

Simulating Quantum Circuits with Tree Tensor Networks using Density-Matrix Renormalization Group Algorithm

Aditya Dubey,^{1,2,*} Zeki Zeybek,^{2,3,†} and Peter Schmelcher^{2,3,‡}

¹*Department of Physics, BITS-Pilani, K K Birla Goa Campus, Zuarinagar, Goa 403726, India*

²*Zentrum für Optische Quantentechnologien, Universität Hamburg,
Luruper Chaussee 149, 22761 Hamburg, Germany*

³*The Hamburg Centre for Ultrafast Imaging, Universität Hamburg,
Luruper Chaussee 149, 22761 Hamburg, Germany*

Quantum computing offers the potential for computational abilities that can go beyond classical machines. However, they are still limited by several challenges such as noise, decoherence, and gate errors. As a result, efficient classical simulation of quantum circuits is vital not only for validating and benchmarking quantum hardware but also for gaining deeper insights into the behavior of quantum algorithms. A promising framework for classical simulation is provided by tensor networks. Recently, the Density-Matrix Renormalization Group (DMRG) algorithm was developed for simulating quantum circuits using matrix product states (MPS). Although MPS is efficient for representing quantum states with one-dimensional correlation structures, the fixed linear geometry restricts the expressive power of the MPS. In this work, we extend the DMRG algorithm for simulating quantum circuits to tree tensor networks (TTNs). To benchmark the method, we simulate random and QAOA circuits with various two-qubit gate connectivities. For the random circuits, we devise tree-like gate layouts that are suitable for TTN and show that TTN requires less memory than MPS for the simulations. For the QAOA circuits, a TTN construction that exploits graph structure significantly improves the simulation fidelities. Our findings show that TTNs provide a promising framework for simulating quantum circuits, particularly when gate connectivities exhibit clustering or a hierarchical structure.

I. INTRODUCTION

Quantum computing promises revolutionary computational capabilities, offering the potential to solve problems that are intractable for classical machines [1–3]. However, despite the allure of exponentially faster algorithms, quantum computers today are still hampered by several challenges [4]. Current devices are limited by noise, decoherence, and gate errors, all of which constrain their reliability. As a result, efficient classical simulation of quantum systems is vital for validating and benchmarking quantum hardware and gaining deeper insights into the behavior of quantum algorithms [5–7].

A promising framework for classical simulation is provided by tensor networks (TNs) [8–11]. They offer a way to manage the exponential complexity of quantum states by exploiting the structure inherent in many-body entanglement, rendering them ideal for the representation and manipulation of quantum states. Consequently, TN algorithms have become cornerstones in the numerical study of strongly correlated quantum systems, addressing problems ranging from ground state search [12, 13] to real-time dynamics [14–16].

Tensor networks have also become a versatile tool in quantum information theory and quantum computing [17]. For example, they provide a framework for loading

classical data into a quantum circuit [18–23]. TN techniques also provide valuable insights into the structure and performance of quantum algorithms [6, 7, 24]. Efficient classical simulation of quantum circuits is equally vital for the validation and benchmarking of quantum hardware. TN methods have proven particularly effective in this context by refuting quantum supremacy claims with classical simulation results that rival or surpass proposed quantum advantages [5, 25–28].

A wide variety of tensor network architectures have been employed for simulating quantum circuits [5, 26, 29–31]. In particular, the DMRG algorithm using matrix product states was generalized to simulate quantum circuits with a finite fidelity [26]. The algorithm essentially takes the resulting state after the application of the gates and compresses it into an MPS. Although MPS is efficient for compressing quantum states that admit one-dimensional correlation structure, the fixed chain-like geometry restricts the expressive power of the MPS. Therefore, an appropriate kind of tensor network should be considered for the DMRG algorithm to represent states exhibiting multi-scale correlations.

Tree tensor networks have emerged as a powerful tool in both many-body physics and quantum chemistry, with applications spanning ground state calculations [32–38], real-time dynamics [39–45], and the study of complex molecular systems [46–49]. However, TTNs have not been explored extensively in the context of quantum circuit simulation. Their structural flexibility allows for adapting the ansatz to match the underlying connectivity or correlation patterns of the quantum system. In this work, we extend the DMRG algorithm for simulating

* f20200627@goa.bits-pilani.ac.in

† zeki.zeybek@uni-hamburg.de

‡ peter.schmelcher@uni-hamburg.de

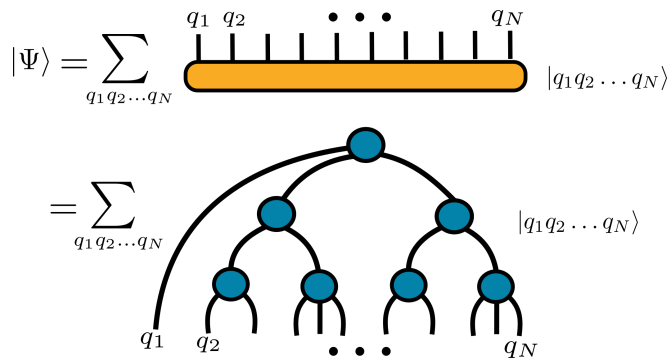


FIG. 1. Illustration of the TTN ansatz is given. TTN represents the coefficient tensor (orange bar) as a hierarchical network of tensors with a tree structure. Fully contracting the network down to the leaf nodes with a specific set of physical indices yields the coefficient of the corresponding basis state.

quantum circuits to TTNs. The key implication of our work is that TTNs provide a promising framework for simulating quantum circuits, especially when gate connectivities exhibit some form of clustering or hierarchical structure. Therefore, the flexibility of TTNs allows them to adapt to a wide range of quantum circuit architectures. We focus on random circuits and QAOA circuits with various connectivities, including a special class of circuits referred to as *tree-like* circuits. For tree-like random circuits, TTNs require less memory than MPS for comparable accuracy due to their ability to capture long-range correlations. In QAOA simulations for MaxCut on 3-regular and tree-like connected graphs, a naive TTN construction exploiting graph structure significantly improves fidelity compared to a blind approach. We achieve near-perfect fidelity for structured bridged 3-regular and tree-like connected graphs despite having small bond dimensions.

The paper is organized as follows. In Sec. II A, we introduce the TTN ansatz and outline its key features. Section II B presents the DMRG algorithm as applied to the simulation of quantum circuits. In Sec. III, we report our results for the simulation of both random and QAOA circuits across various qubit connectivity configurations. Finally, in Sec. IV we provide our conclusions and outlook.

II. THEORY

In the following, the tree tensor network (TTN) ansatz is introduced and we explain the DMRG algorithm for simulating quantum circuits.

A. Tree tensor network

An arbitrary quantum state $|\Psi\rangle$ describing a system of N qubits is expressed as

$$|\Psi\rangle = \sum_{q_1 q_2 \dots q_N} \Psi_{q_1 q_2 \dots q_N} |q_1 q_2 \dots q_N\rangle, \quad (1)$$

where $\Psi_{q_1 q_2 \dots q_N}$ is the coefficient tensor and $|q_1 q_2 \dots q_N\rangle$ denotes the basis states. The coefficient tensor requires 2^N parameters to be specified exhibiting exponential scaling with the system size. A TTN decomposes this large tensor into a product of smaller tensors with a hierarchical tree structure where the individual tensors are located at the nodes of a connected acyclic graph and the edges between nodes indicate how these tensors are contracted as shown in Fig. 1. The TTN ansatz is given as

$$\Psi_{q_1 q_2 \dots q_N} = \sum_{\{\chi\}} T_{\{q_1, \{\chi\}_1}^{[1]} \dots T_{\{q_v, \{\chi\}_v}^{[v]} \dots T_{\{q_l, \{\chi\}_l}^{[l]}, \quad (2)$$

where the summation is over all the virtual indices collectively denoted by $\{\chi\}$. $T_{\{\chi\}_v}^{[v]}$ is the tensor associated with the node v on the tree network, $\{\chi\}_v$ is the set of virtual indices that are connected to the node and $\{q\}_v$ represents the set of physical indices that are connected to the node. Note that $\{q\}_v$ can be an empty set, whereas $\{\chi\}_v$ is always non-empty as every node in the TTN is connected to the rest of the network through its virtual bonds. In this study, the total number of indices (i.e., elements of $\{q\}_v \cup \{\chi\}_v$) for each tensor in the TTNs is at most five.

