IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 20 November 2024, accepted 12 December 2024, date of publication 18 December 2024,
date of current version 27 December 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3519676

SURVEY

Survey on Computational Applications of Tensor-Network Simulations

MARCOS DÍEZ GARCÍA[1] **AND ANTONIO MÁRQUEZ ROMERO**[2]
[1]Fujitsu Research of Europe Ltd., SL1 2BE Slough, U.K.
[2]Fujitsu Research of Europe Ltd., Pozuelo de Alarcón, 28224 Madrid, Spain

Corresponding author: Marcos Díez García (marcos.diezgarcia@fujitsu.com)

ABSTRACT Tensor networks are a popular and computationally efficient approach to simulate general quantum systems on classical computers and, in a broader sense, a framework for dealing with high-dimensional numerical problems. This paper presents a broad but not exhaustive literature review of state-of-the-art applications of tensor networks and related topics across many research domains, including: machine learning, mathematical optimization, materials science, quantum chemistry and quantum circuit simulation. This review aims to clarify which classes of relevant applications have been proposed for which class of tensor networks, as well as to highlight main application results and limitations compared with other classical or quantum simulation methods. We intend this review to be a high-level tour on tensor network applications that is easy to read by non-experts, so basic technical details of tensor networks are summarized.

INDEX TERMS Applications, machine learning, materials science, mathematical optimization, quantum chemistry, quantum circuit simulation, tensor network.

I. INTRODUCTION

Many computational applications have been developed based on the simulation of quantum physical systems and inspired others outside the field of quantum computing. In a seminal lecture in the 1980s, Richard Feynman [1] already discussed the simulation of subatomic-scale physical systems by means of computations based on Turing machines [2] and questioned the extent to which these machines can really simulate quantum physical systems. This motivated the proposal of quantum Turing machines by Deutsch [3], from which quantum complexity theory [4] and quantum computing [5] developed into their present form.

Can classical computers *efficiently* simulate quantum physical systems? This question is problematic because quantum mechanical theory represents quantum physical systems by vectors in a (finite) Hilbert space, which are called state vectors [5], [6]. Due to the quantum mechanical principle of superposition, state vectors are represented as linear combinations of a given basis in Hilbert space. That is, two complex numbers are needed to represent state vectors of any two-state quantum physical system, or qubit; however, 2^n complex numbers are needed to represent state vectors of any composite system with n qubits. Therefore, just representing the states of quantum systems in a classical computer quickly becomes unfeasible for an increasing number of qubits, let alone simulating their behavior over time.

A different but related question, which is a key open challenge in quantum computing, asks instead if classical Turing machines can efficiently simulate *quantum circuits* [7], [8]. Quantum circuits, originally called quantum computational networks [5], [9], are one of the most common models of quantum computation to simulate quantum physical systems. Indeed, there exist quantum circuits which classical Turing machines can efficiently simulate: quantum circuits where all quantum gates are Clifford gates, as shown by Gottesman in 1998 [10]; and, quantum circuits based on ''matchgates,'' as shown later by Valiant [11].

Another approach to efficiently simulate quantum circuits, not based on restricting the type of quantum gates, consists in representing quantum circuits as *tensor networks* (TNs) [12], [13]. TNs can operationally represent a quantum circuit by decomposing it into simpler circuit elements as tensors [14]. That is, quantum states, quantum gates, and the operations between them, are defined in terms of tensors and tensor algebra. A simple definition of tensor is that of an ordered

The associate editor coordinating the review of this manuscript and approving it for publication was Alba Amato.

sequence of values indexed by zero, one, or more indices; scalars, vectors, and matrices, are all specific cases of the general notion of tensor. (Other mathematical notions and related historical remarks of tensors can be found in [14].) Thus, a TN representing a given quantum circuit does not change *what* the circuit computes but *how* such computation is performed.

TNs made it possible to mathematically characterize classes of quantum circuits that can be efficiently simulated on classical computers. One such class are quantum circuits with low entanglement [13]; that is, where the number of entangled qubits grows at most polynomially with the total number of qubits. Another example is the class of quantum circuits where the number of gates grows polynomially, depth grows logarithmically, and all qubit interactions are spatially localized [12]. In fact, localized qubit interactions and low entanglement are properties of well-known antiferromagnetic material models in condensed matter physics [15] that have been described via TNs [16].

A. MOTIVATION AND CONTRIBUTION

Many literature reviews have been published over the last decade discussing:

- theory of TN methods [16], [17], [18], [19], [20], of which the most widely used is the density-matrix renormalization group algorithm [21], [22];
- software for TN methods as standalone packages [23] or part of larger software projects [24], [25], [26];
- partly theory and software of TNs [27], [28], [29];
- TN applications for a *specific* research domain, such as: data analysis in machine learning [30], [31], numerical analysis of continuous multivariate functions [32], molecular orbitals in quantum chemistry [33], [34], variational quantum algorithms in computational fluid dynamics [35], or open quantum systems [36]; and,
- simulation of quantum physical systems by means of classical or quantum computers though not specifically using TNs [8], [37], [38].

Unlike the previous reviews, this paper presents a broad review on state-of-the-art computational applications of TNs across many different research domains, including: machine learning, mathematical optimization, materials science, quantum chemistry, and large-scale simulation of quantum circuits. However, we do not intend to review all TN applications (whether classical or quantum) in those research domains. Our choice of topics is motivated by the aforesaid literature reviews, most of which concern TNs in quantum-based and quantum-inspired computing. For instance, open quantum systems [36] are a fruitful research area, but we exclude them from this review since many TN applications still consider closed quantum systems based on quantum circuits with unitary dynamics. The overall aim of this paper is to clarify what kind of applications have been proposed for which TNs, their major application results, and their limitations, compared with other classical or quantum simulation methods. This review touches upon certain performance[1] aspects of TNs, but this review does not aim to discuss TNs computational complexity (e.g., runtime analysis), which would make our review harder to follow, deviate from our focus, and require separate experimental research. We believe this will make it easier for readers who are not familiar with TNs but are interested in TN applications.

This paper is organized as follows: Section II highlights related state-of-the-art methods and applications in quantum circuit simulation and other areas outside quantum computing. Section III summarizes fundamental notions of TNs which are used in subsequent sections. We then review applications based on well-known classes of TNs: image classification (Section IV), mathematical optimization (Section V), materials science and quantum chemistry (Section VI), and other emerging applications of TN simulations (Section VII). Section VIII presents a general discussion of the TNs applications reviewed and some future work directions. Finally, Section IX concludes this paper, including a handy summary table of the TNs applications reviewed.

