

Digital Discovery

Accepted Manuscript

This article can be cited before page numbers have been issued, to do this please use: E. Eklund and N. Ananth, *Digital Discovery*, 2026, DOI: 10.1039/D5DD00381D.



This is an Accepted Manuscript, which has been through the Royal Society of Chemistry peer review process and has been accepted for publication.

Accepted Manuscripts are published online shortly after acceptance, before technical editing, formatting and proof reading. Using this free service, authors can make their results available to the community, in citable form, before we publish the edited article. We will replace this Accepted Manuscript with the edited and formatted Advance Article as soon as it is available.

You can find more information about Accepted Manuscripts in the [Information for Authors](#).

Please note that technical editing may introduce minor changes to the text and/or graphics, which may alter content. The journal's standard [Terms & Conditions](#) and the [Ethical guidelines](#) still apply. In no event shall the Royal Society of Chemistry be held responsible for any errors or omissions in this Accepted Manuscript or any consequences arising from the use of any information it contains.

Cite this: DOI: 00.0000/xxxxxxxxxx

Hybrid Quantum Algorithm for Simulating Real-Time Thermal Correlation Functions

Elliot Eklund^{ab} and Nandini Ananth^bReceived Date
Accepted Date

DOI: 00.0000/xxxxxxxxxx

We present a hybrid Path Integral Monte Carlo (hPIMC) algorithm to calculate real-time quantum thermal correlation functions for condensed-phase systems. The hPIMC algorithm leverages the successes of classical PIMC as a computational tool for high-dimensional system studies by exactly simulating dissipation using the Feynman-Vernon influence functional on a classical computer. Our method does not rely on variational quantum algorithms and offers an alternative path for near-term quantum algorithm design. We show that by using a quantum computer to compute short-time matrix elements of the quantum propagator, our hybrid algorithm achieves an asymptotic quantum speedup over its classical counterpart, given an efficient oracle for the system Hamiltonian. We employ the recently developed Probabilistic Imaginary-Time Evolution (PITE) algorithm to simulate imaginary-time evolution accurately, and we introduce a novel low-depth circuit that simulates evolution under the kinetic energy operator using an approximate Discrete Variable Representation (DVR). We investigate the accuracy of combining PITE with an approximate DVR by computing the position-position thermal correlation function of a proton transfer reaction on a classical computer. We estimate the circuit depth and CNOT gate count to compute short-time matrix elements for a 1D system with our method to be on the order of 10^5 with a space overhead of 30 qubits.

1 Introduction

Real-time thermal correlation functions (TCF) are a valuable tool for chemists as they provide insight into the dynamics of physical systems and are used to interpret spectroscopic data. However, computing real-time TCFs is difficult because one must propagate the system of interest under its time evolution operator (TEO), which is generally expensive to compute exactly using classical computers. As a result, the computation of exact, real-time TCFs is often limited to low-dimensional systems.^{1–4}

One approach to compute real-time TCFs is to use a path integral formulation in which the TEOs are split into products of many short-time TEO that can be evaluated accurately. In the case of open quantum systems, the path integral framework includes the effects of the environment degrees of freedom exactly through the Feynman-Vernon influence functional.^{5–7} The numerical success of the path integral relies on accurate evaluation of the TEO through the short-time Trotter approximation that becomes exact in the limit of a very small time interval.^{8,9} Additionally, the path integral formulation allows one to use a host of

Path Integral Monte Carlo (PIMC) methods to compute the necessary integrals of the TCF.^{10–13} Since their introduction, significant advances have been made in the numerical implementations of classical PIMC approaches for real-time quantum TCFs.^{14,15} However, this typically corresponds to a significant increase in the number of *environment* degrees of freedom with the system TEO being computationally feasible for only a relatively low dimensional model where, at most, a couple of continuous degrees of freedom coupled to multiple electronic states can be described.

Within the field of quantum computing the situation is different. Here, several well-known quantum algorithms exist that evolve a given quantum system under an approximation to the TEO using only a polynomial amount of resources, providing a quantum speedup over classical algorithms.^{16–23} In principle, one could construct an algorithm to compute real-time TCFs in which the TEO matrix elements are evaluated using these quantum algorithms. Algorithms designed for currently available quantum computers, referred to as noisy intermediate-scale quantum (NISQ) computers, have to account for strict hardware limitations, such as small qubit counts and short coherence times.^{24–28} One way this is achieved is by using hybrid quantum algorithms in which inexpensive subroutines are performed classically while expensive, quantum-in-nature calculations are offloaded to a quantum computer. This allows quantum circuits to be kept shallow and mitigates error accumulation due to hardware noise.^{27,29–37}

We propose the first hybrid PIMC (hPIMC) algorithm and show

^a School of Chemistry, University of Sydney, NSW 2006, Australia; E-mail: elliot.eklund@sydney.edu.au

^b Department of Chemistry and Chemical Biology, Cornell University, Ithaca, New York 14853, USA; E-mail: ananth@cornell.edu

† Supplementary Information available: [details of any supplementary information available should be included here]. See DOI: 00.0000/00000000.



that it achieves an asymptotic speedup for computing real-time TCFs of open quantum systems compared to a classical implementation of the same algorithm assuming an efficient oracle for the system Hamiltonian. The central ideas of the hPIMC approach are to use a quantum computer to push the boundaries of the system size limitations working within the path integral representation where only short-time TEO matrix elements are required. Computing these matrix elements using relatively shallow quantum circuits can mitigate errors that result from deep circuits. The resulting hybrid Monte Carlo algorithm performs operations such as sampling random paths and evaluating acceptance/rejection criteria classically, while using a low-depth quantum circuit to perform the expensive task of computing TEO matrix elements.

The fundamental algorithm design of hPIMC is similar to a recently proposed hybrid algorithm by Layden *et al* for quantum-enhanced Monte Carlo simulations.³⁸ Specifically, they introduce a hybrid algorithm based on Markov chain Monte Carlo and Metropolis-Hastings acceptance/rejection criteria for sampling the Boltzmann distribution of an Ising spin model and demonstrate a significant speedup over classical methods. Both our method and Layden's represent a new class of hybrid algorithm design that is characterized by two important features: They are constructed starting from classical Monte Carlo-based methods, and they do not call subroutines that use variational quantum algorithms (VQA).^{27,30–37,39–41} As a result, these hybrid methods do not face the same limitations as VQA-based methods, such as the emergence of barren plateaus and strict limitations on circuit depth.^{42–45}

In this work, we introduce the hPIMC algorithm to compute real-time quantum TCFs in an arbitrary representation and show that it achieves an asymptotic speedup compared with classical PIMC. We then adapt the hPIMC algorithm to condensed-phase systems modeled by the Caldeira-Leggett (CL) Hamiltonian in position space and introduce an influence functional to capture dissipative effects.⁴⁶ We use the Probabilistic Imaginary Time Evolution (PITE) algorithm^{47,48} for imaginary-time evolution of the TEO matrix elements. We further propose a new approach to approximate system propagation under the kinetic energy operator using a discrete variable representation (DVR)^{49–51} and prove an algorithm for doing so. We test the accuracy of this approach with classical simulations of the position TCF of a proton transfer reaction and estimate the circuit depth of hPIMC.

2 Theory

We start this section by briefly reviewing the classical PIMC formalism for computing TCFs, then introduce our algorithm for hPIMC. Next, we specialize hPIMC to open quantum systems described by the Caldeira-Leggett Hamiltonian. Finally, we give subroutines used during hPIMC to compute TEO matrix elements on a quantum computer.

The symmetrized real-time TCF between two operators, A and B , at time t for a system at equilibrium is⁵²

$$C_{AB}(t) = \frac{1}{Z} \text{Tr} \left(U^\dagger(t_c) A U(t_c) B \right), \quad (1)$$

where $U(t_c) = e^{-iHt_c}$ is the complex-time evolution operator, H

is the Hamiltonian of the system, $t_c = t - i\beta/2$, $\beta = 1/k_B T$, k_B is the Boltzmann constant, and T is the temperature. Throughout, we work in atomic units so that $\hbar = k_B = 1$. In a classical PIMC calculation, the partition function, Z , can be computed through a separate normalization calculation^{3,53} and we do not explicitly address methods of computing it.

