Real-Time Dynamics in Two Dimensions with Tensor Network States via Time-Dependent Variational Monte Carlo

Yantao Wu[1, *]

[1]*Institute of Physics, Chinese Academy of Sciences, Beijing 100190, China*
(Dated: December 11, 2025)

Reliably simulating two-dimensional many-body quantum dynamics with projected entangled pair states (PEPS) has long been a difficult challenge. In this work, we overcome this barrier for low-energy quantum dynamics by developing a stable and efficient time-dependent variational Monte Carlo (tVMC) framework for PEPS. By analytically removing all gauge redundancies of the PEPS manifold and exploiting tensor locality, we obtain a numerically well-conditioned stochastic reconfiguration equation amenable to robust solution using the efficient Cholesky decomposition, enabling long-time evolution in previously inaccessible regimes. We explain how the difficulties associated with the traditional PEPS algorithms for dynamics are naturally resolved within tVMC. We demonstrate the power and generality of the method through four representative real-time problems in two dimensions: (I) chiral edge propagation in a free-fermion Chern insulator; (II) fractionalized charge transport in a fractional Chern insulator; (III) vison confinement dynamics in the Higgs phase of a $\mathbb{Z}_2$ lattice gauge theory coupled to a Higgs field; and (IV) superfluidity and critical velocity in interacting bosons. All simulations are performed on 12×12 or 13×13 lattices with evolution times $T = 10$ to 12 using modest computational resources — 1 to 5 days on a single GPU card. Where exact benchmarks exist (case I), PEPS–tVMC matches free-fermion dynamics with high accuracy up to $T = 12$. These results establish PEPS–tVMC as a practical and versatile tool for real-time quantum dynamics in two dimensions. The method significantly extends the reach of classical tensor-network simulations for studying elementary excitations in quantum many-body systems and provides a valuable computational counterpart to emerging quantum simulators.

I. INTRODUCTION

Understanding real-time dynamics of interacting quantum many-body systems in two dimensions (2D) is a fundamental challenge that spans condensed-matter physics, particle physics, and quantum simulation. While powerful classical methods exist for ground states—including tensor networks [1, 2] and quantum Monte Carlo [3, 4], real-time evolution in 2D remains notoriously difficult. Exact diagonalization is limited to small clusters; quantum Monte Carlo suffers from a dynamical sign problem. Recently, the combination of time-dependent variational Monte Carlo (tVMC) [5–10] with neural quantum states (NQS) has emerged as a promising alternative [11–17]. NQS provide remarkable variational flexibility, while tVMC offers a principled formulation of real-time evolution. At present, extending NQS–tVMC to large 2D systems remains an active area of research, where one must balance expressive parametrizations with efficiency and stable optimization [18].

Projected entangled pair states (PEPS) [19] provide a complementary variational family whose locality and multi-linearity make them particularly appealing for tVMC. Within VMC, only a single layer of PEPS tensors needs to be contracted, making

the method substantially more efficient than conventional PEPS update schemes [20–22]. Furthermore, VMC allows parallel accumulation of samples, making the approach well suited for modern GPU architectures. In fact, PEPS-VMC has recently achieved considerable success for ground states, giving state-of-the-art results for the ground states of frustrated spins [23–25], fermion Hubbard model [26], and lattice gauge theory [27], owing to a foundational breakthrough in the sampling method of PEPS [28].

Although PEPS obey an area law for entanglement entropy [29] and cannot represent volume-law states produced by fully thermalizing dynamics, they are well suited for *local quenches* in which the injected energy does not scale with system size. In this regime, the energy density remains zero (in the thermodynamic limit) and we expect the dynamics to preserve area-law entanglement [30]. Thus PEPS naturally targets low-energy local-quench dynamics — precisely the regime relevant for elementary excitations and real-time response functions, and directly related to experiments including scanning tunneling microscopy [31], angle resolved photoemission spectroscopy [32], particle scattering processes, and more. A controlled method capable of simulating such dynamics provides a valuable tool for studying 2D quantum systems.

While dynamical simulations with matrix product states (MPS) are enormously successful in 1D [33–36], the situation in 2D is far more challeng-

ing. In contrast to the extensive 1D literature, only a small number of works—represented by Refs. [37–46], have reported real-time simulations using PEPS or iPEPS, illustrating the general difficulty of the problem. Traditional approaches to dynamical PEPS have relied on full-update-like methods based on double-layer contraction [20–22], which face several challenges: (a) high cost of double layer contraction, (b) difficulty in fixing the PEPS gauge freedom, and (c) loss of positivity in the norm matrix approximately obtained from the contraction of the double-layer environment. We explain, in Sec. III D, that the last two issues are naturally resolved while the first one is significantly alleviated within PEPS–tVMC, at the expense of manageable statistical noise.

Despite the long development of PEPS [1], their combination with tVMC for real-time dynamics has received little attention, aside from our recent exploratory attempt in a simple setting [27]. Here, we develop PEPS–tVMC systematically, analyze its structure, and demonstrate its ability to simulate large-scale 2D dynamics. Our approach builds on stochastic reconfiguration (SR) [5], identifies and removes the gauge redundancy of PEPS parameters, and exploits tensor locality to simplify and precondition the SR equations. After gauge removal, the SR equation exhibits remarkable numerical stability: the SR residual remains as low as 10^{-9} to 10^{-25} in the examples studied. The central message of this work is that by removing PEPS gauge freedom and exploiting locality, the tVMC equations become surprisingly stable to solve, and that, as a result, PEPS—long viewed as difficult for real-time evolution—can in fact provide a powerful and controllable variational framework for 2D dynamics. This resolves long-standing challenges that have hindered dynamical simulations with PEPS since their introduction [19].

We illustrate the versatility of PEPS–tVMC through four physically motivated examples. First, we simulate the propagation of chiral edge modes in a Chern insulator and visualize their unidirectional boundary motion. Second, we track quantized charge transport in a fractional quantum Hall state, revealing the real-time flow of fractional charge. Third, we study vison dynamics in the Higgs phase of a $\mathbb{Z}_2$ lattice gauge theory, observing vison confinement in real time. Finally, we investigate superflow in interacting bosons around a local obstacle and observe signatures consistent with a critical velocity. All simulations are performed on 12×12 or 13×13 lattices with evolution times $T = 10$ to 12, using a single GPU card. These examples illustrate that the method is not specialized to a single model class.

The paper is organized as follows. In Sec. II, we review the tVMC framework and set up the notation. In Sec. III, we analyze the structure of PEPS–tVMC and present the ingredients necessary for its stability and efficiency. In Sec. IV, we demonstrate the method through four representative examples. In Sec. V, we summarize our findings and outline future directions.

II. TIME-DEPENDENT VARIATIONAL MONTE CARLO

Consider a many-body lattice quantum system with computational basis $\mathbf{s} = (s_1, \ldots, s_N)$. To perform a variational simulation, we adopt a holomorphic parametrization of the *unnormalized* quantum state $\Psi_{\boldsymbol{\theta}}(\mathbf{s})$ with N_p complex variational parameters $\boldsymbol{\theta} = \{\theta_\alpha : \alpha = 1, 2, \ldots, N_p\}$. At time t, the exact evolution under a Hamiltonian H for a short interval dt induces an evolution from $\boldsymbol{\theta}(t)$ to $\boldsymbol{\theta}(t + dt)$ that minimizes the Fubini–Study distance [47] between $\Psi_{\boldsymbol{\theta}(t+dt)}$ and $e^{-iHdt}\Psi_{\boldsymbol{\theta}(t)}$. This yields the SR equation:

$$\sum_{\beta=1}^{N_p} S_{\alpha\beta}\, \dot{\theta}_\beta = -i\, g_\alpha, \tag{1}$$

where the SR matrix $S_{\alpha\beta}$ and the energy gradient g_α are given by

$$S_{\alpha\beta} = [O_\alpha^*(\mathbf{s})O_\beta(\mathbf{s})] - [O_\alpha^*(\mathbf{s})][O_\beta(\mathbf{s})], \tag{2}$$

$$g_\alpha = [O_\alpha^*(\mathbf{s})\, H_{\mathrm{loc}}(\mathbf{s})] - [O_\alpha^*(\mathbf{s})][H_{\mathrm{loc}}(\mathbf{s})]. \tag{3}$$

Here $[\cdot]$ denotes a statistical average over configurations drawn from $P_{\boldsymbol{\theta}}(\mathbf{s}) = |\Psi_{\boldsymbol{\theta}}(\mathbf{s})|^2 / \sum_{\mathbf{s}} |\Psi_{\boldsymbol{\theta}}(\mathbf{s})|^2$, $H_{\mathrm{loc}}(\mathbf{s}) = \langle \mathbf{s}|H|\Psi\rangle / \langle \mathbf{s}|\Psi\rangle$ is the local energy, and

$$O_\alpha(\mathbf{s}) \equiv \frac{\partial \log \Psi_{\boldsymbol{\theta}}(\mathbf{s})}{\partial \theta_\alpha} \tag{4}$$

is the log-derivative.