II. RELATED WORK

This section contextualizes our review by highlighting other popular methods and applications related to tensor-network simulation.

A. CLASSICAL SIMULATION OF QUANTUM CIRCUITS

Two general methodologies for classically simulating quantum circuits are commonly used, based on two well-known formulations of quantum mechanics: Schrödinger's state-vector formulation [5] and Feynman's sum-over-paths (or path integral) formulation [39]. Schrödinger's formulation is followed by most quantum-circuit simulators [26]. Therein, a simulation consists in state-vector transformations defined by the unitary operations of quantum gates [5]. By contrast, simulations under Feynman's formulation focus on computing single probability amplitudes (one for each possible measurement outcome) associated with the final quantum state of a given circuit. Although both methodologies incur a computational time complexity that grows exponentially in the worst case, Feynman's formulation requires a computational space complexity that only grows polynomially in the number of qubits and quantum gates [7].

These two methodologies set a basis for a range of specific and more efficient techniques developed to classically simulate quantum circuits. Briefly:

- *Massively parallel computing* made it possible in 2018 to sample up to 2^{28} probability amplitudes from a state-vector simulation of quantum circuits with

[1] We use the phrase "computationally intractable" to refer indifferently to any decision problem or analogous optimization problem for which no known classical or quantum algorithm solves it and the computational time/space complexity of the algorithm is polynomially upper-bounded.

64 qubits and circuit depth of 22 [40] at a notably lower computational cost than before [41]. Such a number of qubits is already beyond the 50-qubit scale that others in 2019 [42] and 2022 [8] thought to be the limit. To surpass it, however, required remarkable advances in parallel computer hardware with CPUs and GPUs, distributed/shared memory management, as well as quantum circuit partitioning techniques to distribute the entries of state vectors and quantum gates' matrices across cluster nodes.

- Efficient *data compression* techniques for floating-point data [42] have been recently integrated into Intel-QS [43] distributed, full state-vector, classical simulator of quantum circuits. This enabled a leap in full state-vector simulations from 45 up to 61 qubits for Grover's quantum search algorithm and, at the same time, reduce memory usage from $32 \cdot 10^{18}$ bytes (without data compression) down to $768 \cdot 10^{12}$ bytes [42]. Benchmarks up to 45 qubits across random circuit sampling, quantum approximate optimization algorithm, and quantum Fourier transform, show that memory usage can be reduced by approximately 4–21 times the original thanks to such data compression while maintaining 97.6% qubit simulation fidelity [42].
- *Decision diagrams* and *TNs* are the most popular techniques to represent quantum circuits in a computationally more efficient way than using full state vectors [44]. Both use data structures allowing quantum circuits to be conveniently decomposed: TNs use tensors [14], whereas decision diagrams use directed acyclic graphs similar to binary decision diagrams [45]. However, the simulation performance achieved by these data structures is rather dependent on the class of quantum circuits. On the one hand, TNs are not expected to perform well for deep and highly-entangled quantum circuits [12]. On the other hand, decision diagrams are not expected to perform well if the original state vectors and quantum gates' matrices contain few redundant entries (e.g., if most amplitudes are distinct) [46], [47], [48]. There is also significantly more quantum circuit simulators available for TNs than for decision diagrams [23]. However, making an informed choice between simulators is challenging because the software is often redundantly developed, and there is a lack of common standards for development as well as documentation.
- *Hybrid* techniques have been proposed to exploit the relative successes and computational performance trade-offs of the above techniques. For example: Schrödinger-Feynman simulations hybridized with massively parallel computing [49] or decision diagrams [50], tensor-based circuit cutting [51], and tensor-based decision diagrams [52]. Despite their promising results, these hybrid techniques are at preliminary research stage, and their true performance advantages over the well-established techniques above are yet to be clarified.

B. CLASSICAL EMULATION OF QUANTUM CIRCUITS

In contrast with classical simulation techniques, in 2016 the Institute of Theoretical Physics in Zurich, Intel, and Microsoft Research [53] proposed a classical *emulator* of quantum circuits. Such an emulator allows, in principle, to test and debug quantum circuits at a comparatively lower computational cost than classical simulators. However, this involves a fundamental change of paradigm: the proposed classical emulator is required to compute the same output from a given quantum circuit as performed by a quantum computer but, unlike classical simulators, it is not limited to do so by performing quantum gate operations. This means quantum-gate logic can be replaced by faster and functionally equivalent classical subroutines to avoid the overhead costs of simulating reversible gates as well as associated ancillary qubits. Benchmarks [53] for arithmetical operations, quantum Fourier transform, and quantum phase estimation show that running times of the classical emulator are significantly lower than some state-of-the-art classical simulators of quantum circuits.

C. TENSOR NETWORKS BEYOND QUANTUM COMPUTING

TNs encompass a vast number of methods and applications for dealing with high-dimensional numerical problems which are not limited to quantum circuit simulation or the field of quantum computing more generally [28], [29], [30].

Besides classical problems in machine learning like supervised image classification, which we review in Section IV, there exist challenging problems in unsupervised machine learning where tensor-based methods are widely used. One such problem is subspace clustering [54], [55]. It consists in finding low-dimensional representations that approximate a given set of unlabeled data samples well, where the data can be high-dimensional, heterogeneous (e.g., text, image, video, or sound), and contain errors due to random noise, corrupt or missing information. Several TN algorithms [56], [57] have been proposed for subspace clustering which, in essence, remove irrelevant and redundant features from the initial dataset to achieve an approximate yet computationally efficient representation. This computational advantage is achieved by carefully integrating other techniques [58], [59], [60]: low-rank tensors, self-representation learning, sample diversity learning, certain assumptions about linearity and sparsity of the data, as well as generalizations of singular value decomposition (SVD), which is similar but not identical to principal component analysis or the Karhunen-Loève transform [61].

Other classical computing applications have benefited from the computational efficiency and adaptability of TNs and low-rank tensors to compactly represent different types of complex data. Some examples are: improving token

prediction in large language models [62], automated reasoning in knowledge graphs and ontologies [63], [64], reducing the number of parameters to train deep neural networks [65], and code generation for fast tensor algebra operations [66], [67], [68].

III. TENSOR NETWORKS OVERVIEW

This section defines the key notions of a tensor, tensor contraction, and TN representations of state vectors, recalled in later sections.