Within the classical PIMC framework, $U(t_c)$ and $U^\dagger(t_c)$ are both discretized into a product of N short-time TEOs, $U(\Delta t_c)$ and $U^\dagger(\Delta t_c)$, respectively, called Trotter steps. We use the terms Trotter step and short-time TEO interchangeably. The discretized TCF is

$$C_{AB}(t) = \frac{1}{Z} \text{Tr} \left(\left(\prod_{k=1}^N U^\dagger(\Delta t_c) \right) A \left(\prod_{k=1}^N U(\Delta t_c) \right) B \right). \quad (2)$$

where N is the number of Trotter steps, and $\Delta t_c = t_c/N$. To evaluate $C_{AB}(t)$, we insert copies of identity $I = \sum_j |\phi_j\rangle\langle\phi_j|$, where $\{\phi_j\}$ is a complete basis, into Eq. (2). Here, the choice of basis is arbitrary. The resulting expression is

$$C_{AB}(t) = \frac{1}{Z} \sum_J \left(\prod_{k=N+2}^{2N+1} \langle\phi_{j_{k+1}}|U^\dagger(\Delta t_c)|\phi_{j_k}\rangle \right) \langle\phi_{j_{N+2}}|A|\phi_{j_{N+1}}\rangle \left(\prod_{k=1}^N \langle\phi_{j_{k+1}}|U(\Delta t_c)|\phi_{j_k}\rangle \right) \langle\phi_{j_1}|B|\phi_{j_0}\rangle \quad (3)$$

where $J = (j_0, \dots, j_{2N+2})$ is a set of indices that specify the elements of the basis set. We say that J defines a path within the basis $\{\phi_j\}$ and refer to the set of all possible paths as path space. Note that $\phi_{j_0} = \phi_{j_{2N+2}}$ due to evaluating the trace in Eq. (2). For a given Trotter step, we sum the matrix elements $\langle\phi_{j_{k+1}}|U(\Delta t_c)|\phi_{j_k}\rangle$ over the entire basis set.

Equation (3) can be expressed in the conventional classical PIMC form of an expectation value over a distribution W ¹³

$$C_{AB}(t) = \frac{F}{Z} \langle \langle\phi_{j_{N+2}}|A|\phi_{j_{N+1}}\rangle \langle\phi_{j_1}|B|\phi_{j_0}\rangle \text{sgn}\Theta(\phi) \rangle_W, \quad (4)$$

where $W = |\Theta|/F$,

$$\Theta(\phi) = \left(\prod_{k=N+2}^{2N+1} \langle\phi_{j_{k+1}}|U^\dagger(\Delta t_c)|\phi_{j_k}\rangle \right) \left(\prod_{k=1}^N \langle\phi_{j_{k+1}}|U(\Delta t_c)|\phi_{j_k}\rangle \right), \quad (5)$$

and

$$F = \sum_J |\Theta(\phi)|, \quad (6)$$

with $\phi = (\phi_{j_1}, \dots, \phi_{j_{2N+2}})$.

At a high level, $C_{AB}(t)$ is computed using classical PIMC by the following steps. To begin, the short-time TEO matrix elements are computed by diagonalizing $U(\Delta t_c)$ within the complete basis set. The computational cost of this step typically limits classical PIMC to low-dimensional systems. Next, a Monte Carlo importance sampling subroutine is repeated until a user-specified convergence criterion is met. The subroutine is as follows. First, a random path J is sampled from the distribution W . Second, $\Theta(\phi)$ is evaluated using the short-time TEO matrix elements. Next, $W(J)$ is computed and compared to the most recently accepted path using the standard Metropolis-Hastings acceptance/rejection criterion. If the path is accepted, then $\text{sgn}\Theta(\phi)$ and the matrix elements of A and B are computed and accumulated in an estimator,



ending the subroutine. Otherwise, the path is rejected and these quantities are not accumulated, which also concludes the subroutine.

We propose to implement hPIMC following the same steps given in the preceding paragraph to implement classical PIMC, with one modification. Instead of obtaining the short-time TEO matrix elements by constructing and diagonalizing the matrix representation of $U(\Delta t_c)$, we directly compute individual matrix elements as they are sampled. During each iteration of the Monte Carlo importance sampling subroutine, after J is sampled, each of the corresponding matrix elements are computed individually on the quantum computer, then $\Theta(\phi)$ is computed by multiplying the matrix elements on a classical computer. Quantities that depend on $\Theta(\phi)$ are also computed on the classical computer.

We turn now to analyze the performance of classical PIMC compared with hPIMC. To do so, we derive asymptotic expressions for the computational resources needed to compute the TCF to within a specified error. Throughout, we assume that the operators A and B can be computed on a classical computer with negligible cost.

In general, we cannot compute $C_{AB}(t)$ exactly using either classical PIMC or hPIMC. Instead, we compute an approximation $\tilde{C}_{AB}(t)$ to $C_{AB}(t)$ such that the error of the approximation $\varepsilon = |C_{AB}(t) - \tilde{C}_{AB}(t)|$ is bounded. For classical PIMC, ε consists of two terms,

$$\varepsilon = \varepsilon_{MC} + \varepsilon_C. \quad (7)$$

The first term arises from approximating $C_{AB}(t)$ using Monte Carlo and depends on the Monte Carlo sampling rate of convergence. In general, for M iterations of the Monte Carlo importance sampling subroutine, this error is given by¹³

$$\varepsilon_{MC} = O(1/\sqrt{M}). \quad (8)$$

Thus, to achieve an accuracy of ε_{MC} , we require $M = O(1/\varepsilon_{MC}^2)$ iterations. Although Eq. (8) holds in the asymptotic limit, the true rate of convergence is more complicated and not easy to express since for small systems with no significant dissipation, the well-established sign problem in quantum dynamics must be considered.⁵⁴ Additionally, this error will include a contribution from computing the partition function in Eq. (4).

The second term results from the fact that we generally cannot compute $U(t_c)$ exactly and must instead approximate it using $\tilde{U}(t_c)$. We show in Supplementary Information (SI) Section I that

$$\varepsilon_C = O(\|A\|_1 \|B\|_1 \|e^{-\beta H/2}\|_1 \varepsilon_U / Z), \quad (9)$$

provided that for a given ε_U ,

$$\|U(t_c) - \tilde{U}(t_c)\|_1 \leq \varepsilon_U \quad (10)$$

is satisfied. Here, $\|\cdot\|_1$ is trace norm. Typically, ε_C is related to the number of Trotter steps N , but the explicit form of this relationship depends on the choice of approximation used to implement \tilde{U} . For example, if $H = H_1 + H_2$ with non-commuting H_1 and H_2 , $U(t_c)$ can be approximated with a $2k$ th-order Suzuki product.⁵⁵

In this case, an accuracy of ε_C is guaranteed provided that

$$N = O\left(\left(\|A\|_1 \|B\|_1 \|e^{-\beta H/2}\|_1 \Omega^{2k+1} |t_c|^{2k+1} / Z \varepsilon_C\right)^{1/2k}\right), \quad (11)$$

where

$$\Omega = \max\{\|H_1\|, \|H_2\|\}, \quad (12)$$

and $\|\cdot\|$ is the spectral norm. Equation (11) comes from combining Eq. (9) with a previously introduced expression for ε_U .⁵⁵

The space and runtime complexities of approximately computing $C_{AB}(t)$ for a d -dimensional system and a basis set of size P using classical PIMC are

$$\text{PIMC space} = O(P^d \times P^d), \quad (13)$$

$$\text{PIMC runtime} = O(P^{3d} + MN). \quad (14)$$

The space complexity comes from constructing the matrix representation of $\tilde{U}(\Delta t_c)$, and the leading order term in the runtime complexity comes from the one-time cost of diagonalizing this matrix. The second term in the runtime complexity comes from performing M Monte Carlo iterations. For each iteration, two random strings each consisting of N matrix elements, $\{\langle \phi_{j_{k+1}} | \tilde{U}(\Delta t_c) | \phi_{j_k} \rangle\}$ and $\{\langle \phi_{j_{k+1}} | \tilde{U}^\dagger(\Delta t_c) | \phi_{j_k} \rangle\}$, referred to as the forward and backward paths, respectively, are sampled and the TCF estimators are computed. We assume that the cost to compute and store $\langle \phi_{j_{k+1}} | A | \phi_{j_k} \rangle$ and $\langle \phi_{j_{k+1}} | B | \phi_{j_k} \rangle$ is at most on the same order of $\tilde{U}(\Delta t_c)$.