In practice, S and $\mathbf{g}$ are evaluated stochastically by sampling configurations $\{\mathbf{s}\}$ from $P_{\boldsymbol{\theta}}$. Given N_s samples, we form an $N_s \times N_p$ matrix O with elements $O_{s\alpha} = O_\alpha(\mathbf{s})$ and an N_s column vector h with entries $h_s = H_{\mathrm{loc}}(\mathbf{s})$. Subtracting the sample average O_{avg} and h_{avg},

$$\mathscr{O} = O - O_{\mathrm{avg}}, \qquad \hbar = h - h_{\mathrm{avg}}, \tag{5}$$

the SR matrix and the energy gradient can be approximated as

$$S \approx \frac{1}{N_s}\, \mathscr{O}^\dagger \mathscr{O}, \qquad \mathbf{g} \approx \frac{1}{N_s}\, \mathscr{O}^\dagger \hbar, \tag{6}$$

where the approximation reflects finite-sample statistical fluctuations. The centering in Eq. 5 removes

the component parallel to the wavefunction, ensuring projective invariance of the SR metric.

The tVMC algorithm therefore consists of: (1) sampling configurations $\mathbf{s} \sim P_{\boldsymbol{\theta}}$; (2) constructing S and $\mathbf{g}$; (3) solving Eq. (1) for $\dot{\boldsymbol{\theta}}$; and (4) integrating the parameter flow on the variational manifold via a Runge–Kutta method [48].

A major challenge in tVMC is that the SR matrix S is often severely ill-conditioned, making Eq. (1) difficult to solve stably. For ground-state searches, where the precise time trajectory is irrelevant, one may add a diagonal regulator $S \to S + \epsilon I$ with $\epsilon \sim 10^{-3}$. However, in real-time dynamics a large ϵ introduces an unphysical bias, whereas too small a value leaves the system unstable. In NQS-tVMC, several sophisticated regularization schemes have been proposed [13, 16], but none fully resolves the problem [18].

In PEPS, the situation is complicated by gauge redundancy, which introduces many null vectors into O and hence into S. The crucial feature, however, is that these null directions are *analytically identifiable*. As we will show in Sec. III, they can be removed *before* solving the SR equation, providing an exact preconditioning. After this gauge-removal step, the SR equation becomes remarkably well conditioned, and in all examples studied we find it can be solved with exceptional numerical accuracy. This stability is a key factor of the success of PEPS–tVMC.

III. STOCHASTIC RECONFIGURATION WITH PEPS

In this section we derive the SR equations for PEPS, analyze the structure of their null space, and describe how gauge redundancy can be eliminated efficiently in closed form, in both the SR and its reduced variant minSR [49] form.

A PEPS defines a variational wavefunction $\Psi_{\boldsymbol{\theta}}(\mathbf{s})$ parametrized by the tensor elements $A[\mathbf{x}]^p_{lrdu}$ at each lattice site $\mathbf{x}$, as illustrated in Fig. 1(a,b). Each index tuple $(\mathbf{x}, p, l, r, d, u)$ corresponds to one variational parameter, replacing the abstract index α introduced earlier. The physical index p labels the local Hilbert-space configuration (e.g., $p = 0, 1$ for vacuum or occupation), while the virtual indices (l, r, d, u) take values from 1 to D (except on boundaries).

a. Gauge redundancy. The PEPS representation contains a large local gauge freedom associated with internal virtual bonds. As shown in Fig. 1(c),
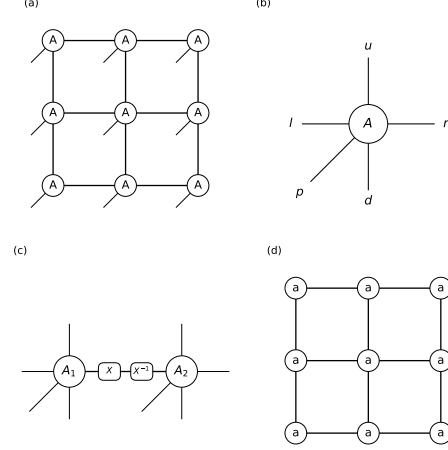

FIG. 1: PEPS structure and gauge freedom. (a) A 3×3 PEPS network with local tensors A. (b) A single tensor A with physical index p and virtual indices (l, r, d, u). (c) Local gauge transformation on a virtual bond: inserting an invertible matrix X and its inverse X^{-1} between two adjacent tensors leaves the global PEPS wavefunction invariant. (d) Wavefunction amplitude $\langle \mathbf{s}|\Psi \rangle$, computed via the contraction of the single layer network of sliced PEPS tensors: $a[\mathbf{x}]_{lrdu} = A[\mathbf{x}]^{p=\mathbf{s}(\mathbf{x})}_{lrdu}$.

the transformation

$$A_{1\,lrdu}^{p} \to \sum_{r'=1}^{D} A_{1\,lr'du}^{p} X_{r'r}, \qquad (7)$$

$$A_{2\,lrdu}^{p} \to \sum_{l'=1}^{D} (X^{-1})_{ll'} A_{2\,l'rdu}^{p}, \qquad (8)$$

leaves the global wavefunction invariant for any invertible $D \times D$ matrix X. The infinitesimal generators of such transformations define *gauge vectors*. Each gauge vector $\mathbf{v}$ is a null vector of O:

$$\sum_{\alpha} O_{s\alpha} v_{\alpha} = 0 \qquad \forall\, s.$$

Hence every gauge vector also lies in the null space of $S = \mathcal{O}^{\dagger} \mathcal{O}$, and must be removed before solving the SR equation

For an $L \times L$ PEPS there are $2L(L-1)$ internal bonds, yielding $2L(L-1)D^2$ gauge vectors. Let E^{ij} be the standard basis of $\mathbb{C}^{D \times D}$ with a single nonzero element at (i, j). For the bond in Fig. 1(c), the corresponding gauge vector $\mathbf{v}$ has components

$$\mathbf{v}[\mathbf{x}_1]^p_{lrdu} = \sum_{r'} A_{1\,lr'du}^{p} E^{ij}_{r'r}, \qquad (9)$$

$$\mathbf{v}[\mathbf{x}_2]^p_{lrdu} = -\sum_{l'} E^{ij}_{ll'} A_{2\,l'rdu}^{p}, \qquad (10)$$

and vanishes for all other parameters.

b. Linear dependence of gauge vectors. The $2L(L-1)D^2$ gauge vectors are not linearly independent. Consider a closed loop of virtual bonds and take E^{ij} to be the identity matrix on each bond. The sum of the corresponding gauge vectors vanishes, indicating linear dependence. These relations are generated by the $(L-1)^2$ elementary plaquettes of the square lattice. We denote the associated constraint vectors by $\mathbf{w}$, which are known exactly in the basis of the gauge vectors and independent of tensor content.

c. Additional null vectors. In addition to gauge redundancy, two further null transformations of $\mathscr{O}$ exist: (i) a global rescaling $A \to aA$, reflecting the projective nature of quantum mechanics; and (ii) for fixed particle number M, a relative rescaling $A^{p=0} \to aA^{p=0}$ and $A^{p=1} \to bA^{p=1}$ with $a^{L^2-M}b^M = 1$. We denote their generators as $\mathbf{u}_1$ and $\mathbf{u}_2$. Note that $\mathbf{u}_1$ is not a null direction of O, but only of $\mathscr{O}$.