An arbitrary tensor of complex numbers is an element $\mathbf{v}$ in a set $\mathbb{C}^{I_1\times\cdots\times I_d}$, where $I_1,\ldots I_d$ are index sets such that $I_j = \{1,\ldots,m_j\}$ for all $j \in \{1,\ldots,d\}$ given any fixed natural numbers m_j and d. Such tensor $\mathbf{v}$ is called an order-d tensor because d indices must be specified to retrieve a single complex number from $\mathbf{v}$. For example, an arbitrary matrix with two rows and three columns is specified as an order-2 tensor $\mathbf{v}$, where the rows are indexed by $I_1 = \{1,2\}$ with $m_1 = 2$ and the columns are indexed by $I_2 = \{1,2,3\}$ with $m_2 = 3$. The element in the first row and third column is specified as $\mathbf{v}[1,3]$ with indices denoted in square brackets rather than subscripts. We use subscripts to label different tensors (e.g., $\mathbf{v}_1$ and $\mathbf{v}_2$ are two different tensors).

Every state vector of a n-qubit composite system can be represented by a linear combination of basis state vectors in a product Hilbert space as follows:

$$|\psi\rangle = \sum_{\{i_1,\ldots,i_n\}} \mathbf{c}[i_1,\ldots,i_n]\,(|i_1\rangle\otimes\cdots\otimes|i_n\rangle)\quad, \tag{1}$$

for all $i_k \in I_k$ and $k \in \{1,\ldots,n\}$. The operator $\otimes$ denotes the Kronecker product, each I_k is an index set, $\mathbf{c}$ is an order-n tensor, and each $\mathbf{c}[i_1,\ldots,i_n]$ is a complex number coefficient given by indices $i_1,\ldots,i_n$. To use computational basis states, one may choose $I_k = \{0,1\}$ so that $|i_k\rangle$ is either $|0\rangle = (1,0)$ or $|1\rangle = (0,1)$. For the case of a two-qubit system where $n = 2$,

$$|\psi\rangle = \sum_{\{i_1,i_2\}} \mathbf{c}[i_1,i_2]\,|i_1,i_2\rangle \tag{2}$$

expands into

$$\begin{aligned}&|\psi\rangle\\ &= \mathbf{c}[0,0]|0,0\rangle + \mathbf{c}[0,1]|0,1\rangle + \mathbf{c}[1,0]|1,0\rangle + \mathbf{c}[1,1]|1,1\rangle\end{aligned} \tag{3}$$

using the shorthand notation $|i_1,i_2\rangle = |i_1\rangle\otimes|i_2\rangle$.

TN representations of $|\psi\rangle$ consist in expressing an order-n tensor $\mathbf{c}$, see (1), in terms of a network of lower-order tensors that use fewer indices. That is, TNs are a natural way to efficiently represent and computationally exploit the underlying structures in high-dimensional data [69].

A contraction operation over a certain index linking two or more tensors within a tensor network is the summation over all elements involved under such index. The order in which we contract those tensors is called a contraction path. Different contraction paths may require a different number of operations, so their respective computational complexity will also be different in general. A TN can be depicted as a graph diagram with nodes and edges displaying the different contractions over the indexed tensors building up the network. For example, let us consider a quantity Q of the form

$$Q = \sum_{\{i,j,k,\ell\}} \mathbf{A}[i,j]\;\mathbf{B}[i,k]\;\mathbf{C}[j,k,\ell]\;\mathbf{D}[\ell], \tag{4}$$

whose TN representation is given in Fig. 1 (a). Contractions over indices ℓ and j are depicted in Fig. 1 (b) and Fig. 1 (c), respectively. Q is finally obtained after contracting indices i and k (in practice, both can be flattened to a single index).

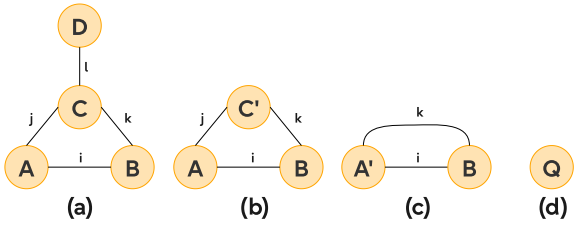

FIGURE 1. Step-by-step contraction of a TN with four tensors (a) to obtain a single scalar Q. The contraction path is defined by the sequence of indices $\ell \to j \to i, k$.

The simplest TN representation is called the matrix product state (MPS) [13], [16]. A general MPS representation of $|\psi\rangle$ is defined by expressing each coefficient $\mathbf{c}[i_1,\ldots,i_n]$ in (1) as a summation of tensor products

$$\mathbf{c}[i_1,\ldots,i_n] = \sum_{\{\alpha_1,\ldots,\alpha_{n-1}\}} C(\alpha_1,\ldots,\alpha_{n-1})\quad, \tag{5}$$

where

$$\begin{aligned}C(\alpha_1,\ldots,\alpha_{n-1}) = {}& \mathbf{c}_1[i_1,\alpha_1]\;\mathbf{c}_2[i_2,\alpha_1,\alpha_2]\\ &\times\cdots\times \mathbf{c}_{n-1}[i_{n-1},\alpha_{n-2},\alpha_{n-1}]\;\mathbf{c}_n[i_n,\alpha_{n-1}],\end{aligned} \tag{6}$$

$\mathbf{c}_1$ and $\mathbf{c}_n$ are order-2 tensors (i.e. matrices), and $\mathbf{c}_2$ through $\mathbf{c}_{n-1}$ are order-3 tensors. The indices $\alpha_1,\ldots,\alpha_{n-1}$ shared between tensors are called virtual or bond indices, which are mathematical artifacts of the representation and do not have a physical meaning per se. By analogy, indices i_k are called physical indices because they relate to the physical degrees of freedom of a quantum state. Each α takes values in $\{1,\ldots,\chi\}$, where χ is a natural number called bond dimension. The bond dimension χ is normally considered as a parameter of the TN representation: increasing χ will increase the size of the corresponding tensor (i.e. number of elements it contains). The expression in (5) is a tensor contraction over such virtual indices. Note that the multiplication of two matrices is a form of tensor contraction when any two given order-2 tensors, say $\mathbf{a}$ and $\mathbf{b}$, are contracted over a single shared index: $\mathbf{c}[u,v] = \sum_w \mathbf{a}[u,w]\mathbf{b}[w,v]$, where $\mathbf{c}[u,v]$ is the element at row u and column v of $\mathbf{c}$. Fig. 2 illustrates a MPS representation for a five-qubit system in diagrammatic form, and Fig. 3 illustrates a quantum circuit

layout for such MPS representation by using two-qubit gates that act on consecutive pairs of qubits.

