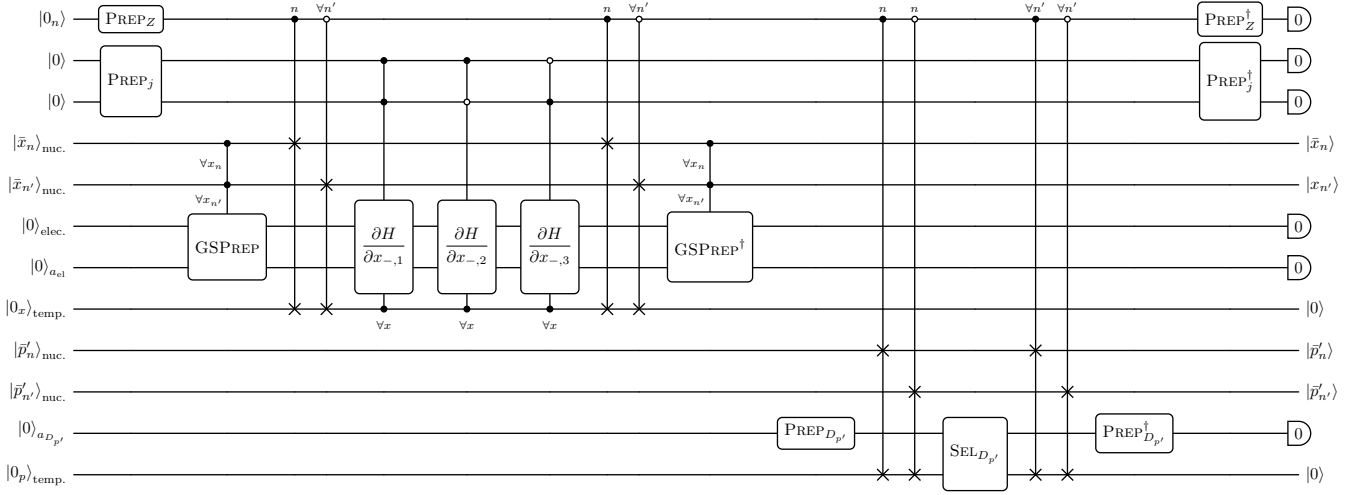
FIG. 5. *Block encoding implementation of the electronic Liouvillian.* We combine the implementation of $D_{n,j}^{\text{el}}$ from Figure 4 and $D_{p'}$ as shown in Ref. [11] for all n and j via LCU. Note that the nuclear-register-controlled ground state preparation operator is implemented as a single operator and need not be repeated for different sets of $\boldsymbol{x}$.

By Lemma B.3, we can construct a $(\tau_n \alpha_{D_{p'}}, a_{D_E} + a_{D_{p'}}, \tau_n \varepsilon_{D_{p'}} + \alpha_{D_{p'}} \varepsilon_{D_E})$-block-encoding of $D_{n,j}^{\text{el}} \otimes D_{p'_{n,j}}$.

Noting that by Lemma A.2, $\forall n, n', \frac{\tau_n}{Z_n} = \frac{\tau_{n'}}{Z_{n'}}$, we let $\tau = \frac{\tau_n}{Z_n}$. We can then notice that for each (n, j) pair, we have a

$$\left(\tau \alpha_{D_{p'}}, a_{D_E} + a_{D_{p'}}, \tau \varepsilon_{D_{p'}} + \frac{\alpha_{D_{p'}} \varepsilon_{D_E}}{Z_n} \right)\text{-block-encoding}$$

of $\frac{1}{Z_n} D_{n,j}^{\text{el}} \otimes D_{p'_{n,j}}$. Given that the minimum value of Z_n is 1, we can also set the error term to be $\tau \varepsilon_{D_{p'}} + \alpha_{D_{p'}} \varepsilon_{D_E}$ for consistency over all (n, j) pairs. We can then construct the weighted sum of

$$i \sum_{n=1}^{N} \sum_{j=1}^{3} Z_n \frac{D_{n,j}^{\text{el}} \otimes D_{p'_{n,j}}}{Z_n} = L_{\text{el}} \qquad (C.25)$$

to construct the electronic Liouvillian. The additional i factor can be added by $XSXS$ gates, or in the controlled version, $(X \otimes H)CX(X \otimes H)CX(\mathbb{I} \otimes S)CX(\mathbb{I} \otimes S)$ gates. Given a $(\alpha_Z, a_Z, \varepsilon_Z)$-state-preparation-pair that prepares the state $\sum_{n=1}^{N} \sqrt{\frac{Z_n}{\sum_{n'=1}^{N} Z_{n'}}} |n\rangle \otimes \frac{1}{\sqrt{3}}(|00\rangle + |01\rangle + |10\rangle)$, where $\alpha_Z = 3 \sum_{n=1}^{N} Z_n \in \mathcal{O}(NZ_{\max})$ and $a_Z \in \mathcal{O}(\log N)$, by Lemma B.2, we construct a $(\alpha_{L_{\text{el}}}, a_{L_{\text{el}}}, \varepsilon_{L_{\text{el}}})$-block-encoding of the electronic Liouvillian, where

$$\alpha_{L_{\text{el}}} \in \mathcal{O}(NZ_{\max}\tau \alpha_{D_{p'}}) \qquad (C.26)$$

$$a_{L_{\text{el}}} \in a_{D_E} + a_{D_{p'}} + \mathcal{O}(\log N) \qquad (C.27)$$

$$\varepsilon_{L_{\text{el}}} \in \mathcal{O}(NZ_{\max}\tau \varepsilon_{D_{p'}} + NZ_{\max} \alpha_{D_{p'}} \varepsilon_{D_E} + \tau \alpha_{D_{p'}} \varepsilon_Z). \qquad (C.28)$$

Plugging in the values of τ and $\alpha_{D_{p'}}$, we note that we

get for the scaling factor

$$\alpha_{L_{\text{el}}} \in \mathcal{O}\left(\frac{N\tilde{N}Z_{\max} \log d_{p'}}{h_{\text{el}}^2 h_{p'}} \right). \qquad (C.29)$$

Next, as the number of ancillae is dependent on error, we check the error terms first. Starting with the largest bound on $\varepsilon_{L_{\text{el}}}$ and working our way down, we obtain the following:

$$\varepsilon_{D_{p'}} \in \mathcal{O}\left(\frac{\varepsilon_{L_{\text{el}}}}{NZ_{\max}\tau} \right) \subseteq \mathcal{O}\left(\frac{h_{\text{el}}^2 \varepsilon_{L_{\text{el}}}}{N\tilde{N}Z_{\max}} \right) \qquad (C.30)$$

$$\varepsilon_{D_E} \in \mathcal{O}\left(\frac{\varepsilon_{L_{\text{el}}}}{NZ_{\max}\alpha_{D_{p'}}} \right) \subseteq \mathcal{O}\left(\frac{h_{p'} \varepsilon_{L_{\text{el}}}}{NZ_{\max} \log d_{p'}} \right)$$
$$\qquad (C.31)$$

$$\varepsilon_Z \in \mathcal{O}\left(\frac{\varepsilon_{L_{\text{el}}}}{\tau \alpha_{D_{p'}}} \right) \subseteq \mathcal{O}\left(\frac{h_{\text{el}}^2 h_{p'} \varepsilon_{L_{\text{el}}}}{\tilde{N} \log d_{p'}} \right) \qquad (C.32)$$

$$\varepsilon_{\text{el}} \in \tilde{\mathcal{O}}\left(\frac{\delta^2 \gamma^2 \varepsilon_{D_E}^4}{\lambda \tau^4 Z_{\max}^4} \right) \subseteq \tilde{\mathcal{O}}\left(\frac{\delta^2 \gamma^2 h_{p'}^4 h_{\text{el}}^8 \varepsilon_{L_{\text{el}}}^4}{\lambda N^4 \tilde{N}^4 Z_{\max}^4 \log^4 d_{p'}} \right)$$
$$\qquad (C.33)$$

$$\varepsilon_{\text{force}} \in \mathcal{O}(\varepsilon_{D_E}) \subseteq \mathcal{O}\left(\frac{h_{p'} \varepsilon_{L_{\text{el}}}}{NZ_{\max} \log d_{p'}} \right). \qquad (C.34)$$

Moving on to the ancillae, we note that

$$a_{L_{\text{el}}} \in a_{D_E} + a_{D_{p'}} + \mathcal{O}(\log N)$$
$$\subseteq \mathcal{O}\left(\tilde{N} \log B + \log \frac{N d_{p'}}{\varepsilon_{\text{el}}} \right)$$
$$\subseteq \mathcal{O}\left(\tilde{N} \log B + \log \frac{NZ_{\max} d_{p'}}{\delta \gamma h_{p'} h_{\text{el}} \varepsilon_{L_{\text{el}}}} \right). \qquad (C.35)$$

Note that we can rewrite this expression as

$$a_{L_{\text{el}}} \in \mathcal{O}\left(\tilde{N} \log B + \log \frac{d_{p'}}{\delta \gamma} + \log \frac{\alpha_{L_{\text{el}}}}{\varepsilon_{L_{\text{el}}}} \right). \qquad (C.36)$$

We also make use of $\mathcal{O}(N \log g_x)$ ancillae for swapping in nuclear positions and another $\mathcal{O}(\log g_{p'})$ pure ancillae for momentum positions.

Lastly, to calculate the number of Toffoli gates, we first go through the PREP operator. Per Lemma B.1, using the PREP operator to prepare a state preparation pair up to ε_Z, we note that this would cost $\mathcal{O}\left(N \log \frac{N Z_{\max}}{\varepsilon_Z}\right)$ Toffoli gates. Next, moving on to the SEL operator, we note that for all (n, j) pairs, we prepare the same ground state; we only need to query the ground state preparation algorithm twice – once for computing and once for uncomputing the ground state. Further, given that we control-SWAP the nuclear positions into an ancilla register and compute the force operator, we query the force operator $\frac{\partial H_{\mathrm{el}}}{x_{n,j}}$ thrice – once for each value of j. If we further decompose the force operator, we note that only the single qubit rotation controlled by $|\nu\rangle$ is different – the other operations may be shared. Nevertheless, the asymptotics are the same. Then, for the $D_{p'_{n,j}}$ operator, we again only need to query once, as we can also control-SWAP the momentum positions into an ancilla register. Thus, we obtain the total number of gates as follows:

$$
\begin{aligned}
\#_{\mathrm{Toffolis}} = &\; \mathcal{O}\left(\frac{\lambda}{\delta\gamma} \log \frac{\tau Z_{\max}}{\delta \varepsilon_{D_E}}\right) \\
&\times \mathcal{O}\left(N + \log B\left(\tilde{N} + \log \frac{B g_x}{\varepsilon_{\mathrm{el}}}\right)\right) \\
&+ \mathcal{O}\left(\frac{\lambda}{\delta\gamma} \log \frac{\tau Z_{\max}}{\delta \varepsilon_{D_E}}\left(a_{\mathrm{el}} + \log\left(\frac{\lambda \tau Z_{\max}}{\delta\gamma \varepsilon_{D_E}}\right)\right)\right) \\
&+ \mathcal{O}\left(N \log g_x + \log B\left(\tilde{N} + \log \frac{B g_x}{\varepsilon_{\mathrm{force}}}\right)\right) \\
&+ \mathcal{O}\left(N \log g_{p'} + d_{p'} \log \frac{g_{p'} \log d_{p'}}{h_{p'} \varepsilon_{D_{p'}}}\right) \\
&+ \mathcal{O}\left(N \log \frac{N Z_{\max}}{\varepsilon_Z}\right),
\end{aligned}
\tag{C.37}
$$

where the first two terms are from ground state preparation, the third term is from the force operator, the fourth term is from the momentum derivative operator, and the last term is from the state preparation pair. Cleaning up the result, we note that a total of

$$
\begin{aligned}
\tilde{\mathcal{O}}\Bigg(&\frac{\lambda}{\delta\gamma} \log \frac{\alpha_{L_{\mathrm{el}}}}{\varepsilon_{L_{\mathrm{el}}}}\left(N + \tilde{N} \log B + \log B \log \frac{B g_x \alpha_{L_{\mathrm{el}}}}{\varepsilon_{L_{\mathrm{el}}}}\right) \\
&+ N \log g_{p'} + d_{p'} \log \frac{g_{p'} \alpha_{L_{\mathrm{el}}}}{\varepsilon_{L_{\mathrm{el}}}}\Bigg)
\end{aligned}
\tag{C.38}
$$

Toffoli gates are used, where, further cleaning up constant values, we obtain

$$
\begin{aligned}
&\tilde{\mathcal{O}}\left(\frac{\tilde{N} N_{\mathrm{tot}}}{\delta\gamma} \log \frac{1}{\varepsilon_{L_{\mathrm{el}}}}\left(N_{\mathrm{tot}} + \log \frac{1}{\varepsilon_{L_{\mathrm{el}}}}\right)\right) \\
&\subseteq \tilde{\mathcal{O}}\left(\frac{\tilde{N} N_{\mathrm{tot}}^2}{\delta\gamma} \log^2 \frac{1}{\varepsilon_{L_{\mathrm{el}}}}\right).
\end{aligned}
\tag{C.39}
$$

$\square$

To block-encode the final Liouvillian, we require the block encoding of the classical Liouvillian under the NVT ensemble as shown in Ref. [11].