A. Gauge-fixing within SR

We now outline the procedure for eliminating null vectors within the SR formulation. First, we construct a projection matrix P onto the subspace orthogonal to the constraint vectors $\mathbf{w}$, thereby reducing the number of independent gauge vectors to

$$N_{\mathrm{gv}} = 2L(L-1)D^2 - (L-1)^2.$$

We then form an $N_p \times (N_{\mathrm{gv}} + 2)$ matrix T whose columns consist of these reduced gauge vectors together with $\mathbf{u}_1$ and $\mathbf{u}_2$. Performing a *complete* QR decomposition $T = \widetilde{Q}R$, we define Q as the last $N_p - N_{\mathrm{gv}} - 2$ columns of $\widetilde{Q}$. Since T has full column rank, the remaining columns of $\widetilde{Q}$ span the orthogonal complement to the column space of T. By construction, Q projects onto the subspace orthogonal to all null directions.

The SR equation is then solved in this reduced space:

$$Q^\dagger \mathscr{O}^\dagger \mathscr{O} Q\, y = -i\, Q^\dagger \mathscr{O}^\dagger \hbar, \qquad (11)$$

and the parameter update is obtained as $\dot{\boldsymbol{\theta}} = Qy$. Only a single QR decomposition is required, which is numerically stable and GPU efficient.

B. Gauge-fixing within minSR

When the number of samples is less than the column rank of $\mathscr{O}$, i.e. $N_s < N_p - N_{\mathrm{gv}} - 2$, the SR

equation can be equivalently solved in the reduced minSR form [49]:

$$\mathscr{O}\,\mathscr{O}^\dagger y = \hbar, \qquad \dot{\boldsymbol{\theta}} = \mathscr{O}^\dagger y. \qquad (12)$$

We observe that the PEPS gauge redundancy does *not* appear at the minSR level, because the parameter index is contracted away, thus gauge-fixing is automatically achieved. Assuming decorrelation between samples, the matrix $\mathscr{O}^\dagger$ has a single null vector $[1, 1, \ldots, 1]^T$, which we remove with the same QR projection described above.

More generally, the null vector of $\mathscr{O}^\dagger$ is the weight vector W used in performing the statistical average: $\langle A \rangle \approx \sum_{s=1}^{N_s} W_s A_s / N_s$, which in our case is simply the identity weight. A non-trivial weight vector would arise if reweighing is used in the Monte Carlo estimate. The removal of this null vector may be also benificial to NQS-tVMC in the minSR form.

Finally, we comment on the choice between SR and minSR. As the system size gets larger, N_p eventually overtakes N_s and one should use minSR for efficiency. However, it is always advised to study a problem first at small size and ideally benchmark against exact methods, in which case typically $N_s > N_p$ and it is only correct to use SR.

C. Small-o trick in minSR

In the minSR case, a small-o trick can be specifically applied to PEPS to alleviate the memory cost in storing the full matrix O. The key observation is that the *locality* of PEPS forces most elements $O_{s\alpha}$ to vanish. For a given sample configuration $\mathbf{s}$,

$$O_{s\alpha} = \frac{\partial \log \Psi(\mathbf{s})}{\partial A[\mathbf{x}]^p_{lrdu}} = 0 \quad \text{if } p \neq s(\mathbf{x}). \qquad (13)$$

Thus each site contributes only the physical slice compatible with the sample. We therefore define the "small-o" object

$$o[\mathbf{x}]_{lrdu}(\mathbf{s}) \equiv O[\mathbf{x}]_{lrdu}^{p=s(\mathbf{x})}(\mathbf{s}), \qquad (14)$$

from which the full minSR matrix $\mathscr{O}\mathscr{O}^\dagger$ can be reconstructed with the memory cost of o instead of O (see the Supplementary Material [50] for the explicit GPU-efficient construction).

In tVMC, the memory bottleneck is storing O, which can be a serious issue on GPUs. For a PEPS with local dimension d, this "small-o" trick reduces memory cost by a factor of d. For the $\mathbb{Z}_N$ gauge-invariant PEPS used in Sec. IV C, the reduction is very substantial—a factor of dN^3. In practice, this saving often determines whether a large-scale GPU simulation is doable at all.

D. Comparison with double-layer methods

We briefly discuss the difficulties of double-layer methods mentioned in the introduction: computational cost, gauge fixing, and positivity of the norm matrix. Although manifested differently, they reflect the same underlying difficulty.

a. Computational cost. Consider a PEPS with bond dimension D. Let D' denote the bond dimension of the boundary MPS used to contract a single-layer PEPS in VMC, cf. Fig. 1 (d), and likewise the boundary matrix product operator used in full-update double-layer contraction. The dominant computational costs scale as $D^4 D'^2$ for VMC [28] and $D^4 D'^3$ for full update [22]. Typical choices are $D' \sim D$ to D^2 for VMC, and $D' \sim D^2$ to D^3 for full update, leading to effective complexities of D^6–D^8 and D^{10}–D^{13}, respectively. This gap is substantial. VMC additionally incurs sampling overhead, but even choosing D' far larger than D^3 does not reliably stabilize the full-update procedure due to the next two issues.

b. Gauge fixing. Unlike MPS in 1D [1], PEPS does not possess a canonical form that fixes the gauge freedoms. Ref. [22] introduces local gauge fixing for full-update gate applications, but no globally consistent gauge-fixing procedure is known. In fact, as an alternative, a variational subfamily — isometric PEPS (isoPEPS) [41, 46, 51] — is introduced to explicitly preserve the isometric gauge at the expense of representative expressiveness. However, the necessary shifting of isometric centers introduces significant errors and leads to suboptimal results. In contrast, PEPS–tVMC achieves global gauge fixing efficiently through a single `QR` projection or simply using the minSR formulation, without any loss of representational power.

c. Normalization. Because PEPS ψ is an unnormalized ansatz, to perform a variational update, one needs to also form the norm matrix N accounting for the norm $\langle \psi | \psi \rangle$. In practice, approximate contraction leads to violations of positivity of N, which destabilizes the generalized eigenvalue problems used in variational updates, especially in time evolution. Tiny errors in the boundary contraction can lead to large deviations in the time trajectory. tVMC avoids this issue entirely: normalization is enforced simply by subtracting O_{avg} and h_{avg} in Eq. 5, requiring no explicit norm matrix.

In the end, we mention that any conservation law in the computational basis can be implemented, in VMC, by sampling in the subsector of configurations with a definite total charge. This is equivalent to acting to the state a global projector of the desired charge sector, which significantly increases the representability of the underlying bare PEPS.

IV. RESULTS

We now apply PEPS–tVMC to study four distinct two-dimensional models—bosonic and fermionic, free and interacting, and including systems with gauge symmetry or nontrivial topology. PEPS configurations are sampled using the sequential sampling algorithm, which was first introduced in Ref. [28], and later proved to satisfy detailed balance in Ref. [52]. The local energies and log-derivatives are computed via the contraction of the single-layer network shown in Fig. 1(d) (see Refs. [27, 28, 52] for methodological details). All simulation parameters are given in the Supplementary Material [50].

A. Free Fermion Chern insulator

As a first test, to obtain nontrivial exact benchmark, we consider a 12×12 free-fermion Hofstadter model with flux $\phi = 2\pi/3$,

$$H_{\mathrm{chern}} = -\sum_{x,y} c^\dagger_{x,y} c_{x+1,y} + e^{i\frac{2\pi x}{3}} c^\dagger_{x,y} c_{x,y+1} + \mathrm{h.c.}$$

We fix the particle number to $2/3$ filling, populating two of the three magnetic subbands. With open boundaries, the system hosts gapless chiral edge modes. Using fermion PEPS [52–54], we prepare the ground state in a weak attractive potential (strength -1) placed at the upper-left corner, which creates a localized edge-density bump. The system is then evolved in real time under H_{chern} with the potential removed.