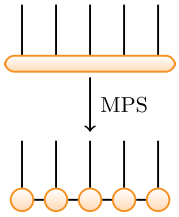

FIGURE 2. Diagram of a MPS representation (bottom) of an order-5 tensor (top). Every node corresponds to one tensor, every edge to one virtual index and every edge incident to only one node corresponds to one physical index.

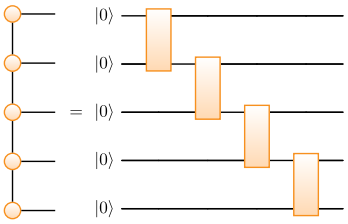

FIGURE 3. Quantum circuit diagram (right), consisting of two-qubit gates ('square boxes'), for a MPS of an order-5 tensor (left).

State vectors (1) require 2^n numbers to fully describe them (or more precisely, $2^n - 1$ numbers taking into account normalization). The MPS representation can be obtained by applying consecutive SVD operations to the state-vector matrix. For example, for a state vector with $n = 2$ qubits

$$|\psi\rangle = \psi_{ij} = \begin{pmatrix} \psi_{11} & \psi_{12} \\ \psi_{21} & \psi_{22} \end{pmatrix} = U_{ix} S_x V_{jx}, \tag{7}$$

where U, V are unitary matrices (also referred to as isometric in the literature) and S is a vector storing the singular values. The structure of the singular values is fundamental to the optimization of the tensor network. Usually, only a few are significant enough to contribute, and we can include only those singular values in the contraction to retrieve $|\psi\rangle$. In other words, we can truncate the TN representation of $|\psi\rangle$ by truncating the bond dimension, and this plays a crucial role in the scalability of TNs. As we can see in (6), the amount of data scales as $O(np\chi^2)$, where n is the number of qubits (equal to the number of tensors $\mathbf{c}$ in the representation), χ is the bond dimension, and p is the physical dimension. Therefore, an MPS representation of state vectors can reduce an exponentially-scaling number of amplitudes to polynomial scaling as long as the bond dimension is truncated to a "sufficiently small" number based on the quantum system or problem at hand.

The configuration of the singular values in the vector S of the SVD dictates the precision in the approximation performed by truncating the bond dimension, and it is closely related to the quantum entanglement of the decomposition. Let us illustrate the relationship with a two-qubit product state $|00\rangle$, whose qubits are not entangled:

$$|00\rangle = |0\rangle \otimes |0\rangle = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix}. \tag{8}$$

After applying SVD to this matrix, we obtain the singular-value vector

$$S_x = \begin{pmatrix} 1 \\ 0 \end{pmatrix}. \tag{9}$$

Thus, restricting the bond dimension to $\chi = 1$ in this simple unentangled case makes the approximation exact since only one singular value contributes to the contraction as seen in (7). However, if we apply SVD to a maximally-entangled Bell state

$$|\psi_B^+\rangle = \frac{1}{\sqrt{2}}(|01\rangle + |10\rangle) = \begin{pmatrix} 0 \\ 1 \\ 1 \\ 0 \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \tag{10}$$

we obtain

$$S_x = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 \\ 1 \end{pmatrix}. \tag{11}$$

In this case, any truncation in the bond dimension results in a poor approximation of the state vector since all singular values contribute equally. Thus, TN representations exploit the low entanglement exhibited in certain parts of the state vector to make a well-controlled approximation, where only the most significant partitions of the system contribute. This extends from quantum mechanics to classical systems by analogy, where instead of entanglement between partitions of the system, there are appropriate correlation functions or couplings [30], [70].

The previous example illustrates why TN representations are inherently limited in the sense that they are not necessarily convenient nor computationally efficient for accurately describing highly-entangled systems (e.g., deep quantum circuits involving a large number of entangling gates). Highly-entangled systems need tailored TN representations based on a careful analysis of their entanglement distribution [16].

If one thinks of each tensor in a MPS as a "particle," then Fig. 2 clearly shows why MPS is a convenient representation of a quantum physical system where particles only interact with their nearest neighbors, as in the Affleck-Kennedy-Lieb-Tasaki model [15], [16]. Other quantum physical systems with more intricate interaction patterns can be represented and simulated by TNs that generalize MPS via higher order tensors. Well-known generalizations of MPS include: tree tensor network (TTN) [71], [72], projected entangled-pair state (PEPS) [73], a form of PEPS called isometric tensor network (isoTNS) [74] where tensors are similar to unitary matrices, and multiscale entanglement renormalization ansatz (MERA) [20]. Fig. 4 illustrates examples of these in

diagrammatic form. Details about their formulations can be found in the aforementioned references.

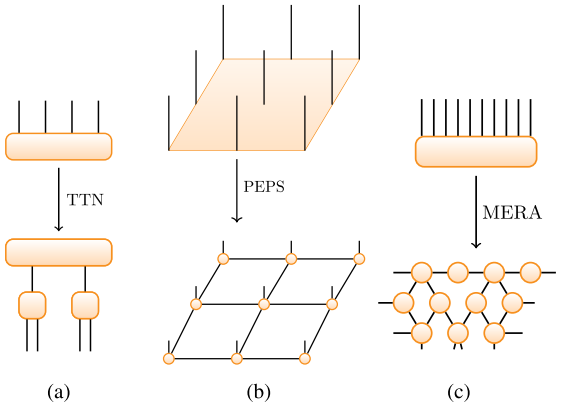

FIGURE 4. Diagrams of generalized tensor networks: tree tensor network (a), projected entangled-pair state (b), multiscale entanglement renormalization ansatz (c).

The computational time and space complexities of TN representations generally depend on several factors, including: the magnitude of the tensors' order involved, the magnitude of the bond dimension, the order in which tensor products are performed during a contraction, and whether one truncates some of the tensors to approximately (rather than exactly) represent the original state vector $|\psi\rangle$. Finding an optimal way to contract a TN is a NP-hard problem in general [12], [75]. This is proved by reduction from the NP-completeness of the tree-width problem [76], [77] or subset-sum problem [78]. Sometimes it is possible to efficiently find tensor decompositions using low-order tensors to make a "good" approximation [79], [80] and heuristic methods are often used to find contraction paths in TNs [27], [75], [81], [82], [83], [84].

