Three-flavor supernova neutrino simulation using a hybrid quantum-classical algorithm with qutrits

Daniel J. Heimsoth,[1,*] Pooja Siwach,[2,3] and A. Baha Balantekin[1,†]

[1]*Department of Physics, University of Wisconsin–Madison, Madison, Wisconsin 53706, USA*
[2]*Department of Physics, University of Arizona, Tucson, Arizona 85721, USA*
[3]*Facility for Rare Isotope Beams, Michigan State University, East Lansing, Michigan 48824, USA*

(Received 5 May 2026; accepted 21 July 2026; published 21 August 2026)

We simulate a self-interacting three-flavor neutrino system within a core-collapse supernova using a hybrid classical-quantum algorithm on a qutrit computer. Based on the Dirac-Frenkel evolution equations, we employ a variation of the quantum-assisted simulator to calculate the system's time evolution operator by performing qutrit Hadamard tests to find expectation values of unitary operators in the Hamiltonian. The time evolution simulation is then done classically. We find that the hybrid algorithm produces results comparable to an exact numerical integration out to times of $t \approx 30\omega_0^{-1}$ with time step $\delta t = 0.005\omega_0^{-1}$, where ω_0 is the energy scale of the single neutrino vacuum oscillations. We discuss the lessons learned in simulating neutrino systems using this hybrid quantum-classical algorithm, along with the advantages it offers over quantum Trotterization.

DOI: 10.1103/8kcg-l79c

I. INTRODUCTION

The realization of low-noise, scalable quantum computers promises to allow the computation of complex problems that classical methods struggle with, either due to processing, memory, or time limitations. These quantum machines are still on the mid- to far-term horizon, however, leaving current research efforts with noisy, intermediate-scale quantum (NISQ) devices. These quantum computers, while useful in solving various types of problems of interest to scientists across fields, struggle with relatively low fidelities on entangling gates, the very operations that give quantum computers their true power over classical methods. Thus, it is advantageous to limit the depth of quantum circuits to minimize the number of entangling gates applied in succession.

One method to achieve this is to actually *not* use the quantum computer, at least in portions of the calculation that a classical computer can solve efficiently. The quantum computer is used only where it provides some scaling or processing advantage over classical methods, in particular for calculating matrix elements of unitary operators. These hybrid classical-quantum algorithms have been relatively popular in the era of NISQ devices to circumvent their limitations due to poor entangling gate fidelities and issues scaling up the number of qubits [1,2].

Time evolution of physical systems that easily map onto spin states is a prime candidate for such a hybrid calculation, since classical computers are very good at calculating time-evolved states (through matrix multiplication) and quantum computers are capable of calculating expectation values of spin operators at fairly low circuit depth. Collective neutrino flavor oscillations are such a system. (For reviews, see Refs. [3–9]). In collective neutrino oscillations, a single neutrino in the two-flavor approximation is a realization of SU(2) and thus maps directly onto a qubit state. Therefore, the collective many-body neutrino oscillation phenomenon beyond mean-field limit has been extensively studied with both classical [10–25] and quantum simulation methods [8,26–30] but mostly under the two-flavor approximation. From experiment, however, we know that there are (at least) three active flavors of neutrinos, and the oscillations among them live in SU(3). Pauli operators are substituted with Gell-Mann matrices, two energy levels become three, and the algebraic structure of the system becomes more complicated [31]. Thus, the natural quantum device for studying three-flavor neutrino states is instead the three-level qutrit [32], which brings complications to the calculation that do not show up in the qubit case [33–35]. Our goal in this paper is to identify and overcome these obstacles while simulating the time evolution of interacting three-flavor neutrinos in a core-collapse supernova (CCSN) environment.

In the following section, we introduce the mathematical framework that we will use to study supernova neutrino

*Contact author: dheimsoth@wisc.edu
†Contact author: baha@physics.wisc.edu

Published by the American Physical Society under the terms of the Creative Commons Attribution 4.0 International license. Further distribution of this work must maintain attribution to the author(s) and the published article's title, journal citation, and DOI. Funded by SCOAP³.

interactions. Then, in Sec. III, we translate our Hamiltonian onto a collection of qutrits, three-level quantum states that match the behavior of three-flavor neutrino oscillations and describe the process of implementing the Dirac-Frenkel quantum-assisted simulator. We present our results from a Dirac-Frenkel hybrid calculation in Sec. IV, along with a quantification of errors from this approach. We follow with Sec. V, a discussion of the benefits, drawbacks, and future developments of this hybrid algorithm in the context of the current and mid-term quantum computing landscape.

II. SUPERNOVA NEUTRINOS

Neutrino oscillations have a very important role in determining the dynamics and observables of core-collapse supernovae, as neutrinos are the main source of energy transport both within and out of the collapsing star [36]. They are predicted to dictate nucleosynthesis rates, electron fraction, and protoneutron star mass, among other quantities [37]. Because of their relatively weak interaction with the surrounding matter, neutrinos are the only direct probe of dynamics in the core of the collapsing star. The nature of their flavor oscillations has a nontrivial effect on supernova processes, as only electron neutrinos (and antineutrinos) have charged current interactions with the large mass of electrons in the core and surrounding matter. Further, these flavor oscillations affect the detectability of supernova neutrinos in Earth-based experiments since different detection channels are sensitive to different neutrino flavors [38–40].

Modeling neutrino interactions within a CCSN is a nontrivial task, as on average a CCSN is predicted to produce around 10^{53} erg in neutrinos which interact both with themselves and with the surrounding matter [41]. Further, the strength of the neutrino-neutrino interaction depends on the angle of the incoming particles, neutrino energy, and the neutrino density, all of which may be different for every interaction. To make this problem more tractable, some averaging is required; commonly, the single-angle approximation and neutrino bulb model are used [42–45]. The single-angle approximation assumes that the flavor evolution of the neutrinos is independent of their trajectories, which allows us to replace the angle-dependent interaction coupling with one that only depends on the radius, r. The neutrino bulb model then gives an approximation for this r dependence, as we expand upon shortly.

The Hamiltonian for neutrino-neutrino interactions at n evenly spaced discrete momenta in the single-angle approximation is [31,46]

$$H = \sum_{q=1}^{n} q\vec{B}\cdot\vec{Q}_q + \mu(t)\sum_{q<q'}\vec{Q}_q\cdot\vec{Q}_{q'}, \tag{1}$$

where $(Q_q)_i = \frac{1}{2}\sum_{j,k=1}^{3} a_j^\dagger(\lambda_{i,q})_{jk}a_k$ are the generators of SU(3) in terms of the Gell-Mann matrices λ_i on the qth qutrit and $\vec{B} = \{0, 0, \omega, 0, 0, 0, 0, \frac{2}{\sqrt{3}}\Omega\}$, with

$$\omega = -\frac{\delta m^2}{2E}, \qquad \Omega = -\frac{\Delta m^2}{2E}. \tag{2}$$

Here, $\delta m^2 = |m_2^2 - m_1^2|$ and $\Delta m^2 \approx |m_3^2 - m_1^2| \approx |m_3^2 - m_2^2|$ are the mass-squared differences of the neutrino mass states and E is the average neutrino energy. The time-dependent coupling strength $\mu(t)$ is determined by the aforementioned neutrino bulb model,

$$\mu(t) = \mu_0\left(1 - \sqrt{1 - \left(\frac{R_\nu}{r(t)}\right)^2}\right)^2; \tag{3}$$

assuming that neutrinos travel at the speed of light, $r(t) = r_0 + ct$. Here, $r(t)$ is the distance from the center of the neutrinosphere, and R_ν is the radius of the neutrinosphere. Properties of this simplified model such as entanglement, spectral splitting, and flavor oscillation have been extensively studied to date [10,17,31,47,48].

Our goal is now mathematically simple and quite familiar: given an initial state $|\psi_0\rangle$, what is the state at a later time t, $|\psi(t)\rangle$? How can we best leverage quantum computing's strengths to achieve this?