TTNs provide a loop-free tensor network representation that allows full exploitation of tensor network gauge freedom. Smart gauging can significantly reduce the number of required tensor contractions and thereby improve computational efficiency [50, 51]. This feature is crucial for the DMRG algorithm used in this study as it greatly simplifies the compression step, discussed in Sec. II B. In contrast to MPS, the distance between physical indices in a TTN scales logarithmically with the system size as $\mathcal{O}(\log N)$, whereas in MPS, it scales linearly $\mathcal{O}(N)$. Since connected correlation functions typically decay exponentially with the path length in a tensor network, the shorter paths in TTNs allow for a more faithful representation of long-range correlations compared to MPS [29, 32]. This property makes TTNs more efficient at representing systems with long-range interactions or quantum circuits with non-local gates. In this work, we leverage this flexibility to simulate a wide class of circuits with a carefully constructed TTN.

B. Quantum circuit simulation

In the DMRG algorithm for simulating quantum circuits, the main idea is to approximate the evolving quan-

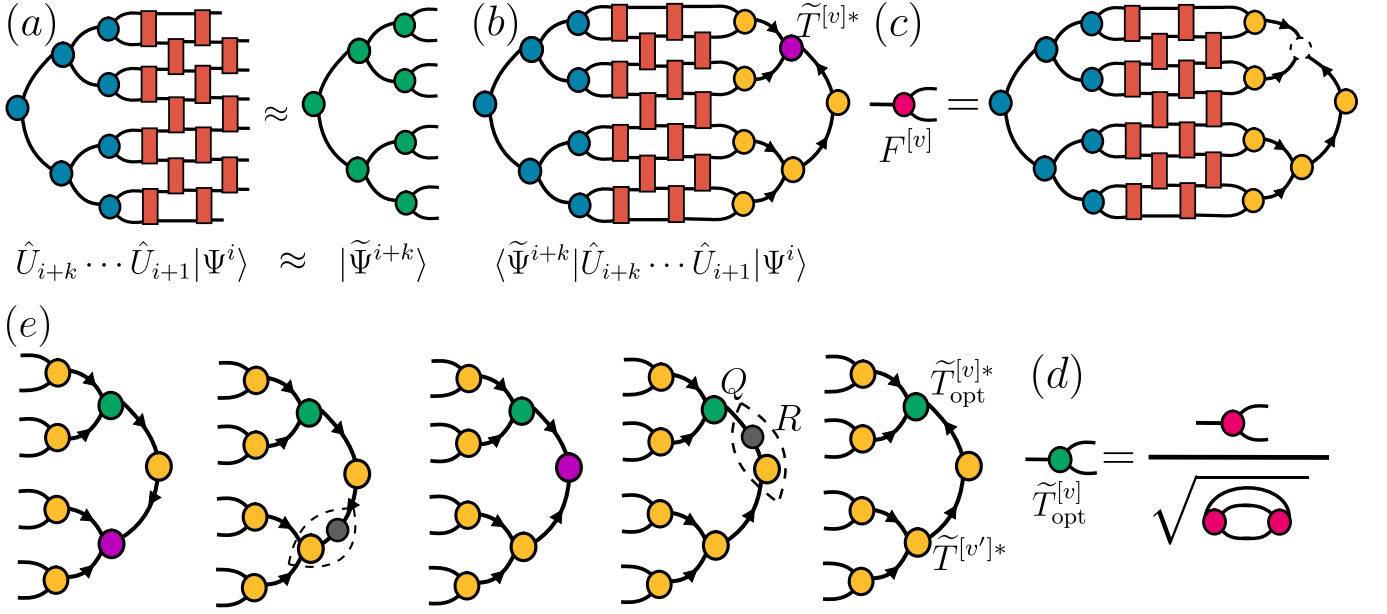


FIG. 2. DMRG algorithm for simulating quantum circuits is illustrated. (a) A schematic of the compression step: k gates are applied to the initial state $|\Psi^i\rangle$, followed by an approximation of the resulting state with a tree tensor network $|\tilde{\Psi}^{i+k}\rangle$. (b) Tensor network diagram of the scalar product being maximized; the purple node indicates the tensor currently being optimized, which serves as the orthogonality center. (c–d) The optimal tensor $\tilde{T}_{\text{opt}}^{[v]}$ is computed from the environment tensor $F^{[v]}$, normalized by $\sqrt{\text{Tr}F^{[v]}F^{[v]*}}$. (e) The orthogonality center is shifted to the next tensor to be optimized, $\tilde{T}^{[v']}$, via a sequence of QR decompositions.

tum state using a TTN ansatz as illustrated in Fig. 2(a). The approximation is performed through iterative compression, where one tensor is locally optimized at a time [8]. At no stage is the full evolved quantum state computed explicitly during this process. Starting with a TTN at step i denoted by $|\Psi^i\rangle$, this ideally approximates the exact quantum state after the application of i gates. When a sequence of k additional gates, $\hat{U}_{i+k} \cdots \hat{U}_{i+1}$, is applied, the updated state is given as

$$|\Psi^{i+k}\rangle = \hat{U}_{i+k} \cdots \hat{U}_{i+1} |\Psi^i\rangle. \quad (3)$$

The goal is to approximate this evolved state with a new TTN $|\tilde{\Psi}^{i+k}\rangle$ of fixed bond dimension by minimizing the distance

$$\delta = \left\| |\Psi^{i+k}\rangle - |\tilde{\Psi}^{i+k}\rangle \right\|^2. \quad (4)$$

This minimization is carried out iteratively by optimizing each tensor $\tilde{T}^{[v]}$ of $|\tilde{\Psi}^{i+k}\rangle$ individually while keeping the others fixed. Each such optimization step reduces to solving a linear system of equations, whose solution yields the optimal tensor $\tilde{T}_{\text{opt}}^{[v]}$. To simplify this optimization, it is advantageous to put $|\tilde{\Psi}^{i+k}\rangle$ into a canonical form with respect to the target tensor $\tilde{T}^{[v]}$, as illustrated in Fig. 2(b). In this form, all other tensors in the network are isometric in the direction of $\tilde{T}^{[v]}$, which serves as the orthogonality center. This ensures that the norm of the TTN is simply given by

$$\langle \tilde{\Psi}^{i+k} | \tilde{\Psi}^{i+k} \rangle = \text{Tr} \tilde{T}^{[v]} \tilde{T}^{[v]*}, \quad (5)$$

where the Tr denotes the full contraction of the network with summation over all indices. The optimal target tensor (up to normalization) is then obtained from the environment tensor $F^{[v]}$, which is computed by contracting the overlap network $\langle \tilde{\Psi}^{i+k} | \Psi^{i+k} \rangle$ without the tensor $\tilde{T}^{[v]*}$, as shown in Fig 2(c–d). The resulting expression for the optimal target tensor is given as

$$\tilde{T}_{\text{opt}}^{[v]} = \frac{F^{[v]}}{\sqrt{\text{Tr}F^{[v]}F^{[v]*}}}. \quad (6)$$

To monitor convergence during the optimization of each tensor, the distance can be monitored $\delta_v = 1 - f_v$, where the partial fidelity f_v is defined as $f_v = \text{Tr}F^{[v]}F^{[v]*}$. This fidelity also serves as a measure of how well the updated TTN approximates the evolved state,

$$f_v = \left\| \langle \tilde{\Psi}^{i+k} | \hat{U}_{i+k} \cdots \hat{U}_{i+1} |\Psi^i\rangle \right\|^2, \quad (7)$$

enabling us to quantitatively assess the progress of the optimization process during individual tensor updates or across multiple sweeps. In this iterative process, the partial fidelities increase monotonically as we continuously improve the TTN by updating the tensors.