Nevertheless, TNs can be adapted to describe different quantum or classical systems. Once a suitable TN representation is chosen, TN parameters are straightforward to change. Moreover, once a contraction path is known (whether optimal or not), it can be reused in other TNs with the same structure but different tensors. This inherent flexibility of TN representations makes them ideal for studying the time evolution of Hamiltonian models [85] and characterizing phase transitions [86].

IV. IMAGE CLASSIFICATION

Many machine-learning applications have been developed between 2019 and 2023 for classifying 2D digital images based on different TN representations:

- Classification of handwritten digits taken from the modified National Institute of Standards and Technology (MNIST) database [87], which is a subset of the NIST Special Database 19 [88], by means of MPS [89] or TTN [90], [91].
- Classification of clothes taken from the Fashion-MNIST database [92], by means of MPS [89], TTN [90] or PEPS [93].
- Classification of vehicles and animals taken from the ten-class Canadian Institute for Advanced Research (CIFAR-10) database [94], by means of TTN [91].
- Classification of COVID-19 pneumonia in X-ray chest images [95], by means of PEPS [93].
- Classification of top quarks and discrimination of quantum chromodynamics background noise in calorimeter images, by means of MPS [96], [97], TTN [97], or MERA [97].

The seminal work of Stoudenmire and Schwab in 2016 [98] introduced TNs for supervised machine learning. Their model consists in finding an optimal weight tensor $\mathbf{w}$ for a real-valued decision function $f(x) = \mathbf{w} \cdot \Phi(x)$ that classifies any input image $x = (x_1, \ldots, x_n)$ given by n grey-scale pixels $x_j \in (\mathbb{R} \cap [0, 1])$. The weight tensor encodes the strength of correlations between pixel values, and a MPS tensor network was proposed to represent it. The function $\Phi(x)$ is called feature map and transforms each pixel into a point with coordinates $\left(\cos\left(x_j \pi/2\right), \sin\left(x_j \pi/2\right)\right)$ in a unit circle, meaning that white pixels $x_j = 0$ map to the vector $(1, 0)$ and black pixels $x_j = 1$ map to the vector $(0, 1)$. These two vectors can be interpreted as qubits $|0\rangle$ and $|1\rangle$, respectively.

All the image-classification applications reviewed in this section build upon Stoudenmire and Schwab's work, using different TN representations of the weight tensor $\mathbf{w}$. The main performance metric used to benchmark these TN models is the classification accuracy achieved on unseen data samples (i.e., test accuracy). Other relevant aspects, such as training time, tensor-contraction time or memory usage are not benchmarked.

TNs have also been proposed for image generation, but this research is rather scarce and preliminary compared with image classification. Two examples are image generation for MNIST handwritten digits using TTN [99] and generation of phase diagram images for a 2D frustrated Heisenberg Hamiltonian by means of PEPS [100]. A general quantum machine learning model [101] was also proposed using a PEPS tensor network algorithm [102], which in theory can be applied to both classify and generate images.

A. MNIST, FASHION-MNIST, AND CIFAR-10

Novel TN representations may not always improve test accuracy on certain benchmarks. For the MNIST dataset [87], low-rank TTNs [90] achieve 98.3% test accuracy (for bond dimension $\chi = 8$ or higher), whereas hierarchical TTNs [91] achieve almost 95% test accuracy (for bond dimension $\chi = 10$). Yet both scores are lower than the 99% test accuracy already achieved by Stoudenmire and Schwab [98] for an MPS with $\chi = 120$.

In fact, achieving a test accuracy of 99% on MNIST is not challenging, as demonstrated by a convolutional neural

network called LeNet-5 [87] in 1998. Also, MNIST is not as computationally challenging for machine-learning methods as benchmarks like Fashion-MNIST [92] and CIFAR-10 [94] proposed more recently. On these benchmarks, the test accuracy achieved by TN models is lower: multilayered PEPS [93] obtain 90.44% on Fashion-MNIST (for $\chi = 5$); low-rank TTNs with tensor dropout [90] obtain 90.3% on Fashion-MNIST (for $\chi = 16$); and, hierarchical TTNs [91] obtain 75% on CIFAR-10 (for $\chi = 6$ or higher). For MPS with $\chi = 5$, an average classification accuracy of 92.2% can be achieved on Fashion-MNIST [89] at a lower computational cost than the aforementioned TTN and PEPS variants. This accuracy, however, was measured via the AUROC[2] metric: it is no longer recommended and can lead to overoptimistic results [103].

B. COVID-19 PNEUMONIA

Regarding the COVID-19 radiography dataset [95], multilayered PEPS[3] can achieve a test accuracy up to 91.63%, which is above the 87.08% test accuracy by standard PEPS [93]. This shows higher-dimensional TNs can achieve an image classification accuracy that is higher than TNs with simpler structures. However, the same authors [93] show that image classification models based on multilayered PEPS will also require more training parameters than those based on simpler TNs like PEPS, TTN, or MPS. The authors report, for instance, that PEPS requires 1 064 964 parameters for bond dimension $\chi = 4$, but a two-layer PEPS requires 1 394 102 parameters for $\chi = 3$ and 10 750 902 parameters for $\chi = 5$. The convolutional neural network GoogleLeNet (Inception v1) [105] requires 6 797 700 parameters[4] in total and achieves a 92.75% test accuracy on the COVID-19 radiography dataset [93]. Therefore, GoogleLeNet can achieve higher classification accuracy using fewer training parameters than a multilayered PEPS. Also, GoogleLeNet was released in 2015 [105] and has been superseded by newer convolutional neural networks [106], so the performance trade-offs between multilayered PEPS and simpler TNs or convolutional neural networks for classifying images are not clear yet.

C. TOP QUARKS

In contrast to the purely classical models above, Araz and Spannowsky [97] propose a quantum machine-learning model to classify the heaviest known elementary particles, called top quarks, in images produced by calorimeters at CERN's Large Hadron Collider. The ATLAS detector is used to generate such images, based on energy measurements from particles' collisions. In this model, input images are first encoded into an initial quantum state and then classified by a quantum circuit, with gates arranged in a TN topology, according to a decision function similar to Stoudenmire and Schwab's proposal [98]. Araz and Spannowsky [97] showed that image classification by quantum circuits with MPS, TTN, or MERA topologies involves notably fewer training parameters to achieve approximately the same or higher classification accuracy than a corresponding simulation on a classical computer (see Table 1). However, the accuracy was measured via the not-so-reliable AUROC metric [103] and therefore the validity of Araz and Spannowsky's benchmark results needs clarification.