III. HYBRID QUTRIT ALGORITHM

As mentioned in the Introduction, NISQ devices struggle with entangling gate fidelity, forcing researchers to explore ways to limit excessive use of two-qudit gates in algorithms [49–52]. This effectively leads to circuit depth minimization; we would rather run many short high-fidelity circuits than run a few long circuits that produce untrustworthy results. This decision can be validated in the CCSN neutrino problem. If we want to perform a simple first-order Trotterization for some Hamiltonian H on a quantum computer, we must find a quantum circuit for the time evolution operator $\exp[-iH\delta t]$. This could be done with an algorithm such as time evolution block decimation (TEBD), where the diagonalization circuit for each block will require two-qubit gates [53]. The number of blocks (and consequently the number of diagonalization circuits required) increases with the number of neutrinos; even for modestly sized systems, the required number of entangling gates can be on the order of ten to 100 per time step. With current entangling gate fidelities hovering around 99%, these circuits are untenable [51,52].

For long time integrations of quantum systems, then, other methods are required in the near term. In this paper, we explore the feasibility of a quantum-assisted simulator (QAS) based on the Dirac-Frenkel variational principle for the time evolution of the supernova neutrino system laid out in Sec. II. As we will show, a QAS has the advantage of performing the time evolution classically, leveraging

classical computers' efficiency at matrix multiplication. This leaves the task of calculating the time evolution operator matrix elements to the quantum computer.

A. Unitary form of Hamiltonian

To find a quantum circuit for our neutrino Hamiltonian H in Eq. (1), it is natural to use qutrits, which transform in SU(3) identically to neutrino three-flavor states. Unfortunately, unlike the generators of SU(2) (the Pauli matrices), the Gell-Mann matrices are not unitary and cannot be used as gates on qutrits. We thus need to rewrite the Hamiltonian in terms of unitary matrices. In analogy with SU(2), we can define generalized X and Z gates for qutrits [54–56]

$$X = \begin{pmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix}, \qquad Z = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \beta & 0 \\ 0 & 0 & \beta^2 \end{pmatrix}, \tag{4}$$

where $\beta = \exp 2\pi i/3$ is a third root of unity. The X gate simply cycles the basis states $|0\rangle \to |1\rangle \to |2\rangle$, while the Z gate adds phases to the $|1\rangle$ and $|2\rangle$ states. It is straightforward to show that these matrices are unitary, and, using the identity $1 + \beta + \beta^2 = 0$, we see that both are traceless. We can then find a set of operators $\vec{\Sigma}$ composed of products of X and Z that are in one-to-one correspondence with the Gell-Mann matrices:

$$\vec{\Sigma} = \{X, Z, X^2, \beta XZ, Z^2, \beta^2 XZ^2, X^2Z, X^2Z^2\}.$$

Expressions converting the Gell-Mann matrices to these unitary operators and vice versa can be found in the Appendix.

We can now rewrite the Hamiltonian in terms of matrices in $\vec{\Sigma}$. Writing $H = H_\nu + \mu(t)H_{\nu\nu}$ and defining $\omega_q = q\omega$ and $\Omega_q = q\Omega$, with ω and Ω defined in Eq. (2), we derive

$$H_\nu = \sum_{q=1}^{n} q\,(\varepsilon Z_q + \varepsilon^* Z_q^2), \tag{5}$$

where

$$\varepsilon = \omega\frac{1-\beta^2}{6} - \Omega\frac{\beta}{3}.$$

We calculate the interaction term as

$$\begin{aligned} H_{\nu\nu} &\equiv \sum_{q<q'} \vec{\lambda}_q \cdot \vec{\lambda}_{q'} = \sum_{q\neq q'}\sum_{i=1}^{8} \lambda_{i,q} \otimes \lambda_{i,q'} \\ &= \sum_{q<q'} \frac{2}{3}[X_q \otimes X_{q'}^2 + X_q^2 \otimes X_{q'} + Z_q \otimes Z_{q'}^2 + Z_q^2 \otimes Z_{q'} \\ &\quad + \beta(X_q Z_q^2 \otimes X_{q'}^2 Z_{q'} + X_q^2 Z_q \otimes X_{q'} Z_{q'}^2) \\ &\quad + \beta^2(X_q Z_q \otimes X_{q'}^2 Z_{q'}^2 + X_q^2 Z_q^2 \otimes X_{q'} Z_{q'})], \end{aligned} \tag{6}$$

where X_q, Z_q act on the qth neutrino and the tensor products are assumed to also contain identity operators for all other momentum states $k \neq q, q'$. It is not immediate that $H_{\nu\nu}$ is Hermitian, but using the relations $XZ = \beta^2 ZX$ and $\beta^* = \beta^2$, this can be verified. For n neutrinos, there will be $8 \cdot \frac{n(n-1)}{2} = 4n(n-1)$ unitary operators in $H_{\nu\nu}$. These operators also have trivial circuits composed only of single-qutrit gates [e.g., $X \otimes X^2 = (X \otimes I) \cdot (I \otimes X^2)$], simplifying the forthcoming calculations.

B. Dirac-Frenkel time evolution

Time evolution of our neutrino system on a quantum computer would normally follow some form of Trotterization, where the state vector is updated over some time step δt by the application of a time evolution operator circuit, $e^{-iH\delta t}$. This often leads to very deep circuits with many entangling gates for even modest numbers of neutrinos and time steps [34,57]. We employ a different method to avoid this problem, instead using a quantum-assisted simulator based on the Dirac and Frenkel variational principle [2]. Below, we provide a short explanation of the procedure; for a lengthier discussion, see Refs. [29,58].

Given a state of our system at time t, $|\psi(t)\rangle$, we wish to calculate the state at some later $t + \delta t$. We first introduce an ansatz basis of states $\{|\phi_i\rangle\}_i$ and corresponding variational parameters $\{\alpha_i(t) \in \mathbb{C}\}_i$ that accurately represent the state $|\psi(t)\rangle$:

$$|\psi(t)\rangle = \sum_i \alpha_i(t)|\phi_i\rangle \equiv |\phi(t)\rangle. \tag{7}$$

The time-evolved state $|\psi(t+\delta t)\rangle$ may not in general be in the vector space of the ansatz basis states, but for small δt, the approximation is good:

$$|\psi(t+\delta t)\rangle \approx |\phi(t+\delta t)\rangle \approx |\phi(t)\rangle - i\delta t H|\phi(t)\rangle. \tag{8}$$

On the other hand, we can expand $|\phi(t+\delta t)\rangle$ in powers of $\delta\alpha_i$ to obtain

$$\begin{aligned} |\phi(t+\delta t)\rangle &\approx |\phi(t)\rangle + \sum_i \frac{\partial|\phi(t)\rangle}{\partial\alpha_i}\delta\alpha_i \\ &= |\phi(t)\rangle + \sum_i \delta\alpha_i(t)|\phi_i\rangle. \end{aligned} \tag{9}$$

Thus, we need to equate $-i\delta t H|\phi(t)\rangle$ and $\sum_i \delta\alpha_i(t)|\phi_i\rangle$, which can be achieved by projecting directly onto the ansatz space:

$$\left(\sum_i \delta\alpha_i^\dagger(t)\langle\phi_i|\right)\left(\frac{d}{dt} + iH\right)|\phi(t)\rangle = 0. \tag{10}$$

This leads to a matrix equation for the time derivatives of the α_i parameters,

$$\mathcal{E}\dot{\vec{\alpha}}(t) = -i\mathcal{D}\vec{\alpha}(t), \tag{11}$$

where $\vec{\alpha} = \{\alpha_i\}_i$ and

$$\mathcal{E}_{ij} = \langle\phi_i|\phi_j\rangle, \qquad \mathcal{D}_{ij} = \langle\phi_i|H|\phi_j\rangle,$$

along with the unitarity condition $\vec{\alpha}^\dagger\mathcal{E}\vec{\alpha} = 1$. Thus, our dynamical variables are now the variational parameters $\alpha_i(t)$, and the resulting differential equation can be easily solved classically. Of course, we are left with the tasks of choosing an ansatz basis and finding the matrix elements of $\mathcal{E}$ and $\mathcal{D}$; we achieve the latter by performing Hadamard tests on a quantum computer.