For hPIMC, ε consists of three terms

$$\varepsilon = \varepsilon_C + \varepsilon_{MC} + \varepsilon_Q. \quad (15)$$

The first two terms are the same as Eqs. (8) and (9). The third term comes from approximately computing the matrix elements $\langle \phi_i | \tilde{U}(\Delta t_c) | \phi_j \rangle$ on a quantum computer. We propose to compute the short-time TEO matrix elements using a Hadamard test as this approach often yields shallow quantum circuits.⁵⁶ Given access to an oracle that prepares the states $|\phi_j\rangle$ on the quantum computer and an oracle that implements $\tilde{U}(\Delta t_c)$, the Hadamard test estimates matrix elements by repeatedly performing measurements on an ancilla qubit.⁵⁶ Because the Hadamard test is shot-noise limited, the error of a given matrix element is $\varepsilon_S = O(1/\sqrt{S})$, where S is the number of circuit repetitions.⁵⁷ Each path that contributes to the sum in Eq. (3) consists of a product of $2N$ matrix elements. Propagating the error from each matrix element results in a total error for each path ε_P given by⁵⁸

$$\varepsilon_P = O\left(\sqrt{N} \varepsilon_S \frac{|\Theta(\phi)|_{\max}}{|\langle \phi_i | \tilde{U}(\Delta t_c) | \phi_j \rangle|_{\min}}\right), \quad (16)$$

where

$$|\Theta(\phi)|_{\max} = \max_j |\Theta(\phi)|, \quad (17)$$

$$|\langle \phi_i | \tilde{U}(\Delta t_c) | \phi_j \rangle|_{\min} = \min_{ij} |\langle \phi_i | \tilde{U}(\Delta t_c) | \phi_j \rangle|. \quad (18)$$

Because there are M iterations, each contributing an error ε_P , we have that $\varepsilon_Q = O(M\varepsilon_P)$.



All three errors that contribute to ϵ are controllable. To decrease ϵ_C , we increase the number of Trotter steps N , to decrease ϵ_{MC} , we increase M and run more Monte Carlo iterations, and to decrease ϵ_Q , we increase S and run more repetitions of the Hadamard test circuit. Hence, the total error made by hPIMC can be systematically reduced by using more classical and quantum computational resources.

The space and runtime complexities of hPIMC are

$$\text{hPIMC space} = O(dn) + \text{Anc.}(\tilde{U}) \quad (19)$$

$$\text{hPIMC runtime} = O(MNSQ(\tilde{U})). \quad (20)$$

The first term of the space complexity is the result of using $n = \log(P)$ qubits to represent a given $|\phi_j\rangle$ on the quantum computer. Because we assume a d -dimensional system, we use d quantum registers each consisting of n qubits. The second term represents the number of ancilla qubits needed to implement the oracles used in the Hadamard test circuit. The runtime complexity comes from performing M Monte Carlo iterations, in which $2N$ matrix elements are computed per iteration. Each matrix element uses S repetitions of the Hadamard test circuit, and $Q(\tilde{U})$ represents the runtime cost to implement the oracles used in the Hadamard test circuit. We make the same assumption about the cost to compute matrix elements of A and B as for classical PIMC.

The efficiency of hPIMC depends on the efficiency of the oracles used by the Hadamard test. For many chemical systems, efficient implementations of $\tilde{U}(\Delta t_c)$ are known.^{16–23,30,36,47,48,59–62} In Sections 2.2 and 2.3, we provide an efficient algorithm to implement $\tilde{U}(\Delta t_c)$ for a 1D proton transfer reaction modeled by a symmetric double-well potential. Assuming that $Q(\tilde{U})$ and $\text{Anc.}(\tilde{U})$ scale polynomially, we see that hPIMC scales exponentially better than classical PIMC in terms of space and runtime complexity.

The performance of hPIMC can be improved by storing TEO matrix elements in a classical look-up table and querying the table each Monte Carlo iteration. If the table contains a queried matrix element, the value is pulled from the table instead of recomputing the matrix element on the quantum computer. Thus, the distribution between classical and quantum resources becomes increasingly more classical as the calculation progresses and the table fills up. This method is only successful when a small subset of matrix elements are needed to converge the Monte Carlo simulation. If all P^{2d} matrix elements are required, then an exponential amount of memory will be required to store the table. In the next section, we argue that this situation is avoided for condensed-phase systems at finite temperature when working in a position space representation.

2.1 Open Quantum Systems

Much of the success of classical PIMC has been in the context of dissipative systems at finite temperature where the rapid damping of quantum system oscillations by coupling to a large number of environment degrees of freedom significantly reduces the dynamical sign problem. This motivates us to adapt hPIMC for condensed-phase dynamics described by the CL Hamiltonian using a position space representation.

The CL Hamiltonian describes a primary system consisting of η modes in d physical dimensions bilinearly coupled to an isotropic bath approximated by f harmonic modes,⁴⁶

$$H = H'_s + H_b, \quad (21)$$

where H'_s is the system sub-Hamiltonian shifted along the adiabatic path,

$$H'_s = H_s - \sum_{j=1}^f \frac{\mathbf{F}_j(\mathbf{x}_1, \dots, \mathbf{x}_\eta) \cdot \mathbf{F}_j(\mathbf{x}_1, \dots, \mathbf{x}_\eta)}{2M_j\omega_j^2}, \quad (22)$$

and the system sub-Hamiltonian defined for η modes is

$$H_s = \sum_{i=1}^{\eta} \frac{\mathbf{p}_i \cdot \mathbf{p}_i}{2m_i} + V(\mathbf{x}_1, \dots, \mathbf{x}_\eta). \quad (23)$$

Here, the i th system mode has d -dimensional position and momentum vectors, \mathbf{x}_i and \mathbf{p}_i , respectively, and mass, m_i . V is a time-independent potential. The j th harmonic oscillator bath mode has mass M_j and frequency ω_j . The coupling term is given by

$$\mathbf{F}_j(\mathbf{x}_1, \dots, \mathbf{x}_\eta) = \sum_{i=1}^{\eta} \mathbf{f}_j(\mathbf{x}_i). \quad (24)$$

Here, we choose a linear, isotropic coupling in which $\mathbf{f}_j(\mathbf{x}_i) = c_j \mathbf{x}_i$.

The bath sub-Hamiltonian is modified to include the system-bath coupling term and is given by²

$$H_b = \sum_{j=1}^f \left[\frac{\mathbf{P}_j \cdot \mathbf{P}_j}{2M_j} + \frac{1}{2} M_j \omega_j^2 \left(\mathbf{Q}_j - \frac{\mathbf{F}_j(\mathbf{x}_1, \dots, \mathbf{x}_\eta)}{M_j \omega_j^2} \right)^2 \right]. \quad (25)$$

Here, the j th oscillator has d -dimensional position and momentum vectors, \mathbf{Q}_j and \mathbf{P}_j .

It has been shown that for this choice of sub-Hamiltonians the error made by approximating the TEO as

$$e^{-iH\Delta t_c} \approx e^{-iH_b\Delta t_c/2} e^{-iH'_s\Delta t_c} e^{-iH_b\Delta t_c/2} \quad (26)$$

goes as $O(\Lambda|\Delta t_c|^3)$, where Λ depends on the adiabaticity of the bath and goes to zero in the limit that the bath responds instantaneously to the system.^{1,63} Thus, the approximation in Eq. (26) becomes more accurate as Λ becomes smaller.

Another reason for expressing the CL Hamiltonian in this way is that in a position space representation, all terms involving contributions from the bath can be collected and integrated analytically, yielding the Feynman-Vernon influence functional.^{1–3,53,63} Assuming A and B are functions only of position, the TCF of Eq. (21) becomes

$$\tilde{C}_{AB}(t) = \frac{1}{Z} \int d\mathbf{x} P_b(\mathbf{x}_b) A(\mathbf{x}_N) P_f(\mathbf{x}_f) B(\mathbf{x}_0) I(\mathbf{x}), \quad (27)$$

where $I(\mathbf{x})$ is the influence functional and we define the forward path, $\mathbf{x}_f = (\mathbf{x}_0, \dots, \mathbf{x}_N)$, the backward path, $\mathbf{x}_b = (\mathbf{x}_N, \dots, \mathbf{x}_0)$, and $\mathbf{x} = (\mathbf{x}_f, \mathbf{x}_b)$. The explicit form the influence function is given by Makri.⁶⁴ We also define the product of forward and backward



path matrix elements

$$P_f(\mathbf{x}_f) = \prod_{k=0}^{N-1} \langle \mathbf{x}_{k+1} | e^{-iH'_s \Delta t_c} | \mathbf{x}_k \rangle \quad (28)$$

and

$$P_b(\mathbf{x}_b) = \prod_{k=N}^{2N-1} \langle \mathbf{x}_{k+1} | e^{iH'_s \Delta t_c} | \mathbf{x}_k \rangle, \quad (29)$$

respectively.