This procedure allows the simulation of an entire quantum circuit by decomposing it into a sequence of compression steps. The final value of the partial fidelity $f^{[\kappa]}$ obtained at each compression step κ can be used to define

the quantity

$$\tilde{\mathcal{F}} = \prod_{\kappa} f^{[\kappa]}. \quad (8)$$

As demonstrated in Refs. [5, 26], $\tilde{\mathcal{F}}$ serves as an accurate approximation of the exact fidelity between the exact and compressed states. Since we employ the same algorithm as in Ref. [26], this approximation remains valid in our case. Based on this, we define the error rate per two-qubit gate in our simulation as

$$\epsilon = 1 - \tilde{\mathcal{F}}^{1/N_{2g}}, \quad (9)$$

where N_{2g} denotes the total number of two-qubit gates. This quantity is commonly employed as a benchmark for evaluating the performance of NISQ devices [52–54].

III. RESULTS

In the following, the results of simulations for random circuits and QAOA circuits across various qubit connectivity configurations are presented. For all the simulations, the initial state is set to $|00 \dots 00\rangle$.

A. Random circuits

Although random circuits are not typically designed to solve practical computational problems, they play a crucial role as benchmarks for demonstrating quantum advantage on near-term quantum devices [52, 55, 56]. Their significance lies in the high degree of entanglement they generate, which poses a major challenge for classical simulation methods such as tensor networks that rely on low-entanglement structures. In these simulations, we focus on the error per gate ϵ , a metric commonly employed for benchmarking NISQ devices [52–54]. A very recent preprint by Quantinuum demonstrated random circuit implementations on arbitrary geometries [57] which they benchmarked using the DMRG algorithm with MPS using error per gate ϵ as the metric. However, the linear structure of MPS limits its expressive power for such circuits. In this context, TTN may offer a more suitable framework for benchmarking these devices.

In the following, we test the method on random circuits where each two-qubit gate making up the circuit is generated through a randomized two-step procedure. Initially, the real and imaginary components of a 4×4 matrix are independently drawn from a normal distribution over the interval $[0, 1]$. Next, the sampled matrix undergoes QR decomposition, yielding $A = QR$, where Q is a unitary matrix and R is an upper triangular matrix. The resulting unitary matrix Q is then used as the two-qubit gate.

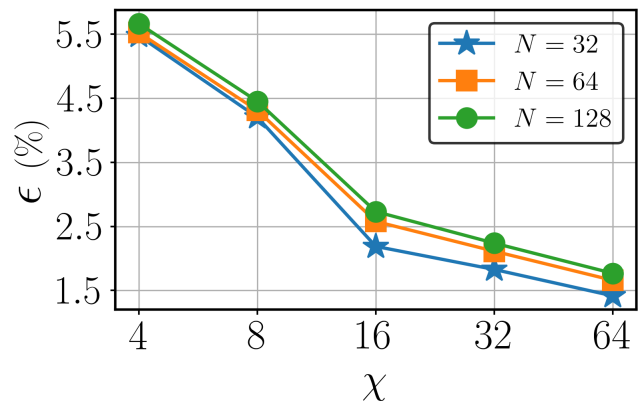


FIG. 3. Error per gate ϵ as a function of bond dimension χ obtained from the simulation of random brick-wall circuits. The results are shown for $N_{2g} = 310, 630$, and 1270 two-qubit gates, corresponding to systems of $N = 32, 64$, and 128 qubits, respectively.

1. Nearest-neighbor connectivity

We first consider quantum circuits with nearest-neighbor connectivity among two-qubit gates. In these circuits, the gates are placed in a brick-wall layout with each two-qubit gate acting on adjacent qubits arranged in a chain-like geometry. In terms of two-qubit connectivity, this can be considered as a linear graph where the vertices denote the qubits and the edges between them are the two-qubit gates. At a given circuit depth D , the total number of two-qubit gates is given by $N_{2g} = N - 1$, which is the total number of edges on the two-qubit gate connectivity graph. To simulate these circuits, we employ a TTN with a binary structure with leaf nodes accommodating four qubits instead of two. In this TTN ansatz, the maximum bond dimension χ achievable is given by $\chi_{\max} = 2^{N/2}$.

Fig. 3 shows the error per gate ϵ as a function of the bond dimension χ for random circuits with a depth of $D = 10$ and $N = 32, 64$, and 128 number of qubits, where the total number of two-qubit gates are given by $N_{2g} = 310, 630$, and 1270 , respectively. At a moderate bond dimension of $\chi = 64$, which is much smaller than χ_{\max} , errors per gate of $\epsilon \approx 1.5\%$ is achieved for all the simulation instances. For the largest system size with $N = 128$, the behavior of the error per gate ϵ as a function of the circuit depth D for various bond dimensions χ is shown in Fig. 4. As expected, increasing the bond dimension leads to a reduction in error growth rates with respect to the circuit depth. It is also worth mentioning that matrix product states (MPS) would likely be a more suitable choice than TTNs for simulating these circuits. The linear geometry of the MPS ansatz does not impose any bias as to what group of qubits would be more entangled by the nearest-neighbor two-qubit connectivity. Instead, it provides a more natural tensor network representation for one-dimensional chains with short-ranged

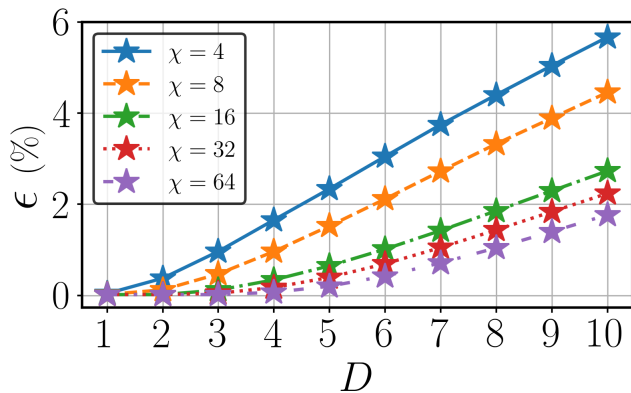


FIG. 4. Error per gate ϵ as a function of the circuit depth D obtained from the simulation of random brick-wall circuits with 128 qubits. Each depth contains 127 two-qubit gates, resulting in a total of $N_{2g} = 1270$. Different curves correspond to different values of the bond dimension χ ranging from 4 to 64.

interactions. With a given TTN structure, a given hierarchical grouping of the qubits may not be optimal for representing the entanglement generated by random circuits with nearest-neighbor connectivity.

2. Tree-like circuit connectivity

In this section, we consider a circuit with a built-in structure that admits a hierarchical clustering of the qubits. Specifically, we investigate a circuit layout that mirrors the TTN structure that represents the quantum state. We refer to such circuits as *tree-like circuits* [29] since their gate connectivity forms a graph exhibiting a hierarchical tree-like clustering as shown in Fig. 5 for a ternary tree structure with $N = 27$ qubits. In these circuits, qubits associated with the nodes of the bottom layer of the TTN are all-to-all connected, forming clusters. Each of these clusters represents a region of strong interaction, where two-qubit gates are applied among all qubits within the cluster. These initial clusters are then merged into larger clusters at the next hierarchical level. In our construction, the inter-cluster connectivity at every level is provided through a single two-qubit gate. A single two-qubit gate acting on the first qubit of each cluster establishes the couplings between the larger clusters. This process is repeated iteratively, where each layer aggregates smaller clusters into larger ones, forming a hierarchy of clusters that reflects the TTN structure.