C. Qutrit Hadamard test

We can decompose the matrix elements of $\mathcal{D}$ by noticing that the terms in our Hamiltonian given in Eqs. (5) and (6) are sums of unitary operators, i.e., H is of the form

$$H = \sum_k \beta_k U_k + \mu(t)\sum_l \gamma_l V_l. \tag{12}$$

Thus, we can write $\mathcal{D}$ as time-independent and time-dependent parts, $\mathcal{D} = \mathcal{D}^I + \mu(t)\mathcal{D}^D$, with

$$\mathcal{D}^I_{ij} = \sum_k \beta_k\langle\phi_i|U_k|\phi_j\rangle, \qquad \mathcal{D}^D_{ij} = \sum_l \gamma_l\langle\phi_i|V_l|\phi_j\rangle. \tag{13}$$

To calculate both $\mathcal{E}$ and $\mathcal{D}$, we must calculate expectation values of the form $\langle\phi_i|U|\phi_j\rangle$ for some unitary operator U (in the case of $\mathcal{E}$, the operator is simply the identity matrix). This can be achieved using the well-known Hadamard test, which calculates expectation values of unitary gates $\langle\psi|U|\psi\rangle$, but for qutrits instead of qubits [59]. Figure 1 shows the qutrit circuit to calculate the real part of the expectation value using the qutrit Hadamard gate,

$$\mathrm{H} = \frac{1}{\sqrt{3}}\begin{pmatrix} 1 & 1 & 1 \\ 1 & \beta & \beta^2 \\ 1 & \beta^2 & \beta \end{pmatrix}.$$

$\mathrm{Re}\langle\psi|U|\psi\rangle$ is derived from the probability of measuring the three computational basis states of the ancillary qutrit:

$$\begin{aligned} P(0) &= \frac{1}{9}[5 + 2\langle\psi|(U^\dagger + U)|\psi\rangle], \\ P(1) &= \frac{1}{9}[2 - \langle\psi|(U^\dagger + U)|\psi\rangle], \\ P(2) &= P(1). \end{aligned}$$

Thus,

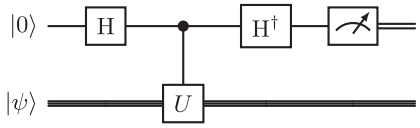


FIG. 1. Qutrit Hadamard test to calculate $\mathrm{Re}\langle\psi|U|\psi\rangle$. The expectation value is derived from the measurement probabilities on the ancillary qutrit; see text for the exact equation. Note that the qutrit Hadamard gate is not Hermitian ($\mathrm{H} \neq \mathrm{H}^\dagger$).

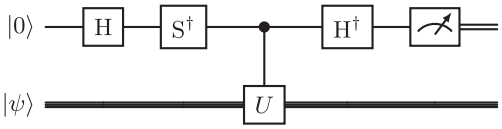


FIG. 2. Qutrit Hadamard test to calculate $\mathrm{Im}\langle\psi|U|\psi\rangle$.

$$\begin{aligned} \mathrm{Re}\langle\psi|U|\psi\rangle &= \frac{1}{2}\langle\psi|(U^\dagger + U)|\psi\rangle \\ &= P(0) - \frac{5}{4}(P(1) + P(2)). \end{aligned} \tag{14}$$

The imaginary part of $\langle\psi|U|\psi\rangle$ can be calculated with a similar circuit, shown in Fig. 2. This circuit uses the Hermitian conjugate of the qutrit S gate,

$$\mathrm{S} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \sqrt{\beta} & 0 \\ 0 & 0 & \beta \end{pmatrix}.$$

It can be shown straightforwardly that

$$\mathrm{Im}\langle\psi|U|\psi\rangle = \frac{3\sqrt{3}}{4}[P(0) - P(2)]. \tag{15}$$

Now, we are almost ready to use the Hadamard test circuits to calculate $\mathcal{E}$, $\mathcal{D}^I$, and $\mathcal{D}^D$. We notice, however, that the Hadamard tests assume the same state sandwiching the unitary operator, while our matrices' off-diagonal elements contain different $\langle\phi_i|$ and $|\phi_j\rangle$. Thus, we must express the $|\phi_i\rangle$ in terms of a common state, i.e., $|\phi_i\rangle = A_i|\phi_0\rangle$ for some unitary matrix A_i and state $|\phi_0\rangle$. Then, to calculate $\langle\phi_i|U|\phi_j\rangle = \langle\phi_0|A_i^\dagger U A_j|\phi_0\rangle$, we replace the unitary operator U in the Hadamard test with $A_i^\dagger U A_j$. Luckily, our procedure for building the ansatz basis states gives us these A_i directly.

D. Ansatz basis

The central idea behind building the ansatz basis for the quantum-assisted simulator is to find states that are likely to strongly overlap with the "true" wave function $|\psi(t)\rangle$, given an initial state $|\psi(0)\rangle$. For a time-dependent Hamiltonian such as ours in Eq. (1), the time evolution operator $e^{-iH\delta t}$ can

be written as a sum of commutators of the Hamiltonian with itself (e.g., in a Magnus expansion [60]). Thus, our initial state $|\psi(0)\rangle$ will be acted on by unitary matrices contained in such operators, so it would be advantageous for our ansatz to be comprising states of the form

$$|\phi_i\rangle = A_i|\psi(0)\rangle,$$
$$A_i \in \left\{ U \in U(3):\ U \in \sum_k c_k C_k, c_k \in \mathbb{C} \right\},$$
$$C_k \in \{I, H_\nu, [H_\nu, H_{\nu\nu}], [H_\nu, [H_\nu, H_{\nu\nu}]], \ldots\}.$$

Here, $U(3)$ denotes the group of 3×3 unitary matrices. That is, we want the A_i to be unitary operators that are contained somewhere in the time evolution operator. We set $A_0 = I$ so that $|\phi_0\rangle = |\psi(0)\rangle$; it will be this state that will be the initial prepared state in the Hadamard tests discussed in the previous section.

While this procedure produces a set of states that can be whittled down to a useable basis $\{A_i\}$, we still need to find quantum circuits for each A_i to calculate $\mathcal{E}$ and $\mathcal{D}$. This is achievable given that all the A_i are some combination of commutators of unitary matrices in the Hamiltonian, but procedurally generating them for any number of neutrinos is nontrivial. To simplify this, we choose to define our ansatz basis such that the A_i are easily translated into quantum circuits.

Defining the set $\mathcal{X} = \{I, X, X^2\}$, we build the A_i by taking every possible tensor product combination of the elements of $\mathcal{X}$ for n neutrinos. Thus, $A_i = X^{i_1} \otimes X^{i_2} \otimes \ldots \otimes X^{i_n}$, with the exponents $i_j \in \{0, 1, 2\}$, $j = 1, \ldots, n$. Again, using this procedure $A_0 = I$ ($i_j = 0\ \forall\ j$). Further, this choice of basis makes the circuits for the A_i trivial (simply applications of X and X^2 gates on requisite qutrits). From our testing, we find this choice of basis does not produce appreciable errors, at least compared to other sources of errors. We also looked at different definitions of $\mathcal{X}$, for example, using powers of XZ instead of X, and found no significant difference in performance. We note that this ansatz basis is not orthogonal, which is not a particular problem since the Dirac-Frenkel method does not require orthogonality as any overlap in states is taken into account in the $\mathcal{E}_{ij} = \langle\phi_i|\phi_j\rangle$ matrix. However, this does mean that our 3^n states do not fully span the entire Hilbert space; by building the basis with X gates applied to our initial state, it is expected that the basis states will have high overlap with the state at time t, $|\psi(t)\rangle = e^{-iHt}|\psi(0)\rangle$, given that the time evolution operator contains a sum of commutators of the Hamiltonian with itself (the terms in C_k above).