Integrating out the bath degrees of freedom via the influence functional results in large computational savings. Before integration, the integral in Eq. (27) is $2Nd(\eta + f)$ dimensional. Afterwards, the dimensionality is reduced to $2Nd\eta$. This reduction in dimensionality comes at the cost of evaluating the influence functional. For most spectral densities, $I(\mathbf{x})$ can be evaluated analytically or numerically on a classical computer at a cost of $O(dN^2)$, which tends to be negligible compared to the cost of computing matrix elements.^{65,66} Because $f \gg \eta$ is typically required to accurately model the bath using Eq. (21), this reduction in dimensionality generally outweighs the cost of computing $I(\mathbf{x})$. The influence functional also mediates dissipation from the system to the bath, damping oscillatory phases in the integrand and thereby alleviating the dynamical sign problem, which improves Monte Carlo convergence.

We can compute Eq. 27 using either classical PIMC or hPIMC by modifying Eq. 5:

$$\tilde{C}_{AB}(t) = \frac{F}{Z} \langle A(\mathbf{x}_N) B(\mathbf{x}_0) I(\mathbf{x}) \text{sgn} \Theta(\mathbf{x}) \rangle_W, \quad (30)$$

where the function Θ becomes

$$\Theta(\mathbf{x}) = P_f(\mathbf{x}_f) P_b(\mathbf{x}_b). \quad (31)$$

Algorithmically, the procedure to compute Eq. 30 using classical PIMC is the same as for Eq. 5 (described in the paragraph following Eq. 5), except that now the product $A(\mathbf{x}_N) B(\mathbf{x}_0) I(\mathbf{x})$ includes the influence functional that is computed and accumulated in the estimator whenever a path is accepted. To compute Eq. 30 with hPIMC, the same procedure as for classical PIMC is followed, except that the TEO matrix elements of $\Theta(\mathbf{x})$ are computed on a quantum computer.

For bound systems at finite temperature, we only need to sample matrix elements that connect nearby points rather than sample all possible displacements in position space, leading to enhanced sampling efficiency. This is because, at finite temperatures, the TEO matrix elements contain an imaginary time contribution that causes their magnitude, as well as the magnitudes of $P_f(\mathbf{x}_f)$ and $P_b(\mathbf{x}_b)$, to decay exponentially with increasing energy. Additionally, for many bound systems, localized states tend to be lower in energy, causing $P_f(\mathbf{x}_f)$ and $P_b(\mathbf{x}_b)$ to be relatively large, while deocalized states tend to be higher in energy, causing $P_f(\mathbf{x}_f)$ and $P_b(\mathbf{x}_b)$ to be relatively small.⁶³ As a result, only paths containing TEO matrix elements that connect points close in position space contribute significantly to Eq. (27).

We note that after the arXiv version of this paper was published,⁶⁷ a path integral quantum algorithm that leverages the influence functional formalism has been reported for spin-boson

systems that, unlike the present work, cannot be used in the context of Hamiltonians where the system includes continuous degrees of freedom.⁶⁸

2.2 Quantum Circuit

We can approximate the continuous integrals in Eq. (27) as discrete sums over a finite-width grid of evenly spaced grid points. In one dimension, let L be the total length of the grid and D the number of grid points. Any position on the grid is given by $x_j = -\frac{L}{2} + j\Delta x$, where $\Delta x = L/D$, $j \in G$, and G is the set of integers $G = [0, D-1]$. We represent the grid on a quantum computer by mapping j to the computational basis:

$$|j\rangle = |b_{j_{n-1}}\rangle \otimes |b_{j_{n-2}}\rangle \otimes \dots \otimes |b_{j_0}\rangle, \quad (32)$$

where b_{j_i} are the binary expansion coefficients of j and $n = \lceil \log D \rceil$. Here, $|j\rangle$ represents a single quantum register. For ηd dimensions, each position coordinate is mapped to its own quantum register and the overall state is the direct product of all ηd registers. The constant offset, $-L/2$, and grid spacing, Δx , are absorbed into the implementation of the potential energy.⁶⁹ Because we are mapping positions on a grid directly to the computational basis, each register can be efficiently initialized with a single layer of Pauli-X gates.

We compute the short-time TEO matrix elements $\langle \mathbf{x}_{k+1} | U(\Delta t_c) | \mathbf{x}_k \rangle$, where now $U(\Delta t_c) = e^{-iH'_s \Delta t_c}$, on a quantum computer using a Hadamard test.⁵⁶ A circuit diagram that implements the Hadamard test is given in Fig. 1. Within the Hadamard test, we split $U(\Delta t_c)$ into real- and imaginary-time components,

$$U(\Delta t_c) = e^{-iH'_s \Delta t} e^{-\frac{\Delta \beta}{2} H'_s}, \quad (33)$$

where $\Delta t = t/N$ and $\Delta \beta = \beta/N$. The imaginary-time component is computed using the PITE method, and the real-time component is computed using Eq. (35).

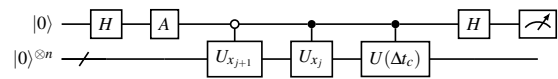


Fig. 1 Hadamard test to compute TEO matrix elements. Here, H is a Hadamard gate, U_{x_j} is a unitary that prepares the state $|x_j\rangle$, and $U(\Delta t_c)$ performs time evolution under H'_s for an amount Δt_c . To measure the real or imaginary component of the matrix element, A is either set to identity or S^\dagger , respectively, where S is the phase gate.

For an initial state $|\psi\rangle$ and a single ancilla qubit, the PITE algorithm takes as input a circuit that executes real-time evolution, $e^{-iH'_s \tau}$, and approximates evolution in imaginary-time by the unitary U_{PITE} such that^{47,48}

$$U_{PITE} |\psi\rangle |0\rangle \rightarrow m_0 (1 - H'_s \tau) |\psi\rangle \otimes |0\rangle + \left(\gamma + \frac{m_0^2}{\gamma} H'_s \tau \right) |\psi\rangle \otimes |1\rangle + O(\tau^2), \quad (34)$$

where $\tau = s_1 \Delta \beta / 2$, $\gamma = \sqrt{1 - m_0^2}$, $s_1 = m_0 / \gamma$, and m_0 is a real parameter. The PITE quantum circuit diagram is given in Fig. 2.



When the ancillary qubit is measured in the state $|0\rangle$, $|\psi\rangle$ is evolved in imaginary-time by a first-order approximation to $e^{-\frac{\Delta\beta}{2}H'_s}$.

We note that the probability of successfully measuring $|0\rangle$ increases for smaller values of τ . PITE was originally designed for finding the ground state of an input Hamiltonian where successive applications of the circuit are needed to reach larger τ , each of which lowers the overall probability of success. However, in hPIMC, τ is inherently small to ensure that Eq. (26) is a good approximation. Thus, we will only require a single application of PITE per TEO matrix element, and the additional overhead from calling PITE until a successful outcome is measured is minimal.

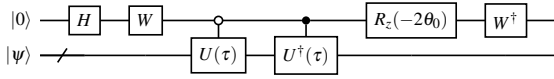


Fig. 2 PITE quantum circuit diagram for evolving an input state $|\psi\rangle$ in imaginary time by $\Delta\beta/2$. The gate W implements $W = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & -i \\ \sqrt{2} & i \end{pmatrix}$, $R_z(2\theta_0)$ is a rotation about the z -axis of the Bloch sphere by an angle $2\theta_0$, where $\theta_0 = \kappa \arccos[(m_0 + \gamma)/\sqrt{2}]$. $\kappa = \text{sgn}(m_0 - 1/\sqrt{2})$ and m_0 is an adjustable real parameter with conditions $0 < m_0 < 1$ and $m_0 \neq 1/\sqrt{2}$. $U(\tau)$ performs evolution under H'_s for time $\tau = s_1 \frac{\Delta\beta}{2}$, where $s_1 = m_0/\gamma$.