As shown in Fig. 2, the excitation propagates unidirectionally along the edge. PEPS–tVMC faithfully reproduces the edge motion, its speed, and its spatial profile, in excellent agreement with the exact free-fermion numerics. For context, isoPEPS simulations of a similar chiral edge motion (e.g., Fig.6(b) of Ref.[46]) already provided state-of-the-art results at the time; however, their accuracy was ultimately limited by the errors due to the shifting of isometric centers. The present PEPS–tVMC approach achieves noticeably higher accuracy. We further note that the numerical difficulty of tensor network simulations is largely dictated by the smaller of L_x and L_y. Thus, although the isoPEPS study used a 9×18 lattice, the 12×12 system considered here is in fact the more challenging geometry, making this benchmark particularly stringent.

Finally, we emphasize that the PEPS used in tVMC is entirely agnostic to the Gaussianity in the free fermion state. Thus, despite the model being noninteracting, its non-trivial dynamics makes a demanding test for PEPS, especially at this lattice size and time scale.
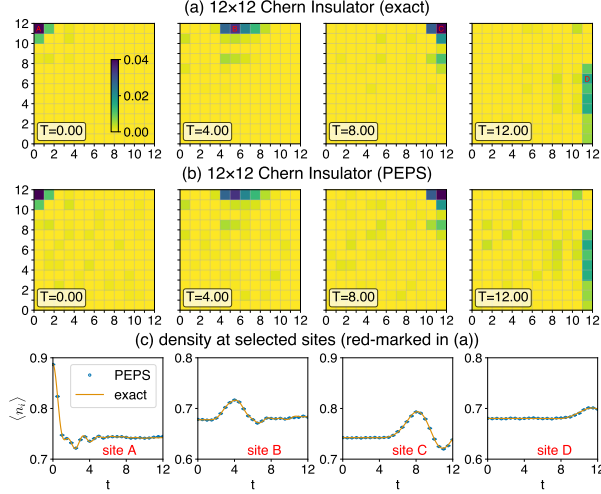
FIG. 2: Chiral edge dynamics in a 12×12 Chern insulator. (a) Exact real-time evolution obtained from free-fermion numerics. The color scale shows the density deviation $\langle n_i(t) \rangle - \langle n_i \rangle_{\mathrm{gs}}$. A density perturbation created near the boundary at $T = 0$ propagates unidirectionally along the edge at later times $T = 4, 8, 12$. (b) Same protocol computed using PEPS–tVMC. (c) Time evolution of the density at four selected sites red-marked in (a), chosen so that the chiral pulse passes them at $T = 0, 4, 8, 12$. The error bars are of order 7×10^{-4}. All panels share the same vertical scale of 0.2.

B. Bosonic fractional quantum Hall effect

Next we study the remarkable phenomena of fractional quantum Hall effect [55, 56]. We consider a bosonic Hofstadter model with flux $\phi = \pi/2$ and hardcore constraint, on a 12×12 lattice

$$H_{\mathrm{fqhe}} = -\sum_{x,y} b^{\dagger}_{x,y} b_{x+1,y} + e^{i\frac{\pi x}{2}} b^{\dagger}_{x,y} b_{x,y+1} + \mathrm{h.c.}$$

We consider the 1/8 density, where the lowest magnetic band is half filled, corresponding to filling $\nu = 1/2$. With the filling belonging to the Jain sequence [57], composite fermion theory [58, 59] predicts that the ground state to be a bosonic analogue of the Laughlin state [56] with quasi-particle charge 1/2. The lattice structure [60] also complicates the theory which is initially developed for the continuum. We now explicitly check the fractional transport in real-time.

To account for the finite boundary, in practice, we use particle number $(L_x - 1)(L_y - 1)/8 \approx 15$ [61]. As shown in Fig. 3 (a), the ground state VMC gives a charge distribution that is uniformly 1/8 in the bulk.
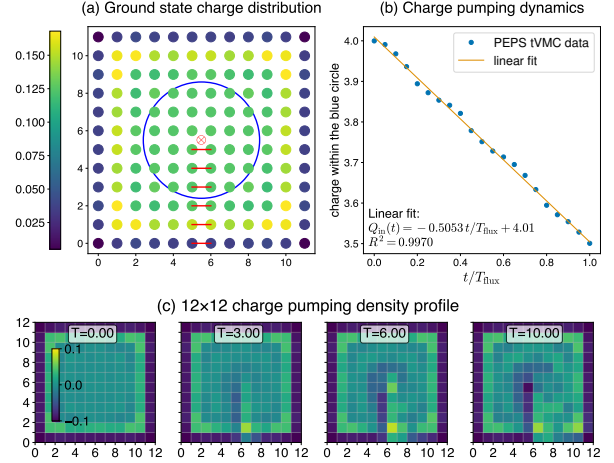

FIG. 3: Charge pumping in the bosonic Hofstadter model at flux $\phi = \pi/2$. (a) Ground-state charge distribution on a 12×12 lattice with total particle number 15. A localized external flux is threaded through the central plaquette (marked by $\otimes$). The red horizontal bonds indicate the Peierls-phase path along which the time-dependent vector potential $A_{ij}(t)$ is ramped. The blue circle denotes the region over which the enclosed charge Q_{in} is monitored. (b) Total charge within the blue circle during one flux-insertion cycle of duration $T_{\mathrm{flux}} = 10$. (c) Evolution of real-space density, subtracted by 1/8, during the pump.

To probe the fractional charge transport, we thread an external time-dependent flux $\Phi(t)$ through the central plaquette, cf. the red $\otimes$ in Fig. 3 (a), increasing linearly from 0 to 2π over a period T_{flux}. The external flux is implemented via a Peierls phase on the red horizontal bonds shown in Fig. 3 (a). This generates a steady electromotive force around any closed loop enclosing the plaquette and induces a Hall transport from the middle of the system to the edge. If the system is indeed a fractional quantum Hall state, a quantized charge of $\Delta Q = \nu = 1/2$ is expected to be transported per pumping cycle [62, 63]. Indeed, Fig. 3 (b) shows that PEPS–tVMC confirms this quantized transfer of charge. A linear fit of the charges within a central region, cf. the blue circle in Fig. 3 (a), versus t/T_{flux} gives a slope of 0.5053, close to the expected Hall response of 1/2.

The density profile during the pumping is shown in Fig. 3 (c). Because the flux is time-dependent, different Peierls paths are not gauge-equivalent at the level of time evolution, and our choice breaks the C_4 symmetry of the lattice. Although the instantaneous density flow seems structureless, the integrated pumped charge remains amazingly quantized, attesting to the robustness of the Hall response to

microscopic details of the system.

This result provides a direct real-time confirmation of the composite-fermion prediction of fractional quasiparticle charge. While previous numerical studies of fractional Chern insulators relied primarily on static diagnostics, such as entanglement spectra or density profiles of pinned quasiholes, PEPS–tVMC enables us to directly visualize the transport of fractional charge in two dimensions. This establishes a dynamical counterpart to the usual static signatures of topological order. It would be exciting to study other anyon dynamics with PEPS–tVMC in the future.

C. $\mathbb{Z}_2$ lattice gauge theory with Higgs field

Next, using the gauge-invariant PEPS–VMC framework recently introduced in Ref. [27], we examine the confinement dynamics in the $\mathbb{Z}_2$ gauge theory coupled to a Higgs field,

$$H_{\text{lgt}} = -\sum_i \sigma_i^z - \sum_p B_p - J\sum_l \sigma_{l_-}^x X_l \sigma_{l_+}^x - g\sum_l Z_l$$

subject to the Gauss-law constraint $\sigma_i^z \prod_{l \in i} Z_l = 1$. X_l, Z_l are the Pauli x, z matrices on the link l, representing the gauge field. σ_i^x, σ_i^z are the Pauli matrices on the site i, representing the Higgs field. $l_\pm$ are the sites at the ends of link l. Here $B_p = \prod_{l \in p} X_l$ measures the magnetic flux through plaquette p, and a violation of $B_p = 1$ defines a *vison* excitation. The system has three ground state phases: confined (small J and large g), Higgs (large J and small g), and deconfined (small J and small g) phases [64]. When $g = 0$, the phase transition point between the deconfined phase and the Higgs phase is at $J_c \approx 0.328$ [65]. In the Higgs phase, the visons are expected to have a linearly attractive interaction, i.e. vison confinement. In the deconfined phase, the visons are expected to propagate freely, i.e. vison deconfinement.