Lemma C.4 (Block encoding of the discretized classical NVT Liouvillian – Lemma 2, revised, [11]). *There exists a $(\alpha_{\mathrm{NVT}}, a_{\mathrm{NVT}}, \varepsilon_{\mathrm{NVT}})$-block-encoding of the discretized classical Liouvillian L_{NVT} with scaling factor*

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

and the number of ancilla qubits

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

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

$$
\tilde{\mathcal{O}}\left((N + d) \log \frac{g \alpha_{NVT}}{\varepsilon_{\mathrm{NVT}}} + \log^{\log 3} \frac{\alpha_{NVT}}{\varepsilon_{\mathrm{NVT}}}\right)
$$

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

Lastly, we arrive at the block encoding of the full Liouvillian. Note that in the practice of implementing the block encoding of the hardware, there is no need to separate the classical and electronic Liouvillians. One can simply LCU the operator $D_{n,j}^{\mathrm{el}}$ to the nuclear force operator V and then tensor-product $D_{p'}$. This implementation is shown in Figure 6. However, in the proofs, the choice to separate classical and electronic Liouvillian is simply due to the ease of theoretical proofs, without having to perform repetitive calculations regarding the operators in the classical Liouvillian. Nevertheless, we note that due to the scaling factor being the ℓ_1-norm of components of an LCU, the scaling factor of the Liouvillian should not be affected. The runtime and ancilla count *may* be affected slightly due to the changes in how error propagates, but as error has only a logarithmic dependency in both the runtime and ancilla count, the dependencies of the multiplicative factors acting on the error would be hidden within polylogarithmic factors not shown in the asymptotics. We now present the following theorem, where the implementation is based on separate classical and electronic Liouvillians.

Proposition C.5 (Block encoding of the NVT canonical Liouvillian). *Suppose we have a nuclear-position-controlled electronic Hamiltonian that can be block-encoded in the first quantization scheme with scaling factor λ satisfying Assumption C.1 in regards to spectral gap γ, ground state upper bound μ. We are given an approximate ground state preparation oracle U_I that can prepare*

FIG. 6. *Block encoding implementation of the NVT Liouvillian L.* As in Ref. [11], the Liouvillian is broken into separate operators that can be individually implemented by PREP–SEL–PREP† operators. PREP are denoted as P while SEL are denoted as S. These block-encoded operators include the discrete derivative operators for position U_{D_x}, momentum $U_{D_{p'}}$, and bath variables U_{D_s}, $U_{D_{p_s}}$, the momentum operator $U_{p'}$, the nuclear force operator U_V, the momentum operator for the heat bath U_{p_s}, the kinetic energy force operator of the heat bath U_s, and the potential energy force operator of the heat bath $U_{\frac{1}{s}}$. In addition to the nine operators previously block-encoded in Ref. [11], we provide the block encoding of $U_{D_{el}}$ as shown in Lemma C.2 and Figure 4, which corresponds to the electronic force operator. When applying the tensor product with $D_{p'}$, this forms the electronic Liouvillian shown in Proposition C.3 and Figure 5, which is highlighted in blue. In addition, we also require the use of P_m to encode the nuclear masses and P_Z to encode the nuclear charges to create the correct weighted sum of block encodings, as well as P_f and P_{fs} to combine the force terms correctly and P_{out} to combine the Liouvillian terms correctly. Note that for simplicity, the figure only displays the implementation used for a single $(n, \hat{j})$ pair. Implementations for all $(n, \hat{j})$ would make use of SWAP gates to reduce the number of queries to individual block encodings, as illustrated in Figure 5 for the electronic Liouvillian.

ground states up to an overlap of $1 - \delta$. Then there exists a $(\alpha_L, a_L, \varepsilon_L)$-block-encoding of the NVT canonical Liouvillian L, where

$$\alpha_L \in \mathcal{O}\Bigg(N\tilde{N}\frac{Z_{\max}}{h_{\mathrm{el}}^2}\frac{\log d_{p'}}{h_{p'}} + N^2 \frac{Z_{\max}^2 x_{\max}}{\Delta^3}\frac{\log d_{p'}}{h_{p'}}$$
$$+ N\frac{p'_{\max}}{m_{\min}s_{\min}^2}\frac{\log d_x}{h_x} + \frac{p_{s,\max}}{Q}\frac{\log d_s}{h_s}$$
$$+ \left(N\frac{p'^2_{\max}}{m_{\min}s_{\min}^3} + \frac{N_f k_B T}{s_{\min}} \right)\frac{\log d_{p_s}}{h_{p_s}} \Bigg)$$
$$a_L \in \mathcal{O}\left(\tilde{N}\log B + \log\frac{d}{\delta\gamma} + \log\frac{\alpha_L}{\varepsilon_L} \right)$$

and $d := \max(d_x, d_{p'}, d_s, d_{p_s})$, can be prepared with $\mathcal{O}(\frac{1}{\delta})$ queries to U_I, and

$$\tilde{\mathcal{O}}\left(\left(\frac{\lambda}{\delta\gamma}\left(N + \log B\left(\tilde{N} + \log\frac{Bg\alpha_L}{\varepsilon_L} \right) \right) + d \right)\log\frac{g\alpha_L}{\varepsilon_L} \right)$$

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

Proof. To take the combination of the electronic and classical Liouvillian, we first note that by Proposition C.3 and Lemma C.4, we have a $(1, a_{L_{\mathrm{el}}}, \frac{\varepsilon_{L_{\mathrm{el}}}}{\alpha_{L_{\mathrm{el}}}})$-block-encoding of $\frac{L_{\mathrm{el}}}{\alpha_{L_{\mathrm{el}}}}$ and a $(1, a_{\mathrm{NVT}}, \frac{\varepsilon_{\mathrm{NVT}}}{\alpha_{\mathrm{NVT}}})$-block-encoding of $\frac{L_{\mathrm{NVT}}}{\alpha_{\mathrm{NVT}}}$. Let $\alpha_L = \alpha_{L_{\mathrm{el}}} + \alpha_{\mathrm{NVT}}$. Then, given a state preparation unitary that prepares the state

$$\sqrt{\frac{\alpha_{L_{\mathrm{el}}}}{\alpha_{L_{\mathrm{el}}} + \alpha_{\mathrm{NVT}}}}\,|0\rangle + \sqrt{\frac{\alpha_{\mathrm{NVT}}}{\alpha_{L_{\mathrm{el}}} + \alpha_{\mathrm{NVT}}}}\,|1\rangle \quad (C.40)$$

that functions as a $(\alpha_L, 1, \varepsilon_{\mathrm{rot}})$-state-preparation-pair, we can construct via Lemma B.2 a $(\alpha_L, a_L, \varepsilon_L)$-block-encoding of the Liouvillian L such that

$$\alpha_L = \alpha_{L_{\mathrm{el}}} + \alpha_{\mathrm{NVT}}, \quad (C.41)$$
$$a_L = \max(a_{L_{\mathrm{el}}}, a_{\mathrm{NVT}}) + 1, \quad (C.42)$$
$$\varepsilon_L = \alpha_L \cdot \max\left(\frac{\varepsilon_{L_{\mathrm{el}}}}{\alpha_{L_{\mathrm{el}}}}, \frac{\varepsilon_{\mathrm{NVT}}}{\alpha_{\mathrm{NVT}}} \right) + \varepsilon_{\mathrm{rot}}. \quad (C.43)$$

Dealing with the errors, we set the following to satisfy the above bounds

$$\varepsilon_{\mathrm{rot}} \in \mathcal{O}(\varepsilon_L), \quad (C.44)$$
$$\varepsilon_{L_{\mathrm{el}}} \in \mathcal{O}\left(\frac{\alpha_{L_{\mathrm{el}}}\varepsilon_L}{\alpha_L} \right), \quad (C.45)$$
$$\varepsilon_{\mathrm{NVT}} \in \mathcal{O}\left(\frac{\alpha_{\mathrm{NVT}}\varepsilon_L}{\alpha_L} \right). \quad (C.46)$$

We can then use this to capture the bounds on the ancillae, where

$$a_L = \max(a_{L_{\mathrm{el}}}, a_{\mathrm{NVT}}) + 1$$
$$\in \mathcal{O}\Bigg(\max\bigg(\left(\tilde{N}\log B + \log\frac{d_{p'}}{\delta\gamma} + \log\frac{\alpha_{L_{\mathrm{el}}}}{\varepsilon_{L_{\mathrm{el}}}} \right),$$
$$\left(\log d + \log\frac{\alpha_{NVT}}{\varepsilon_{\mathrm{NVT}}} \right) \bigg) \Bigg)$$

$$\subseteq \mathcal{O}\left(\tilde{N}\log B + \log\frac{d\alpha_L}{\delta\gamma\varepsilon_L} \right). \quad (C.47)$$

Now, for the cost of implementation, we note that we simply have to add the implementation cost of both the classical and electronic Liouvillian, plus the cost for the single rotation gate (which is negligible), and adjust for errors, which we can find to be as follows:

$$\tilde{\mathcal{O}}\left(\left(\frac{\lambda}{\delta\gamma}\left(N + \log B\left(\tilde{N} + \log\frac{Bg\alpha_L}{\varepsilon_L} \right) \right) + d \right)\log\frac{g\alpha_L}{\varepsilon_L} \right). \quad (C.48)$$
$\square$

3. Implementing the Liouvillian simulation algorithm

The final step in the Liouvillian simulation algorithm involves Hamiltonian simulation of L. We refer the reader to Algorithm 1 for the full algorithmic description that includes the construction of the electronic Liouvillian in prior sections.

Theorem C.6 (Liouvillian simulation). *Suppose we have a nuclear-position-controlled electronic Hamiltonian that can be block-encoded in the first quantization scheme with scaling factor λ satisfying Assumption C.1 in regards to spectral gap γ, ground state upper bound μ. We are given an approximate ground state preparation oracle U_I that can prepare ground states up to an overlap of $1 - \delta$. Given an initial state $|\rho_0\rangle$ that encodes the initial discretized phase-space density and a discretized Liouvillian under the NVT canonical ensemble, there exists a quantum algorithm that outputs a quantum state that is ε-close in ℓ_2 distance to $|\rho_t\rangle = e^{-iLt}|\rho_0\rangle$ using*

$$\tilde{\mathcal{O}}\Bigg(\left(\alpha_L t + \log\frac{1}{\varepsilon} \right)\left(\frac{\lambda}{\delta\gamma}\log\frac{g}{\varepsilon} \right.$$
$$\left. \times\left(N + \tilde{N}\log B + \log B\log\frac{Bg}{\varepsilon} \right) + d\log\frac{g}{\varepsilon} \right) \Bigg)$$

Toffoli gates,

$$\tilde{\mathcal{O}}\left(\left(\alpha_L t + \log\frac{1}{\varepsilon} \right)\frac{1}{\delta} \right)$$

queries to U_I, and

$$\tilde{\mathcal{O}}\left(N\log g + \tilde{N}\log B + \log\frac{d}{\delta\gamma} + \log\frac{\alpha_L t}{\varepsilon} \right)$$

qubits, where α_L is as defined in Proposition C.5, $d := \max(d_x, d_{p'}, d_s, d_{p_s})$ and $g := \max(g_x, g_{p'}, g_s, g_{p_s})$.

Proof. Set $\varepsilon_L = \frac{\varepsilon}{2t}$, and apply the block encoding constructed in Proposition C.5 to Lemma B.5 and Corollary B.5.1. This provides the number of Toffoli gates directly. $\square$

Note that if we consider only the particle numbers, the spectral gap, approximation ground state preparation overlap, and error precision, the runtime can be written as

$$\tilde{\mathcal{O}}\left(\frac{\tilde{N}N_{\text{tot}}}{\delta\gamma}\log\frac{1}{\varepsilon}\left(N_{\text{tot}}+\log\frac{1}{\varepsilon}\right)\left(NN_{\text{tot}}t+\log\frac{1}{\varepsilon}\right)\right)$$

$$\subseteq\tilde{\mathcal{O}}\left(\frac{N\tilde{N}N_{\text{tot}}^3 t}{\delta\gamma}\log^3\frac{1}{\varepsilon}\right),\qquad\text{(C.49)}$$

as shown in the informal version of the theorem (Theorem 1) in the main text.

4. Block encoding results for the NVE microcanonical ensemble

Here we present the results for the NVE microcanonical ensemble.