In Fig. 6, we analyze the error per gate versus memory footprint M for the simulation of these circuits. The memory footprint M is defined as the total number of tensor elements $M = \sum_v \prod_i \dim_i(T^{[v]})$. Here, $\dim_i(T^{[v]})$ refers to the dimension of index i of the tensor associated with the node v . This analysis focuses on simulating random tree-like circuits with $N_{2g} = 390$ for $N = 27$ qubits

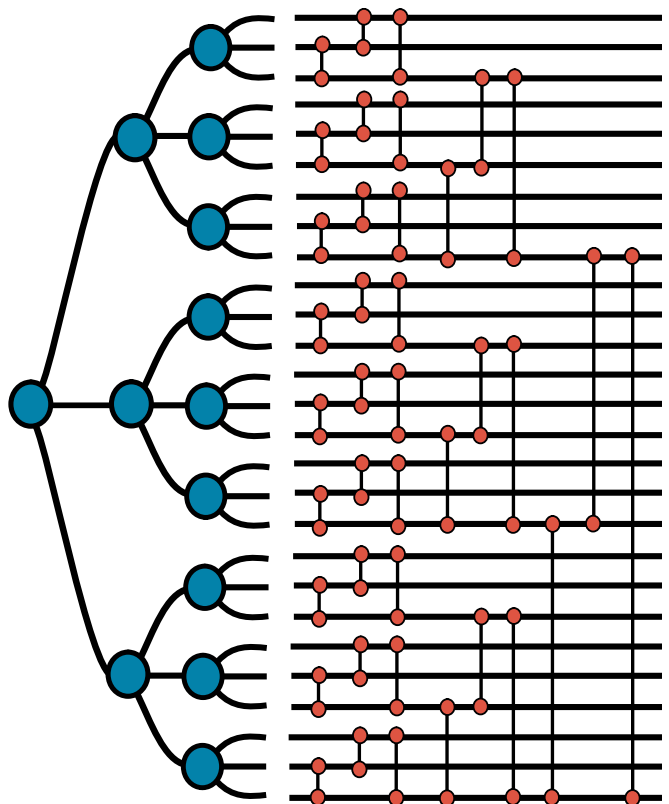


FIG. 5. Single depth of a tree-like circuit resulting from a ternary tree tensor network of 27 qubits. In this circuit, qubits associated with the nodes of the bottom layer of the TTN are fully connected, forming nine clusters of three qubits each. These clusters are then connected to clusters associated with the same parent node in the second layer of the TTN via single two-qubit gates, specifically acting on the first qubit of each cluster. This results in larger clusters comprising nine qubits. At the highest level of the hierarchy, these nine-qubit clusters are further connected using a single two-qubit gate between the first qubit of each cluster, thereby completing one depth of the tree-like circuit.

using MPS and TTN. These circuits pose a significant challenge for MPS, as achieving comparable gate errors to TTN requires substantially more memory. This difficulty stems from the inherent limitation of MPS in efficiently capturing non-local correlations induced by long-range gates due to its strictly linear topology. Specifically, when two qubits that are far apart in the MPS chain are entangled by a gate then the intermediate tensors must carry this non-local information. To faithfully encode these correlations, the bond dimension must grow across the path connecting the two qubits, resulting in a greater memory footprint. In contrast, TTN leverages the underlying connectivity of the circuit to construct a more expressive ansatz that allows it to capture correlations more effectively while requiring less memory. The results highlight that when gate connectivity can be exploited, TTN can be a more efficient choice than MPS in simulating such quantum circuits.

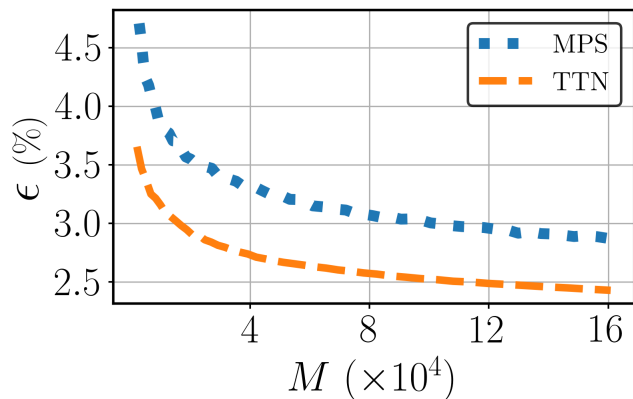


FIG. 6. Comparison of error rates ϵ as a function of memory footprint M obtained from the simulation of random tree-like circuits using MPS and TTN. The results are for a system of $N = 27$ qubits with $N_{2g} = 390$ two-qubit gates.

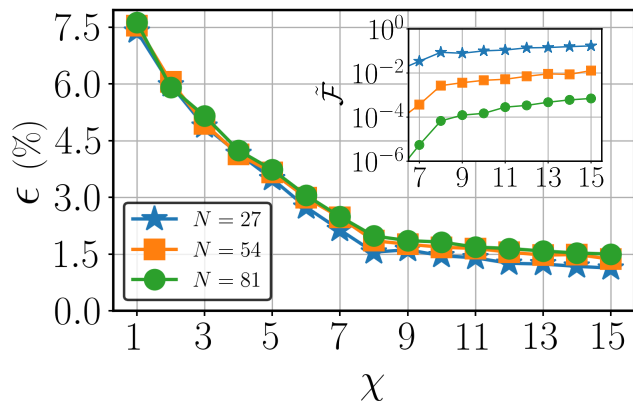


FIG. 7. Error per gate ϵ as a function of bond dimension χ obtained from the simulation of random tree-like circuits. The results are shown for $N_{2g} = 156, 326,$ and 480 two-qubit gates, corresponding to systems of $N = 27, 54,$ and 81 qubits. The inset shows fidelity \tilde{F} as a function of bond dimension χ obtained from the same simulations.

In Fig. 7, the error per gate is shown as a function of the bond dimension χ for the simulation of random tree-like circuits with system sizes of $N = 27, 54,$ and 81 qubits. The corresponding total number of two-qubit gates is $N_{2g} = 156, 316,$ and 480 , respectively. Notably, an error of $\epsilon \approx 1.5\%$ is achieved even for a small bond dimension of $\chi = 15$ for all the simulations. The inset displays the fidelities \tilde{F} from the same simulations. This suggests that having similar error values ϵ across circuits with different qubit counts does not ensure their fidelities will be comparable. Therefore, as the qubit count increases, low fidelities are likely at such small bond dimensions due to the high entanglement generated in these circuits.

To understand how the error per gate ϵ depends on the degree of inter-cluster connectivity, tree-like circuits with different numbers of inter-cluster gate counts are simu-

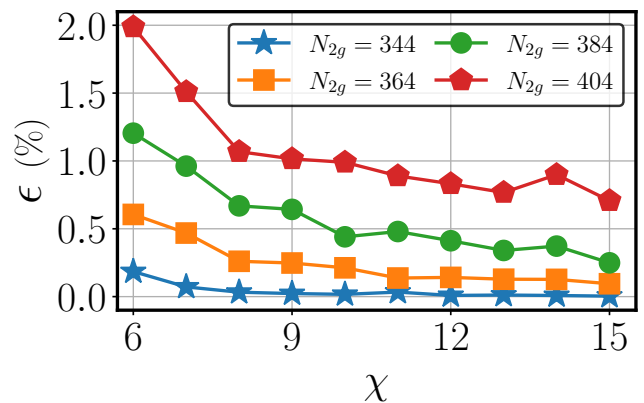


FIG. 8. Error per gate ϵ as a function of bond dimension χ obtained from the simulation of random tree-like circuits for a system of $N = 81$ qubits with varying number N_{2g} of two qubit gates.

lated. Specifically, we focus on a tree-like circuit based on a ternary tree structure with $N = 81$ qubits. We randomly select a specific number of two-qubit gates to connect different clusters. This selection is made uniformly from all possible connections between clusters with repetitions allowed. In all cases, the total number of intra-cluster two-qubit gates is kept all-to-all, and only the inter-cluster gates are varied. In Fig. 8, the error per gate ϵ as a function of χ for different numbers of two-qubit gates is shown. Despite having a random circuit, TTN allows us to achieve very low error per gate of $\epsilon \approx 0.003\%$ by appropriately grouping strongly interacting qubits in one cluster as shown by the results for $N_{2g} = 344$ (Fig. 8). As the number of inter-cluster gates increases, the error per gate increases, as is expected due to the corresponding increase in the entanglement of inter-cluster qubits. It should also be noted that the increase in error per gate cannot be solely attributed to the number of inter-cluster gates. The all-to-all connectivity within each cluster is a key contributor because the qubits forming the inter-cluster links act as a bridge. Through this single connecting qubit, any qubit in one cluster can indirectly influence, and be influenced by, all the qubits in the other cluster. This may necessitate an increase in the bond dimension of the virtual bond connecting the clusters for an accurate representation of the updated state. This issue is analogous to the time evolution of many-body systems where correlations spread throughout the system as time progresses. Accurately capturing these long-range correlations often requires increasing the bond dimension in the simulation.