IV. RESULTS

We considered a four neutrino system with three flavors described by Eq. (1) in the initial state $|ee\mu\tau\rangle$, with the first neutrino having the lowest energy (in this case, the first ν_e). In the following analysis, we use the variable $\omega_0 = 10^{-10}$ eV as the energy scale of the neutrino vacuum oscillations and redefined dimensionful quantities in terms of this scale. Table I lists the values of physical parameters used in the calculation. We calculated the $\mathcal{E}$, $\mathcal{D}^I$, and $\mathcal{D}^D$ matrices by simulating quantum Hadamard test circuits for each matrix element as described in Sec. III C. We simulated each Hadamard circuit 2,000,000 times using the QuForge [61] qudit simulation package to obtain estimates of the measurement probabilities on the ancillary qutrit, and each matrix element was independently calculated 65 times. Then, for each run, we simulated the time evolution of the $\vec{\alpha}(t)$ [Eq. (11)] classically with time step $\delta t = 0.005\omega_0^{-1}$ out to final time $t_f = 5000\omega_0^{-1}$ using the SCIPY.INTEGRATE.SOLVE_IVP method with the DOP853 eighth-order Runge-Kutta method [62]. This time step δt was chosen to be small enough to fully resolve the flavor oscillations, and we chose t_f such that the mass state probability and entropy reach constant, asymptotic values for each neutrino. We also solved the Schrödinger equation explicity, again using SCIPY.INTEGRATE.SOLVE_IVP to find $|\psi(t)\rangle$ in the interval $[0, t_f]$. In the following text and figures, the Dirac-Frenkel method is referred to as "DF" and/or drawn in solid lines, while the numerical Schrödinger equation solution is called "ODE" or "Exact ODE" and/or drawn in dashed lines.

TABLE I. Values of quantities used in this analysis. We assume natural units $c = \hbar = 1$, and units are shown in parentheses.

ω_0 (eV)	$\mu_0\,(\omega_0)$	$R_\nu\,(\omega_0^{-1})$	$r_0\,(\omega_0^{-1})$	E (MeV)
10^{-10}	3.62×10^4	32.2	210.64	10
θ_{12} (deg)	θ_{23} (deg)	θ_{13} (deg)	δm^2 (eV^2)	Δm^2 (eV^2)
33.41	49.1	8.54	7.4×10^{-5}	2.51×10^{-3}

Figure 3 shows the survival probability of our initial $|ee\mu\tau\rangle$ state, defined as $P(\text{surv}) = |\langle ee\mu\tau|\psi(t)\rangle|^2$, at early times (left) and late times (right) calculated with the DF hybrid algorithm. We see that the median of the hybrid algorithm runs (shown in black) matches the numerical solution (red dotted line) fairly well, although as the integration continues, the amplitude of the oscillations becomes damped. This damping is caused by slight variations in the time evolution matrices $\mathcal{E}$ and $\mathcal{D}$ of the 65 runs that are being averaged over and taken to late times causes a complete loss of any signal. However, each individual run still has nontrivial oscillations in the survival probability, as highlighted by the wide range of hybrid algorithm data (blue band in the figure), even at late times.

Figure 4 shows the probability of finding each neutrino q from the initial $|ee\mu\tau\rangle$ state in the ith mass state $|\nu_i\rangle$ over time, defined as $P_{m_i,q}(t) = \text{tr}[\rho_q(t)\Pi_{i,q}]$, where $\Pi_{i,q} = (\bigotimes_{k\neq q} \mathbb{1}_k) \otimes |\nu_i\rangle\langle\nu_i|_q$ and ρ_q is the reduced density matrix

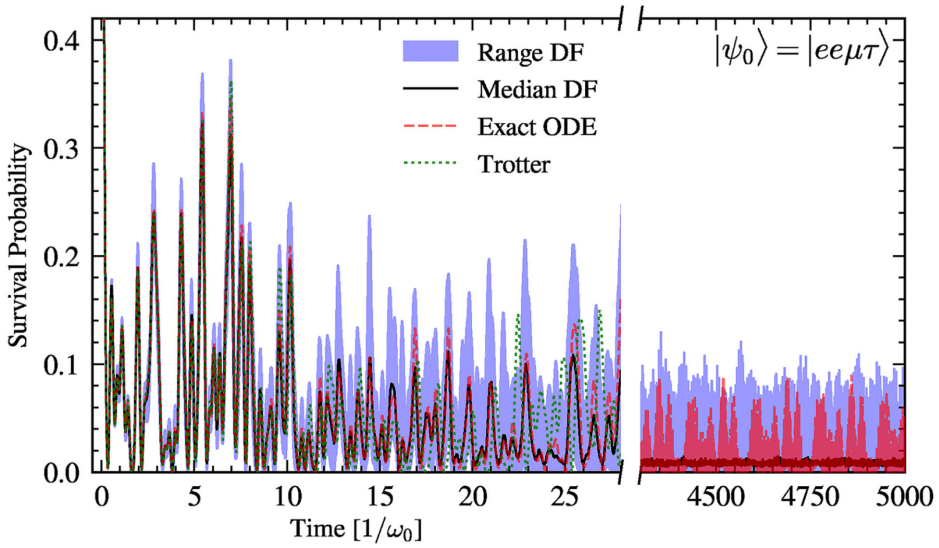


FIG. 3. Survival probability of an initial $|ee\mu\tau\rangle$ state, calculated with the Dirac-Frenkel quantum-assisted simulator (median in black and total range in blue), first-order simulated Trotterization (green dotted), and a numerical integration (red dashed). Each Hadamard test in the hybrid simulator used $n_{\text{shots}} = 2000000$ shots, and the simulation was repeated $n_{\text{runs}} = 65$ times. The Trotter circuit was calculated with $\delta t = 0.005\omega_0$. At late times, the hybrid algorithm loses any resemblance to the exact calculation.

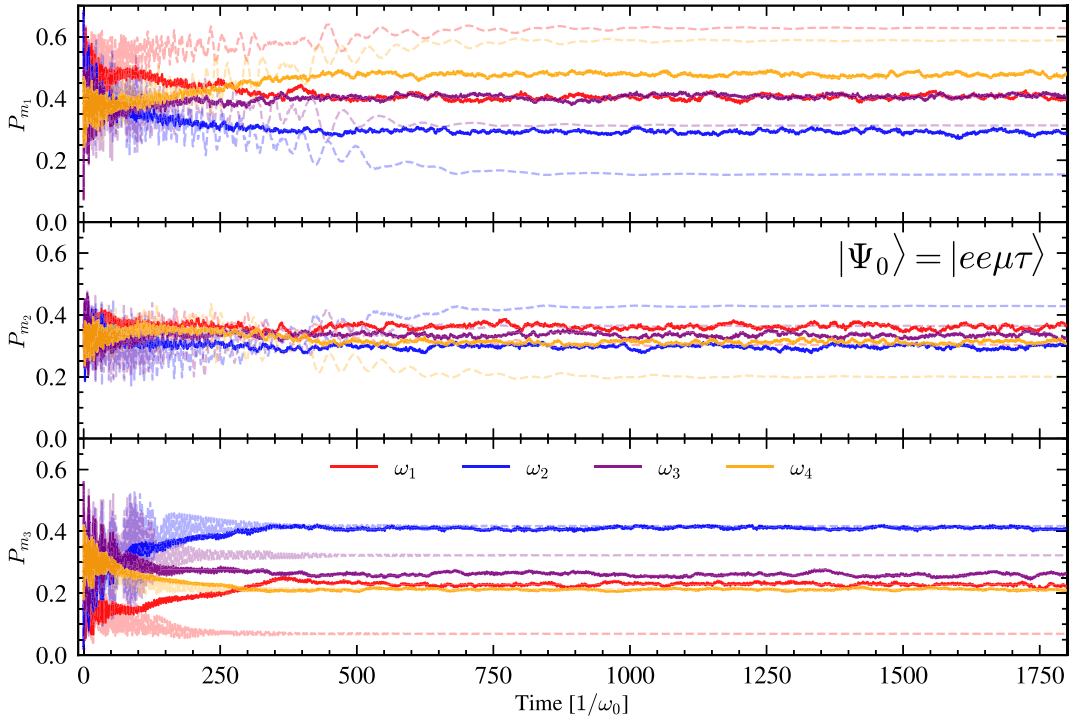


FIG. 4. Mass state probability of each particle from an initial $|ee\mu\tau\rangle$ state, calculated with the Dirac-Frenkel quantum-assisted simulator (solid lines) and a numerical integration (dotted lines). Data from the same run as in Fig. 3.

for the qth neutrino. We see a similar phenomenon occurring here as in Fig. 3, where the hybrid algorithm slowly deviates from the exact solution (dotted lines) due to small deviations in the time evolution matrices. By the time we reach the asymptotic region, the data from the various hybrid algorithm runs have been smoothed out and no longer resembles the expected behavior.