Proposition C.7 (Block encoding of the microcanonical NVE Liouvillian). *Suppose we have a nuclear-position-controlled electronic Hamiltonian that can be block-encoded in the first quantization scheme with scaling factor λ satisfying Assumption C.1 in regards to spectral gap γ, ground state upper bound μ. We are given an approximate ground state preparation oracle U_I that can prepare ground states up to an overlap of $1-\delta$. Then there exists a $(\alpha_L, a_L, \varepsilon_L)$-block-encoding of the microcanonical NVE Liouvillian L where*

$$\alpha_L\in\mathcal{O}\left(N\tilde{N}\frac{Z_{\max}}{h_{\text{el}}^2}\frac{\log d_p}{h_p}+N^2\frac{Z_{\max}^2 x_{\max}}{\Delta^3}\frac{\log d_p}{h_p}\right.$$
$$\left.+N\frac{p_{\max}}{m_{\min}}\frac{\log d_x}{h_x}\right)$$
$$a_L\in\mathcal{O}\left(\tilde{N}\log B+\log\frac{d}{\delta\gamma}+\log\frac{\alpha_L}{\varepsilon_L}\right)$$

and $d:=\max(d_x, d_p)$, can be prepared with $\mathcal{O}(\frac{1}{\delta})$ queries to U_I, and

$$\tilde{\mathcal{O}}\left(\left(\frac{\lambda}{\delta\gamma}\left(N+\log B\left(\tilde{N}+\log\frac{Bg\alpha_L}{\varepsilon_L}\right)\right)+d\right)\log\frac{g\alpha_L}{\varepsilon_L}\right)$$

Toffoli gates, where $g:=\max(g_x, g_p)$.

Proof. Proof follows trivially by replacing the NVT Liouvillian with the NVE Liouvillian in Theorem C.6. $\square$

Appendix D: Construction and proof of the thermodynamic integration algorithm

In the thermodynamic integration algorithm, we integrate the nuclear Hamiltonian difference over an interpolated set of Liouvillian-equilibrated states, which can be achieved by a Hadamard test upon the difference of nuclear Hamiltonians. We first discuss the bounds on the discretization error from taking the Riemann sum as an estimation instead of integration. Next, we implement the interpolated Liouvillian evolution states, followed by the implementation of the nuclear Hamiltonian difference block encoding. Lastly, we calculate the cost of the full thermodynamic integration algorithm.

Although thermal averages of systems are not time-dependent, the algorithm that we implement produces the free energy estimation results in the computed value being dependent on the evolution time. To minimize the amount of dependencies and better error bounds from taking the Riemann sum across interpolated systems in the thermodynamic integration algorithm, we fix the equilibration time t_{eq} across different Λ-dependent Liouvillian simulations in our proof and construction of the algorithm. Given the potentially different scaling factors of the Liouvillian block encodings, we need a method to perform different Hamiltonian simulations in superposition.

However, we note that in practice, as long as the equilibration time is long enough and assuming that the evolved state settles into the thermal state such that the computed values do not fluctuate much, given the time-independence of thermal averages, the equilibration time t_{eq} need not be consistent across different Λ-s, and one can simply use the same Hamiltonian simulation circuit across all Λ-s, provided that all states are sufficiently equilibrated. Regardless, the techniques developed in this section are still valuable outside of the convenience of theoretical guarantees, as they can be slightly modified to allow for coherent superposed evolutions over different evolution times for better time-independent estimates of free energy, as mentioned in the main text.

1. Upper bounds on the discretization error

In this section, we take a deeper look at the discretization error from approximating the thermodynamic integration with a Riemann sum approximation.

Lemma D.1 (Discretization error of thermodynamic integration by left Riemann sum). *Given two systems A and B, and suppose for both systems, we are given Liouvillians L_A and L_B under the NVT canonical ensemble encoding the dynamics, as well as nuclear Hamiltonians H_A and H_B. Further, we are given an initial state $|\rho_0\rangle$ that encodes the initial discretized phase-space density as well as an equilibration time t_{eq} for Liouvillian simulation by Theorem C.6. Then the discretization error from approximating the thermodynamic integration algorithm by computing the Riemann sum of N_Λ discretized points with the left rectangle rule is upper bounded by*

$$\frac{\|L_B-L_A\|\cdot\|H_B-H_A\|t_{\text{eq}}}{N_\Lambda}.$$

Proof. We begin by observing that the thermodynamic integration algorithm obtains the free energy difference of two systems A and B via the following equation:

$$\Delta F_{A \to B} = \int_0^1 \langle H_B - H_A \rangle_\Lambda \, d\Lambda. \tag{D.1}$$

By the left rectangle rule, we can approximate the above integral as follows:

$$\Delta \tilde{F} = \frac{1}{N_\Lambda} \sum_{\bar{\Lambda}=0}^{N_\Lambda - 1} \langle H_B - H_A \rangle_\Lambda, \tag{D.2}$$

where $\bar{\Lambda} = N_\Lambda \cdot \Lambda$. The discretization error arising from this approximation can then be upper-bounded as follows:

$$\left| \Delta F_{A \to B} - \Delta \tilde{F} \right| \leq \left| \frac{\partial \langle H_B - H_A \rangle_\Lambda}{\partial \Lambda} \right| \frac{1}{2N_\Lambda}. \tag{D.3}$$

Evaluating the first fraction, we obtain by the triangle inequality,

$$
\left| \frac{\partial \langle H_B - H_A \rangle_\Lambda}{\partial \Lambda} \right|
$$
$$
= \left| \frac{\partial}{\partial \Lambda} \langle \rho_0 | e^{iL_\Lambda t_{\text{eq}}} (H_B - H_A) e^{-iL_\Lambda t_{\text{eq}}} | \rho_0 \rangle \right|
$$
$$
= \left| \langle \rho_0 | \left(\frac{\partial}{\partial \Lambda} e^{iL_\Lambda t_{\text{eq}}} \right) (H_B - H_A) e^{-iL_\Lambda t_{\text{eq}}} | \rho_0 \rangle \right.
$$
$$
\left. + \langle \rho_0 | e^{iL_\Lambda t_{\text{eq}}} (H_B - H_A) \left(\frac{\partial}{\partial \Lambda} e^{-iL_\Lambda t_{\text{eq}}} \right) | \rho_0 \rangle \right|
$$
$$
\leq \left| \langle \rho_0 | \left(\frac{\partial}{\partial \Lambda} e^{iL_\Lambda t_{\text{eq}}} \right) (H_B - H_A) e^{-iL_\Lambda t_{\text{eq}}} | \rho_0 \rangle \right|
$$
$$
+ \left| \langle \rho_0 | e^{iL_\Lambda t_{\text{eq}}} (H_B - H_A) \left(\frac{\partial}{\partial \Lambda} e^{-iL_\Lambda t_{\text{eq}}} \right) | \rho_0 \rangle \right|. \tag{D.4}
$$

We evaluate the first term such that by the tracial matrix Hölder's inequality [101–103] and unitarily invariant properties of spectral/Schatten norms [104], we can obtain

$$
\left| \langle \rho_0 | \left(\frac{\partial}{\partial \Lambda} e^{iL_\Lambda t_{\text{eq}}} \right) (H_B - H_A) e^{-iL_\Lambda t_{\text{eq}}} | \rho_0 \rangle \right|
$$
$$
= \left| \text{Tr} \left(\left(\frac{\partial}{\partial \Lambda} e^{iL_\Lambda t_{\text{eq}}} \right) (H_B - H_A) e^{-iL_\Lambda t_{\text{eq}}} | \rho_0 \rangle \langle \rho_0 | \right) \right|
$$
$$
\leq \left\| \left(\frac{\partial}{\partial \Lambda} e^{iL_\Lambda t_{\text{eq}}} \right) (H_B - H_A) e^{-iL_\Lambda t_{\text{eq}}} \right\| \text{Tr} \left(\| |\rho_0 \rangle \langle \rho_0 | \| \right)
$$
$$
\leq \left\| \frac{\partial}{\partial \Lambda} e^{iL_\Lambda t_{\text{eq}}} \right\| \| H_B - H_A \| \left\| e^{-iL_\Lambda t_{\text{eq}}} \right\| \text{Tr} \left(\| |\rho_0 \rangle \langle \rho_0 | \| \right)
$$
$$
= \left\| \frac{\partial}{\partial \Lambda} e^{iL_\Lambda t_{\text{eq}}} \right\| \| H_B - H_A \|. \tag{D.5}
$$

Then by Lemma 50 of Ref. [44], which states

$$\left\| e^{itH} - e^{itH'} \right\| \leq \| H - H' \| t. \tag{D.6}$$

We find that

$$
\left\| \frac{\partial}{\partial \Lambda} e^{iL_\Lambda t_{\text{eq}}} \right\|
$$
$$
= \lim_{\Delta\Lambda \to 0} \frac{\left\| e^{i(L_A + (\Lambda + \Delta\Lambda)(L_B - L_A)t_{\text{eq}}} - e^{i(L_A + \Lambda(L_B - L_A)t_{\text{eq}}} \right\|}{\Delta\Lambda}
$$
$$
\leq \lim_{\Delta\Lambda \to 0} \frac{(\Lambda + \Delta\Lambda)\| L_B - L_A \| t_{\text{eq}} - \Lambda \| L_B - L_A \| t_{\text{eq}}}{\Delta\Lambda}
$$
$$
= \| L_B - L_A \| t_{\text{eq}}. \tag{D.7}
$$

We can then see that

$$
\left| \langle \rho_0 | \left(\frac{\partial}{\partial \Lambda} e^{iL_\Lambda t_{\text{eq}}} \right) (H_B - H_A) e^{-iL_\Lambda t_{\text{eq}}} | \rho_0 \rangle \right|
$$
$$
\leq \| L_B - L_A \| \cdot \| H_B - H_A \| t_{\text{eq}}. \tag{D.8}
$$

Likewise, we can show that

$$
\left| \langle \rho_0 | e^{iL_\Lambda t_{\text{eq}}} (H_B - H_A) \left(\frac{\partial}{\partial \Lambda} e^{-iL_\Lambda t_{\text{eq}}} \right) | \rho_0 \rangle \right|
$$
$$
\leq \| L_B - L_A \| \cdot \| H_B - H_A \| t_{\text{eq}}. \tag{D.9}
$$

Thus, one can show that the discretization error is upper-bounded as shown

$$
\left| \Delta F_{A \to B} - \Delta \tilde{F} \right| \leq \left| \frac{\partial \langle H_B - H_A \rangle_\Lambda}{\partial \Lambda} \right| \frac{1}{2N_\Lambda}
$$
$$
\leq \frac{\| L_B - L_A \| \cdot \| H_B - H_A \| t_{\text{eq}}}{N_\Lambda}. \tag{D.10}
$$

$\square$

While one can obtain tighter bounds by obtaining high-order numerical integration approximation methods and showing the proof via iteratively applying the Lie-Trotter formula [41], this is not required unless QSVT phase angles are loaded into the quantum circuit via lookup tables, which would incur an additional $\mathcal{O}(N_\Lambda)$ cost per query to the Liouvillian. If the angles are computed on separate quantum registers, or avoided via QSP without angle finding, then the dependency of the runtime would be at most $\mathcal{O}(\text{poly} \log N_\Lambda)$ per query to the Liouvillian, to which higher-order approximations would not help much as the dependency is already logarithmic.

2. Derivations of Λ-dependent Liouvillian simulation

In this section, we further describe methods for Λ-dependent Liouvillian simulation discussed in Section IV. First, as mentioned in the main text, we can implement a Λ-dependent Liouvillian simulation by implementing the entire QSVT algorithm in superposition, controlling both the Liouvillian and the phase angles by Λ. Phase angles are computed in a separate register that implements phase-finding algorithms, the details of which are outside the scope of this paper.
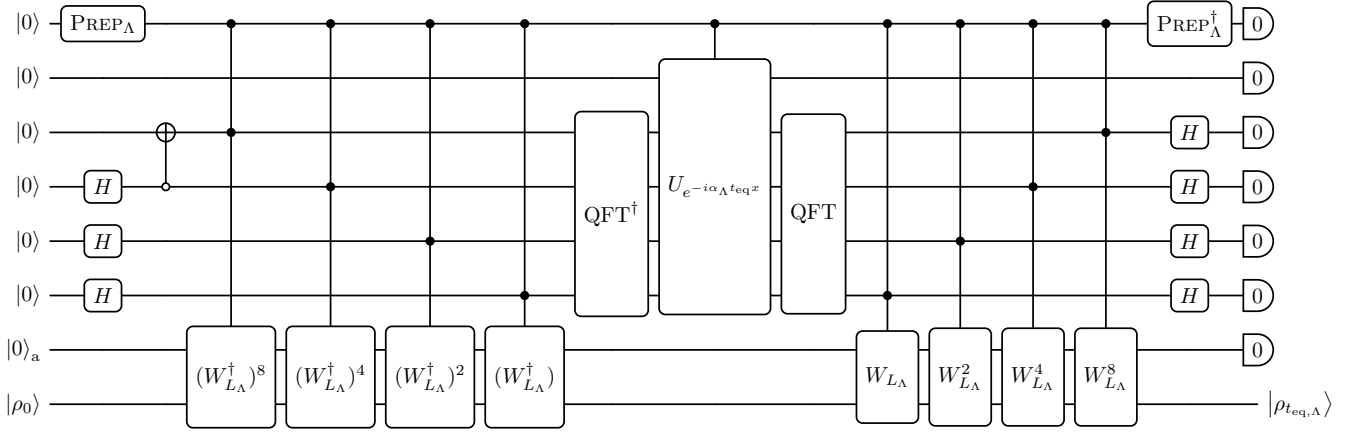
FIG. 7. Λ-*dependent Hamiltonian simulation of L_Λ using QSP without angle finding* [61]. The central Λ-controlled unitary is diagonal with the entries $\exp(-i\alpha_\Lambda t \cos\frac{2\pi k}{4\mathcal{D}})$ and can be performed using $\log^2(\frac{N_\Lambda}{\varepsilon})$ gates. In this example figure, we assume polynomial degree $\mathcal{D} = 4$. The gate W_{L_Λ} is the qubitization of the block-encoded Liouvillian U_{L_Λ} such that $W_{L_\Lambda} = (2\Pi_a - I_a)U_{L_\Lambda}$.