We create a pair of visons on top of the ground state in both the Higgs and the deconfined regimes, and simulate the subsequent dynamics under H_{lgt}, shown in Fig. 4. In the deconfined phase, the initially localized visons rapidly spread into extended flux clouds that delocalize across the lattice, consistent with free, deconfined excitations and the absence of a confining string. In the Higgs phase, the visons remain bound. The absence of spreading reflects a finite string tension and the confining nature of the Higgs phase.

Recently, considerable progress has been made in implementing real-time gauge-theory dynamics on quantum devices [66–68]. Our results highlight that classical tensor-network approaches can serve
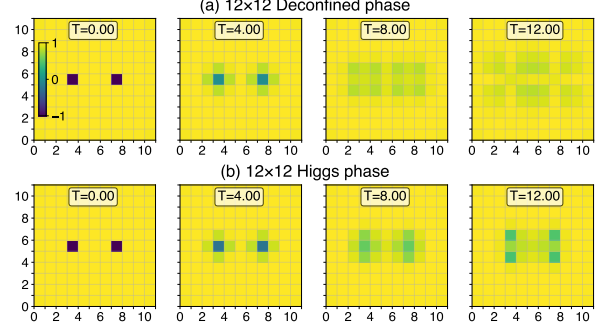

FIG. 4: Real-time vison dynamics in the $\mathbb{Z}_2$ lattice gauge theory on a 12×12 lattice. Shown are snapshots of the plaquette operator $B_p(t)$, which highlights magnetic-flux (vison) excitations created at $T = 0$. (a) Deconfined phase ($J = 0.1, g = 0.1$). (b) Higgs phase ($J = 0.5, g = 0.1$).

as a highly effective and complementary platform for these studies. It would not be impossible for PEPS–tVMC to probe the intricate dynamics of glueballs in the (2+1)D Yang-Mills theory, after the gauge-invariant PEPS–VMC [27] is extended to the non-Abelian case.

D. Superfluid bosons

Finally, we study real-time flow in a superfluid of hardcore bosons on a 13×13 lattice,

$$H_{\text{sf}} = -\sum_{\langle ij \rangle} b_i^\dagger b_j \tag{15}$$

at half-filling. This system is known to be in the superfluid phase due to Bose-Einstein condensation. Although its static properties are well understood from quantum Monte Carlo [69, 70], its real-time dynamics—including the onset of dissipation and the critical velocity—remain much less explored, especially in two dimensions.

Here we study the dynamics of the bosons in the presence of an obstacle, interacting via a repulsive potential:

$$H_{\text{obs}}(t) = 2\sum_i b_i^\dagger b_i \exp\left(-\frac{|\mathbf{r}_i - \mathbf{R}(t)|^2}{2}\right) \tag{16}$$

where $\mathbf{R}(t)$ is the position of the obstacle at time t. By Galilean invariance, we keep the superfluid at rest and drag the obstacle rightward at constant velocity: $\mathbf{R}(t) = \mathbf{R}_0 + \mathbf{v}t$. $H_{\text{obs}}(t)$ can also be realized via a laser in cold-atom simulations. The initial state is the ground state of $H_{\text{sf}} + H_{\text{obs}}(0)$. As shown in Fig.
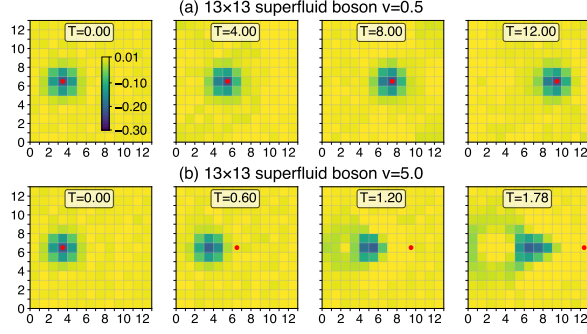
FIG. 5: Real-time density response of a 2D bosonic superfluid to a moving obstacle (the red dot). Snapshots show the deviation of the local density from 0.5 on a 13×13 lattice for different times T as the obstacle is dragged along the x direction. (a) $|\mathbf{v}| = 0.5$. (b) $|\mathbf{v}| = 5.0$.

5, for slow motion $|\mathbf{v}| = 0.5$, the density perturbation remains localized and approximately symmetric around the impurity, with no pronounced wake, consistent with dissipationless flow below the critical velocity. For fast motion $|\mathbf{v}| = 5.0$, the impurity leaves behind a pronounced density wake and radiates excitations throughout the system above the critical velocity. In particular, the density depletion caused by the obstacle closely follows the obstacle when $\mathbf{v}$ is small, while it lags far behind when $\mathbf{v}$ is large. This qualitative difference signals the presence of a critical superfluid velocity, as anticipated by Landau [71].

In the future, it would be exciting to apply PEPS–tVMC for the dynamical studies of phase coherence, vortex nucleation, and other hallmarks of superfluidity and superconductivity.

E. Error in solving the SR equation

An important factor underlying the accuracy and efficiency of PEPS–tVMC is the numerical stability with which the SR equation—or its reduced minSR variant—can be solved. All simulations in this work solve the SR equation using a Cholesky decomposition [72] of the SR matrix with a diagonal regulator of magnitude 10^{-8}. Compared to the rank-revealing singular value decomposition (SVD) solver, the Cholesky approach is substantially faster and more parallelizable on GPUs, but it lacks the robustness of SVD when applied to ill-conditioned matrices. Recent NQS–tVMC studies often resort to the SVD solver for this reason [13, 16, 18], trading computational efficiency for numerical robustness. The fact that Cholesky remains stable in all simulations indicates that once gauge redundancies

are analytically removed, the PEPS variational manifold becomes exceptionally stable for real-time local quench dynamics.
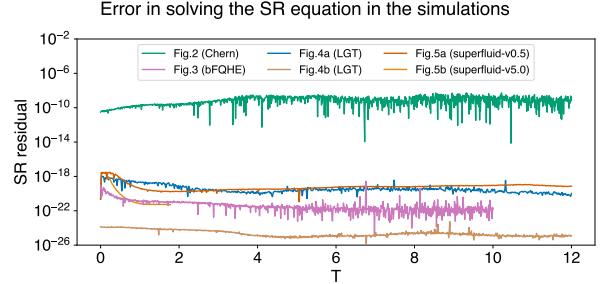

FIG. 6: Residual error $\varepsilon = \|S\dot{\boldsymbol{\theta}} - i\mathbf{g}\|^2 / \|\mathbf{g}\|^2$ during PEPS–tVMC time evolution for the models considered in this work: the fermionic Chern insulator (Fig. 2), the bosonic fractional quantum Hall effect (Fig. 3), the $\mathbb{Z}_2$ lattice gauge theory (Fig. 4), and the superfluid at velocities $|\mathbf{v}| = 0.5$ and 5 (Fig. 5).

To quantify the accuracy of the Cholesky solution, we monitor the SR residual with the regulator removed,

$$\varepsilon = \frac{\|S\dot{\boldsymbol{\theta}} - (-i\mathbf{g})\|^2}{\|\mathbf{g}\|^2}, \tag{17}$$

which measures the relative error in satisfying the SR equation. As shown in Fig. 6, the residual remains exceptionally small—typically between 10^{-9} and 10^{-25}, depending on the model. These results confirm the effectiveness of the preconditioning strategies introduced earlier, ensuring that the SR step does not contribute appreciable error during real-time evolution.