For evolution in real-time, we approximate the TEO using a second-order Trotter splitting,⁸

$$e^{-iH'_s\Delta t} \approx e^{-iV\Delta t/2} e^{-iT\Delta t} e^{-iV\Delta t/2}. \quad (35)$$

In the position basis, V is diagonal and $e^{-iV\Delta t/2}$ can be constructed efficiently if there exists a classical algorithm that efficiently implements $e^{-iV\Delta t/2}$.⁷⁰ For example, quadratic potentials with linear coupling can be constructed with $O(\eta dn^2)$ one- and two-qubit gates⁶⁹.

2.3 DVR Approximation

We propose to evaluate time evolution under the kinetic energy operator with a discrete variable representation (DVR). The kinetic energy operator, T , can be expressed using a sinc basis set as⁴⁹

$$T^{DVR} = \sum_{\alpha=1}^{\eta} T_{\alpha}^{DVR}, \quad (36)$$

where

$$T_{\alpha}^{DVR} = I_1^{\otimes d} \otimes \dots \otimes I_{\alpha-1}^{\otimes d} \otimes DVR_{\alpha}^{\otimes d} \otimes I_{\alpha+1}^{\otimes d} \otimes \dots \otimes I_{\eta}^{\otimes d}. \quad (37)$$

Here, $I^{\otimes d}$ is a tensor product of d identity matrices and the DVR of the α th system mode in one dimension is defined by the matrix elements⁵¹

$$DVR_{\alpha,ij} = \frac{(-1)^{i-j}}{2m_{\alpha}(\Delta x)^2} \begin{cases} \frac{\pi^2}{3}, & i = j \\ \frac{2}{(i-j)^2}, & i \neq j \end{cases} \quad (38)$$

where m_{α} is the mass of the α th system mode. In the limit where Δx goes to zero, this approximation is exact.

Ordinarily, one diagonalizes the DVR Hamiltonian, $H^{DVR} = T^{DVR} + V$, and performs dynamics in the basis of DVR eigenvectors. Instead, we propose to approximate time evolution under

the kinetic energy operator in the position basis by directly applying $e^{-iT^{DVR}\Delta t}$ to a state $|\psi\rangle$. This approximation is justified because, in the limit that Δx goes to zero, the sinc basis becomes the position basis⁵⁰. This allows us to avoid having to diagonalize H^{DVR} . The resulting unitary is given by

$$e^{-iT\Delta t} \approx e^{-iT^{DVR}\Delta t} = \prod_{\alpha=1}^{\eta} e^{-iT_{\alpha}^{DVR}\Delta t}. \quad (39)$$

We present results in the next section that validate the use of this approximation.

In general, we cannot efficiently implement $e^{-iDVR_{\alpha}\Delta t}$ on a quantum computer because DVR_{α} is dense. At the same time, we can see from Eq. (38) that the matrix elements decrease in magnitude moving away from the main diagonal, and should eventually become small enough to safely neglect. In SI Section II, we prove that this is indeed the case and find that for a given error tolerance δ , DVR_{α} can be made sparse by neglecting all diagonals, v , where $v \geq \ell$, and ℓ is given by

$$\ell = \left\lceil \frac{(2K)^{2/3}}{\delta^{2/3}} \right\rceil, \quad (40)$$

with $K = 1/m_{\alpha}(\Delta x)^2$. Here, $v = 1$ is the main diagonal, $v = 2$ are the adjacent upper and lower diagonals, and so on. Formally, this expression results from a lower bound on error. However, we show in SI Section II that this bound is tight and therefore use it to approximate the true error.

We can then simulate time evolution under DVR_{α} using a first-order Trotter splitting about the diagonals of DVR_{α} :

$$e^{-iDVR_{\alpha}\Delta t} \approx \prod_{v=1}^{\ell} e^{-i\text{diag}(DVR_{\alpha},v)\Delta t}, \quad (41)$$

where $\text{diag}(\cdot, v)$ returns the input matrix with all elements neglected except those on the v th diagonals. Each matrix $\text{diag}(DVR_{\alpha}, v)$ is 2-sparse and in SI Section III we prove that each of these 2-sparse matrices can be decomposed into a sum of at most two 1-sparse Hermitian matrices. Our method yields a factor of 12 improvement compared to the results of⁵⁵ for a 2-sparse matrix. Using a first-order Trotter splitting about each 1-sparse matrix yields

$$e^{-iDVR_{\alpha}\Delta t} \approx \prod_{v=1}^{\ell} \prod_{\sigma=1}^2 e^{-i\text{diag}(DVR_{\alpha},v)_{\sigma}\Delta t}, \quad (42)$$

where σ indexes the 1-sparse matrices resulting from the decomposition.

In SI Section V, we construct a quantum circuit that performs time evolution under Eq. (42) using the sparse matrix simulation approach detailed in Aharonov and Ta-Shma.⁷⁰ The main idea of this approach is to use the fact that 1-sparse Hermitian matrices can always be represented in block-diagonal form where each block acts on a two-dimensional subspace of the overall Hilbert space. Time evolution can be performed by constructing an oracle that identifies each subspace, then applying the appropriate two-dimensional time evolution operator for a given subspace. We show how to construct the necessary oracles and two-dimensional



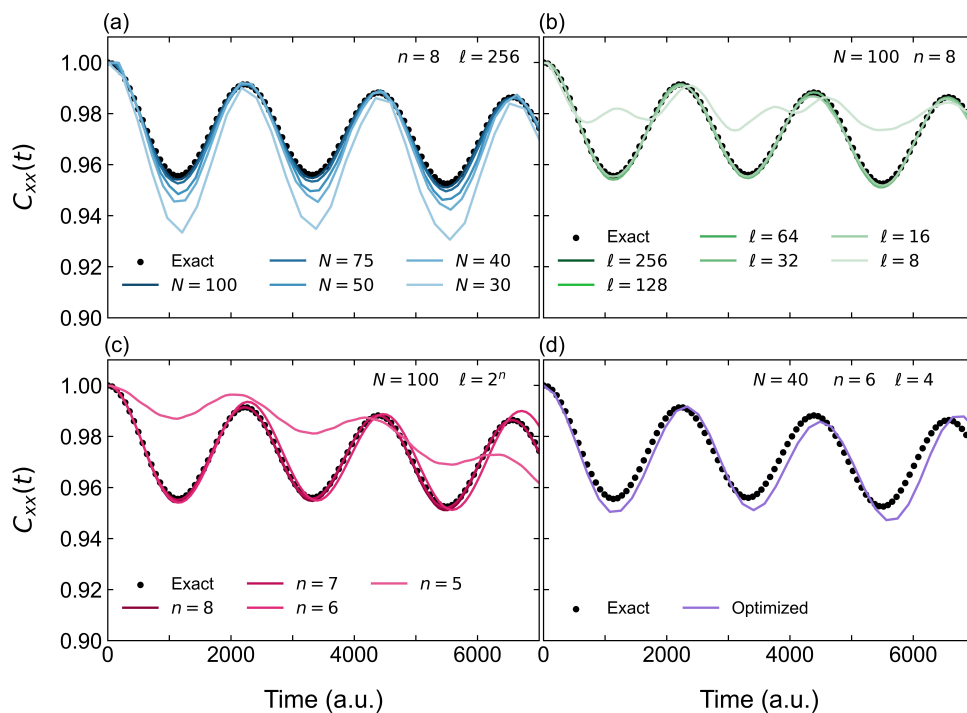


Fig. 3 TCF of symmetric double-well potential. We evaluate real-time TEO matrix elements using a second-order Trotter splitting about T and V , and QFT for evolution under the kinetic energy operator. For imaginary-time evolution, we use a first-order Trotter splitting about T and V , and Eq. (42) for evolution under the kinetic energy operator. For each plot, $m = 1836$ a.u., $1/\beta = 350$ K, $\omega_b = 500$ cm $^{-1}$, $V_0 = 1500$ cm $^{-1}$, and $L = 30$ a.u. In (a) we vary the number of Trotter steps N , (b) we approximate the DVR by neglecting all diagonals $v \geq \ell$, and in (c) we vary the number of grid points, $D = 2^n$. In (d), we plot the TCF using optimized parameters so that the resource requirements are minimized while maintaining reasonable error. Each plot is normalized by the magnitude of its maximum value.

time evolution operators for Eq. (42) in SI Section V, and demonstrate that simulating a single $\text{diag}(DVR_{\alpha}, v)_{\sigma}$ for a 1D system can be performed with $O(n)$ ancilla qubits, $O(n \log(n))$ CNOT gates, and a circuit depth of $O(n)$.