Alternatively, one can apply the robust Hamiltonian simulation in superposition by utilizing quantum signal processing without angle finding as shown in Propositions B.8 and B.9. Corresponding to the methodology of constructing Hamiltonian simulation in superposition for standard QSVT, which is to control the phase angles, in angleless QSP, to construct Hamiltonian simulation in superposition, we produce the functional diagonal block encoding $U_{f,4\mathcal{D}}$ in superposition, such that we have

$$\sum_\Lambda |\Lambda\rangle\langle\Lambda| \otimes U_{f_\Lambda,4\mathcal{D}}, \qquad (D.11)$$

where $\mathcal{D}$ is chosen to be the maximum degree for polynomial expansions for all Λ, or

$$\mathcal{D} \in \mathcal{O}\left(\max(\alpha_{L_A}, \alpha_{L_B})t_{eq} + \log\frac{1}{\varepsilon}\right), \qquad (D.12)$$

and

$$f_\Lambda(x) = \exp(i\alpha_\Lambda t_{eq} \cos x). \qquad (D.13)$$

For simplicity, in the angleless QSP setting, we approximate the thermodynamic integration algorithm via the left Riemann sum per Lemma D.1 such that

$$N_\Lambda = \frac{\|L_B - L_A\|\|H_B - H_A\|t_{eq}}{\varepsilon} \in \mathcal{O}\left(\frac{\alpha_L\alpha_H t_{eq}}{\varepsilon}\right). \qquad (D.14)$$

To detail the implementation of Equation (D.11), we first prepare the state

$$\sum_{\bar{\Lambda}=0}^{N_\Lambda-1}\sum_{k=0}^{4\mathcal{D}-1} |\Lambda\rangle|k\rangle. \qquad (D.15)$$

Then by implementing the CORDIC algorithm [105] via quantum arithmetic circuits [106] and pre-stored classical

values on lookup tables, whose values can just be directly provided on ancilla registers, or via a modification of the quantum CORDIC algorithm [107], one can implement the cos function up to ε additive precision with $\mathcal{O}(\log^2\frac{\mathcal{D}}{\varepsilon})$ Toffoli gates. We then produce the state

$$\sum_{\bar{\Lambda}=0}^{N_\Lambda-1}\sum_{k=0}^{4\mathcal{D}-1} |\Lambda\rangle|k\rangle\left|\cos\frac{2\pi k}{4\mathcal{D}}\right\rangle. \qquad (D.16)$$

By further use of quantum arithmetic circuits, we can also obtain

$$\sum_{\bar{\Lambda}=0}^{N_\Lambda-1}\sum_{k=0}^{4\mathcal{D}-1} |\Lambda\rangle|k\rangle|\alpha_\Lambda\rangle\left|\cos\frac{2\pi k}{4\mathcal{D}}\right\rangle\left|\alpha_\Lambda t_{eq}\cos\frac{2\pi k}{4\mathcal{D}}\right\rangle \qquad (D.17)$$

using an additional $\mathcal{O}(\log\frac{\mathcal{D}N_\Lambda}{\varepsilon})$ Toffoli gates. Then controlled on the final fixed-point register of $\alpha_\Lambda t_{eq}\cos\frac{2\pi k}{4\mathcal{D}}$, we can perform a rotational Z transform to an ancilla qubit or via the alternating sign trick [59] such that we have (ignoring intermediate states)

$$\sum_{\bar{\Lambda}=0}^{N_\Lambda-1}\sum_{k=0}^{4\mathcal{D}-1} |\Lambda\rangle|k\rangle\left|\alpha_\Lambda t_{eq}\cos\frac{2\pi k}{4\mathcal{D}}\right\rangle e^{i\alpha_\Lambda t_{eq}\cos\frac{2\pi k}{4\mathcal{D}}}|0\rangle \qquad (D.18)$$

with $\mathcal{O}(\log^2\frac{\mathcal{D}N_\Lambda}{\varepsilon})$ Toffoli gates. Then, uncomputing the arithmetic circuits, we obtain

$$\sum_{\bar{\Lambda}=0}^{N_\Lambda-1}\sum_{k=0}^{4\mathcal{D}-1} |\Lambda\rangle|k\rangle e^{i\alpha_\Lambda t_{eq}\cos\frac{2\pi k}{4\mathcal{D}}}|0\rangle \quad,. \qquad (D.19)$$

Notice that this can then be used to provide a $(1, 1, \frac{\varepsilon}{24\sqrt{2}})$-block-encoding of $D_{f_\Lambda,4\mathcal{D}}$ in superposition as shown in Equation (D.11) for which the implementation costs

$$\tilde{\mathcal{O}}\left(\log^2\frac{\alpha_L\alpha_H t_{eq}}{\varepsilon}\right) \qquad (D.20)$$

Toffoli gates.

Further accounting for the gates to prepare the weighted sum between L_A and L_B, we note that an additional $\mathcal{O}(\log^2 \frac{N_\Lambda}{\varepsilon})$ gates are needed for state preparation per access to L_Λ. Hence, the runtime for preparing the L_Λ simulation in superposition of Λ in total would still be asymptotically the same as preparing a single Liouvillian simulation when polylogarithmic factors are disregarded. We provide a circuit for this approach in Figure 7 (based on a circuit from Ref. [61]). Lastly, we note that although controlled versions of the qubitized Liouvillian are needed, as the ground state oracle is only used for state preparation that is then uncomputed, controlled versions of the approximate ground state oracle are not required.

Lastly, we note that it is also possible to rescale either L_A or L_B to have matching scaling factors $\alpha_{\max} = \max(\alpha_{L_A}, \alpha_{L_B})$ by tensor-producting a rotational Y gate on an ancilla qubit where $\theta = \arccos(\frac{\min(\alpha_{L_A}, \alpha_{L_B})}{\max(\alpha_{L_A}, \alpha_{L_B})})$ such that the block encoding is multiplied by $\cos(\theta)$, followed by the same polynomial expansion for $e^{-i\alpha_{\max}xt}$. However, there is no asymptotic advantage of doing so, and intentionally scaling down a block encoding would lead to higher errors. Hence, we choose not to adopt the block-encoding descaling method for implementation.

3. Implementation of nuclear Hamiltonian

The nuclear Hamiltonian H_{nuc} can be decomposed into H_{kin}, H_{pot}, H_{gse}, which correspond to the kinetic energy, potential energy, and ground state energy values at relevant points in phase-space:

$$H_{\mathrm{kin}} = \sum_{n,j} \sum_{\bar{p}'_{n,j}} \sum_{\bar{s}} \frac{p'^2_{n,j}}{m_n(s+s_{\min})^2} |\bar{p}'_{n,j}\rangle\langle\bar{p}'_{n,j}| \otimes |\bar{s}\rangle\langle\bar{s}|, \tag{D.21}$$

$$H_{\mathrm{pot}} = \sum_{n'\neq n} \sum_{\bar{x}_n} \sum_{\bar{x}_{n'}} \frac{Z_n Z_{n'}}{(\|x_n - x_{n'}\|^2 + \Delta^2)^{1/2}} |\bar{x}_n\rangle\langle\bar{x}_n|$$
$$\otimes |\bar{x}_{n'}\rangle\langle\bar{x}_{n'}|, \tag{D.22}$$

$$H_{\mathrm{gse}} = \sum_{\bar{\boldsymbol{x}}} E_{\mathrm{el}}(\boldsymbol{x}) |\bar{\boldsymbol{x}}\rangle\langle\bar{\boldsymbol{x}}|. \tag{D.23}$$

In this section, we review results on block-encoding the kinetic and potential energy operators from Ref. [11], and we present an improved implementation of the ground state energy operator H_{gse}.

Lemma D.2 (Block encoding of H_{kin} – Lemma 22, [11]). *There exists a* $(\alpha_{\mathrm{kin}}, a_{\mathrm{kin}}, \varepsilon_{\mathrm{kin}})$-*block-encoding of* H_{kin} *with normalization constant*

$$\alpha_{\mathrm{kin}} \in \mathcal{O}\left(N \frac{p'^2_{\max}}{m_{\min} s^2_{\min}}\right) \tag{D.24}$$

and a number of ancilla qubits

$$a_{\mathrm{kin}} \in \mathcal{O}\left(\log \frac{\alpha_{\mathrm{kin}}}{\varepsilon_{\mathrm{kin}}}\right) \tag{D.25}$$

that can be implemented using

$$\mathcal{O}\left(N \log \frac{g_{p'}\alpha_{\mathrm{kin}}}{\varepsilon_{\mathrm{kin}}} + \log^{\log 3} \frac{\alpha_{\mathrm{kin}}}{\varepsilon_{\mathrm{kin}}}\right) \tag{D.26}$$

Toffoli gates, where $g_{p'}$ *is the number of discrete momenta considered in the classical part of the Liouvillian.*

Lemma D.3 (Block encoding of H_{pot} – Lemma 23, [11]). *There exists a* $(\alpha_{\mathrm{pot}}, a_{\mathrm{pot}}, \varepsilon_{\mathrm{pot}})$-*block-encoding of* H_{pot} *with normalization constant*

$$\alpha_{\mathrm{pot}} \in \mathcal{O}\left(N^2 \frac{Z^2_{\max}}{\Delta}\right) \tag{D.27}$$

and the number of ancilla qubits

$$a_{\mathrm{pot}} \in \mathcal{O}\left(\log \frac{\alpha_{\mathrm{pot}}}{\varepsilon_{\mathrm{pot}}}\right). \tag{D.28}$$

This block encoding can be implemented using

$$\mathcal{O}\left(N \log \frac{g_x \alpha_{\mathrm{pot}}}{\varepsilon_{\mathrm{pot}}} + \log^{\log 3} \frac{\alpha_{\mathrm{pot}}}{\varepsilon_{\mathrm{pot}}}\right) \tag{D.29}$$

Toffoli gates, where g_x *is the number of discrete positions considered in the classical part of the Liouvillian.*

Given that the electronic Hamiltonian can be accessed in superposition, to construct H_{gse}, one can obtain the ground state energy of the electronic Hamiltonian $H_{\mathrm{el}}(\boldsymbol{x})$ for different values of $\bar{\boldsymbol{x}}$ in superposition. Ref. [11] achieves this by preparing a separate set of electronic registers, and preparing the ground states of the electronic Hamiltonian via a ground state preparation algorithm by Lin and Tong [10] for different nuclear positions in superposition. Then, using phase estimation, fixed point representations of the ground state energies are extracted, after which one can then approximately encode the fixed point representations into amplitudes of the quantum state via the alternating sign trick [59]. However, this construction is largely bottlenecked by phase estimation, which requires $\mathcal{O}(\frac{1}{\varepsilon})$ applications of the block encoding of the electronic Hamiltonian $U_{H_{\mathrm{el}}}$. In contrast, our implementation requires only $\mathcal{O}(\mathrm{poly} \log \frac{1}{\varepsilon})$ applications of $U_{H_{\mathrm{el}}}$.

In our work, similar to Ref. [11], we implement H_{gse} by first preparing the ground state with the preparation algorithm by Lin and Tong [10], but instead of using phase estimation to extract the eigenvalues, we then directly apply the block encoding of the Hamiltonian such that the scaled eigenvalues are directly encoded in the amplitude of the ground states in superposition, followed lastly by uncomputing the ground state, similar to our strategy of implementing the force operator in Lemma A.2.

This produces a block encoding implementation of H_{gse} that is only dependent on the precision logarithmically in terms of runtime. The following is therefore an improved version of Lemma 24 of Ref. [11].