B. QAOA

To test our method on a useful task, we simulate the QAOA circuit on various graphs for the MaxCut problem. The MaxCut problem involves partitioning the ver-

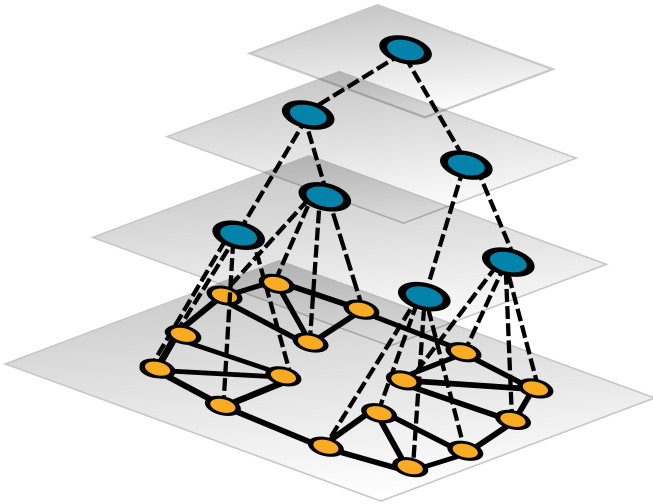


FIG. 9. The bridged 3-regular graph representing the two-qubit gate connectivity is embedded into a hierarchical tree structure. Clusters of qubits (orange nodes) in the lowest level make up the distinct leaf nodes of the tree tensor network. These clusters are then coupled to form the internal nodes of the upper layer while capturing the inter-cluster two-qubit gate connectivity.

tices of a graph into two subsets to maximize the number of edges crossing between these subsets. The QAOA circuit for the MaxCut problem on a graph with N vertices with depth p and $2p$ parameters is expressed as

$$|\boldsymbol{\theta} \equiv \{\boldsymbol{\beta}, \boldsymbol{\gamma}\}\rangle = \prod_{\ell=1}^p U_{\beta_{\ell}} U_{\gamma_{\ell}} H^{\otimes N} |0\rangle^{\otimes N}, \quad (10)$$

where $U_{\beta_{\ell}} = \prod_j \exp(-i\frac{\beta_{\ell}}{2} X_j)$ with X_j the Pauli operator on qubit j , $U_{\gamma_{\ell}} = \exp(-i\frac{\gamma_{\ell}}{2} C)$ with $C = \sum_{\{i,j\} \in E} w_{ij} Z_i Z_j$ the cost function operator of the MaxCut problem running over the edges $\{i, j\}$ on the graph and H the Hadamard gate applied on the individual qubits. Here, we want to probe how accurately the QAOA circuit can be simulated. Therefore, the variational parameters of the circuit are not optimized. Instead, the $2p$ parameters are sampled from uniform distributions: $\gamma \in [0, 2\pi)$ and $\beta \in [0, \pi)$. We simulate the QAOA algorithm on two types of graphs: 3-regular graphs and graphs with two-qubit gate connectivity that align with the structure of tree-like circuits. For both the graphs, weights are set to be uniform as $w_{ij} = w = 1$.

1. 3-regular graphs

A 3-regular graph with N vertices is a graph where each vertex is connected to three other and distinct vertices at random without having self-loops or parallel edges. A TTN can be constructed to effectively leverage the underlying connectivity of such graphs. This approach allows obtaining high fidelity even at relatively

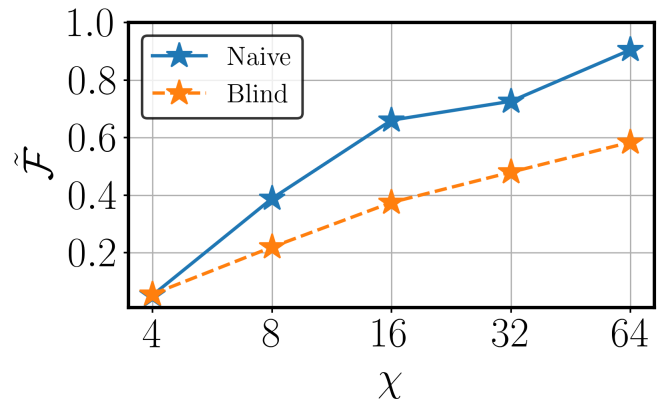


FIG. 10. Comparison of fidelities $\tilde{\mathcal{F}}$ as a function of bond dimension χ obtained from the simulation of QAOA circuits on random 3-regular graphs using TTN constructed via blind and naive approaches. The results are shown for $N = 64$ at depth $p = 1$ with $N_{2q} = 96$ two-qubit gates.

small bond dimensions. An example of such a construction is shown in Fig. 9, where clusters of four qubits are identified based on their edge density. These clusters then form the nodes associated with the lowest layer of the TTN. These clusters are then coupled together to form the nodes of the upper layer while capturing the inter-cluster connectivity.

For random 3-regular graphs, we employ a “naive” clustering approach to construct the TTN. Clusters are first identified within the graph such that there is a high density of edges within each cluster and only a few edges connecting different clusters. This clustering is achieved using the algorithm described in [58], which is designed for community detection in networks. Once the clusters are identified, qubits belonging to the same cluster are positioned closer together in the TTN. However, this approach does not account for inter-cluster relationships, which limits its ability to fully exploit the graph’s connectivity. We compare this naive approach to a “blind” one that doesn’t use any structural information about the graph during TTN construction. Despite its simplicity, the naive method demonstrates significant improvements in fidelity compared to the blind approach, as shown in Fig. 10. The results are presented for the simulation of a QAOA circuit with 64 qubits at depth $p = 1$ on random 3-regular graphs. The naive approach achieves better results with $\chi = 16$ than the blind approach with $\chi = 64$, highlighting the effectiveness of leveraging even basic structural information in TTN construction can enhance algorithm performance. These findings suggest that further improvements in fidelity can be achieved with more sophisticated TTN constructions that exploit both intra- and inter-cluster relationships.

A subclass of 3-regular graphs that has built-in clustering is the bridged 3-regular graphs. In such graphs, nodes form tightly connected clusters that are arranged in a circular chain. This inherent clustering allows for

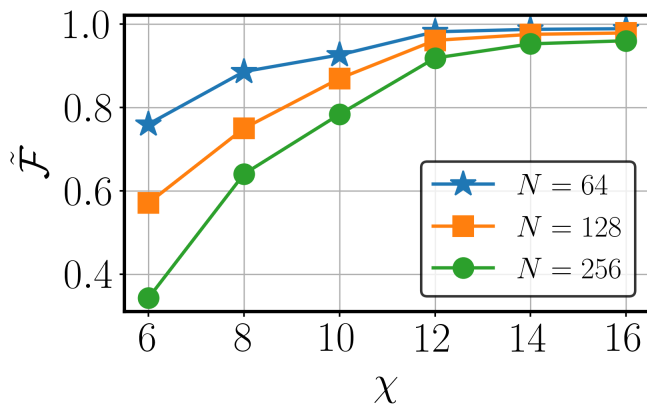


FIG. 11. Fidelity $\tilde{\mathcal{F}}$ as a function of bond dimension χ obtained from the simulation of QAOA circuits on bridged 3-regular graphs at circuit depth $p = 4$. The results are shown for $N_{2g} = 384, 768$, and 1536 two-qubit gates, corresponding to systems of $N = 64, 128$, and 256 qubits, respectively.

constructing a TTN that aligns with the graph connectivity as illustrated in Fig. 9 for a bridged 3-regular graph with sixteen vertices. To evaluate the performance of our method, we perform QAOA simulations on such graphs. Fig. 11 shows results at depth $p = 4$ for $N = 64, 128$, and 256 qubits. We achieve fidelities of $\tilde{\mathcal{F}} \approx 1$ with very small bond dimensions across all system sizes. These results demonstrate the efficiency of TTNs when applied to graphs that admit hierarchical embedding into a tree network.