V. DISCUSSION AND OUTLOOK

In this work we have shown that projected entangled pair states, when combined with time-dependent variational Monte Carlo, provide a stable and efficient framework for simulating real-time dynamics in two-dimensional quantum many-body systems. The key technical insight is that the ill-conditioning traditionally associated with PEPS—stemming from gauge redundancy—can be removed analytically in closed form. Once gauge directions are eliminated, the SR equation can be solved efficiently with high numerical stability over long evolution times. This stability, together with the locality and single-layer contraction structure of PEPS within VMC, makes PEPS–tVMC a powerful clas-

sical method for studying low-energy dynamics in 2D.

The examples explored in Sec. IV demonstrate the breadth of physical phenomena accessible with this approach. These results establish PEPS–tVMC as a general-purpose tool for simulating elementary excitations, transport phenomena, and real-time response in a wide range of quantum systems with realistic computational resources. The improvements developed here apply equally to ground-state PEPS–VMC calculations [23–26]. Extensions to excited states, finite temperature [73], and open quantum systems are also promising avenues. In the case of finite temperature, if purification-based ansatz is used, the small-o trick can save the memory cost by a factor of 16 for spinful fermions!

An important conceptual question concerns the regime of validity of PEPS for real-time dynamics. While PEPS cannot represent volume-law states, our results indicate that for local quenches and low-energy excitations the entanglement growth remains compatible with PEPS bond dimensions accessible on current hardware. A more systematic understanding of entanglement-light-cone structure in 2D local quenches, and of the interplay between PEPS bond dimension and dynamical fidelity, would provide valuable theoretical guidance. In addition, studying late-time thermalization or hydrodynamic transport within PEPS–tVMC may reveal whether approximate area-law manifolds can capture emergent long-wavelength physics even when the exact state exhibits extensive entanglement.

Finally, PEPS–tVMC offers a classical compu-tational counterpart to emerging quantum simulation platforms. Real-time experiments increasingly probe 2D dynamics, and a method capable of providing controlled, large-scale classical benchmarks will be valuable for validating quantum devices and interpreting experimental data. We expect PEPS–tVMC to play a significant role in this context, bridging analytical theory, classical simulation, and quantum hardware.

Overall, our work demonstrates that PEPS, long regarded as difficult for time evolution, can in fact support accurate and stable simulations of large-scale 2D quantum dynamics at low energy. We hope that the ideas introduced here can inspire new developments in both variational algorithms and tensor-network theory, ultimately expanding the reach of classical simulation deep into the low-energy real-time dynamics of quantum matter.

ACKNOWLEDGMENTS

[1] J. I. Cirac, D. Pérez-García, N. Schuch, and F. Verstraete, Matrix product states and projected entangled pair states: Concepts, symmetries, theorems, Rev. Mod. Phys. 93, 045003 (2021).

[2] T. Xiang, *Density Matrix and Tensor Network Renormalization* (Cambridge University Press, 2023).

[3] A. W. Sandvik, Computational studies of quantum spin systems, in *AIP Conference Proceedings*, Vol. 1297 (2010) pp. 135–338.

[4] F. Becca and S. Sorella, *Quantum Monte Carlo Approaches for Correlated Systems* (Cambridge University Press, 2017).

[5] S. Sorella, Green function monte carlo with stochastic reconfiguration, Phys. Rev. Lett. 80, 4558 (1998).

[6] F. Becca and S. Sorella, *Quantum Monte Carlo Approaches for Correlated Systems* (Cambridge University Press, 2017).

[7] S. Sorella, Generalized lanczos algorithm for variational quantum monte carlo, Phys. Rev. B 64, 024512 (2001).

[8] G. Carleo, F. Becca, M. Schiró, and M. Fabrizio, Localization and glassy dynamics of many-body quantum systems, Scientific Reports 2, 243 (2012).

[9] G. Carleo, F. Becca, L. Sanchez-Palencia, S. Sorella, and M. Fabrizio, Light-cone effect and supersonic correlations in one- and two-dimensional bosonic superfluids, Phys. Rev. A 89, 031602 (2014).

[10] K. Ido, T. Ohgoe, and M. Imada, Time-dependent many-variable variational monte carlo method for nonequilibrium strongly correlated electron systems, Phys. Rev. B 92, 245106 (2015).

[11] G. Carleo, L. Cevolani, L. Sanchez-Palencia, and M. Holzmann, Unitary dynamics of strongly interacting bose gases with the time-dependent variational monte carlo method in continuous space, Phys. Rev. X 7, 031026 (2017).

[12] G. Carleo, F. Becca, M. Schiró, and M. Fabrizio, Localization and glassy dynamics of many-body quantum systems, Scientific Reports 2, 243 (2012).

[13] M. Schmitt and M. Heyl, Quantum many-body dy-

namics in two dimensions with artificial neural networks, Phys. Rev. Lett. **125**, 100503 (2020).

[14] I. L. Gutiérrez and C. B. Mendl, Real time evolution with neural-network quantum states, Quantum **6**, 627 (2022).

[15] A. Sinibaldi, C. Giuliani, G. Carleo, and F. Vicentini, Unbiasing time-dependent Variational Monte Carlo by projected quantum evolution, Quantum **7**, 1131 (2023).

[16] M. Medvidović and D. Sels, Variational quantum dynamics of two-dimensional rotor models, PRX Quantum **4**, 040302 (2023).

[17] J. Nys, G. Pescia, A. Sinibaldi, and G. Carleo, Ab-initio variational wave functions for the time-dependent many-electron schrödinger equation, Nature Communications **15**, 9404 (2024).

[18] H. Lange, A. Van de Walle, A. Abedinnia, and A. Bohrdt, From architectures to applications: a review of neural quantum states, Quantum Science and Technology **9**, 040501 (2024).

[19] F. Verstraete and J. I. Cirac, Renormalization algorithms for quantum-many body systems in two and higher dimensions, arXiv:cond-mat/0407066 (2004).

[20] J. Jordan, R. Orús, G. Vidal, F. Verstraete, and J. I. Cirac, Classical simulation of infinite-size quantum lattice systems in two spatial dimensions, Phys. Rev. Lett. **101**, 250602 (2008).

[21] P. Corboz, R. Orús, B. Bauer, and G. Vidal, Simulation of strongly correlated fermions in two spatial dimensions with fermionic projected entangled-pair states, Phys. Rev. B **81**, 165104 (2010).

[22] M. Lubasch, J. I. Cirac, and M.-C. Bañuls, Algorithms for finite projected entangled pair states, Phys. Rev. B , 064425 (2014).

[23] W.-Y. Liu, S.-S. Gong, Y.-B. Li, D. Poilblanc, W.-Q. Chen, and Z.-C. Gu, Gapless quantum spin liquid and global phase diagram of the spin-1/2 $J_1 - J_2$ square antiferromagnetic Heisenberg model, Science Bulletin **67**, 1034 (2022).

[24] W.-Y. Liu, J. Hasik, S.-S. Gong, D. Poilblanc, W.-Q. Chen, and Z.-C. Gu, Emergence of gapless quantum spin liquid from deconfined quantum critical point, Phys. Rev. X **12**, 031039 (2022).

[25] W.-Y. Liu, X.-T. Zhang, Z. Wang, S.-S. Gong, W.-Q. Chen, and Z.-C. Gu, Quantum criticality with emergent symmetry in the extended shastry-sutherland model, Phys. Rev. Lett. **133**, 026502 (2024).

[26] W.-Y. Liu, H. Zhai, R. Peng, Z.-C. Gu, and G. K.-L. Chan, Accurate simulation of the hubbard model with finite fermionic projected entangled pair states, Phys. Rev. Lett. **134**, 256502 (2025).

[27] Y. Wu and W.-Y. Liu, Accurate gauge-invariant tensor-network simulations for abelian lattice gauge theory in $(2+1)$D: Ground-state and real-time dynamics, Phys. Rev. Lett. **135**, 130401 (2025).

[28] W.-Y. Liu, Y.-Z. Huang, S.-S. Gong, and Z.-C. Gu, Accurate simulation for finite projected entangled pair states in two dimensions, Phys. Rev. B **103**, 235155 (2021).