We again find a similar story in Fig. 5, which shows the von Neumann entropy of the four neutrinos as a function of time. The von Neumann entropy for the ith neutrino is defined as

$$S_{VN,i}(t) = -\text{tr}(\rho_i \ln \rho_i),$$

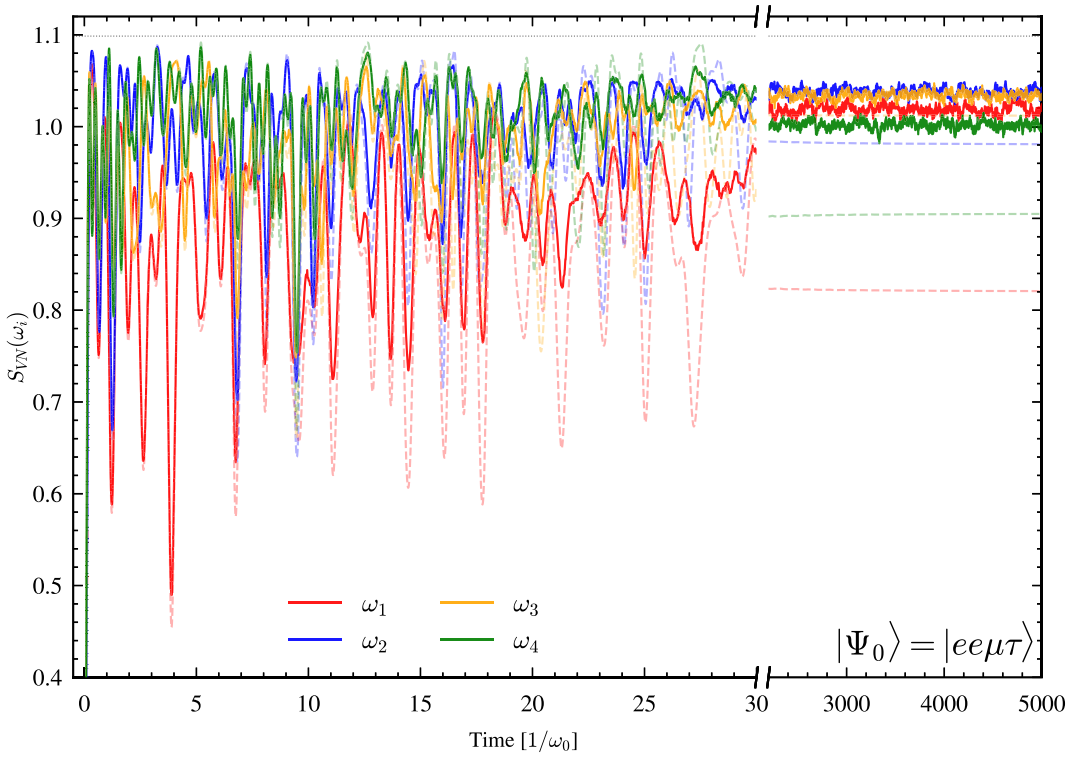


FIG. 5. Von Neumann entropy of each particle from an initial $|ee\mu\tau\rangle$ state, calculated with the Dirac-Frenkel quantum-assisted simulator (solid lines) and a classical numerical integration (dotted lines). Data from the same run as in Fig. 3.

where ρ_i is the reduced density matrix for the ith neutrino. The oscillations in entropy become damped over time such that at late times the asymptotic values deviate from the exact solution.

A. Error analysis

Despite the deviations from the exact numerical integration results that we saw in the previous section, the Dirac-Frenkel quantum variational method is still useful to avoid common pitfalls in current noisy quantum time evolution simulations. Circuit depth that scales with the number of time steps and Trotterization errors from building circuits for the time evolution operator $e^{-iH\delta t}$ are two barriers that limit quantum computers' effectiveness in the near term, and the hybrid algorithm avoids both. However, the hybrid algorithm, like any numerical computation, has its own limitations. The dimension of the $\mathcal{E}$ and $\mathcal{D}$ matrices still scales as 3^n, and the number of unitary operators in the Hamiltonian increases with the system size as well. Thus, the number of Hadamard tests that need to be performed increases faster than 3^n, requiring much more quantum computing time for larger systems.

Another issue that arose during our study was the error of $\mathcal{E}$ and $\mathcal{D}$ matrix elements due to the probabilistic nature of quantum measurement. The α-norm is defined as $\vec{\alpha}^\dagger \mathcal{E} \vec{\alpha}$ is equal to 1, assuming $\mathcal{E}$ and $\mathcal{D}$ are Hermitian. To quantify the effect of matrix errors on the simulation results, we performed the following analysis: 1) the $\vec{\alpha}$-vector was calculated exactly using a classical numerical method, denoted $\vec{\alpha}_{cl}$; 2) "noisy" $\mathcal{E}$ and $\mathcal{D}$ matrices were produced by injecting random complex-valued Gaussian noise into each matrix element of the classically calculated matrices for multiple different magnitudes of noise; 3) $\vec{\alpha}(t)$ was calculated using the noisy evolution matrices by the same numerical integration used from the Dirac-Frenkel method in the main analysis of this paper; and 4) the overlap $\vec{\alpha}_{cl}^\dagger \mathcal{E}_{cl} \vec{\alpha}(t)$ was calculated, where $\mathcal{E}_{cl}$ is the $\mathcal{E}$ with no injected noise. Crucially, we maintain the Hermiticity of the noisy $\mathcal{E}$ and $\mathcal{D}$ matrices by producing random noise only for the upper triangular portion of the matrix and adding the complex conjugate of that noise to the lower triangular part. This matches how we calculate these matrices in the Dirac-Frenkel method, where we assume the matrices are Hermitian and thus only perform Hadamard tests for the upper triangular part of the matrices. [As an example, for two ansatz basis states $|\phi_i\rangle$ and $|\phi_j\rangle$, $(\mathcal{D}_{ij})^\dagger = (\langle\phi_i|H|\phi_j\rangle)^\dagger = \langle\phi_j|H|\phi_i\rangle = \mathcal{D}_{ji}$].

The deviation of the noisy values $\vec{\alpha}(t)$ from the noiseless $\vec{\alpha}_{cl}$ can be quantified with the measure $1 - \vec{\alpha}_{cl}^\dagger \mathcal{E}_{cl} \vec{\alpha}(t)$. In Fig. 6, we plot the absolute value of this measure as a function of integration time for different magnitudes of injected matrix error. Not surprisingly, the larger the error in the evolution matrices, the larger the deviation from the numerical results. Also, this deviation

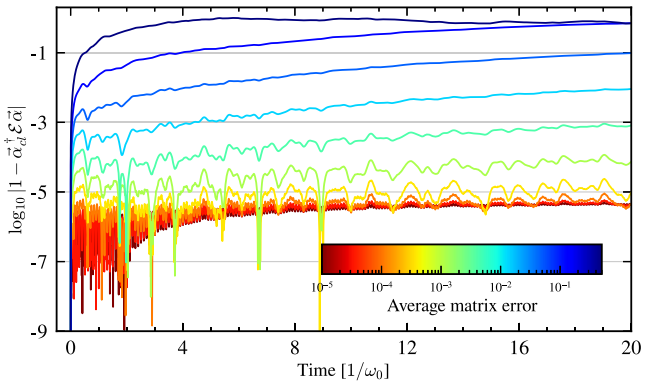


FIG. 6. Error in the norm $\vec{\alpha}_{cl}^{\dagger}\mathcal{E}\vec{\alpha}$ as a function of integration time for different evolution matrix error scales, where $\vec{\alpha}_{cl}$ is the α vectors computed with classically calculated evolution matrices (as opposed to Hadamard test circuits as done in the main analysis). We take the initial state of the system to be $|ee\mu\tau\rangle$ as in the main analysis. As the error in the matrix elements and integration time increase, so too does the deviation from the classical result.

increases with time, setting an effective maximum time limit for the simulation.