3 Results

Our proposed method to simulate the system TEO involves several approximations: imaginary-time evolution is approximated by PITE, real-time evolution is approximated using a second-order Trotter splitting, the kinetic energy operator is approximated using a DVR, the DVR is approximated using a truncated number of diagonals, and the Hadamard test is approximated using a finite number of circuit repetitions. Although the error from some of these approximations can be analyzed individually, rigorously studying the cumulative error is not straightforward.

Instead, we investigate the resulting error numerically by computing the position-position TCF of a 1D model proton transfer reaction using the approach outlined above to compute TEO matrix elements and compare the approximate results with exact results obtained by diagonalizing the DVR matrix. We model the donor and acceptor sites of the reaction using a standard symmetric double-well potential:

$$V(x) = -\frac{1}{2}m\omega_b^2x^2 + \frac{m^2\omega_b^4}{16V_0}x^4. \quad (43)$$

Real-time evolution is approximated with a second-order Trotter

splitting, as in Eq. (35), and imaginary-time evolution is approximated with a first-order Trotter splitting to replicate PITE.

Our focus is to study how the error introduced by approximating TEO matrix elements using the algorithm proposed in the previous section propagates through the calculation of the overall TCF. To isolate the error from approximating the TEO matrix elements, we use quadrature on a classical computer to compute all TCF integrals in Eq. (1). Computing the TCF this way eliminates the error that would be introduced from Monte Carlo sampling or using a finite number of circuit repetitions. The results presented here are equivalent to what one would obtain from a fully converged Monte Carlo and a fully converged Hadamard test using noiseless hardware.

Initially, we used Eq. (42) to approximate evolution under the kinetic energy operator for both real-time and imaginary-time. However, we found that $N > 100$ was needed to achieve accurate results, which is relatively large. Instead, we found that using Eq. (42) for imaginary-time only, and using a standard Quantum Fourier Transform (QFT) approach to diagonalize the kinetic energy operator in real-time achieved the same accuracy with a smaller N . Because a smaller N results in a shorter hPIMC runtime, we used this mixed approach for evolution under the kinetic energy operator to obtain the results presented in Fig. 3. Interestingly, we also found that using QFT for both real- and imaginary-time evolution required an even larger N than using Eq. (42) for both, and that the results do not systematically improve with in-



creasing N as one would expect.

In Fig. 3, panel (a), we see that for sufficiently large N the approximate results match the exact results. As N decreases, the error from Trotter splitting increases, and we see that the amplitude of the TCF begins to deteriorate, although the frequency of the oscillations remains accurate. In panel (b), we approximate T by neglecting all diagonals v , where $v \geq \ell$, from DVR_α . We see that the approximate results lie directly on top of the exact results up to $\ell = 16$. In other words, 87.5% of the diagonals in DVR_α can be neglected without impacting the results of the calculation. This is important because each additionally neglected diagonal results in a shallower circuit.

In panel (c), we compare approximate and exact results for different numbers of grid points, $D = 2^n$. For $n = 8$ and 7 , the difference is indistinguishable. At $n = 6$, the approximate results slightly deviate, and for $n = 5$ the grid spacing becomes too large and all features of the TCF are lost. In panel (d), we pick a set of parameters that is optimized to use minimal computational resources: $N = 40$, $n = 6$, and $\ell = 4$. We see that our method of computing the TEO matrix elements is robust to the cumulative error resulting from varying multiple parameters simultaneously. These results are encouraging and demonstrate that complex-time evolution can be well approximated by directly exponentiating DVR_α to simulate imaginary-time evolution and simulating real-time evolution using a QFT approach. Further, we see that the Trotter error inherent in Eq. (42) and introduced through PITE is manageable for $N = 40$, and only 6 qubits and $l = 4$ diagonals are needed to achieve results that recover the amplitude and frequency of the position TCF of a proton transfer model.

In SI Section VI, we estimate the resources needed to simulate time evolution under Eq. (43) on a quantum computer. Our approach uses the same method described by Beneti and Strini⁶⁹ for simulating quadratic potentials. We find that simulating Eq. (43) for a 1D system can be performed with a depth and CNOT gate count of $O(n^4)$, and a constant number of ancilla qubits. Because time evolution under the kinetic energy operator using the DVR approximation and conventional QFT approach can be performed in $O(\ell n \log n)$ and $O(n^2)$, respectively, we approximate the cost of the hPIMC circuit as the cost of simulating the double-well potential. Using the constant factor scalings that we derive in SI Section VI, we estimate that a 1D proton transfer reaction using $n = 6$ qubits can be simulated with a depth and CNOT count of 2.96×10^5 , and 24 ancilla qubits. As a comparison, accounting for the DVR circuit only increases the CNOT count by 4.2×10^4 , and the depth by 3.0×10^4 for $\ell = 16$.

4 Discussion

Our results show that for a 1D proton transfer reaction, computing the TEO matrix elements to be used in hPIMC requires circuits with depth and CNOT gate count of 2.96×10^5 . For context, this is comparable to other recently proposed NISQ algorithms which estimate $O(10^7)$ CNOT gates to simulate a spin-boson system using Trotterization.³⁹ For currently available quantum computers with two-qubit error rates on the order of 10^{-3} , a circuit of depth $O(10^5)$ is likely infeasible without significant error accumulation.⁷¹ For reference, an error rate of 10^{-3} means that we

expect on average one error to occur for every 10^3 two-qubit operations.⁷¹ Thus, we can expect on average $O(10^2)$ errors to occur for a circuit consisting of $O(10^5)$ CNOT gates. However, there are many strategies to further reduce the costs of the present algorithm that we will explore moving forward.

The dominant cost to compute the TEO for a proton transfer reaction comes from our method of computing a double-well potential, which scales as $O(n^4)$. An alternative approach is to first compute x^2 into a new register, then evolve under the quadratic term via phase kickback.^{19,72} Next, the x^4 term can be computed into an additional register using quantum arithmetic to square x^2 , and then evolving under the quartic term via phase kickback again. The ensuing algorithm would be $O(n^2)$. The circuit we propose in SI Section VI does not explicitly include coupling to the bath degrees of freedom. However, adding this effect does not increase the cost of the circuit. For linear, isotropic couplings, we see from Eq. (22) that incorporating coupling to the bath adds a term that is quadratic in the system position coordinates. This term can be grouped with the other quadratic terms and simulated in one go without additional resources.

Our circuit for time evolution under the DVR kinetic energy approximation is more encouraging with $O(\ln \log n)$ and $O(\ln)$ scaling in terms of CNOT gates and depth, respectively. We have demonstrated in Fig. 3 and SI Section II that l can be made quite small while retaining accurate results. We point out that the numerical study in Section 3 was for a proton. However, from Eq. (40), we see that for a given δ , ℓ can be made smaller for particles with larger mass. Thus, we expect our approach to perform better for heavier particles. Despite the relatively low asymptotic scaling, the associated constant factors found in SI Section V are quite high. This is because many of the primitive circuits used to construct the circuit are not optimal. Improvements can be made by replacing them with the currently best known methods. Additionally, optimizing the circuit over all ℓ of the $\text{diag}(DVR_\alpha, v)$ operators simultaneously rather than on a case-by-case basis could make it easier to detect and remove redundant operations.

In Section 3, we report that the best results were obtained by simulating evolution under the kinetic energy operator using a QFT approach for real time and DVR for imaginary time. We suggest using this pairing when implementing hPIMC as it performed better than using QFT or DVR for both real and imaginary time evolution, or DVR for real time and QFT for imaginary time. Further investigation is needed to understand why this particular combination gives the best results.

hPIMC experiences a steep penalty from the error ϵ_S of computing matrix elements using a Hadamard test. This is because ϵ_S is compounded both from multiplying TEO matrix elements together, then adding these products together to compute $\tilde{C}_{AB}(t)$. Ideally, we would like ϵ_S to be as small as possible, but this comes at the expense of more circuit repetitions. We note that several error mitigation schemes have been designed to help offset the effect of noise and decoherence that contributes to the error of performing NISQ calculations.^{29,73–75} Applying these schemes to hPIMC will help to keep ϵ_S as small as possible. Although hPIMC was originally conceived for NISQ computing, it is worth considering its application for near-term fault tolerant quantum com-



puting. In this framework, implementing circuits of $O(10^5)$ CNOT gates is more feasible. In addition, coherent techniques such as amplitude estimation, which scale more favorably with the error of the computation, can be explored.⁷⁶ This would significantly reduce the overall error from performing hPIMC.