One challenge faced by machine-learning models, particularly those based on gradient-descent methods, are barren plateaus or flat regions associated with the loss function used for training the model (see Section V-A). However, Araz and Spannowsky [97] also argue that their classification model is unlikely to suffer from barren plateaus if simulated on a quantum circuit whether using MPS, TTN, or MERA topologies. To show this, the authors analyse the eigenvalue distribution of the empirical Fisher information matrix [107] for the quantum TNs and their corresponding simulation on a classical computer. Such eigenvalue distribution can be used as an indirect measure of flatness in the loss function [107]: a model suffering from barren plateaus will have an increasing number of these eigenvalues around zero as the number of qubits of the model increases.

TABLE 1. Classification accuracy by tensor networks for top quark discrimination from background radiation noise in 2D calorimeter images.

Simulation	Tensor Network	# Training Parameters	Best Accuracy (%)
Classical	MPS	2 150	89.4
	TTN	14 800	89.6
	MERA	18 200	90.1
Quantum Circuit	MPS	9	88.6
	TTN	9	89.3
	MERA	17	91.4

V. OPTIMIZATION AND LOCAL HAMILTONIANS

A class of optimization problems that is central to quantum computing consists in finding a minimum eigenvalue λ and associated eigenvector $|\psi\rangle$ of a given Hamiltonian operator $H : \mathcal{H}^{\otimes n} \to \mathcal{H}^{\otimes n}$, on a Hilbert space $\mathcal{H}$ of n-qubit states, so that $H|\psi\rangle = \lambda|\psi\rangle$ holds [5], [108], [109]. H represents the energy function of a physical system, mapping a given ground state $|\psi\rangle$ to its corresponding ground-state energy value λ. Usually, the Hamiltonian can be expressed as a sum of sub-functions $H = \sum_{j=1}^{r} H_j$, where each sub-function H_j is locally defined on k qubits at most (given a fixed $k \leq n$) and the number of terms r is polynomial in n. In other words, one often can expand Hamiltonians as a finite series without exponentially many terms and limit the maximum number of interacting qubits to k. This refers to the class of k-local

[2]AUROC stands for "area under the receiver operating characteristic curve" [103].

[3]Multilayered PEPS [93] are a generalized class of TN representations based on PEPS. It should not be confused with the use of the term "two-layer PEPS" in [104] that refers to the contraction of two PEPS.

[4]Authors [93] incorrectly report 5 604 004 as GoogleLeNet's total number of parameters.

Hamiltonians [109], which is a generalization of maximum k-satisfiability problems for the quantum complexity class of QMA-complete problems (analogous to the NP-complete complexity class).

The subclass of 2-local Hamiltonians is noteworthy for two reasons. First, this subclass suffices to prove that any adiabatic quantum computation, performed by quantum annealers [110] for instance, can be efficiently simulated by the quantum circuit model and vice versa [109]. This equivalence suggests that, for all k-local Hamiltonians with $k \geq 2$, TN simulations of a quantum circuit can probably be performed by an adiabatic quantum computer (though this review focuses on quantum circuits). Second, 2-local Hamiltonians include popular Hamiltonian models such as the Lenz-Ising model of spin glasses based on Sherrington and Kirkpatrick's work [111]. This model has applications in physics, chemistry, biology, and combinatorial optimization [112]. In fact, many constrained optimization problems can be reformulated as spin-glass models or quadratic unconstrained binary optimization (QUBO) problems [112], [113], [114].

In the following, we cover TN applications in QUBO and related methods for dynamic portfolio problems in finance [115], a variational quantum algorithm for QUBO problem solving [116], and analysis of barren plateaus in variational quantum optimization [117]. Additionally, one can find spin-glass models for the so-called "dose optimization problem" in cancer radiotherapy, which can be solved via a TTN algorithm [118]. This is preliminary research, and we exclude it for the lack of clear performance advantages over state-of-the-art algorithms and comparison with alternative TN representations.

A. VARIATIONAL QUANTUM OPTIMIZATION AND BARREN PLATEAUS

Variational quantum algorithms (VQAs) are hybrid quantum-classical methods that iteratively approximate an optimal solution to QUBO problems, spin-glass models or k-local Hamiltonian problems more generally [108], [119]. In essence, VQAs use a parametrized quantum circuit to first generate an initial quantum state and compute its energy for a given Hamiltonian as an expectation value. This estimated value is used as an upper-bound of the ground-state energy or optimal solution. Based on this guess, a separate search algorithm run on a classical computer will heuristically update the circuit parameter values to generate an improved guess in the next VQA iteration until some convergence criteria are satisfied.

Unfortunately, convergence of VQAs towards an optimal solution can fail because of barren plateaus [120] among other major issues such as quantum noise, parameter initialization, and quantum state initialization [108], [119], [121]. Put simply, a barren plateau occurs when the expectation value of the energy or cost function resembles a flat surface, which leaves the VQA with no useful heuristic information (e.g., gradients) to update the circuit parameters, possibly leading to random search behavior [120].

Nevertheless, recent research [122] strongly suggests that barren plateaus can be avoided for certain VQAs if the:

1) circuit depth is restricted to grow no faster than logarithmically in the number of qubits (i.e., shallow circuits); and,
2) maximum number of qubit interactions at any one time is fixed to a finite and preferably "small" number (e.g., nearest neighbors), thus restricting the class of cost functions and associated optimization problems that the VQA can possibly solve.

These conditions were assumed by separate research to prove the absence of barren plateaus in TN representations of quantum circuits with MPS [123], [124], TTN, and MERA topologies [117], [123]. Moreover, shallow quantum circuits with local interactions can be efficiently simulated on classical computers by means of TN contractions [12], [13]. In short, the topology of TNs and the computational complexity of contractions can provide valuable analytic insight about the presence of barren plateaus in VQAs.