We see that below a matrix error of $\sim 10^{-2}$ the error in the α-norm remains below 1% out to time $t \approx 20\omega_0^{-1}$, while errors 10^{-2} and larger quickly deviate from the classical values (over 10% by $t = 20\omega_0^{-1}$). Thus, in order to achieve satisfactory results from the Dirac-Frenkel hybrid algorithm, errors in the time evolution matrices must be kept very small. The simplest method to decrease these matrix errors is to simply run more shots in the Hadamard tests that calculate the time evolution matrix elements. In our calculations of the four neutrino system above, we achieved matrix errors averaging about 10^{-2} with 2×10^6 shots; this produced results reasonably close to the classical calculation out to times on the order of $10 - 30\omega_0^{-1}$. The addition of gate noise would of course build upon the shot noise from the measurement of the Hadamard circuits. Our analysis here is agnostic to the source of the noise, so we can determine an estimate for the accuracy of a particular simulation given the number of shots per Hadamard test and gate fidelity. For example, a two-qutrit gate fidelity of 99.9% in the Hadamard tests would introduce to the time evolution matrices an error of ~1%, which according to Fig. 6 would still keep the simulation accurate to ~1% level out to $t = 20\omega_0^{-1}$ (assuming no shot noise). See Sec. V for further discussion of the impact of two-qutrit (entangling) gate fidelity on the performance of the Hadamard tests in the Dirac-Frenkel method.

V. DISCUSSION

The Dirac-Frenkel quantum-assisted simulator, in the current context of quantum computing, provides a workaround for current NISQ quantum devices: instead of attempting to evaluate a long, deep time evolution circuit, we can perform many short Hadamard test circuits to calculate the time evolution operator which is used for classical time integration.

Nevertheless, there are many algorithmic and computational hurdles that still exist for the Dirac-Frenkel hybrid simulator, some of which we have highlighted here in this study. Most glaringly, the results from this method depend strongly on how accurately the time evolution matrix elements are calculated by the quantum Hadamard tests. Increased accuracy will allow for longer time integrations, but determining this required accuracy for a given time interval is a nontrivial task. Because of the probabilistic nature of measuring Hadamard tests, better accuracy can be achieved simply by running more shots per circuit, as in the limit $n_{\text{shots}} \to \infty$ the matrix elements should approach the classically calculated values. Thus, unlike errors in other quantum time evolution algorithms such as Trotterization, there is a very clear path to mitigating these matrix element errors–throw more computing time at the problem.

The Dirac-Frenkel hybrid algorithm also does not solve the issue of classically simulating a Hilbert space that scales exponentially with the number of neutrinos. As mentioned in the previous section on error analysis, the number of degrees of freedom of the system scales as 3^n, which still needs to be time evolved on a classical computer. The Dirac-Frenkel equations comprise a system of coupled first-order ordinary differential equations, which, while certainly easy to solve in the grand scheme of differential equations, still require considerable compute time and memory as the number of equations increases. For a 20 neutrino system, the matrix dimension would already be over 3 billion. One way to mitigate this exponential scaling is to cull the size of the ansatz basis to keep only states with amplitudes over some threshold within the first few time steps. Indeed, this is perhaps the greatest strength of this type of algorithm: we are free to choose our ansatz basis as we please, as long as that choice preserves the data of the system up to some tolerance. For our small four neutrino state, we chose not to decrease the size of the basis; as the number of neutrinos increases, it seems almost mandatory to do so to make the problem tractable. We leave the study of strategies for ansatz basis culling to a later date.

A. Comparison to Trotterization

As mentioned in the error analysis subsection above, the more pressing scaling issue with this hybrid algorithm is the exponential increase in the number of Hadamard tests required in order to calculate the time evolution matrices $\mathcal{E}$ and $\mathcal{D}$. The number of Hadamard tests needed is equal to $n_{\text{tests}} = n_U \cdot n_{\text{basis}} \cdot n_{\text{shots}} \cdot n_{\text{runs}}$, where n_U is the number of unitary operators in the Hamiltonian, n_{basis} is the size of the ansatz basis, n_{shots} is the number of shots per circuit,

and n_{runs} is the number of independent runs of the system to average over. In our study, with four neutrinos, $n_U = 56$, $n_{\text{shots}} = 2 \times 10^6$, and $n_{\text{runs}} = 65$, we had to take $\sim 5.8 \times 10^{11}$ measurements of Hadamard circuits. Of course, simulating these tests on a classical computer allows for some parallelization, but applying this method to quantum hardware does not benefit from this speedup.

Despite these drawbacks, the Dirac-Frenkel hybrid algorithm offers an improvement in time evolution calculations over Trotterization on NISQ devices for several reasons. First, there is no requirement to build a circuit for the time evolution operator $\exp(-iH\delta t)$, which in general is a matrix sum of nested commutators of the Hamiltonian with itself. This almost always requires some form of approximation, for example in TEBD, which introduces errors that scale polynomially with the time step. Second, the hybrid algorithm avoids deep Trotterization circuits that are plagued by unsatisfactory fidelities on entangling gates. While it is true that the Hadamard test also contains entangling gates (since the Hamiltonian unitary operator and ansatz basis operators needs to be controlled by the ancilla qutrit), the number of these gates stays relatively small.

Quantitatively, for our particular system, the maximum number of entangling gates for a Hadamard test of $\mathcal{D}_{ij} \supset \langle\phi_i|U|\phi_j\rangle = \langle\phi_0|A_i^\dagger U A_j|\phi_0\rangle$ is 12: four each for the $A_i^\dagger$ and A_j circuits (which are at most four X or X^2 gates) and four from one of the unitary operators in the interaction Hamiltonian $H_{\nu\nu}$. In contrast, a first-order Trotterization circuit representing $e^{-iH\delta t}$ also requires 12 entangling gates. The crucial difference is that any Trotterization error continues to accumulate with each time step, while the Hadamard tests are all independent, short calculations. Also importantly, the size and depth of the Hadamard tests increase linearly with the number of neutrinos (due to the application of the ansatz basis operators $A_i^\dagger$ and A_j), since our Hamiltonian only contains unitary operations between at most two neutrinos. In contrast, for a first-order Trotter circuit of the neutrino collective oscillation Hamiltonian, the number of entangling gates required goes as $n(n-1)$ per time step, and Trotter error still scales as $\mathcal{O}(\delta t^2)$. To reiterate, however, the Dirac-Frenkel method achieves this better scaling by requiring many more (shorter) circuits to be run, although these circuits contain very basic gates such as controlled-X and controlled-Z gates.

VI. CONCLUSIONS

We performed a hybrid quantum-classical time evolution simulation based on the Dirac-Frenkel variational principle for a core-collapse supernova neutrino system. We found that for our initial four neutrino coherent flavor state the hybrid algorithm matches the exact numerical result out to times of around $t \approx 20$–$30\omega_0^{-1}$ with time step $\delta t = 0.005\omega_0^{-1}$, where ω_0 is the energy scale of the single neutrino vacuum oscillations. This time resolution over this integration time is currently impossible using a Trotterization circuit on NISQ devices given the state of entangling gate fidelities.

While the Dirac-Frenkel hybrid algorithm has its own drawbacks and exponential scaling issues, it offers a usable and easily implementable method to study physical systems on current quantum hardware. Further, improvements to the results shown in this paper can be achieved by increasing the computation time theoretically to any precision, as the error in the quantum-calculated time evolution matrices goes to zero as the number of measurement shots goes to infinity. While we hope and expect quantum computers will become more reliable and capable of longer computations in the future, the Dirac-Frenkel hybrid algorithm and quantum-assisted simulators like it are one of the best choices for leveraging quantum hardware today to study physical systems.

ACKNOWLEDGMENTS

This material is based upon work supported in part by the U.S. Department of Energy, Office of Science, Office of Nuclear Physics, under the FRIB Theory Alliance Grant No. DE-SC0013617. It was also supported in part by the U.S. National Science Foundation Grant No. PHY-2411495.

DATA AVAILABILITY

The data that support the findings of this article are not publicly available upon publication because it is not technically feasible and/or the cost of preparing, depositing, and hosting the data would be prohibitive within the terms of this research project. The data are available from the authors upon reasonable request.