We do not estimate the number of shots needed to converge the Hadamard test to a given accuracy in this work. Although hPIMC performs asymptotically better than classical PIMC, this estimate is needed to conduct a more rigorous comparison at the level of constant factors. Taking this into consideration, it is likely that the turnover point where hPIMC begins to outperform classical PIMC occurs for higher-dimensional systems. We leave this analysis to future work.

An important distinction between hPIMC and VQA-based hybrid approaches is that hPIMC is non-variational and therefore avoids trainability issues such as barren plateaus.⁴² Additionally, the dominant sources of systematic error in hPIMC, Trotterization and the approximation of the DVR operator, are systematically controllable. Decreasing Δt_c reduces the Trotter error, and including more diagonals in Eq. (41) reduces the error associated with the DVR approximation. For VQAs, there is typically no controllable parameter that guarantees convergence to the exact result because the error depends on the expressiveness of the chosen ansatz.

The CL Hamiltonian described in this paper involves a single electronic state potential energy surface. However, hPIMC can easily be extended to nonadiabatic systems that employ a diabatic representation for multiple potential energy surfaces.³⁰ We expect hPIMC to provide a useful tool for studies of condensed-phase adiabatic systems as well as nonadiabatic systems in which the electronic state transitions are strongly coupled to multiple vibrational degrees of freedom. Extending such systems to multiple dimensions challenges current classical methodology.

5 Conclusion

In this initial paper, we propose and develop the theory of hPIMC for computing real-time TCFs. hPIMC is based on Monte Carlo simulation and does not use variational methods to perform simulations. We also propose a new method to perform time evolution under the kinetic energy operator using a DVR approximation. We provide an efficient algorithm for our DVR approach with $O(\ln \log n)$ scaling in CNOT count and $O(\ln)$ in circuit depth. We perform classical numerical studies on a 1D proton transfer reaction and show that accurate results can be obtained using a DVR approximation. We estimate that the space overhead needed to simulate the DVR circuit is low, using only 6 state qubits and 24 ancilla qubits.

Moving forward, additional algorithm optimizations and tests are needed to enable hPIMC for use on currently available NISQ machines. We identify several opportunities to optimize our circuit and reduce the circuit depth and CNOT gate count from $O(10^5)$. In particular, error mitigation schemes, optimal circuit primitives, and an $O(n^2)$ algorithm for simulating the double-well potential have all yet to be explored. We are optimistic that these improvements will allow for the study of higher-dimensional condensed-phase systems via hPIMC that are currently inaccessible with classical algorithms.

sible with classical algorithms.

Author contributions

EE and NA conceptualized the main algorithm and wrote and edited the manuscript. EE wrote the software and prepared figures for visualization. NA supervised the project and secured funding.

Conflicts of interest

There are no conflicts to declare.

Data availability

The code used to simulate a proton transfer reaction, create Fig. 3, and analyze the DVR error in SI Fig. 1 can be found at https://github.com/AnanthGroup/hPIMC_figures with DOI 10.5281/zenodo.16900433.

Acknowledgements

The authors acknowledge funding for this work through NSF EA-GER: QCA-QSA grant number CHE-2038005. The authors thank Peter McMahon for insightful discussion and guidance.

References

- 1 M. Topaler and N. Makri, *J. Chem. Phys.*, 1992, **97**, 9001.
- 2 M. Topaler and N. Makri, *Chem. Phys. Lett.*, 1993, **210**, 285.
- 3 M. Topaler and N. Makri, *Chem. Phys. Lett.*, 1993, **210**, 448.
- 4 F. Matzkies and U. Manthe, *J. Chem. Phys.*, 1997, **106**, 2646.
- 5 R. P. Feynman, *Rev. Mod. Phys.*, 1948, **20**, 367.
- 6 R. P. Feynman and A. R. Hibbs, *Quantum Mechanics and Path Integrals*, Dover Publications, Emended edition edn, 2010.
- 7 R. P. Feynman, *Int. J. Theor. Phys.*, 1982, **21**, 467.
- 8 H. F. Trotter, *Proc. Amer. Math. Soc.*, 1959, **10**, 545.
- 9 H. De Raedt and B. De Raedt, *Phys. Rev. A*, 1983, **28**, 3575.
- 10 B. J. Berne and D. Thirumalai, *Ann. Rev. Phys. Chem.*, 1986, **37**, 401.
- 11 V. Zamalin and G. Norman, *USSR Comp. Math. and Math. Phys.*, 1973, **13**, 169.
- 12 B. Bernu and D. M. Ceperley, *Quantum Simulations of Complex Many-Body Systems: From Theory to Algorithms*, John von Neumann Institute for Computing, 2002, vol. 10, p. 51.
- 13 M. E. Tuckerman, *Statistical Mechanics: Theory and Molecular Simulation*, Oxford University Press, 2nd edn, 2023.
- 14 N. Makri, *Annu. Rev. Phys. Chem.*, 1999, **50**, 167.
- 15 S. Kundu and N. Makri, *Annu. Rev. Phys. Chem.*, 2022, **73**, 349.
- 16 N. C. Rubin, D. W. Berry, A. Kononov, F. D. Malone, T. Khattar, A. White, J. Lee, H. Neven, R. Babbush and A. D. Baczewski, *Proc. Natl. Acad. Sci. U.S.A.*, 2024, **121**, e2317772121.
- 17 R. Babbush, W. J. Huggins, D. W. Berry, S. F. Ung, A. Zhao, D. R. Reichman, H. Neven, A. D. Baczewski and J. Lee, *Nat. Commun.*, 2023, **14**, 4058.
- 18 Y. Su, D. W. Berry, N. Wiebe, N. Rubin and R. Babbush, *PRX Quantum*, 2021, **2**, 040332.