There are specific examples of VQAs, like "QuEnc" [116], designed to solve binary optimization problems with simple linear equality constraints by reformulating them as QUBO problems. Using a MPS representation, the authors show that the running time of classically simulating a shallow five-layer QuEnc circuit scales linearly as the number of qubits increases, taking no more than one second on a regular laptop for 300 qubits [116]. In terms of QUBO problem solving, the same authors also benchmarked QuEnc on a quantum computer (IBM's five-qubit `ibmq_manila`) and compared it against simulated annealing (SA) and Goemans-Williamson's (GW) algorithm. Based on randomized 256-node graph instances of the maximum cut problem, QuEnc's solution quality improves as its circuit depth is increased from 5 to 20. However, even with a 20-layer QuEnc circuit, the solution quality is still worse than SA and GW.

B. DYNAMIC PORTFOLIO OPTIMIZATION

The mean-variance model introduced by Markowitz [125] is the basis of many quantitative approaches to portfolio selection used in finance. It is a constrained optimization problem where the goal is to find a portfolio, that is a vector of proportions of a given capital for investment across assets, maximizing the expected return on the investment while minimizing financial risk. Optimal solutions in Markowitz's model can be found efficiently with classical solvers, but newer realistic models based on discrete formulations with additional constraints turn portfolio optimization into a mixed-integer programming problem that is computationally intractable [126], [127].

One example is dynamic portfolio optimization models, where portfolios are generalized from single-period to multi-period investments over a series of consecutive trading days. Researchers from Multiverse Computing [115] benchmarked a MPS-based algorithm as well as state-of-the-art quantum

and classical solvers against dynamic portfolio problems formulated as QUBO with up to 1 272 variables and all-to-all interaction pattern. The benchmark results show that the MPS algorithm achieved the best Sharpe ratios. Sharpe ratios measure solution quality as the proportion of expected return per unit of risk. The MPS algorithm also outperformed, regarding problem size scalability, a classical solver provided by Python's GEKKO library and two VQAs (implemented by the authors using IBM's quantum platform and Xanadu's PennyLane library). However, the MPS algorithm performed worse than GEKKO's classical solver in terms of solution quality measured via total profit (i.e., returns minus transaction costs) and worse than D-Wave's 2000Q quantum annealer in terms of running time. Overall, the best trade-off between Sharpe ratios and speed for solving dynamic portfolio optimization problems is attained by the MPS algorithm and D-Wave's 2000Q quantum annealer.

VI. MATERIALS SCIENCE AND QUANTUM CHEMISTRY

This section covers several applications proposed between 2013 and 2023 for analysis and discovery of materials, all of which are based on computing ground states of local Hamiltonians via classical TN simulations.

A. ARTIFICIAL GRAPHENE

Graphene is a prime example of nanomaterial that is made of a single layer of carbon atoms forming a 2D hexagonal structure. Graphene has many applications in energy storage, steel coating, and biomedical sensors [128] despite posing risks to biological systems [129]. The so-called ''artificial graphenes'' are materials with graphene-like properties that can be manufactured using alternative substrates such as aluminum gallium arsenide [130].

A quantum circuit was recently proposed to find the ground state of artificial graphene [131] based on a 2-local Hamiltonian model proposed by Hubbard [132]. The proposed circuit's depth grows linearly with the number of qubits, which suggests the circuit is not unreasonably deep even though TNs are more efficient on shallow circuits where depth grows logarithmically [12], [116], [117]. Finding the exact ground state by diagonalizing the Hamiltonian exceeds the memory limitations of the MareNostrum 4 supercomputer for circuits beyond 20 qubits or graphene lattices with more than two hexagons [131]. However, the same authors also show that approximating the ground state via full state-vector simulation is possible up to 32 qubits and, if using a MPS representation of the circuit, up to 36 qubits with 1% accuracy error relative to the true ground state. This result demonstrates that high-accuracy classical simulations of Hubbard's model are possible beyond the 24-qubit limit reached in past experiments with VQAs using no TN representation [133].

B. HYDROGEN CHAINS, ETHANE, AND ATAZANAVIR

China's fastest supercomputer, Sunway TaihuLight (SW26010 Pro), has been recently used to classically simulate a MPS-based VQA [134] for finding ground states of the following molecules: hydrogen, ethane, hydrogen chain of 500 atoms, and atazanavir. Atazanavir is a prescription medicine to treat the human immunodeficiency virus. The authors [134] claim these are the largest quantum-circuit simulations reported to date for a quantum-chemistry problem in terms of the total number of qubits (n) or CNOT gates (n_{CNOT}) involved:

- hydrogen H_2, $n = 92$, $n_{\text{CNOT}} = 1.4 \cdot 10^5$;
- ethane C_2H_6, $n = 32$, $n_{\text{CNOT}} = 4.4 \cdot 10^5$;
- hydrogen chain $(H_2)_{250}$, $n = 1000$, $n_{\text{CNOT}} = 10^6$; and,
- atazanavir $C_{38}H_{52}N_6O_7$, $n = 16$, $n_{\text{CNOT}} = 1.8 \cdot 10^6$.

The benchmarks for hydrogen and hydrogen chain molecules [134] show that MPS-VQA achieves enough chemical accuracy to match exact reference values of ground energies (i.e., full configuration interaction) obtained via Python's PySCF library. However, such level of chemical accuracy is not reported for neither ethane nor atazanavir molecules; in fact, the same authors suggest that improved accuracies can be achieved by using other VQA designs. Moreover, a single iteration of their proposed MPS-VQA takes more than 30 minutes to complete, using 512 cores of Sunway's supercomputer, for a hydrogen chain of 500 atoms. Taking more than 30 minutes for only one MPS-VQA iteration is arguably a long running time, and it aligns with the fact that TNs like MPS are not adequate for such deep VQA circuits (see Section V-A). More importantly, the benchmarks do not show what performance advantages does the proposed MPS-VQA provide over other state-of-the-art methods (whether classical, quantum, based on TNs or not). Full state-vector simulations of many useful quantum circuits, including VQAs, with 45 qubits and more have been successfully demonstrated in 2019 [42].

C. TREE-SHAPED MOLECULES

MPS is the most common representation used in TN algorithms to find the ground states of quantum-chemistry Hamiltonians [33], [34]. However, the electronic interaction pattern in certain tree-shaped molecules is not accurately described by the linear structure characteristic of MPS. Performance advantages of TTNs, in terms of lower running times or lower energy estimation errors during simulation, have been demonstrated using toy examples including: crystalline salts like lithium fluoride [135], Cayley trees formed by hydrogen atoms [136], as well as more realistic examples using nitrogen dimers and stilbenoid dendrimers (naturally occurring in plants) with up to 110 electrons and 110 active orbitals [136].