2. Tree-like circuits

We now focus on graphs whose edges correspond to the two-qubit gate connectivity of tree-like circuits. As previously discussed, these graphs exhibit hierarchical tree-like clustering. This hierarchical structure makes TTN particularly well-suited for simulating QAOA circuits aimed at solving combinatorial problems on such graphs. As shown in Fig. 12, we achieve fidelities of $\tilde{\mathcal{F}} \approx 1$ at a very low bond dimension of $\chi = 8$ across all system sizes considered for a QAOA circuit with depth $p = 4$.

Quantum circuits designed for practical applications often utilize a particular type of parametrized two-qubit gate throughout the circuit. For example, QAOA circuits employ carefully chosen two-qubit gates tailored for a specific problem. This contrasts with random circuits that use two-qubit gates that are completely random unitaries and different from each other. For instance, in the QAOA circuit for the MaxCut problem, the same type of two-qubit gate is applied consistently at each depth p , varying only by a parameterized angle. This inherent structure in the gates results in significantly less entanglement generation compared to random circuits. Consequently, this allows for a more accurate representation of the quantum state using the DMRG algorithm. To high-

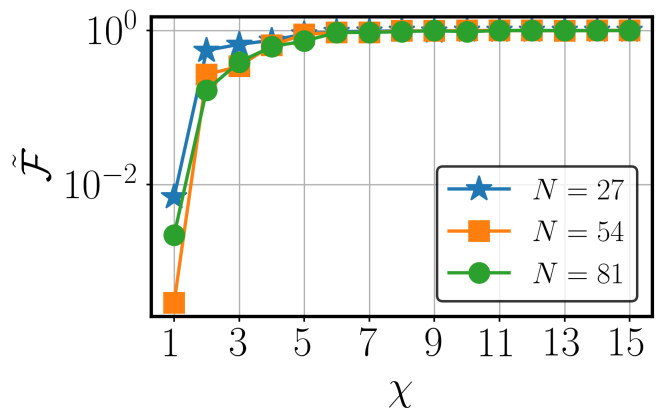


FIG. 12. Fidelity $\tilde{\mathcal{F}}$ as a function of bond dimension χ obtained from the simulation of QAOA tree-like circuits at circuit depth $p = 4$. The results are shown for $N_{2g} = 156, 316$, and 480 two-qubit gates, corresponding to systems of $N = 27, 54$, and 81 qubits, respectively.

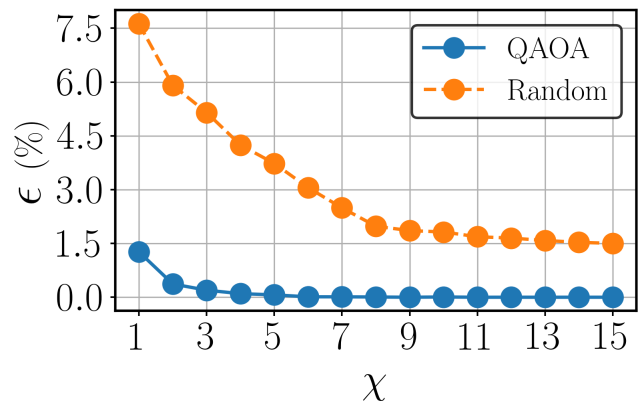


FIG. 13. Comparison of error rates ϵ as a function of bond dimension χ obtained from the simulation of random and QAOA tree-like circuits with $N_{2g} = 480$ two-qubit gates for a system of $N = 81$ qubits.

light this contrast, we compare the error per gate from simulations of tree-like random and tree-like QAOA circuits with $N_{2g} = 480$ two-qubit gates for a system of $N = 81$ qubits. As shown in Fig. 13, the QAOA simulations achieve significantly lower error rates. Specifically, the error is reduced from $\epsilon \approx 1.5\%$ for the random circuit to $\epsilon \approx 0.0004\%$ for the QAOA circuit with $\chi = 15$, corresponding to a reduction by a factor of nearly 4000.

IV. CONCLUSION AND OUTLOOK

Tensor networks offer an ideal framework for classical simulation of quantum circuits and have been successfully applied in various contexts [5–7, 25, 27–29, 31]. In particular, the DMRG algorithm was generalized to simulate quantum circuits with a finite fidelity using ma-

trix product states [26]. However, despite their efficiency, MPS are fundamentally limited by their chain-like structure. This structural constraint restricts their ability to represent quantum states with more complex entanglement patterns. To address this limitation, we extend the DMRG algorithm for quantum circuit simulation to tree tensor networks.

In this study, we considered the simulation of random circuits and QAOA circuits across various two-qubit gate connectivities, including a special class of circuits referred to as *tree-like* circuits. The key implication of our work is that TTNs provide a promising framework for simulating quantum circuits, especially when gate connectivities exhibit some form of clustering or hierarchical structure. For the tree-like circuits with random gates, we showed that MPS requires substantially more memory than TTN to achieve comparable error per gate. This advantage is due to TTN’s ability to better capture long-range correlations than MPS. For QAOA circuits on random 3-regular graphs, we demonstrated that a naive TTN construction approach based on clustering graph nodes greatly improves fidelity compared to a blind approach that ignores structural information. To further emphasize the importance of exploiting structure, we simulated the QAOA circuit for bridged 3-regular graphs and graphs with two-qubit gate connectivity that aligns with the structure of tree-like circuits. For both types of graphs, we achieved high fidelities at very small bond dimensions. Our results also highlight that circuits designed for practical tasks, such as QAOA, are significantly more amenable to compression than random circuits.

Our findings demonstrate that tree tensor networks offer an efficient and flexible framework for simulating

quantum circuits. A natural extension of our work is to consider a two-site version of the compression step. Although the previous work using MPS reported no significant improvement from the two-site variant over the single-site version [26], its potential advantages in the TTN setting remain worth investigating. Further performance gains could be achieved by constructing optimal TTNs using graph embedding techniques or enabling adaptive changes to the TTN structure during simulation, as developed for MPS in [59]. Notably, the compression step in our algorithm is quite general and can be incorporated into other TTN-based quantum circuit algorithms, such as those proposed in Ref. [29]. Recent works, such as Quantinuum’s realization of random quantum circuits in arbitrary geometries and benchmarking them with MPS [57], highlight the relevance of our approach as an alternative benchmarking tool. The framework employed here can also be extended to simulate the time evolution of a quantum many-body system [60]. Finally, for certain applications like QAOA circuits, where the entanglement generated remains relatively low [61], our method offers a particularly efficient and practical solution.

ACKNOWLEDGMENTS

Z.Z. and P.S. are funded by the Cluster of Excellence “CUI: Advanced Imaging of Matter” of the Deutsche Forschungsgemeinschaft (DFG) - EXC 2056 - Project ID 390715994. The tensor contractions were performed using the `cotengra` [62] and `opt_einsum` [63] libraries. The qubit connectivity graphs were generated using the `networkX` [64] library.