- 19 I. Kassal, S. P. Jordan, P. J. Love, M. Mohseni and A. Aspuru-Guzik, *Proc. Natl. Acad. Sci. U.S.A.*, 2008, **105**, 18681.
- 20 M. E. S. Morales, P. C. S. Costa, G. Pantaleoni, D. K. Burgarth, Y. R. Sanders and D. W. Berry, *Quantum Inf. Comput.*, 2025, **25**, 1.
- 21 G. H. Low and I. L. Chuang, *Quantum*, 2019, **3**, 163.
- 22 E. Campbell, *Phys. Rev. Lett.*, 2019, **123**, 070503.
- 23 D. Motlagh, R. A. Lang, P. Jain, J. A. Campos-Gonzalez-Angulo, W. Maxwell, T. Zeng, A. Aspuru-Guzik and J. Miguel Arrazola, *Quantum Sci. Technol.*, 2025, **10**, 045048.
- 24 J. Preskill, *Quantum*, 2018, **2**, 79.
- 25 K. Bharti, A. Cervera-Lierta, T. H. Kyaw, T. Haug, S. Alperin-Lea, A. Anand, M. Degroote, H. Heimonen, J. S. Kottmann, T. Menke, W.-K. Mok, S. Sim, L.-C. Kwek and A. Aspuru-Guzik, *Rev. Mod. Phys.*, 2022, **94**, 015004.
- 26 F. D. Malone, R. M. Parrish, A. R. Welden, T. Fox, M. Degroote, E. Kyoseva, N. Moll, R. Santagati and M. Streif, *Chem. Sci.*, 2022, **13**, 3094.
- 27 X. Xing, A. Gomez Cadavid, A. F. Izmaylov and T. V. Tscherbul, *J. Phys. Chem. Lett.*, 2023, **14**, 6224.
- 28 Y. Wang, E. Mulvihill, Z. Hu, N. Lyu, S. Shivpuje, Y. Liu, M. B. Soley, E. Geva, V. S. Batista and S. Kais, *J. Chem. Theory Comput.*, 2023, **19**, 4851.
- 29 S. Endo, Z. Cai, S. C. Benjamin and X. Yuan, *J. Phys. Soc. Jpn.*, 2021, **90**, 032001.
- 30 P. J. Ollitrault, G. Mazzola and I. Tavernelli, *Phys. Rev. Lett.*, 2020, **125**, 260511.
- 31 P. J. Ollitrault, A. Miessen and I. Tavernelli, *Acc. Chem. Res.*, 2021, **54**, 4229.
- 32 P. J. Ollitrault, S. Jandura, A. Miessen, I. Burghardt, R. Martinazzo, F. Tacchino and I. Tavernelli, *Quantum*, 2023, **7**, 1139.
- 33 J. R. McClean, J. Romero, R. Babbush and A. Aspuru-Guzik, *New J. Phys.*, 2016, **18**, 023023.
- 34 D. Wang, O. Higgott and S. Brierley, *Phys. Rev. Lett.*, 2019, **122**, 140504.
- 35 A. Peruzzo, J. McClean, P. Shadbolt, M.-H. Yung, X.-Q. Zhou, P. J. Love, A. Aspuru-Guzik and J. L. O'Brien, *Nat. Commun.*, 2014, **5**, 4213.
- 36 C.-K. Lee, C.-Y. Hsieh, S. Zhang and L. Shi, *J. Chem. Theory Comput.*, 2022, **18**, 2105.
- 37 Y.-X. Yao, N. Gomes, F. Zhang, C.-Z. Wang, K.-M. Ho, T. Iadecola and P. P. Orth, *PRX Quantum*, 2021, **2**, 030307.
- 38 D. Layden, G. Mazzola, R. V. Mishmash, M. Motta, P. Wocjan, J.-S. Kim and S. Sheldon, *Nature*, 2023, **619**, 282.
- 39 A. Miessen, P. J. Ollitrault and I. Tavernelli, *Phys. Rev. Research*, 2021, **3**, 043212.
- 40 M.-H. Yung, J. Casanova, A. Mezzacapo, J. McClean, L. Lamata, A. Aspuru-Guzik and E. Solano, *Sci. Rep.*, 2014, **4**, 3589.
- 41 F. Benfenati, G. Mazzola, C. Capecci, P. K. Barkoutsos, P. J. Ollitrault, I. Tavernelli and L. Guidoni, *J. Chem. Theory Comput.*, 2021, **17**, 3946.
- 42 S. Wang, E. Fontana, M. Cerezo, K. Sharma, A. Sone, L. Cincio and P. J. Coles, *Nat. Commun.*, 2021, **12**, 6961.
- 43 S. H. Sack, R. A. Medina, A. A. Michailidis, R. Kueng and M. Serbyn, *PRX Quantum*, 2022, **3**, 020365.
- 44 M. Cerezo, A. Arrasmith, R. Babbush, S. C. Benjamin, S. Endo, K. Fujii, J. R. McClean, K. Mitarai, X. Yuan, L. Cincio and P. J. Coles, *Nat. Rev. Phys.*, 2021, **3**, 625.
- 45 A. Tikku and I. H. Kim, *Circuit Depth Versus Energy in Topologically Ordered Systems*, arXiv, 2022, preprint, arXiv:2210.06796.
- 46 A. O. Caldeira and A. J. Leggett, *Phys. Rev. Lett.*, 1981, **46**, 211.
- 47 T. Kosugi, Y. Nishiyama, H. Nishi and Y.-i. Matsushita, *Phys. Rev. Res.*, 2022, **4**, 033121.
- 48 H. Nishi, K. Hamada, Y. Nishiyama, T. Kosugi and Y.-i. Matsushita, *Phys. Rev. Res.*, 2023, **5**, 043048.
- 49 J. C. Light and T. Carrington Jr., in *Discrete-Variable Representations and Their Utilization*, John Wiley and Sons, Ltd, 2000, p. 263.
- 50 R. G. Littlejohn, M. Cargo, T. Carrington, K. A. Mitchell and B. Poirier, *J. Chem. Phys.*, 2002, **116**, 8691.
- 51 D. T. Colbert and W. H. Miller, *J. Chem. Phys.*, 1992, **96**, 1982.
- 52 W. H. Miller, S. D. Schwartz and J. W. Tromp, *J. Chem. Phys.*, 1983, **79**, 4889.
- 53 M. Topaler and N. Makri, *J. Chem. Phys.*, 1994, **101**, 7500.
- 54 S. Caratzoulas and P. Pechukas, *J. Chem. Phys.*, 1996, **104**, 6265.
- 55 D. W. Berry, G. Ahokas, R. Cleve and B. C. Sanders, *Commun. Math. Phys.*, 2007, **270**, 359.
- 56 D. Aharonov, V. Jones and Z. Landau, *Algorithmica*, 2009, **55**, 395.
- 57 S. Polla, G.-L. R. Anselmetti and T. E. O'Brien, *Phys. Rev. A*, 2023, **108**, 012403.
- 58 J. R. Taylor, *An Introduction to Error Analysis: The Study of Uncertainties in Physical Measurements*, University Science Books, Sausalito, CA, 2nd edn, 1997.
- 59 Y. Cao, J. Romero, J. P. Olson, M. Degroote, P. D. Johnson, M. Kieferová, I. D. Kivlichan, T. Menke, B. Peropadre, N. P. D. Sawaya, S. Sim, L. Veis and A. Aspuru-Guzik, *Chem. Rev.*, 2019, **119**, 10856.
- 60 R. Babbush, N. Wiebe, J. McClean, J. McClain, H. Neven and G. K.-L. Chan, *Phys. Rev. X*, 2018, **8**, 011044.
- 61 R. Babbush, D. W. Berry, J. R. McClean and H. Neven, *npj Quantum Inf.*, 2019, **5**, 92.
- 62 A. M. Childs, J. Leng, T. Li, J.-P. Liu and C. Zhang, *Quantum*, 2022, **6**, 860.
- 63 N. Makri, *J. Phys. Chem. A*, 1998, **102**, 4414.
- 64 N. Makri, *J. Math. Phys.*, 1995, **36**, 2430.
- 65 K. Allinger and M. A. Ratner, *Phys. Rev. A*, 1989, **39**, 864.
- 66 T. C. Allen, P. L. Walters and N. Makri, *J. Chem. Theory Comput.*, 2016, **12**, 4169.
- 67 E. C. Eklund and N. Ananth, *Hybrid Quantum Algorithm for Simulating Real-Time Thermal Correlation Functions*, arXiv, 2025, preprint, arXiv:2405.19599v2.
- 68 A. Seneviratne, P. L. Walters and F. Wang, *J. Chem. Phys.*,



- 2025, **162**, 194101.
- 69 G. Beneti and G. Strini, *Am. J. Phys.*, 2008, **76**, 657.
- 70 D. Aharonov and A. Ta-Shma, Proceedings of the Thirty-Fifth Annual ACM Symposium on Theory of Computing, 2003, p. 20.
- 71 D. C. McKay, I. Hincks, E. J. Pritchett, M. Carroll, L. C. G. Govia and S. T. Merkel, *Benchmarking Quantum Processor Performance at Scale*, arXiv, 2023, preprint, arXiv:2311.05933.
- 72 R. Cleve, A. Ekert, C. Macchiavello and M. Mosca, *Proc. R. Soc. Lond. A*, 1998, **454**, 339.
- 73 K. Temme, S. Bravyi and J. M. Gambetta, *Phys. Rev. Lett.*, 2017, **119**, 180509.
- 74 Z. Cai, R. Babbush, S. C. Benjamin, S. Endo, W. J. Huggins, Y. Li, J. R. McClean and T. E. O'Brien, *Rev. Mod. Phys.*, 2023, **95**, 045005.
- 75 D. Bultrini, M. H. Gordon, P. Czarnik, A. Arrasmith, M. Cerezo, P. J. Coles and L. Cincio, *Quantum*, 2023, **7**, 1034.
- 76 G. Brassard, P. Høyer, M. Mosca and A. Tapp, *Quantum Computation and Information*, American Mathematical Society, Providence, RI, 2002, vol. 305, p. 43.



Data Availability Statement

View Article Online
DOI: 10.1039/D5DD00381D

“Hybrid Quantum Algorithm for Simulating Real-Time Thermal Correlation Functions”

Elliot C. Eklund and Nandini Ananth

The code used to simulate a proton transfer reaction, create Fig. 3, and analyze the DVR error in SI Fig. 1 can be found at https://github.com/AnanthGroup/hPIMC_figures with DOI 10.5281/zenodo.16900433.