Lemma D.4 (Block encoding of H_{gse}). *Suppose we have a $(\lambda, a_{\text{el}}, \varepsilon_{\text{el}})$-block-encoding U_H of the nuclear-position-controlled electronic Hamiltonian satisfying As-sumption C.1 in regards to spectral gap γ, ground state upper bound μ, and we are given an approximate ground state preparation oracle U_I that can prepare ground states up to an overlap of $1 - \delta$. If*

$$\varepsilon_{\text{el}} \in \tilde{\mathcal{O}}\left(\frac{\delta^2 \gamma^2 \varepsilon_{\text{gse}}^4}{\lambda^5} \right),$$

then there exists a $(\lambda, a_{\text{gse}}, \varepsilon_{\text{gse}})$-block-encoding of H_{gse} where

$$a_{\text{gse}} \in a_{\text{el}} + \mathcal{O}(\tilde{N} \log B).$$

This block encoding can be prepared with $\mathcal{O}(\frac{\lambda}{\delta\gamma} \log \frac{\lambda}{\delta\varepsilon_{\text{gse}}})$ queries to U_H, $\mathcal{O}(\frac{1}{\delta})$ queries to U_I and an additional $\mathcal{O}\left(\frac{\lambda}{\delta\gamma} \log \frac{\lambda}{\delta\varepsilon_{\text{gse}}} \left(a_{\text{el}} + \log \frac{\lambda}{\delta\gamma\varepsilon_{\text{gse}}} \right) \right)$ Toffoli gates.

Proof. First, we demonstrate the correctness of the block encoding construction. Assuming perfect block encoding and ground state preparation, we note that preparing the ground state in superposition gives us

$$\sum_{\bar{\boldsymbol{x}}} |\bar{\boldsymbol{x}}\rangle |0\rangle_{\text{anc}} \to \sum_{\bar{\boldsymbol{x}}} |\bar{\boldsymbol{x}}\rangle |\psi_0(\boldsymbol{x})\rangle |0\rangle_{\text{anc}} . \quad \text{(D.30)}$$

Then, applying the controlled Hamiltonian, we obtain

$$\sum_{\bar{\boldsymbol{x}}} |\bar{\boldsymbol{x}}\rangle \left(\frac{1}{\lambda} H_{\text{el}}(\boldsymbol{x}) |\psi_0(\boldsymbol{x})\rangle |0\rangle_{\text{anc}} + \sum_j \lambda'_j |\text{bad}_j\rangle |j\rangle_{\text{anc}} \right) . \quad \text{(D.31)}$$

where λ'_j is a normalization factor for some bad state $|\text{bad}_j\rangle$. Noting that $H_{\text{el}}(\boldsymbol{x}) |\psi_0(\boldsymbol{x})\rangle = E_0(\boldsymbol{x}) |\psi_0(\boldsymbol{x})\rangle$, if we then uncompute the ground state, we obtain

$$\sum_{\bar{\boldsymbol{x}}} |\bar{\boldsymbol{x}}\rangle \left(\frac{1}{\lambda} E_0(\boldsymbol{x}) |0\rangle_{\text{anc}} + \sum_j \lambda'_j |\text{bad}'_j\rangle |j\rangle_{\text{anc}} \right) . \quad \text{(D.32)}$$

Then, post-selecting on $|0\rangle$ for the ancilla registers and the electronic register, we see that we would obtain

$$\sum_{\bar{\boldsymbol{x}}} \frac{1}{\lambda} E_0(\boldsymbol{x}) |\bar{\boldsymbol{x}}\rangle . \quad \text{(D.33)}$$

Now, to the factors of the block encoding, the scaling factor λ of the block encoding follows from the scaling factor H_{el}. Given that the electronic register is computed and then uncomputed, in regards to encoding the ground state energy onto the nuclear register, the ancilla qubits are the sum of the ancilla qubits of the H_{el}, or a_{el}, and the size of the electronic register, or $\mathcal{O}(\tilde{N} \log B)$.

Regarding the error, we first prepare the ground state of $H_{\text{el}}^{\text{ctrl}}$ up to fidelity $1 - \varepsilon_{\text{prep}}$ with Proposition B.7. This produces a state $|\phi_0\rangle$ such that $|(\langle\psi_0| \otimes \langle 0|) |\phi_0\rangle| \geq 1 - \varepsilon_{\text{prep}}$. Similar to Lemma A.2, for cleanliness, we denote $|\tilde{\psi}_0\rangle = |\psi_0\rangle \otimes |0\rangle$. Then the ℓ_2 distance between the two states can be found to be

$$\left\| |\phi_0\rangle - |\tilde{\psi}_0\rangle \right\| \leq \sqrt{2\varepsilon_{\text{prep}}} . \quad \text{(D.34)}$$

Let W be the unitary operator that prepares the approximate ground state $|\phi_0\rangle$ from $|0\rangle$, composed of U_I and the ground state preparation procedure in Proposition B.7. Given that the final block encoding is that of a diagonal matrix on different nuclear positions, the error term of the block encoding (spectral norm difference) is less than that of the maximum difference on a single entry. We derive the difference as follows via a series of triangular inequalities and drop the notation on nuclear positions, as the derivation is for a single position:

$$\left| \lambda \langle 0| W^\dagger U_{H_{\text{el}}} W |0\rangle - E_0 \right|$$
$$\leq \lambda \left| \langle 0| W^\dagger U_{H_{\text{el}}} W |0\rangle - \langle 0| W^\dagger U_{H_{\text{el}}} |\tilde{\psi}_0\rangle \right|$$
$$+ \lambda \left| \langle 0| W^\dagger U_{H_{\text{el}}} |\tilde{\psi}_0\rangle - \langle\tilde{\psi}_0| U_{H_{\text{el}}} |\tilde{\psi}_0\rangle \right|$$
$$+ \left| \langle\tilde{\psi}_0| U_{H_{\text{el}}} |\tilde{\psi}_0\rangle - \langle 0| E_0 |0\rangle \right| . \quad \text{(D.35)}$$

We now derive the upper bounds of the four absolute differences. The first term can be bounded by the Cauchy–Schwarz inequality such that

$$\lambda \left| \langle 0| W^\dagger U_{H_{\text{el}}} W |0\rangle - \langle 0| W^\dagger U_{H_{\text{el}}} |\tilde{\psi}_0\rangle \right|$$
$$= \lambda \left| \langle 0| W^\dagger U_{H_{\text{el}}} |\phi_0\rangle - \langle 0| W^\dagger U_{H_{\text{el}}} |\tilde{\psi}_0\rangle \right|$$
$$= \lambda \left\| |\phi_0\rangle - |\psi_0\rangle \right\| . \quad \text{(D.36)}$$

The second term can be similarly bounded as follows:

$$\lambda \left| \langle 0| W^\dagger U_{H_{\text{el}}} |\tilde{\psi}_0\rangle - \langle\tilde{\psi}_0| U_{H_{\text{el}}} |\tilde{\psi}_0\rangle \right| \leq \lambda \left\| |\phi_0\rangle - |\psi_0\rangle \right\| . \quad \text{(D.37)}$$

The third term can be upper-bounded as follows:

$$\left| \langle\tilde{\psi}_0| U_{H_{\text{el}}} |\tilde{\psi}_0\rangle - \langle\tilde{\psi}_0| E_0 |\tilde{\psi}_0\rangle \right|$$
$$= |\langle\psi_0| (\lambda(\mathbb{I} \otimes \langle 0|) U_{H_{\text{el}}} (\mathbb{I} \otimes |0\rangle)) - H) |\psi_0\rangle|$$
$$\leq \|\lambda(\mathbb{I} \otimes \langle 0|) U_{H_{\text{el}}} (\mathbb{I} \otimes |0\rangle)) - H\| \leq \varepsilon_{\text{el}} , \quad \text{(D.38)}$$

When we combine the results, we get:

$$\left| \lambda \langle 0| W^\dagger U_{H_{\text{el}}} W |0\rangle - E_0 \right| \leq 2\lambda \left\| |\phi_0\rangle - |\psi_0\rangle \right\| + \varepsilon_{\text{el}}$$
$$\leq 2\lambda \sqrt{2\varepsilon_{\text{prep}}} + \varepsilon_{\text{el}} . \quad \text{(D.39)}$$

From Ref. [9], we know that λ is given from the ℓ_1 norm of the coefficients α_ℓ from the LCU decomposition of the electronic Hamiltonian $H_{\text{el}} = \sum_{\ell=1} \alpha_\ell H_\ell$, where H_ℓ are unitary. Thus, $\|H_{\text{el}}\|$ can be upper bounded by λ via the

triangular inequality. The error of the block encoding can then be further upper-bounded by

$$2\lambda\sqrt{2\varepsilon_{\text{prep}}} + \varepsilon_{\text{el}}. \qquad (D.40)$$

Set $\varepsilon_{\text{el}} \leq \frac{\varepsilon_{\text{gse}}}{2}$ and $2\lambda\sqrt{2\varepsilon_{\text{prep}}} = \frac{\varepsilon_{\text{gse}}}{2}$. Replacing the bounds of $\varepsilon_{\text{prep}}$ of that obtained in Proposition B.7, where we see that

$$\varepsilon_{\text{el}} \in \tilde{\mathcal{O}}\left(\frac{\delta^2\gamma^2\varepsilon_{\text{prep}}^2}{\lambda}\right) \subseteq \tilde{\mathcal{O}}\left(\frac{\delta^2\gamma^2\varepsilon_{\text{gse}}^4}{\lambda^5}\right). \qquad (D.41)$$

For consistency across different n, we set ε_{el} to a tighter bound. We then let

$$\varepsilon_{\text{prep}} \in \mathcal{O}\left(\frac{\varepsilon_{\text{gse}}^2}{\lambda^2}\right) \qquad (D.42)$$

$$\varepsilon_{\text{el}} \in \tilde{\mathcal{O}}\left(\frac{\delta^2\gamma^2\varepsilon_{\text{prep}}^2}{\lambda}\right) \subseteq \tilde{\mathcal{O}}\left(\frac{\delta^2\gamma^2\varepsilon_{\text{gse}}^4}{\lambda^5}\right), \qquad (D.43)$$

which still satisfies the former bound. The runtime is largely dominated by ground state preparation, hence it has the same asymptotic runtime as the latter. Replacing $\varepsilon_{\text{prep}}$ by its bounds of ε_{gse}, we note that we require $\mathcal{O}(\frac{\lambda}{\delta\gamma}\log\frac{\lambda}{\delta\varepsilon_{\text{gse}}})$ queries to U_H. The number of Toffoli gates needed can be found by Corollary B.7.1 to be

$$\mathcal{O}\left(\frac{\lambda}{\delta\gamma}\log\frac{\lambda}{\delta\varepsilon_{\text{gse}}}\left(a_{\text{el}} + \log\left(\frac{\lambda}{\delta\gamma\varepsilon_{\text{gse}}}\right)\right)\right). \qquad (D.44)$$

$\square$

Note that the requirement on the error bound of the electronic Hamiltonian is tighter than that of the requirements for implementing the force operator in Lemma A.2. However, due to the error being a logarithmic dependency, this is hidden in the asymptotics. Alternatively, we can remove this requirement if we take the approach in Ref. [11] and assume that the electronic Hamiltonian is a $(\lambda, a_{\text{el}}, 0)$-block-encoding of an alternative Hamiltonian $\tilde{H}$, which we can prepare the ground state to since we technically use this Hamiltonian as for ground state preparation. This requires dependency on an alternate spectral gap $\tilde{\gamma}$, which can still be bounded by γ by Weyl's inequality, and an alternative overlap bound $\tilde{\delta}$ on the approximate ground state oracle U_I, which in practice would be roughly δ, but further assumptions are required for theoretical completeness. Since the dependency on error is logarithmic in our scheme, we opt to consider the higher requirements on the electronic Hamiltonian block encoding, in contrast to adding another assumption on the oracle.

We now combine the Hamiltonian terms to produce a block encoding of the nuclear Hamiltonian H_{nuc}.

Proposition D.5 (Block encoding of nuclear Hamiltonian). *Suppose we have a nuclear-position-controlled electronic Hamiltonian that can be block-encoded in the first quantization scheme with scaling factor λ satisfying* Assumption C.1 *in regards to spectral gap γ, ground state upper bound μ. We are given an approximate ground state preparation oracle U_I that can prepare ground states up to an overlap of $1 - \delta$. Then there exists a $(\alpha_H, a_H, \varepsilon_H)$-block-encoding of H_{nuc} where*

$$\alpha_H \in \mathcal{O}\left(\frac{Np_{\text{max}}'^2}{m_{\text{min}}s_{\text{min}}^2} + \frac{N^2Z_{\text{max}}^2}{\Delta} + \lambda\right),$$

$$a_H \in \mathcal{O}\left(\tilde{N}\log B + \log\frac{1}{\delta\gamma} + \log\frac{\alpha_H}{\varepsilon_H}\right).$$

This block encoding can be prepared with $\mathcal{O}(\frac{1}{\delta})$ queries to U_I, and

$$\tilde{\mathcal{O}}\left(\frac{\lambda}{\delta\gamma}\log\frac{g\alpha_H}{\varepsilon_H}\left(N + \tilde{N}\log B + \log B\log\frac{Bg\alpha_H}{\varepsilon_H}\right)\right)$$

Toffoli gates, where $g = \max(g_x, g_{p'})$.