-
- [1] R. P. Feynman, Simulating physics with computers, *Int. J. Theor. Phys.* **21** (1982).
 - [2] P. W. Shor, Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer, *SIAM J. Comput.* **26**, 1484 (1997).
 - [3] I. Kassal, J. D. Whitfield, A. Perdomo-Ortiz, M.-H. Yung, and A. Aspuru-Guzik, Simulating chemistry using quantum computers, *Annu. Rev. Phys. Chem.* **62**, 185 (2011).
 - [4] J. Preskill, Quantum Computing in the NISQ era and beyond, *Quantum* **2**, 79 (2018).
 - [5] Y. Zhou, E. M. Stoudenmire, and X. Waintal, What Limits the Simulation of Quantum Computers?, *Phys. Rev. X* **10**, 041038 (2020).
 - [6] J. Chen, E. Stoudenmire, and S. R. White, Quantum Fourier Transform Has Small Entanglement, *PRX Quantum* **4**, 040318 (2023).
 - [7] E. M. Stoudenmire and X. Waintal, Opening the Black Box inside Grover’s Algorithm, *Phys. Rev. X* **14**, 041029 (2024).
 - [8] U. Schollwöck, The density-matrix renormalization group in the age of matrix product states, *Ann. Phys.* **326**, 96 (2011).
 - [9] R. Orús, A practical introduction to tensor networks: Matrix product states and projected entangled pair states, *Ann. Phys.* **349**, 117 (2014).
 - [10] M. C. Bañuls, Tensor network algorithms: A route map, *Annu. Rev. Condens. Matter Phys.* **14**, 173 (2023).
 - [11] F. Verstraete, T. Nishino, U. Schollwöck, M. C. Bañuls, G. K. Chan, and M. E. Stoudenmire, Density matrix renormalization group, 30 years on, *Nat. Rev. Phys.* **5**, 273 (2023).
 - [12] S. R. White, Density matrix formulation for quantum renormalization groups, *Phys. Rev. Lett.* **69**, 2863 (1992).
 - [13] U. Schollwöck, The density-matrix renormalization group, *Rev. Mod. Phys.* **77**, 259 (2005).
 - [14] G. Vidal, Efficient classical simulation of slightly entangled quantum computations, *Phys. Rev. Lett.* **91**, 147902 (2003).
 - [15] 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).
 - [16] S. Paeckel, T. Köhler, A. Swoboda, S. R. Manmana, U. Schollwöck, and C. Hubig, Time-evolution methods for matrix-product states, *Ann. Phys.* **411**, 167998 (2019).

- (2019).
- [17] A. Berezutskii, A. Acharya, R. Ellerbrock, J. Gray, R. Haghshenas, Z. He, A. Khan, V. Kuzmin, M. Liu, D. Lyakh, D. Lykov, S. Mandrà, C. Mansell, A. Melnikov, A. Melnikov, V. Mironov, D. Morozov, F. Neukart, A. Nocera, M. A. Perlin, M. Perelshtein, R. Shaydulin, B. Villalonga, M. Pfitsch, M. Pistoia, V. Vinokur, and Y. Alexeev, Tensor networks for quantum computing, [arXiv:2503.08626](https://arxiv.org/abs/2503.08626) .
- [18] S. Lu, M. Kanász-Nagy, I. Kukuljan, and J. I. Cirac, Tensor networks and efficient descriptions of classical data, *Phys. Rev. A* **111**, 032409 (2025).
- [19] Z.-Y. Han, J. Wang, H. Fan, L. Wang, and P. Zhang, Unsupervised generative modeling using matrix product states, *Phys. Rev. X* **8**, 031012 (2018).
- [20] E. M. Stoudenmire, Learning relevant features of data with multi-scale tensor networks, *Quantum Sci. Technol.* **3**, 034003 (2018).
- [21] C. Schön, E. Solano, F. Verstraete, J. I. Cirac, and M. M. Wolf, Sequential generation of entangled multi-qubit states, *Phys. Rev. Lett.* **95**, 110503 (2005).
- [22] A. A. Melnikov, A. A. Termanova, S. V. Dolgov, F. Neukart, and M. R. Perelshtein, Quantum state preparation using tensor networks, *Quantum Sci. Technol.* **8**, 035027 (2023).
- [23] K. C. Smith, A. Khan, B. K. Clark, S. Girvin, and T.-C. Wei, Constant-depth preparation of matrix product states with adaptive quantum circuits, *PRX Quantum* **5**, 030344 (2024).
- [24] E. Dumitrescu, Tree tensor network approach to simulating shor’s algorithm, *Phys. Rev. A* **96**, 062322 (2017).
- [25] F. Pan and P. Zhang, Simulation of quantum circuits using the big-batch tensor network method, *Phys. Rev. Lett.* **128**, 030501 (2022).
- [26] T. Ayral, T. Louvet, Y. Zhou, C. Lambert, E. M. Stoudenmire, and X. Waintal, Density-Matrix Renormalization Group Algorithm for Simulating Quantum Circuits with a Finite Fidelity, *PRX Quantum* **4**, 020304 (2023).
- [27] J. Tindall, M. Fishman, E. M. Stoudenmire, and D. Sels, Efficient tensor network simulation of ibm’s eagle kicked ising experiment, *PRX Quantum* **5**, 010308 (2024).
- [28] C. Oh, M. Liu, Y. Alexeev, B. Fefferman, and L. Jiang, Classical algorithm for simulating experimental Gaussian boson sampling, *Nat. Phys.* **20**, 1461 (2024).
- [29] P. Seitz, I. Medina, E. Cruz, Q. Huang, and C. B. Mendl, Simulating quantum circuits using tree tensor networks, *Quantum* **7**, 964 (2023).
- [30] A. V. Berezutskii, I. A. Luchnikov, and A. K. Fedorov, Simulating quantum circuits using the multi-scale entanglement renormalization ansatz, *Phys. Rev. Res.* **7**, 013063 (2025).
- [31] S.-B. B. Lee, H. R. Choi, D. D. Ohm, and S.-S. B. Lee, Scalable simulation of random quantum circuits using projected entangled-pair states, [arXiv:2504.04769](https://arxiv.org/abs/2504.04769) .
- [32] V. Murg, F. Verstraete, O. Legeza, and R. M. Noack, Simulating strongly correlated quantum systems with tree tensor networks, *Phys. Rev. B* **82**, 205105 (2010).
- [33] L. Tagliacozzo, G. Evenbly, and G. Vidal, Simulation of two-dimensional quantum systems using a tree tensor network that exploits the entropic area law, *Phys. Rev. B* **80**, 235127 (2009).
- [34] P. Silvi, V. Giovannetti, S. Montangero, M. Rizzi, J. I. Cirac, and R. Fazio, Homogeneous binary trees as ground states of quantum critical Hamiltonians, *Phys. Rev. A* **81**, 062335 (2010).
- [35] D. Nagaj, E. Farhi, J. Goldstone, P. Shor, and I. Sylvester, Quantum transverse-field ising model on an infinite tree from matrix product states, *Phys. Rev. B* **77**, 214431 (2008).
- [36] E. Macaluso, T. Comparin, R. O. Umucalılar, M. Gerster, S. Montangero, M. Rizzi, and I. Carusotto, Charge and statistics of lattice quasiholes from density measurements: A tree tensor network study, *Phys. Rev. Res.* **2**, 013145 (2020).
- [37] G. Ferrari, G. Magnifico, and S. Montangero, Adaptive-weighted tree tensor networks for disordered quantum many-body systems, *Phys. Rev. B* **105**, 214201 (2022).
- [38] M. Gerster, M. Rizzi, P. Silvi, M. Dalmonte, and S. Montangero, Fractional quantum hall effect in the interacting hofstadter model via tensor networks, *Phys. Rev. B* **96**, 195123 (2017).
- [39] Y.-Y. Shi, L.-M. Duan, and G. Vidal, Classical simulation of quantum many-body systems with a tree tensor network, *Phys. Rev. A* **74**, 022320 (2006).
- [40] W. Li, J. von Delft, and T. Xiang, Efficient simulation of infinite tree tensor network states on the bethe lattice, *Phys. Rev. B* **86**, 195137 (2012).
- [41] S. Debenbrock and F. Pollmann, Phase diagram of the isotropic spin- $\frac{3}{2}$ model on the $z = 3$ bethe lattice, *Phys. Rev. B* **88**, 035138 (2013).
- [42] B. Kloss, D. R. Reichman, and Y. B. Lev, Studying dynamics in two-dimensional quantum lattices using tree tensor network states, *SciPost Phys.* **9**, 070 (2020).
- [43] D. Sulz, C. Lubich, G. Ceruti, I. Lesanovsky, and F. Carollo, Numerical simulation of long-range open quantum many-body dynamics with tree tensor networks, *Phys. Rev. A* **109**, 022420 (2024).
- [44] A. Dubey, Z. Zeybek, F. Köhler, R. Mukherjee, and P. Schmelcher, Ab initio approach to many-body quantum spin dynamics, *Phys. Rev. Res.* **7**, 023039 (2025).
- [45] W. Krinitsin, N. Tausendpfund, M. Rizzi, M. Heyl, and M. Schmitt, Roughening dynamics of interfaces in two-dimensional quantum matter, [arXiv:2412.10145](https://arxiv.org/abs/2412.10145) .
- [46] N. Nakatani and G. K.-L. Chan, Efficient tree tensor network states (TTNS) for quantum chemistry: Generalizations of the density matrix renormalization group algorithm, *J. Chem. Phys.* **138**, 134113 (2013).
- [47] V. Murg, F. Verstraete, R. Schneider, P. R. Nagy, and Ö. Legeza, Tree tensor network state with variable tensor order: An efficient multireference method for strongly correlated systems, *J. Chem. Theory Comput.* **11**, 1027 (2015).
- [48] K. Gunst, F. Verstraete, S. Wouters, Ö. Legeza, and D. Van Neck, T3ns: Three-legged tree tensor network states, *J. Chem. Theory Comput.* **14**, 2026 (2018).
- [49] H. R. Larsson, Computing vibrational eigenstates with tree tensor network states (ttns), *J. Chem. Phys.* **151**, 204102 (2019).
- [50] M. Gerster, P. Silvi, M. Rizzi, R. Fazio, T. Calarco, and S. Montangero, Unconstrained tree tensor network: An adaptive gauge picture for enhanced performance, *Phys. Rev. B* **90**, 125154 (2014).
- [51] P. Silvi, F. Tschirsich, M. Gerster, J. Jünemann, D. Jaschke, M. Rizzi, and S. Montangero, The Tensor Networks Anthology: Simulation techniques for many-body quantum lattice systems, *SciPost Phys. Lect. Notes*