APPENDIX: ZX MATRICES FROM GELL-MANN MATRICES

The operator basis for the ZX algebra is

$$\vec{\Sigma} = \{\mathbb{1}, X, Z, X^2, \beta XZ, Z^2, \beta^2 XZ^2, X^2Z, X^2Z^2\}. \quad \text{(A1)}$$

The basis elements can be written as linear combinations of the Gell-Mann matrices (including an "identity" Gell-Mann matrix λ_0):

$$\mathbb{1} = \lambda_0, \qquad X = \frac{1}{2}[(\lambda_1 - i\lambda_2) + (\lambda_4 + i\lambda_5) + (\lambda_6 - i\lambda_7)], \qquad Z = \frac{1}{2}(1-\beta)(\lambda_3 + i\lambda_8),$$
$$X^2 = \frac{1}{2}[(\lambda_1 + i\lambda_2) + (\lambda_4 - i\lambda_5) + (\lambda_6 + i\lambda_7)], \qquad \beta XZ = \frac{1}{2}[\beta(\lambda_1 - i\lambda_2) + (\lambda_4 + i\lambda_5) + \beta^2(\lambda_6 - i\lambda_7)],$$
$$Z^2 = \frac{1}{2}(1-\beta^2)(\lambda_3 - i\lambda_8), \qquad \beta^2 XZ^2 = \frac{1}{2}[\beta^2(\lambda_1 - i\lambda_2) + (\lambda_4 + i\lambda_5) + \beta(\lambda_6 - i\lambda_7)],$$
$$X^2 Z = \frac{1}{2}[\beta(\lambda_1 + i\lambda_2) + (\lambda_4 - i\lambda_5) + \beta^2(\lambda_6 + i\lambda_7)], \qquad X^2 Z^2 = \frac{1}{2}[\beta^2(\lambda_1 + i\lambda_2) + (\lambda_4 - i\lambda_5) + \beta(\lambda_6 + i\lambda_7)].$$

Note that these operators can also be written in terms of the $\mathfrak{su}(2)$ subalgebra raising and lowering operators $I^{\pm} = \lambda_1 \pm \lambda_2$, $U^{\pm} = \lambda_4 \pm \lambda_5$, and $V^{\pm} = \lambda_6 \pm \lambda_7$ and two "number operators" $N^{\pm} = \lambda_3 \pm i\lambda_8$. It is easy to check that, while these matrices are not Hermitian, they are unitary.

The inverse relations to those above are

$$\lambda_0 = \mathbb{1}, \qquad \lambda_1 = \frac{1}{3}[X(\mathbb{1} + Z + Z^2) + X^2(\mathbb{1} + \beta^2 Z + \beta Z^2)], \qquad \lambda_2 = \frac{i}{3}[X(\mathbb{1} + Z + Z^2) - X^2(\mathbb{1} + \beta^2 Z + \beta Z^2)],$$
$$\lambda_3 = \frac{1}{3}[(1-\beta^2)Z + (1-\beta)Z^2], \qquad \lambda_4 = \frac{1}{3}[X(\mathbb{1} + \beta Z + \beta^2 Z^2) + X^2(\mathbb{1} + Z + Z^2)],$$
$$\lambda_5 = \frac{-i}{3}[X(\mathbb{1} + \beta Z + \beta^2 Z^2) - X^2(\mathbb{1} + Z + Z^2)], \qquad \lambda_6 = \frac{1}{3}[X(\mathbb{1} + \beta^2 Z + \beta Z^2) + X^2(\mathbb{1} + \beta Z + \beta^2 Z^2)],$$
$$\lambda_7 = \frac{i}{3}[X(\mathbb{1} + \beta^2 Z + \beta Z^2) - X^2(\mathbb{1} + \beta Z + \beta^2 Z^2)], \qquad \lambda_8 = \frac{-1}{\sqrt{3}}(\beta Z + \beta^2 Z^2).$$

These can be easily verified using the definitions of X and Z and the relation $1 + \beta + \beta^2 = 0$. Further, the translation matrix U satisfying $\vec{\Sigma} = U\vec{\lambda}$, where $\vec{\Sigma}$ and $\vec{\lambda}$ are 8-vectors of the ZX basis matrices and Gell-Mann matrices (excluding $\mathbb{1}$ and λ_0), respectively, is given by

$$U_{ij} = \frac{1}{2}\mathrm{Tr}(\Sigma_i \lambda_j). \tag{A2}$$

[1] M. Cerezo, K. Sharma, A. Arrasmith, and P. J. Coles, Variational quantum state eigensolver, npj Quantum Inf. **8**, 113 (2022).

[2] X. Yuan, S. Endo, Q. Zhao, Y. Li, and S. C. Benjamin, Theory of variational quantum simulation, Quantum **3**, 191 (2019).

[3] H. Duan, G. M. Fuller, and Y.-Z. Qian, Collective neutrino oscillations, Annu. Rev. Nucl. Part. Sci. **60**, 569 (2010).

[4] S. Chakraborty, R. Hansen, I. Izaguirre, and G. Raffelt, Collective neutrino flavor conversion: Recent developments, Nucl. Phys. **B908**, 366 (2016).

[5] A. B. Balantekin, Symmetries and algebraic methods in neutrino physics, J. Phys. G **45**, 113001 (2018).

[6] I. Tamborra and S. Shalgar, New developments in flavor evolution of a dense neutrino gas, Annu. Rev. Nucl. Part. Sci. **71**, 165 (2021).

[7] M. C. Volpe, Neutrinos from dense environments: Flavor mechanisms, theoretical approaches, observations, and new directions, Rev. Mod. Phys. **96**, 025004 (2024).

[8] A. B. Balantekin, M. J. Cervia, A. V. Patwardhan, E. Rrapaj, and P. Siwach, Quantum information and quantum simulation of neutrino physics, Eur. Phys. J. A **59**, 186 (2023).

[9] L. Johns, S. Richers, and M.-R. Wu, Neutrino oscillations in core-collapse supernovae and neutron star mergers, Annu. Rev. Nucl. Part. Sci. **75**, 399 (2025).

[10] M. J. Cervia, A. V. Patwardhan, A. B. Balantekin, S. N. Coppersmith, and C. W. Johnson, Entanglement and collective flavor oscillations in a dense neutrino gas, Phys. Rev. D **100**, 083001 (2019).

[11] M. J. Cervia, P. Siwach, A. V. Patwardhan, A. B. Balantekin, S. N. Coppersmith, and C. W. Johnson, Collective neutrino oscillations with tensor networks using a time-dependent variational principle, Phys. Rev. D **105**, 123025 (2022).

[12] D. Lacroix, A. B. Balantekin, M. J. Cervia, A. V. Patwardhan, and P. Siwach, Role of non-Gaussian quantum fluctuations in neutrino entanglement, Phys. Rev. D **106**, 123006 (2022).

[13] A. V. Patwardhan, M. J. Cervia, E. Rrapaj, P. Siwach, and A. B. Balantekin, Many-body collective neutrino oscillations: Recent developments, in *Handbook of Nuclear*