Proof. To take the combination of the electronic and classical Liouvillian, we first note that by Lemmas D.2 to D.4, we have a $(1, a_{\text{kin}}, \frac{\varepsilon_{\text{kin}}}{\alpha_{\text{kin}}})$-block-encoding of $\frac{H_{\text{kin}}}{\alpha_{\text{kin}}}$, a $(1, a_{\text{pot}}, \frac{\varepsilon_{\text{pot}}}{\alpha_{\text{pot}}})$-block-encoding of $\frac{H_{\text{pot}}}{\alpha_{\text{pot}}}$, and a $(1, a_{\text{gse}}, \frac{\varepsilon_{\text{gse}}}{\lambda})$-block-encoding of $\frac{H_{\text{gse}}}{\lambda}$. Let $\alpha_H = \alpha_{\text{kin}} + \alpha_{\text{pot}} + \lambda$. Then, given a state preparation unitary that prepares the state

$$\sqrt{\frac{\alpha_{\text{kin}}}{\alpha_H}}\,|00\rangle + \sqrt{\frac{\alpha_{\text{pot}}}{\alpha_H}}\,|01\rangle + \sqrt{\frac{\lambda}{\alpha_H}}\,|10\rangle \qquad (D.45)$$

that functions as a $(\alpha_H, 2, \varepsilon_{\text{rot}})$-state-preparation-pair, we can construct via Lemma B.2 a $(\alpha_H, a_H, \varepsilon_H)$-block-encoding of the Liouvillian L such that

$$\alpha_H = \alpha_{\text{kin}} + \alpha_{\text{pot}} + \lambda, \qquad (D.46)$$

$$a_H = \max(a_{\text{kin}}, a_{\text{pot}}, a_{\text{gse}}) + 2, \qquad (D.47)$$

$$\varepsilon_H = \alpha_H \cdot \max\left(\frac{\varepsilon_{\text{kin}}}{\alpha_{\text{kin}}}, \frac{\varepsilon_{\text{pot}}}{\alpha_{\text{pot}}}, \frac{\varepsilon_{\text{gse}}}{\lambda}\right) + \varepsilon_{\text{rot}}. \qquad (D.48)$$

Dealing with the errors, we set the following to satisfy the above bounds

$$\varepsilon_{\text{rot}} \in \mathcal{O}(\varepsilon_H), \qquad (D.49)$$

$$\varepsilon_{\text{kin}} \in \mathcal{O}\left(\frac{\alpha_{\text{kin}}\varepsilon_H}{\alpha_H}\right), \qquad (D.50)$$

$$\varepsilon_{\text{pot}} \in \mathcal{O}\left(\frac{\alpha_{\text{pot}}\varepsilon_H}{\alpha_H}\right), \qquad (D.51)$$

$$\varepsilon_{\text{gse}} \in \mathcal{O}\left(\frac{\lambda\varepsilon_H}{\alpha_H}\right), \qquad (D.52)$$

$$\varepsilon_{\text{el}} \in \tilde{\mathcal{O}}\left(\frac{\delta^2\gamma^2\varepsilon_H^4}{\lambda\alpha_H^4}\right). \qquad (D.53)$$

We can then use this to capture the bounds on the ancillae, where

$$a_H = \max(a_{\text{kin}}, a_{\text{pot}}, a_{\text{gse}}) + 2$$

$$\in \mathcal{O}\left(\max\left(\log\frac{\alpha_{\mathrm{kin}}}{\varepsilon_{\mathrm{kin}}}, \log\frac{\alpha_{\mathrm{pot}}}{\varepsilon_{\mathrm{pot}}}, \log\frac{\lambda}{\varepsilon_{\mathrm{el}}} + \tilde{N}\log B\right)\right)$$

$$\subseteq \mathcal{O}\left(\tilde{N}\log B + \log\frac{1}{\delta\gamma} + \log\frac{\alpha_H}{\varepsilon_H}\right). \qquad (D.54)$$

Now, for the cost of implementation, we note that we simply have to add the implementation cost of the kinetic, potential, and ground state energy Hamiltonians, plus the cost for the single rotation gate (which is negligible), and adjust for errors, which we can find to be as follows:

$$\tilde{\mathcal{O}}\left(\frac{\lambda}{\delta\gamma}\log\frac{g\alpha_H}{\varepsilon_H}\left(N + \tilde{N}\log B + \log B\log\frac{Bg\alpha_H}{\varepsilon_H}\right)\right). \tag{D.55}$$

Note that the number of ancillae and the number of gates is largely dominated by the ground state energy Hamiltonian, with the other two only affecting a logarithmic term regarding g. $\qquad\square$

If we consider only the particle numbers, the spectral gap, approximation ground state preparation overlap, and error precision, the scaling factor can be written as

$$\tilde{\mathcal{O}}\left(N_{\mathrm{tot}}^2\right). \tag{D.56}$$

while the gate cost can be written as

$$\tilde{\mathcal{O}}\left(\frac{\tilde{N}N_{\mathrm{tot}}}{\delta\gamma}\log\frac{1}{\varepsilon_H}\left(N_{\mathrm{tot}} + \log\frac{1}{\varepsilon_H}\right)\right),$$

$$\subseteq \tilde{\mathcal{O}}\left(\frac{\tilde{N}N_{\mathrm{tot}}^2}{\delta\gamma}\log^2\frac{1}{\varepsilon_H}\right). \tag{D.57}$$

Lastly, given the construction of the nuclear Hamiltonian H_{nuc}, we can construct the internal energy difference between two systems, A and B, as utilized in the thermodynamic integration algorithm by a simple LCU. We note here that, in the context of thermodynamic integration, a large portion of the kinetic and potential Hamiltonian terms cancel each other out. A more efficient block encoding with a smaller scaling factor can be constructed by examining the exact LCU terms and canceling identical terms. However, given that the ground state energy Hamiltonian is constructed by preparing the ground state on an electronic register and applying the electronic Hamiltonian, it may be hard to cancel terms out in this block encoding. Given that the scaling factor of the nuclear Hamiltonian H_{nuc} is primarily dominated by the ground state energy Hamiltonian, we note that even if we can cancel out most of the terms in the kinetic and potential operators, the scaling factor would still be in $\mathcal{O}(\tilde{N}N_{\mathrm{tot}})$ with additional terms dependent on the differing terms of the kinetic and potential Hamiltonians. Hence, for simplicity of theoretical constructions, we do not delve into the cancellation of Hamiltonian terms, but

instead construct the difference using a simple LCU, noting that in practice, the block encoding implementation can be made more efficient with a lower scaling factor.

It is then easy to see that when we are provided two systems A and B, given $H_{\Delta\mathrm{int}} = H_{\mathrm{int}_A} - H_{\mathrm{int}_B}$, by Lemma B.2, there exists a $(\alpha_\Delta, a_\Delta, \varepsilon_\Delta)$-block-encoding of $H_{\Delta\mathrm{int}}$ where

$$\alpha_\Delta \in \mathcal{O}\left(N_{\mathrm{tot},A}^2 + N_{\mathrm{tot},B}^2\right) \tag{D.58}$$

$$a_\Delta \in \mathcal{O}\Bigg(\max(\tilde{N}_A\log B_A, \tilde{N}_B\log B_B) \tag{D.59}$$

$$+ \log\frac{1}{\min(\delta_A\gamma_A, \delta_B\gamma_B)} + \log\frac{\alpha_\Delta}{\varepsilon_\Delta}\Bigg)$$

with cost

$$\tilde{\mathcal{O}}\left(\left(\frac{\tilde{N}_A N_{\mathrm{tot},A}^2}{\delta_A\gamma_A} + \frac{\tilde{N}_B N_{\mathrm{tot},B}^2}{\delta_B\gamma_B}\right)\log^2\frac{1}{\varepsilon_\Delta}\right). \tag{D.60}$$

4. The quantum thermodynamic integration algorithm

Our quantum implementation of the thermodynamic integration algorithm first prepares Λ-dependent Liouvillian simulations in superposition, such that we have

$$\frac{1}{\sqrt{N_\Lambda}}\sum_\Lambda |\Lambda\rangle |\rho_\Lambda\rangle = \frac{1}{\sqrt{N_\Lambda}}\sum_\Lambda |\Lambda\rangle \otimes e^{-iL_\Lambda t_{\mathrm{eq}}} |\rho_0\rangle \tag{D.61}$$

by the methods outlined in Appendix D 2. We then apply the controlled version of the block encoding of ΔH as shown in Appendix D 3 in the form of a Hadamard test circuit, where we then obtain probability

$$\mathbb{P}_{\Delta\tilde{F}} = \frac{1}{2}\left(\frac{\Delta\tilde{F}}{\alpha_\Delta} + 1\right), \tag{D.62}$$

of measuring the zero state on the measurement qubit. One can then use amplitude estimation to obtain an $\varepsilon/6\alpha_\Delta$-close estimate of $p_{\Delta\tilde{F}}$ with constant success probability. This in turn provides a $\varepsilon/3$-close estimate of $\Delta\tilde{F}$, and consequently a ε-close estimate of ΔF when Riemann sum errors and block encoding errors are sufficiently bounded. We refer the reader to Algorithm 2 for the complete algorithm description.

We now obtain the formal statement of the thermodynamic integration algorithm.

Theorem D.6 (Thermodynamic integration). *Given two systems A and B, suppose for both systems, we have a nuclear-position-controlled electronic Hamiltonian that can be block-encoded in the first quantization scheme with scaling factor λ satisfying Assumption C.1 in regards to spectral gap lower bound γ for both systems and that we are each given an approximate ground state preparation oracle $U_{I,A/B}$ that can prepare ground states up to an overlap of $1 - \delta$ for either system. We are given*

block encodings of discretized Liouvillians L_A and L_B under the NVT canonical ensemble encoding the dynamics for systems A and B per Proposition C.5, as well as block encodings of discretized nuclear Hamiltonians H_A and H_B for systems A and B per Proposition D.5. Further, we are given an initial state $|\rho_0\rangle$ that encodes the initial discretized phase-space density as well as an equilibration time $t_{\rm eq}$ for Liouvillian simulation by Theorem C.6. There exists a quantum algorithm that computes the Helmholtz free energy difference between the two systems up to additive ε-precision with success probability $1 - \xi$ using

$$\tilde{\mathcal{O}}\left(\frac{\alpha_H \alpha_L t_{\rm eq} \log g}{\varepsilon} \log \frac{1}{\xi} \right.$$
$$\left. \times \left(\frac{\lambda}{\delta\gamma}(N + \tilde{N}\log B + \log B \log Bg) + d \right) \right)$$

Toffoli gates,

$$\tilde{\mathcal{O}}\left(\frac{\alpha_H \alpha_L t_{\rm eq}}{\delta\varepsilon} \log \frac{1}{\xi} \right)$$

queries to $U_{I,A}$ and $U_{I,B}$, and

$$\tilde{\mathcal{O}}\left(N\log g + \tilde{N}\log B + \log \frac{d}{\delta\gamma} + \log \frac{\alpha_L \alpha_H t_{\rm eq}}{\varepsilon} \right)$$

qubits, where $N = \max(N_A, N_B)$, $\tilde{N} = \max(\tilde{N}_A, \tilde{N}_B)$, $N_{\rm tot} = N + \tilde{N}$, $\alpha_L := \max(\alpha_{L_A}, \alpha_{L_B})$ and α_{L_A} and α_{L_B} are as defined for systems A and B in Proposition C.5, $\alpha_H := \max(\alpha_{H_A}, \alpha_{H_B})$ and α_{H_A} and α_{H_B} are as defined for systems A and B in Proposition D.5, $d := \max(d_x, d_{p'}, d_s, d_{p_s})$ and $g := \max(g_x, g_{p'}, g_s, g_{p_s})$.