[29] J. Eisert, M. Cramer, and M. B. Plenio, Colloquium: Area laws for the entanglement entropy, Rev. Mod. Phys. **82**, 277 (2010).

[30] P. Calabrese and J. Cardy, Quantum quenches in 1+1 dimensional conformal field theories, Journal of Statistical Mechanics: Theory and Experiment **2016**, 064003 (2016).

[31] O. Fischer, M. Kugler, I. Maggio-Aprile, C. Berthod, and C. Renner, Scanning tunneling spectroscopy of high-temperature superconductors, Rev. Mod. Phys. **79**, 353 (2007).

[32] J. A. Sobota, Y. He, and Z.-X. Shen, Angle-resolved photoemission spectroscopy, Rev. Mod. Phys. **93**, 025006 (2021).

[33] J. Haegeman, C. Lubich, I. Oseledets, B. Vandereycken, and F. Verstraete, Unifying time evolution and optimization with matrix product states, Phys. Rev. B **94**, 165116 (2016).

[34] G. Vidal, Efficient classical simulation of slightly entangled quantum computations, Phys. Rev. Lett. **91**, 147902 (2003).

[35] M. P. Zaletel, R. S. K. Mong, C. Karrasch, J. E. Moore, and F. Pollmann, Time-evolving a matrix product state with long-ranged interactions, Phys. Rev. B **91**, 165112 (2015).

[36] G. Vidal, Classical simulation of infinite-size quantum lattice systems in one spatial dimension, Phys. Rev. Lett. **98**, 070201 (2007).

[37] P. Czarnik and J. Dziarmaga, Time evolution of an infinite projected entangled pair state: An algorithm from first principles, Phys. Rev. B **98**, 045110 (2018).

[38] P. Czarnik, J. Dziarmaga, and P. Corboz, Time evolution of an infinite projected entangled pair state: An efficient algorithm, Phys. Rev. B **99**, 035115 (2019).

[39] C. Hubig and J. I. Cirac, Time-dependent study of disordered models with infinite projected entangled pair states, SciPost Phys. **6**, 031 (2019).

[40] C. Hubig, A. Bohrdt, M. Knap, F. Grusdt, and J. I. Cirac, Evaluation of time-dependent correlators after a local quench in iPEPS: hole motion in the t-J model, SciPost Phys. **8**, 021 (2020).

[41] S.-H. Lin, M. P. Zaletel, and F. Pollmann, Efficient simulation of dynamics in two-dimensional quantum spin systems with isometric tensor networks, Phys. Rev. B **106**, 245102 (2022).

[42] J. Dziarmaga, Time evolution of an infinite projected entangled pair state: A gradient tensor update in the tangent space, Phys. Rev. B **106**, 014304 (2022).

[43] R. T. Ponnaganti, M. Mambrini, and D. Poilblanc, Real-time dynamics of a critical resonating valence bond spin liquid, Phys. Rev. B **106**, 195132 (2022).

[44] J. D. Arias Espinoza and P. Corboz, Spectral functions with infinite projected entangled-pair states, Phys. Rev. B **110**, 094314 (2024).

[45] Y. Motoyama, T. Okubo, K. Yoshimi, S. Morita, T. Aoyama, T. Kato, and N. Kawashima, Tenes-v2: Enhancement for real-time and finite temperature simulations of quantum many-body systems, Com-

puter Physics Communications **315**, 109692 (2025).

[46] Z. Dai, Y. Wu, T. Wang, and M. P. Zaletel, Fermionic isometric tensor network states in two dimensions, Phys. Rev. Lett. **134**, 026502 (2025).

[47] J. P. Provost and G. Vallee, Riemannian structure on manifolds of quantum states, Communications in Mathematical Physics **76**, 289 (1980).

[48] J. C. Butcher, *Numerical Methods for Ordinary Differential Equations*, 3rd ed. (Wiley, 2016).

[49] A. Chen and M. Heyl, Empowering deep neural quantum states through efficient optimization, Nature Physics **20**, 1476 (2024).

[50] Y. Wu, Supplemental material (2025).

[51] M. P. Zaletel and F. Pollmann, Isometric tensor network states in two dimensions, Phys. Rev. Lett. **124**, 037201 (2020).

[52] Y. Wu and Z. Dai, Algorithms for variational monte carlo calculations of fermion projected entangled pair states in the swap gates formulation and the detailed balance of tensor network sequential sampling (2025), arXiv:2506.20106 [cond-mat.str-el].

[53] C. V. Kraus, N. Schuch, F. Verstraete, and J. I. Cirac, Fermionic projected entangled pair states, Phys. Rev. A **81**, 052338 (2010).

[54] P. Corboz, R. Orús, B. Bauer, and G. Vidal, Simulation of strongly correlated fermions in two spatial dimensions with fermionic projected entangled-pair states, Phys. Rev. B **81**, 165104 (2010).

[55] D. C. Tsui, H. L. Stormer, and A. C. Gossard, Two-dimensional magnetotransport in the extreme quantum limit, Phys. Rev. Lett. **48**, 1559 (1982).

[56] R. B. Laughlin, Anomalous quantum hall effect: An incompressible quantum fluid with fractionally charged excitations, Phys. Rev. Lett. **50**, 1395 (1983).

[57] J. K. Jain, Composite-fermion approach for the fractional quantum hall effect, Phys. Rev. Lett. **63**, 199 (1989).

[58] A. S. Sørensen, E. Demler, and M. D. Lukin, Fractional quantum hall states of atoms in optical lattices, Phys. Rev. Lett. **94**, 086803 (2005).

[59] G. Möller and N. R. Cooper, Composite fermion theory for bosonic quantum hall states on lattices, Phys. Rev. Lett. **103**, 105303 (2009).

[60] N. Regnault and B. A. Bernevig, Fractional chern insulator, Phys. Rev. X **1**, 021014 (2011).

[61] We thank Bartholomew Andrews for letting us know this practice.

[62] R. B. Laughlin, Quantized hall conductivity in two dimensions, Phys. Rev. B **23**, 5632 (1981).

[63] D. J. Thouless, Quantization of particle transport, Phys. Rev. B **27**, 6083 (1983).

[64] E. Fradkin and S. H. Shenker, Phase diagrams of lattice gauge theories with higgs fields, Phys. Rev. D **19**, 3682 (1979).

[65] F. Wu, Y. Deng, and N. Prokof'ev, Phase diagram of the toric code model in a parallel magnetic field, Phys. Rev. B **85**, 195104 (2012).

[66] E. A. Martinez, C. A. Muschik, P. Schindler, D. Nigg, A. Erhard, M. Heyl, P. Hauke, M. Dalmonte, T. Monz, P. Zoller, and R. Blatt, Real-time dynamics of lattice gauge theories with a few-qubit quantum computer, Nature **534**, 516 (2016).

[67] B. Yang, H. Sun, R. Ott, H.-Y. Wang, T. V. Zache, J. C. Halimeh, Z.-S. Yuan, P. Hauke, and J.-W. Pan, Observation of gauge invariance in a 71-site Bose-Hubbard quantum simulator, Nature **587**, 392 (2020).

[68] M. Meth, J. Zhang, J. F. Haase, C. Edmunds, L. Postler, A. J. Jena, A. Steiner, L. Dellantonio, R. Blatt, P. Zoller, T. Monz, P. Schindler, C. Muschik, and M. Ringbauer, Simulating two-dimensional lattice gauge theories on a qudit quantum computer, Nature Physics **21**, 570 (2025).

[69] A. W. Sandvik, Stochastic series expansion method for quantum spin systems, Phys. Rev. B **59**, R14157 (1999).

[70] S. Heinrichs and W. J. Mullin, Quantum-monte-carlo calculations for bosons in a two-dimensional harmonic trap, Journal of Low Temperature Physics **113**, 231 (1998).

[71] L. Landau, Theory of the superfluidity of helium ii, Phys. Rev. **60**, 356 (1941).

[72] G. H. Golub and C. F. Van Loan, *Matrix Computations - 4th Edition* (Johns Hopkins University Press, Philadelphia, PA, 2013).