Physics, edited by I. Tanihata, H. Toki, and T. Kajino (Springer Nature, Singapore, 2023), pp. 1–16.
[14] M. Illa and M. J. Savage, Multi-neutrino entanglement and correlations in dense neutrino systems, Phys. Rev. Lett. **130**, 221003 (2023).
[15] J. D. Martin, D. Neill, A. Roggero, H. Duan, and J. Carlson, Equilibration of quantum many-body fast neutrino flavor oscillations, Phys. Rev. D **108**, 123010 (2023).
[16] J. D. Martin, A. Roggero, H. Duan, J. Carlson, and V. Cirigliano, Classical and quantum evolution in a simple coherent neutrino problem, Phys. Rev. D **105**, 083020 (2022).
[17] J. D. Martin, A. Roggero, H. Duan, and J. Carlson, Many-body neutrino flavor entanglement in a simple dynamic model, arXiv:2301.07049.
[18] D. Lacroix, A. Bauge, B. Yilmaz, M. Mangin-Brinet, A. Roggero, and A. B. Balantekin, Phase-space methods for neutrino oscillations: Extension to multibeams, Phys. Rev. D **110**, 103027 (2024).
[19] J. Carlson, A. Roggero, and D. Neill, Neutrino flavor evolution in high flux astrophysical environments, arXiv:2603.12192.
[20] D. Neill, H. Liu, J. Martin, and A. Roggero, Scattering neutrinos, spin models, and permutations, Phys. Rev. Res. **7**, 023157 (2025).
[21] I. Chernyshev, C. E. P. Robin, and M. J. Savage, Quantum magic and computational complexity in the neutrino sector, Phys. Rev. Res. **7**, 023228 (2025).
[22] R. Bhaskar, A. Roggero, and M. J. Savage, Timescales in many-body fast-neutrino-flavor conversion, Phys. Rev. C **110**, 045801 (2024).
[23] O. Kiss, I. Tavernelli, F. Tacchino, D. Lacroix, and A. Roggero, Neutrino thermalization via randomization on a quantum processor, arXiv:2510.24841.
[24] Z. Laraib and S. Richers, Many-body simulations of the fast flavor instability, Phys. Rev. D **112**, L101304 (2025).
[25] Z. Laraib and S. Richers, Two-beam multiparticle many-body simulations of inhomogeneous fast flavor instabilities, Phys. Rev. D **112**, 123045 (2025).
[26] K. Yeter-Aydeniz, S. Bangar, G. Siopsis, and R. C. Pooser, Collective neutrino oscillations on a quantum computer, Quantum Inf. Process. **21**, 84 (2022).
[27] V. Amitrano, A. Roggero, P. Luchi, F. Turro, L. Vespucci, and F. Pederiva, Trapped-ion quantum simulation of collective neutrino oscillations, Phys. Rev. D **107**, 023007 (2023).
[28] B. Hall, A. Roggero, A. Baroni, and J. Carlson, Simulation of collective neutrino oscillations on a quantum computer, Phys. Rev. D **104**, 063009 (2021).
[29] P. Siwach, K. Harrison, and A. B. Balantekin, Collective neutrino oscillations on a quantum computer with hybrid quantum-classical algorithm, Phys. Rev. D **108**, 083039 (2023).
[30] F. Turro, I. A. Chernyshev, R. Bhaskar, and M. Illa, Qutrit and qubit circuits for three-flavor collective neutrino oscillations, Phys. Rev. D **111**, 043038 (2025).
[31] P. Siwach, A. M. Suliga, and A. B. Balantekin, Entanglement in three-flavor collective neutrino oscillations, Phys. Rev. D **107**, 023019 (2023).
[32] A. B. Balantekin and A. M. Suliga, On the properties of qudits, Eur. Phys. J. A **60**, 124 (2024).
[33] M. Mangin-Brinet, A. Bauge, and D. Lacroix, Three-flavor neutrino oscillations using the phase space approach, Phys. Rev. D **113**, 036026 (2026).
[34] L. Spagnoli, N. Goss, A. Roggero, E. Rrapaj, M. J. Cervia, A. V. Patwardhan, R. K. Naik, A. B. Balantekin, E. Younis, D. I. Santiago, I. Siddiqi, and S. Aldaihan, Collective neutrino oscillations in three flavors on qubit and qutrit processors, Phys. Rev. D **111**, 103054 (2025).
[35] I. Chernyshev, C. E. P. Robin, and M. J. Savage, Quantum magic and computational complexity in the neutrino sector, Phys. Rev. Res. **7**, 023228 (2025).
[36] G. Martínez-Pinedo, T. Fischer, K. Langanke, A. Lohs, A. Sieverding, and M.-R. Wu, Neutrinos and their impact on core-collapse supernova nucleosynthesis, in *Handbook of Supernovae* (Springer International Publishing, 2017), pp. 1805–1841, 10.1007/978-3-319-21846-5_78.
[37] A. B. Balantekin, M. J. Cervia, A. V. Patwardhan, R. Surman, and X. Wang, Collective neutrino oscillations and heavy-element nucleosynthesis in supernovae: Exploring potential effects of many-body neutrino correlations, Astrophys. J. **967**, 146 (2024).
[38] L. Choi, A. Burrows, and D. Vartanyan, Predicted neutrino signal features of core-collapse supernovae, Phys. Rev. D **111**, 123038 (2025).
[39] A. Mirizzi, I. Tamborra, H.-T. Janka, N. Saviano, K. Scholberg, R. Bollig, L. Hudepohl, and S. Chakraborty, Supernova neutrinos: Production, oscillations and detection, Riv. Nuovo Cimento **39**, 1 (2016).
[40] T. Pitik, D. J. Heimsoth, A. M. Suliga, and A. B. Balantekin, Exploiting stellar explosion induced by the QCD phase transition in large-scale neutrino detectors, Phys. Rev. D **106**, 103007 (2022).
[41] G. M. Fuller and W. C. Haxton, *Neutrinos in Stellar Astrophysics* (2022), 10.1142/9789811282645_0008.
[42] H. Duan, G. M. Fuller, J. Carlson, and Y.-Z. Qian, Simulation of coherent non-linear neutrino flavor transformation in the supernova environment. 1. Correlated neutrino trajectories, Phys. Rev. D **74**, 105014 (2006).
[43] Y. Z. Qian and G. M. Fuller, Neutrino-neutrino scattering and matter enhanced neutrino flavor transformation in supernovae, Phys. Rev. D **51**, 1479 (1995).
[44] N. F. Bell, A. A. Rawlinson, and R. F. Sawyer, Speedup through entanglement: Many body effects in neutrino processes, Phys. Lett. B **573**, 86 (2003).
[45] A. Friedland, B. H. J. McKellar, and I. Okuniewicz, Construction and analysis of a simplified many-body neutrino model, Phys. Rev. D **73**, 093002 (2006).
[46] Y. Pehlivan, A. B. Balantekin, and T. Kajino, Neutrino magnetic moment, *CP* violation and flavor oscillations in matter, Phys. Rev. D **90**, 065011 (2014).
[47] S. Birol, Y. Pehlivan, A. B. Balantekin, and T. Kajino, Neutrino spectral split in the exact many body formalism, Phys. Rev. D **98**, 083002 (2018).
[48] P. Siwach, A. B. Balantekin, A. V. Patwardhan, and A. M. Suliga, Exploring entanglement and spectral split correlations in three-flavor collective neutrino oscillations, Phys. Rev. D **111**, 063038 (2025).

[49] D. C. McKay, C. J. Wood, S. Sheldon, J. M. Chow, and J. M. Gambetta, Efficient Z gates for quantum computing, Phys. Rev. A **96**, 022330 (2017).
[50] T. Tiwari, S. K. Sahu, G. Ribeill, M. Senatore, M. D. LaHaye, R. W. Simmonds, D. L. Campbell, A. Kamal, and L. Ranzani, High-fidelity iSWAP gate with double transmon coupler, arXiv:2604.27080.
[51] W. Huang, C. H. Yang, K. W. Chan, T. Tanttu, B. Hensen, R. C. C. Leon, M. A. Fogarty, J. C. C. Hwang, F. E. Hudson, K. M. Itoh, A. Morello, A. Laucht, and A. S. Dzurak, Fidelity benchmarks for two-qubit gates in silicon, Nature (London) **569**, 532 (2019).
[52] F. Pfeiffer *et al.*, A high-fidelity two-qubit gate for multi-mode superconducting P-mon qubits, arXiv:2606.24772.
[53] G. Vidal, Efficient classical simulation of slightly entangled quantum computations, Phys. Rev. Lett. **91**, 147902 (2003).
[54] M. Howard, E. Brennan, and J. Vala, Quantum contextuality with stabilizer states, Entropy **15**, 2340 (2013).
[55] S. X. Cui and Z. Wang, Universal quantum computation with metaplectic anyons, J. Math. Phys. (N.Y.) **56**, 032202 (2015).
[56] Q. Wang, Qutrit zx-calculus is complete for stabilizer quantum mechanics, Electron. Proc. Theor. Comput. Sci. **266**, 58 (2018).
[57] F. Turro, I. A. Chernyshev, R. Bhaskar, and M. Illa, Qutrit and qubit circuits for three-flavor collective neutrino oscillations, Phys. Rev. D **111**, 043038 (2025).
[58] K. Bharti and T. Haug, Quantum-assisted simulator, Phys. Rev. A **104**, 042418 (2021).
[59] A. Luongo, Quantum algorithms for data analysis, https://quantumalgorithms.org/ (June 24, 2026).
[60] W. Magnus, On the exponential solution of differential equations for a linear operator, Commun. Pure Appl. Math. **7**, 649 (1954).
[61] T. de Souza Farias, L. Friedrich, and J. Maziero, QuForge: A library for qudits simulation, Comput. Phys. Commun. **314**, 109687 (2025).
[62] P. Virtanen *et al.* (SciPy 1.0 Contributors), SciPy 1.0: Fundamental algorithms for scientific computing in Python, Nat. Methods **17**, 261 (2020).