Proof. We now consider the error bounds of the algorithm. From Theorem D.6, we can prepare a state that ε_L-close in ℓ_2 norm to the Liouvillian evolved state $|\rho\rangle$, upon successful post-selection of the ancillae, which requires a constant factor times of reruns. Here, because we need to use the state as input to the Hadamard test, we do not want to perform post-selection. Here we use slightly looser bounds and note that prepared state $|\sigma\rangle$ (inclusive of the ancilla registers) can be prepared to be close enough to the target state $|\tilde{\rho}\rangle = |\rho\rangle |0\rangle_a$ with fidelity up to $1 - \varepsilon_L$. Formally, we write

$$|\langle\sigma|\tilde{\rho}\rangle| \geq 1 - \varepsilon_L. \tag{D.63}$$

The ℓ_2-norm difference can then be bounded as follows:

$$\||\sigma\rangle - |\tilde{\rho}\rangle\| \leq \sqrt{2\varepsilon_L} \tag{D.64}$$

We note that the above bounds applies to all Λ-dependent states $|\rho_\Lambda\rangle$. Then applying the state to the $(\alpha_\Delta, a_\Delta, \varepsilon_\Delta)$-block-encoding U_Δ of the Hamiltonian difference $\Delta H = H_B - H_A$, we note the total block encoding errors can be bounded as follows:

$$|\alpha_\Delta \langle\sigma| U_\Delta |\rho\rangle - \langle\sigma| \Delta H |\rho\rangle|$$

$$\leq \alpha_\Delta |\langle\sigma| U_\Delta |\sigma\rangle - \langle\sigma| U_\Delta |\tilde{\rho}\rangle|$$
$$+ \alpha_\Delta |\langle\sigma| U_\Delta |\tilde{\rho}\rangle - \langle\tilde{\rho}| U_\Delta |\tilde{\rho}\rangle|$$
$$+ |\alpha_\Delta \langle\tilde{\rho}| U_\Delta |\tilde{\rho}\rangle - \langle\rho| \Delta H |\rho\rangle|$$
$$\leq 2\alpha_\Delta \sqrt{2\varepsilon_L} + \varepsilon_\Delta \tag{D.65}$$

To obtain the total error over the state $\frac{1}{\sqrt{N_\Lambda}}\sum_\Lambda |\Lambda\rangle |\rho_\Lambda\rangle$, we simply compute the following

$$\left| \frac{\alpha_\Delta}{N_\Lambda}\left(\sum_\Lambda \langle\Lambda| \langle\sigma_\Lambda| \right) \mathbb{I} \otimes U_\Delta \left(\sum_\Lambda |\Lambda\rangle |\sigma_\Lambda\rangle \right) \right.$$
$$\left. - \frac{1}{N_\Lambda}\left(\sum_\Lambda \langle\Lambda| \langle\rho_\Lambda| \right) \mathbb{I} \otimes \Delta H \left(\sum_\Lambda |\Lambda\rangle |\rho_\Lambda\rangle \right) \right|$$
$$= \frac{1}{N_\Lambda}\left| \sum_\Lambda \alpha_\Delta \langle\sigma_\Lambda| U_\Delta |\sigma_\Lambda\rangle - \langle\rho_\Lambda| \Delta H |\rho_\Lambda\rangle \right|$$
$$\leq \frac{1}{N_\Lambda}\sum_\Lambda |\alpha_\Delta \langle\sigma_\Lambda| U_\Delta |\sigma_\Lambda\rangle - \langle\rho_\Lambda| \Delta H |\rho_\Lambda\rangle|$$
$$\leq \frac{1}{N_\Lambda}\sum_\Lambda (2\alpha_\Delta \sqrt{2\varepsilon_L} + \varepsilon_\Delta)$$
$$= 2\alpha_\Delta \sqrt{2\varepsilon_L} + \varepsilon_\Delta \tag{D.66}$$

From the discretization errors of Riemann sum bounds, we add an additional term of $\varepsilon_{]{\rm disc}}$, which is shown in Appendix D 1. From the Hadamard test circuit, we obtain the probability

$$\mathbb{P}_{\Delta\tilde{F}} = \frac{1}{2}\left(\frac{\Delta\tilde{F}}{\alpha_\Delta} + 1 \right), \tag{D.67}$$

on measurement of the zero state. Viewing the entire circuit (the Λ-Liouvillian evolution and the Hadamard test) as a quantum state $|\psi\rangle$ we note that

$$\text{Tr}\left(|\psi\rangle\langle\psi|(|0\rangle\langle0| \otimes \mathbb{I}) \right) = \mathbb{P}_{\Delta\tilde{F}} \tag{D.68}$$

By amplitude estimation [49, 71], we can output an $\frac{\varepsilon_{qae}}{2\alpha_\Delta}$-close estimation of $p_{\Delta\tilde{F}}$ with $\mathcal{O}(\frac{\alpha_\Delta}{\varepsilon_{qae}})$ accesses to $2|\psi\rangle\langle\psi| - \mathbb{I}$ and $2|0\rangle\langle0| \otimes \mathbb{I} - \mathbb{I}$. Thus, the total error in the estimation of ΔF is then found to be

$$2\alpha_\Delta \sqrt{2\varepsilon_L} + \varepsilon_\Delta + \varepsilon_{\rm disc} + \varepsilon_{\rm qae} \tag{D.69}$$

Upper-bounding the total error by ε, we note that the error terms are bounded as follows:

$$\varepsilon_L \in \mathcal{O}\left(\frac{\varepsilon^2}{\alpha_\Delta^2} \right) \tag{D.70}$$

$$\varepsilon_\Delta, \varepsilon_{\rm qae}, \varepsilon_{\rm disc} \in \mathcal{O}(\varepsilon) \tag{D.71}$$

Plugging these results into the runtime, we note that the cost of block-encoding the Hamiltonian in Proposition D.5 is overshadowed by the cost of block-encoding the Λ-dependent Liouvillian evolution in Theorem C.6. Then, factoring the cost of amplitude estimation to precision $\varepsilon/\alpha_\Delta$, we multiply the cost of Liouvillian simulation by the inverse of the precision, while noting that

$\alpha_\Delta \in \mathcal{O}(\alpha_H)$. This, in turn, overshadows the polylogarithmic error terms in the Liouvillian simulation. Lastly, we include the additional cost of using median boosting to raise the success probability to $1 - \xi$, which yields the result in the theorem. The number of qubits can be found by plugging in the error bounds of ε_L as provided above to Theorem C.6. $\qquad\square$

Note that if we consider only the particle numbers, the spectral gap, approximation ground state preparation overlap, and error precision, the runtime can be written as

$$\tilde{\mathcal{O}} \left(\frac{N \tilde{N} N_{\text{tot}}^5 t_{\text{eq}}}{\delta \gamma \varepsilon} \log \frac{1}{\xi} \right) \qquad (D.72)$$

as shown in the informal version of the theorem (Theorem 3) in the main text.

Appendix E: Discretization errors of the Liouvillian

We provide a qualitative discussion of the results of errors in the discretization of the Liouvillian. From Theorems C.6 and D.6, we note that the number of grid points in the nuclear registers g and the number of plane wave bases in the electronic register B are both only logarithmically dependent within the runtime. Hence, one can potentially decrease the error caused by discretization by increasing g and B to a large number without making polynomial increases to the runtime.

Regarding approximating the derivative of position and momentum via central finite difference to order d, we note that the discretization error of a function f can be upper bounded [108] by

$$\varepsilon \in \mathcal{O} \left(\max_{\xi \in [a,b]} \left| f^{(2d+1)}(\xi) \right| \left(\frac{eh}{2} \right)^{2d} \right), \qquad (E.1)$$

for which $d \in \mathcal{O}(\log \frac{1}{\varepsilon})$ if h is constant. Thus, given that the dependency of the central finite difference orders d in the runtime of Theorems C.6 and D.6 is only linear, the error stemming from this discretization error can potentially be suppressed, provided that there exists a constant $\mathcal{C}$ such that

$$\mathcal{C} \geq \sup_{d \geq 0} \max_{\xi \in [a,b]} \left| f^{(2d+1)}(\xi) \right|^{\frac{1}{2d}}. \qquad (E.2)$$

However, this assumption is, in fact, difficult to validate in general, as the maximum scale of the higher-order derivatives is difficult to bound in full generality as the continuum limit is approached. This discretization issue remains an important consideration within the field [109] and the development of alternative differential operators that do not rely on central finite difference would serve as value material for future work.

[1] J. D. Chodera, D. L. Mobley, M. R. Shirts, R. W. Dixon, K. Branson, and V. S. Pande, Alchemical free energy methods for drug discovery: progress and challenges, Curr. Opin. Struct. Biol. **21**, 150 (2011).

[2] A. de Ruiter, D. Petrov, and C. Oostenbrink, Optimization of alchemical pathways using extended thermodynamic integration, J. Chem. Theory Comput. **17**, 56 (2021).

[3] D. Frenkel and B. Smit, *Understanding Molecular Simulation: From Algorithms to Applications* (Academic Press, Inc., USA, 1996).

[4] J. Kästner, H. M. Senn, S. Thiel, N. Otte, and W. Thiel, QM/MM free-energy perturbation compared to thermodynamic integration and umbrella sampling: Application to an enzymatic reaction, J. Chem. Theory Comput. **2**, 452 (2006).

[5] Z. Cournia, B. Allen, and W. Sherman, Relative binding free energy calculations in drug discovery: Recent advances and practical considerations, J. Chem. Inf. Model. **57**, 2911 (2017).

[6] M. R. Shirts, D. L. Mobley, and J. D. Chodera, Chapter 4 Alchemical free energy calculations: Ready for prime time?, in *Annual Reports in Computational Chemistrys*, Vol. 3, edited by D. Spellmeyer and R. Wheeler (Elsevier, 2007) pp. 41–59.

[7] A. S. Mey, B. K. Allen, H. E. Bruce Macdonald, J. D. Chodera, D. F. Hahn, M. Kuhn, J. Michel, D. L. Mobley, L. N. Naden, S. Prasad, A. Rizzi, J. Scheen, M. R. Shirts, G. Tresadern, and H. Xu, Best practices for alchemical free energy calculations, Living J. Comp. Mol. Sci. **2** (2020).

[8] D. H. de Jong, L. V. Schäfer, A. H. de Vries, S. J. Marrink, H. J. C. Berendsen, and H. Grubmüller, Determining equilibrium constants for dimerization reactions from molecular dynamics simulations, J. Comput. Chem **32**, 1919–1928 (2011).

[9] 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).

[10] L. Lin and Y. Tong, Near-optimal ground state preparation, Quantum **4**, 372 (2020).

[11] S. Simon, R. Santagati, M. Degroote, N. Moll, M. Streif, and N. Wiebe, Improved precision scaling for simulating coupled quantum-classical dynamics, PRX Quantum **5**, 010343 (2024).

[12] 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).

[13] R. Cleve, A. Ekert, C. Macchiavello, and M. Mosca, Quantum algorithms revisited, Proc. R. Soc. A **454**, 339–354 (1998).

[14] A. M. Childs and N. Wiebe, Hamiltonian simulation using linear combinations of unitary operations, Quantum

Info. Comput. **12**, 901–924 (2012).

[15] G. H. Low and I. L. Chuang, Optimal Hamiltonian simulation by quantum signal processing, Phys. Rev. Lett. **118**, 010501 (2017).

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

[17] A. Gilyén, Y. Su, G. H. Low, and N. Wiebe, Quantum singular value transformation and beyond: Exponential improvements for quantum matrix arithmetics, in *Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing*, STOC 2019 (Association for Computing Machinery, New York, NY, USA, 2019) pp. 193–204.

[18] A. Y. Kitaev, Quantum measurements and the abelian stabilizer problem (1995), arXiv:quant-ph/9511026 [quant-ph].

[19] H. Hellmann, *Einführung in die Quantenchemie* (Franz Deuticke, Leipzig, 1937).

[20] R. P. Feynman, Forces in molecules, Phys. Rev. **56**, 340 (1939).

[21] J. G. Kirkwood, Statistical mechanics of fluid mixtures, J. Chem. Phys. **3**, 300 (1935).

[22] R. W. Zwanzig, High-temperature equation of state by a perturbation method. II. Polar gases, J. Chem. Phys. **23**, 1915 (1955).

[23] C. H. Bennett, Efficient estimation of free energy differences from Monte Carlo data, J. Comput. Phys. **22**, 245 (1976).

[24] M. P. Allen and D. J. Tildesley, *Computer Simulation of Liquids* (Oxford University Press, 1987).

[25] B. J. Williams-Noonan, E. Yuriev, and D. K. Chalmers, Free energy methods in drug design: Prospects of "alchemical perturbation" in medicinal chemistry, J. Med. Chem. **61**, 638 (2018).

[26] N. Hansen and W. F. van Gunsteren, Practical aspects of free-energy calculations: A review, J. Chem. Theory Comput. **10**, 2632 (2014).

[27] R. P. Feynman, Simulating physics with computers, Int. J. Theor. Phys. **21**, 467–488 (1982).

[28] S. Lloyd, Universal quantum simulators, Science **273**, 1073–1078 (1996).

[29] A. Caesura, C. L. Cortes, W. Pol, S. Sim, M. Steudtner, G.-L. R. Anselmetti, M. Degroote, N. Moll, R. Santagati, M. Streif, and C. S. Tautermann, Faster quantum chemistry simulations on a quantum computer with improved tensor factorization and active volume compilation (2025), arXiv:2501.06165 [quant-ph].

[30] G. H. Low, R. King, D. W. Berry, Q. Han, A. E. D. III, A. White, R. Babbush, R. D. Somma, and N. C. Rubin, Fast quantum simulation of electronic structure by spectrum amplification (2025), arXiv:2502.15882 [quant-ph].

[31] M. Reiher, N. Wiebe, K. M. Svore, D. Wecker, and M. Troyer, Elucidating reaction mechanisms on quantum computers, Proc. Natl. Acad. Sci. U.S.A. **114**, 7555 (2017).

[32] S. McArdle, S. Endo, A. Aspuru-Guzik, S. C. Benjamin, and X. Yuan, Quantum computational chemistry, Rev. Mod. Phys. **92**, 015003 (2020).

[33] V. von Burg, G. H. Low, T. Häner, D. S. Steiger, M. Reiher, M. Roetteler, and M. Troyer, Quantum computing enhanced computational catalysis, Phys. Rev. Res. **3**, 033055 (2021).

[34] C. Outeiral, M. Strahm, J. Shi, G. M. Morris, S. C. Benjamin, and C. M. Deane, The prospects of quantum computing in computational molecular biology, Wiley Interdiscip. Rev. Comput. Mol. Sci. **11**, e1481 (2021).