[73] M. Zhang, H. Zhang, C. Wang, and L. He, Scalable tensor network algorithm for thermal quantum many-body systems in two dimensions, Phys. Rev. B **111**, 075146 (2025).

[74] J. Hauschild and F. Pollmann, Efficient numerical simulations with tensor networks: Tensor network python (tenpy), SciPost Physics Lecture Notes , 005 (2018).

[75] J. Hauschild, J. Unfried, S. Anand, B. Andrews, M. Bintz, U. Borla, S. Divic, M. Drescher, J. Geiger, M. Hefel, K. Hémery, W. Kadow, J. Kemp, N. Kirchner, V. S. Liu, G. Möller, D. Parker, M. Rader, A. Romen, S. Scalet, L. Schoonderwoerd, M. Schulz, T. Soejima, P. Thoma, Y. Wu, P. Zechmann, L. Zweng, R. S. K. Mong, M. P. Zaletel, and F. Pollmann, Tensor network Python (TeNPy) version 1, SciPost Phys. Codebases , 41 (2024).

[76] J. Bradbury, R. Frostig, P. Hawkins, M. J. Johnson, C. Leary, D. Maclaurin, G. Necula, A. Paszke, J. VanderPlas, S. Wanderman-Milne, and Q. Zhang, JAX: composable transformations of Python+NumPy programs (2018).

[77] The small-o trick is the following process. Consider O matrix with shape (N_s, N_p). When computing the minSR matrix $OO^\dagger$, to save memory, one can partition the parameters into K parts of the same size: i) compute log-derivatives of $1/K$ of the paramters, and form a smaller matrix o_k with shape $(N_s, N_p/K)$ for $k = 0, 1, 2, .., K-1$; ii) form $o_k o_k^\dagger$; iii) use an accumulator to obtain the minSR matrix: $OO^\dagger = \sum_{k=0}^{K-1} o_k o_k^\dagger$. O never appears, and one saves the memory by a factor K. For a generic wavefunction ansatz, e.g. an NQS, computing o_k typically takes more time than $1/K$ of the time needed to compute O. Then, one needs to spend more time for the blocking. For TNS, this does not happen,

i.e. there is a structure where one can spend the same amount of time using o_k as using O at once. In fact, if one is on CPU and does not need to worry about the GPU coding practices, one can also save the time by a factor of K.

Supplemental Material

S-1. SMALL-o TRICK

[77] For tensor network states (TNS), for given sample $\mathbf{s}$ and at a given lattice site $\mathbf{x}$, $O[\mathbf{x}](\mathbf{s})_{plrdu} = 0$ if $p \neq \mathbf{s}(\mathbf{x})$. One thus only keeps

$$o[\mathbf{x}](\mathbf{s})_{lrdu} = O[\mathbf{x}](\mathbf{s})_{\mathbf{s}(\mathbf{x})lrdu}$$

This saves memory by a factor of d. We index o with o_{sa}, where a runs over a selected subset of the parameter sets, depending on the sample $\mathbf{s}$. One can construct the minSR matrix with o instead of O, keeping the memory benefit while not sacrificing time cost. We explain this in detail below.

One needs to form the minSR matrix $T = \mathcal{O}\mathcal{O}^\dagger$:

$$T_{ss'} = \sum_\alpha (O_{s\alpha} O_{\alpha s'}^\dagger - \langle O \rangle_\alpha O_{\alpha s'}^\dagger - O_{s\alpha} \langle O \rangle_\alpha^* + \langle O \rangle_\alpha \langle O \rangle_\alpha^*), \qquad \langle O \rangle_\alpha \equiv \frac{1}{N_s} \sum_s O_{s\alpha}$$

1. $\sum_\alpha O_{s\alpha} O_{\alpha s'}^\dagger$:

   To compute this from o, one constructs a sample tensor p s.t. $p_{sa} = \mathbf{s}(\mathbf{x})$ where $\mathbf{s}$ is the sample indexed by s and $\mathbf{x}$ is the site associated with the parameter indexed by a. Then

   $$\sum_\alpha O_{s\alpha} O_{\alpha s'}^\dagger = \sum_a o_{sa} o_{as'}^\dagger (p_{sa} == p_{s'a})$$

   $$= \sum_{k=0}^{K-1} \sum_a o_{sa} o_{as'}^\dagger (p_{sa} == k) \& (p_{s'a} == k)$$

   $$= \sum_{k=0}^{K-1} \sum_a \tilde{o}(k)_{sa} \tilde{o}(k)_{as'}^\dagger$$

   where $\tilde{o}(k)_{sa} \equiv o_{sa} * (p_{sa} == k)$

   To implement this, one loops over k and use an accumulator for the sum. This takes the same memory by a factor of K with no additional time cost. ($oo^\dagger$ is K-times faster than $OO^\dagger$, but one executes K of them in sequence. The memory cost of p_{sa} is neglegible because its dtype is `jnp.int8` compared to O's dtype `jnp.complex128`).

Listing 1: JAX implementation of $OO^\dagger$

```
1   import jax.numpy as jnp
2
3   def get_OOdag(o: jnp.ndarray, p: jnp.ndarray, K: int):
4       r
5       Input:
6           o : (Ns, Np/K)
7           p : (Ns, Np/K)
8           K : int
9       Return:
10          OOdag : (Ns, Ns)
11
12      OOdag = 0
13      for k in range(K):
14          o2 = jnp.where(p == k, o, 0)
15          OOdag += o2 @ o2.T.conj()
16      return OOdag
```

2. $\sum_\alpha O_{s\alpha} \langle O \rangle_\alpha^* = \frac{1}{N_s} \sum_\alpha O_{s\alpha} \sum_{s'} O_{s'\alpha}^* = \frac{1}{N_s} \sum_{s'} (OO^\dagger)_{ss'} = $ `jnp.average(OOdag, axis=1)`.

3. Overall, the minSR matrix is obtained as

Listing 2: JAX implementation of computing the full minSR matrix

```
1  import jax.numpy as jnp
2  def get_minSR_matrix(OOdag):
3      r
4      Input:
5          OOdag : (Ns, Ns), obtained from get_OOdag()
6
7      OOavg_c = jnp.average(OOdag, axis=1)
8      return OOdag - OOavg_c.conj() - OOavg_c[:,None] + jnp.average(OOavg_c)
```

S-2. DETAILS OF TVMC SIMULATIONS

Here we document the simulation details of the tVMC carried out in the main text. The integrator of all simulations is 4th-order Runge-Kutta. The regulator in solving the SR or the minSR equation using the Cholesky decomposition is all 10^{-8}.

The notations are

- D: bond dimension of PEPS

- D': bond dimension of the boundary MPS used to contract the single layer PEPS after the physical legs are sliced by the samples

- dt: step of the time evolution

- T: total evolution time

- N_s: number of samples used to accumulate the gradients

- N_p: number of complex variational parameters

- t_{wall}: wall time of the simulation per time step

- T_{wall}: total wall time of the entire simulation

Model	D	D'	dt	T	N_s	N_p	t_{wall}	T_{wall}	SR or minSR
12×12 Chern insulator	4	16	0.01	12	40960	28224	399s	5.5 days	SR
12×12 Bosonic Hofstadter	4	32	0.01	10	20480	56448	465s	5.3 days	minSR
12×12 $\mathbb{Z}_2$ gauge theory	8	16	0.01	12	20480	451584	300s	4.2 days	minSR
13×13 Superfluid boson	4	16	0.02	12	20480	67712	223s	1.5 days	minSR

T_{wall} should be taken as an upper bound, as there are many algorithmic inefficiencies in our code which can be straightforwardly fixed in the future. For the $\mathbb{Z}_2$ lattice gauge theory, despite the large N_p due to $D = 8$, the time cost and the memory cost are both at the level of a $D = 4$ normal PEPS. The time saving is from our findings in Ref. [27], while the memory saving is due to the small-o trick introduced in the main text (If one stores the full 20480×451584 `complex128` O matrix, the memory cost would be 138GB, which is a significant burden for GPUs).