- , 8 (2019).
- [52] F. Arute, K. Arya, R. Babbush, D. Bacon, J. C. Bardin, R. Barends, R. Biswas, S. Boixo, F. G. S. L. Brandao, D. A. Buell, B. Burkett, Y. Chen, Z. Chen, B. Chiaro, R. Collins, W. Courtney, A. Dunsworth, E. Farhi, B. Foxen, A. Fowler, C. Gidney, M. Giustina, R. Graff, K. Guerin, S. Habegger, M. P. Harrigan, M. J. Hartmann, A. Ho, M. Hoffmann, T. Huang, T. S. Humble, S. V. Isakov, E. Jeffrey, Z. Jiang, D. Kafri, K. Kechedzhi, J. Kelly, P. V. Klimov, S. Knysh, A. Korotkov, F. Kostritsa, D. Landhuis, M. Lindmark, E. Lucero, D. Lyakh, S. Mandrà, J. R. McClean, M. McEwen, A. Megrant, X. Mi, K. Michielsen, M. Mohseni, J. Mutus, O. Naaman, M. Neeley, C. Neill, M. Y. Niu, E. Ostby, A. Petukhov, J. C. Platt, C. Quintana, E. G. Rieffel, P. Roushan, N. C. Rubin, D. Sank, K. J. Satzinger, V. Smelyanskiy, K. J. Sung, M. D. Trevithick, A. Vainsencher, B. Villalonga, T. White, Z. J. Yao, P. Yeh, A. Zalcman, H. Neven, and J. M. Martinis, Quantum supremacy using a programmable superconducting processor, *Nature* **574**, 505 (2019).
- [53] J. M. Pino, J. M. Dreiling, C. Figgatt, J. P. Gaebler, S. A. Moses, M. S. Allman, C. H. Baldwin, M. Foss-Feig, D. Hayes, K. Mayer, C. Ryan-Anderson, and B. Neyenhuis, Demonstration of the trapped-ion quantum CCD computer architecture, *Nature* **592**, 209 (2021).
- [54] Y. Wu, W.-S. Bao, S. Cao, F. Chen, M.-C. Chen, X. Chen, T.-H. Chung, H. Deng, Y. Du, D. Fan, M. Gong, C. Guo, C. Guo, S. Guo, L. Han, L. Hong, H.-L. Huang, Y.-H. Huo, L. Li, N. Li, S. Li, Y. Li, F. Liang, C. Lin, J. Lin, H. Qian, D. Qiao, H. Rong, H. Su, L. Sun, L. Wang, S. Wang, D. Wu, Y. Xu, K. Yan, W. Yang, Y. Yang, Y. Ye, J. Yin, C. Ying, J. Yu, C. Zha, C. Zhang, H. Zhang, K. Zhang, Y. Zhang, H. Zhao, Y. Zhao, L. Zhou, Q. Zhu, C.-Y. Lu, C.-Z. Peng, X. Zhu, and J.-W. Pan, Strong quantum computational advantage using a superconducting quantum processor, *Phys. Rev. Lett.* **127**, 180501 (2021).
- [55] M. J. Bremner, A. Montanaro, and D. J. Shepherd, Average-case complexity versus approximate simulation of commuting quantum computations, *Phys. Rev. Lett.* **117**, 080501 (2016).
- [56] A. J. Daley, I. Bloch, C. Kokail, S. Flannigan, N. Pearson, M. Troyer, and P. Zoller, Practical quantum advantage in quantum simulation, *Nature* **607**, 667 (2022).
- [57] M. DeCross, R. Haghshenas, M. Liu, E. Rinaldi, J. Gray, Y. Alexeev, C. H. Baldwin, J. P. Bartolotta, M. Bohn, E. Chertkov, J. Cline, J. Colina, D. DelVento, J. M. Dreiling, C. Foltz, J. P. Gaebler, T. M. Gatterman, C. N. Gilbreth, J. Giles, D. Gresh, A. Hall, A. Hankin, A. Hansen, N. Hewitt, I. Hoffman, C. Holliman, R. B. Hutson, T. Jacobs, J. Johansen, P. J. Lee, E. Lehman, D. Lucchetti, D. Lykov, I. S. Madjarov, B. Mathewson, K. Mayer, M. Mills, P. Niroula, J. M. Pino, C. Roman, M. Schechter, P. E. Siegfried, B. G. Tiemann, C. Volin, J. Walker, R. Shaydulin, M. Pistoia, S. A. Moses, D. Hayes, B. Neyenhuis, R. P. Stutz, and M. Foss-Feig, The computational power of random quantum circuits in arbitrary geometries, [arXiv:2406.02501](https://arxiv.org/abs/2406.02501) .
- [58] A. Clauset, M. E. J. Newman, and C. Moore, Finding community structure in very large networks, *Phys. Rev. E* **70**, 066111 (2004).
- [59] A. D. Girolamo, P. Facchi, P. Rabl, S. Pascazio, C. Lupo, and G. Magnifico, Dynamical cluster-based optimization of tensor network algorithms for quantum circuit simulations, [arXiv:2502.19289](https://arxiv.org/abs/2502.19289) .
- [60] A. Dubey, Z. Zeybek, and P. Schmelcher, *In preparation*, 2025.
- [61] A. C. Nakhil, T. Quella, and M. Usman, Calibrating the role of entanglement in variational quantum circuits, *Phys. Rev. A* **109**, 032413 (2024).
- [62] J. Gray and S. Kourtis, Hyper-optimized tensor network contraction, *Quantum* **5**, 410 (2021).
- [63] D. G. a. Smith and J. Gray, opt_einsum - a python package for optimizing contraction order for einsum-like expressions, *J. Open Source Softw.* **3**, 753 (2018).
- [64] A. A. Hagberg, D. A. Schult, and P. J. Swart, Exploring network structure, dynamics, and function using networkx, in *Proceedings of the 7th Python in Science Conference*, edited by G. Varoquaux, T. Vaught, and J. Millman (Pasadena, CA USA, 2008) pp. 11 – 15.