[35] N. C. Rubin, D. W. Berry, F. D. Malone, A. F. White, T. Khattar, A. E. DePrince, S. Sicolo, M. Küehn, M. Kaicher, J. Lee, and R. Babbush, Fault-tolerant quantum simulation of materials using Bloch orbitals, PRX Quantum **4**, 040303 (2023).

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

[37] M. Born and R. Oppenheimer, Zur Quantentheorie der Molekeln, Ann. Phys. (Leipzig) **389**, 457 (1927).

[38] 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, Quantum **7**, 1164 (2023).

[39] J. Gonzalez-Conde and A. T. Sornborger, Mixed quantum-semiclassical simulation (2023), arXiv:2308.16147 [quant-ph].

[40] J. Liouville, Sur la théorie de la variation des constantes arbitraires., J. Math. Pures Appl. , 342 (1838).

[41] H. F. Trotter, On the product of semi-groups of operators, Proc. Am. Math. Soc. **10**, 545–551 (1959).

[42] 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).

[43] D. W. Berry, A. M. Childs, R. Cleve, R. Kothari, and R. D. Somma, Simulating Hamiltonian dynamics with a truncated Taylor series, Phys. Rev. Lett. **114**, 090502 (2015).

[44] S. Chakraborty, A. Gilyén, and S. Jeffery, The power of block-encoded matrix powers: Improved regression techniques via faster Hamiltonian simulation, in *46th International Colloquium on Automata, Languages, and Programming (ICALP 2019)*, Leibniz International Proceedings in Informatics (LIPIcs), Vol. 132, edited by C. Baier, I. Chatzigiannakis, P. Flocchini, and S. Leonardi (Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2019) pp. 33:1–33:14.

[45] G. H. Low and N. Wiebe, Hamiltonian simulation in the interaction picture (2018), arXiv:1805.00675 [quant-ph].

[46] M. Suzuki, Generalized Trotter's formula and systematic approximants of exponential operators and inner derivations with applications to many-body problems, Commun. Math. Phys. **51**, 183–190 (1976).

[47] M. Suzuki, General theory of fractal path integrals with applications to many-body theories and statistical physics, J. Math. Phys. **32**, 400–407 (1991).

[48] A. Gilyén and T. Li, Distributional property testing in a quantum world, in *11th Innovations in Theoretical Computer Science Conference (ITCS 2020)*, Leibniz International Proceedings in Informatics (LIPIcs), Vol. 151, edited by T. Vidick (Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl, Germany, 2020) pp. 25:1–25:19.

[49] G. Brassard, P. Høyer, M. Mosca, and A. Tapp, Quantum amplitude amplification and estimation, in *Quantum computation and information*, Contemporary Mathematics, Vol. 305 (American Mathematical Society, Providence, RI, USA, 2002) pp. 53–74.

[50] J. W. Gibbs, *Elementary Principles in Statistical Mechanics: Developed with Especial Reference to the Rational Foundation of Thermodynamics* (Cambridge University Press, 1902).

[51] S. Nosé, A molecular dynamics method for simulations in the canonical ensemble, Mole. Phys. **52**, 255–268 (1984).

[52] S. Nosé, A unified formulation of the constant temperature molecular dynamics methods, J. Chem. Phys. **81**, 511–519 (1984).

[53] J. C. Tully, Perspective on "Zur Quantentheorie der Molekeln", Theor. Chem. Acc. **103**, 173 (2000).

[54] B. Bransden and C. Joachain, *Physics of atoms and molecules*, 2nd ed. (Pearson Education, 2003).

[55] B. O. Koopman, Hamiltonian systems and transformation in Hilbert space, Proc. Natl. Acad. Sci. U.S.A. **17**, 315–318 (1931).

[56] J. von Neumann, Zur Operatorenmethode in der Klassischen Mechanik, Ann. Math. **33**, 587 (1932).

[57] J. von Neumann, Zusatze zur Arbeit "Zur Operatorenmethode"..., Ann. Math. **33**, 789 (1932).

[58] D. Mauro, On Koopman–von Neumann waves, Int. J. Mod. Phys. A **17**, 1301–1325 (2002).

[59] D. W. Berry, A. M. Childs, R. Cleve, R. Kothari, and R. D. Somma, Exponential improvement in precision for simulating sparse Hamiltonians, in *Proceedings of the Forty-Sixth Annual ACM Symposium on Theory of Computing*, STOC '14 (Association for Computing Machinery, New York, NY, USA, 2014) p. 283–292.

[60] S. Simon, M. Degroote, N. Moll, R. Santagati, M. Streif, and N. Wiebe, Amplified amplitude estimation: Exploiting prior knowledge to improve estimates of expectation values (2024), arXiv:2402.14791 [quant-ph].

[61] A. Alase, Quantum signal processing without angle finding (2025), arXiv:2501.07002 [quant-ph].

[62] M. Szegedy, Quantum speed-up of Markov chain based algorithms, in *45th Annual IEEE Symposium on Foundations of Computer Science* (2004) pp. 32–41.

[63] E. Knill, G. Ortiz, and R. D. Somma, Optimal quantum measurements of expectation values of observables, Phys. Rev. A **75**, 012328 (2007).

[64] S. Istrail, Statistical mechanics, three-dimensionality and NP-completeness: I. Universality of intracatability for the partition function of the Ising model across non-planar surfaces (extended abstract), in *Proceedings of the Thirty-Second Annual ACM Symposium on Theory of Computing*, STOC '00 (Association for Computing Machinery, New York, NY, USA, 2000).

[65] R. Chao, D. Ding, A. Gilyen, C. Huang, and M. Szegedy, Finding angles for quantum signal processing with machine precision (2020), arXiv:2003.02831 [quant-ph].

[66] Y. Dong, X. Meng, K. B. Whaley, and L. Lin, Efficient phase-factor evaluation in quantum signal processing, Phys. Rev. A **103** (2021).

[67] J. Wang, Y. Dong, and L. Lin, On the energy landscape of symmetric quantum signal processing, Quantum **6**, 850 (2022).

[68] D. Motlagh and N. Wiebe, Generalized quantum signal processing, PRX Quantum **5**, 020368 (2024).

[69] M. Jorge, N. M. Garrido, A. J. Queimada, I. G. Economou, and E. A. Macedo, Effect of the integration method on the accuracy and computational efficiency of free energy calculations using thermodynamic integration, J. Chem. Theory Comput. **6**, 1018–1027 (2010).

[70] N. Metropolis and S. Ulam, The Monte Carlo method, J. Am. Stat. Assoc. **44**, 335–341 (1949).

[71] A. Montanaro, Quantum speedup of Monte Carlo methods, Proc. R. Soc. A **471**, 20150301 (2015).

[72] L. Verlet, Computer "experiments" on classical fluids. I. Thermodynamical properties of Lennard-Jones molecules, Phys. Rev. **159**, 98 (1967).

[73] M. Tuckerman, B. J. Berne, and G. J. Martyna, Reversible multiple time scale molecular dynamics, J. Chem. Phys. **97**, 1990–2001 (1992).

[74] H. Yoshida, Construction of higher order symplectic integrators, Phys. Lett. A **150**, 262–268 (1990).

[75] N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller, and E. Teller, Equation of state calculations by fast computing machines, J. Chem. Phys. **21**, 1087–1092 (1953).

[76] W. K. Hastings, Monte carlo sampling methods using markov chains and their applications, Biometrika **57**, 97–109 (1970).

[77] S. McArdle, S. Endo, A. Aspuru-Guzik, S. C. Benjamin, and X. Yuan, Quantum computational chemistry, Rev. Mod. Phys. **92**, 015003 (2020).

[78] P. Hohenberg and W. Kohn, Inhomogeneous electron gas, Phys. Rev. **136**, B864 (1964).

[79] W. Kohn and L. J. Sham, Self-consistent equations including exchange and correlation effects, Phys. Rev. **140**, A1133 (1965).

[80] K. Burke, J. Werschnik, and E. K. U. Gross, Time-dependent density functional theory: Past, present, and future, J. Chem. Phys. **123**, 062206 (2005).

[81] R. A. Friesner, Ab initio quantum chemistry: Methodology and applications, Proc. Natl. Acad. Sci. U.S.A. **102**, 6648–6653 (2005).

[82] A. Warshel and M. Levitt, Theoretical studies of enzymic reactions: Dielectric, electrostatic and steric stabilization of the carbonium ion in the reaction of lysozyme, J. Mol. Biol. **103**, 227–249 (1976).

[83] H. M. Senn and W. Thiel, QM/MM methods for biomolecular systems, Angew. Chem. **48**, 1198–1229 (2009).

[84] E. Cancés, F. Castella, P. Chartier, E. Faou, C. L. Bris, F. Legoll, and G. Turinici, Long-time averaging for integrable hamiltonian dynamics, Num. Math. **100**, 211–232 (2005).

[85] E. Cancès, F. Castella, P. Chartier, E. Faou, C. Le Bris, F. Legoll, and G. Turinici, High-order averaging schemes with error bounds for thermodynamical properties calculations by molecular dynamics simulations, J. Chem. Phys. **121**, 10346–10355 (2004).

[86] J. M. Deutsch, Eigenstate thermalization hypothesis, Rep. Prog. Phys. **81**, 082001 (2018).

[87] T. Shirai and T. Mori, Thermalization in open many-body systems based on eigenstate thermalization hypothesis, Phys. Rev. E **101**, 042116 (2020).

[88] C.-F. Chen and F. G. S. L. B. ao, Fast thermalization from the eigenstate thermalization hypothesis (2023), arXiv:2112.07646 [quant-ph].

[89] G. Almeida, P. Ribeiro, M. Haque, and L. Sá, Universality, robustness, and limits of the eigenstate thermalization hypothesis in open quantum systems (2025), arXiv:2504.10261 [cond-mat.stat-mech].

[90] C.-F. Chen, M. J. Kastoryano, F. G. S. L. B. ao, and A. Gilyén, Quantum thermal state preparation (2023), arXiv:2303.18224 [quant-ph].

[91] P. Rall, C. Wang, and P. Wocjan, Thermal state preparation via rounding promises, Quantum **7**, 1132 (2023).

[92] A. Ramkumar and M. Soleimanifar, Mixing time of quantum Gibbs sampling for random sparse Hamiltonians (2024), arXiv:2411.04454 [quant-ph].

[93] C. Rouzé, D. S. França, and Á. M. Alhambra, Optimal quantum algorithm for Gibbs state preparation (2024), arXiv:2411.04885 [quant-ph].

[94] Y. Tong and Y. Zhan, Fast mixing of weakly interacting fermionic systems at any temperature (2025), arXiv:2501.00443 [quant-ph].

[95] Š. Šmíd, R. Meister, M. Berta, and R. Bondesan, Polynomial time quantum Gibbs sampling for Fermi-Hubbard model at any temperature (2025), arXiv:2501.01412 [quant-ph].

[96] J. M. Martyn, Z. M. Rossi, A. K. Tan, and I. L. Chuang, Grand unification of quantum algorithms, PRX Quantum **2**, 040203 (2021).

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

[98] G. H. Low and I. L. Chuang, Hamiltonian simulation by uniform spectral amplification (2017), arXiv:1707.05391 [quant-ph].

[99] H. Weyl, Das asymptotische Verteilungsgesetz der Eigenwerte linearer partieller Differentialgleichungen (mit einer Anwendung auf die Theorie der Hohlraumstrahlung), Math. Ann. **71**, 441–479 (1912).

[100] T. G. Draper, Addition on a quantum computer (2000), arXiv:quant-ph/0008033 [quant-ph].

[101] O. Hölder, Ueber einen Mittelwertsatz, Nachr. von Königl, Ges. Wiss. Georg-Augusts-Univ. Gött. **2**, 38 (1889).

[102] R. A. Horn and C. R. Johnson, *Topics in Matrix Analysis* (Cambridge University Press, 1991).

[103] B. Baumgartner, An inequality for the trace of matrix products, using absolute values (2011), arXiv:1106.6189 [math-ph].

[104] R. Schatten, *Norm Ideals of Completely Continuous Operators*, Ergebnisse der Mathematik und ihrer Grenzgebiete, Vol. 27 (Springer, Berlin, 1960).

[105] J. E. Volder, The CORDIC trigonometric computing technique, IRE Trans. Electron. Comput. **EC-8**, 330 (1959).

[106] V. Vedral, A. Barenco, and A. Ekert, Quantum networks for elementary arithmetic operations, Phys. Rev. A **54**, 147–153 (1996).

[107] I. Burge, M. Barbeau, and J. Garcia-Alfaro, Quantum CORDIC – arcsin on a budget (2024), arXiv:2411.14434 [quant-ph].

[108] J. Li, General explicit difference formulas for numerical differentiation, J. Comput. Appl. Math. **183**, 29–52 (2005).

[109] R. J. LeVeque, *Finite Difference Methods for Ordinary and Partial Differential Equations: Steady-State and Time-Dependent Problems* (Society for Industrial and Applied Mathematics, 2007).