A decade later, research still continues to characterize the classes of quantum circuits where TTN can outperform MPS [137]. Using a single CPU (AMD Ryzen 7 3700) and 32 GB of RAM, it has been shown experimentally [137] up to 37 qubits that TTN scales exponentially better than MPS in terms of wall-clock time and bond dimension provided that: the circuits exactly match a well defined tree layout, are shallow, and have limited entanglement. However, this

result was obtained on artificial and carefully chosen quantum circuits that were an ideal fit for TTN. It is far from clear if and how such TTN performance advantage can be extrapolated to other problems involving approximately the same number of qubits, like artificial graphene with 36 qubits (Section VI-A) or ethane with 32 qubits (Section VI-B), for which top-class supercomputers were needed.

D. DISCOVERY OF PHYSICAL PHASES

Finding the ground state of a physical system can be a challenging task. One main reason is that it entails solving a local Hamiltonian problem [108], [109], which is computationally intractable in general. Another reason is that the ground state itself can vary depending on whether the physical system at hand undergoes sudden phase transitions according to changes in pressure, temperature, or a magnetic field force for example. Therefore, analyzing such phase transitions is a fundamental part of research in quantum computing and quantum physics at large. In fact, the existence of exotic phases, like topological quantum phases [138], provides a theoretical foundation to build universal quantum computers that are intrinsically and fully fault-tolerant at hardware level [139].

Quantum Monte Carlo (QMC) algorithms have been used to find phase transitions of Hamiltonian models but certain shortcomings of QMC recently motivated the use of TNs, such as PEPS for the Shastry-Sutherland model [140] and 2D isometric TNs (similar to PEPS) for the transverse-field Ising model [141]. These TN applications are part of fundamental research to develop new technologies. For instance, the Shastry-Sutherland model [142], with spins arranged on a 2D lattice with next-nearest neighbor interactions, is the only known model for which it is possible to find the exact ground states of an alkaline earth oxide material known as strontium copper borate $SrCu_2(BO_3)_2$. This material is relevant because it is thought to be a Mott-Hubbard insulator that can exhibit superconductivity [143], [144], and unknown phases of $SrCu_2(BO_3)_2$ have been discovered thanks to PEPS [140], [145].

VII. OTHER TRENDS IN TENSOR-NETWORK SIMULATION

This section covers emerging TN applications, proposed between 2014 and 2024, for other selected topics: computational fluid dynamics (Section VII-A), quantum advantage experiments (Section VII-B), and quantum error correction (Section VII-C).

A. COMPUTATIONAL FLUID DYNAMICS

The Navier-Stokes equations are nonlinear partial differential equations that have been traditionally used to model the time-dependent behavior of fluids. Except when simplifying assumptions are made, obtaining solutions to such equations by traditional methods, like direct numerical simulation (DNS), is computationally inefficient for classical computers. This becomes particularly evident if one considers realistic turbulent flows with complex geometries characterized by high Reynolds numbers [146].

A promising alternative to DNS methods was introduced in 2022 to solve the incompressible Navier-Stokes equations, by solving an associated linear optimization problem with a VQA-inspired MPS algorithm [147]. This new TN approach assumes that the local interactions featured in MPS can approximate well the local behavior of turbulent flows and that any correlations between spatially distant points are negligible.

A follow-up work by Kiffner and Jaksch [148] extends said TN approach to simulate flows with nonperiodic boundary conditions. Here, a MPS representation is also used to encode the flow velocities, but the algorithm is based on a classical DNS method instead. As a benchmark problem, Kiffner and Jaksch consider the lid-driven cavity model, a toy model for solving the incompressible Navier-Stokes equations: the flow is confined to a (discrete) square lattice in two spatial dimensions, and the flow density does not change over time. To justify the computational efficiency of this approach, the authors empirically show that the number of parameter variables describing the flow grows proportionally to the bond dimension of MPS. Also, the bond dimension grows logarithmically with simulation time. This leads to faster runtimes compared with DNS for high Reynolds numbers (Re): it achieves up to a 17-fold speedup for $\mathrm{Re} = 60.5 \cdot 10^3$. All benchmarks comparing DNS and their MPS-based method are implemented using MATLAB and run on a single CPU node (Intel Xeon Platinum 8268) of Oxford's Advanced Research Computing facility. However, authors warn that the performance advantage of MPS may degrade if simulation time or bond dimension increases significantly.

B. QUANTUM ADVANTAGE EXPERIMENTS

Quantum computers are expected to perform tasks that are computationally intractable for classical computers, even though it remains unclear which task is most appropriate to benchmark such quantum advantage as well as what quantum computer implementation can achieve it in practice and at what cost. Over two decades of research advances towards fault-tolerant quantum computation [149] elapsed, yet all current physical realizations of quantum computers perform noisy and error-prone quantum computations [8], [108]. These are often called noisy intermediate-scale quantum (NISQ) computers, without a commonly agreed and exact definition of how noisy or large.

One benchmark task for demonstrating quantum advantage is sampling fixed-length bitstrings from the output of a pseudo-random quantum circuit. This was popularized by an experiment in 2019 on Google's Sycamore superconducting quantum processor with 54 qubits arranged on a rectangular grid with nearest-neighbor interactions [150]. For classical computers, this task is regarded as computationally intractable principally due to the highly-entangled quantum states output by such random quantum circuits. Google

researchers [150] estimated that it would take 10 000 years for state-of-the-art supercomputers to compute one million samples from a random quantum circuit with 53 qubits and a depth of 20. However, such claim was conclusively refuted in 2022 [151] after several advances on massively parallel and efficient TN simulators [152], [153]. A TN algorithm, called sparse-state contraction, classically simulated and solved the sampling problem of Google's Sycamore quantum circuit within approximately 15 hours of wall-clock time using a cluster of 512 NVIDIA Tesla V100 GPUs [151]. There is also theoretical evidence arguing that random-circuit sampling can be approximately simulated by a classical algorithm running in polynomial time [154].

Another attempt to demonstrate a quantum advantage was conducted in 2023 for computing expectation values of the energy of a 2D transverse-field Ising Hamiltonian model [155]. Here, IBM's Eagle 127-qubit superconducting processor is benchmarked against MPS and 2D isoTNS classical simulators run on a single 64-core processor and 128 GB of memory. The transverse-field Ising model is chosen because it matches the IBM Eagle processor's topology. The authors [155] argue that running quantum circuits of that many qubits, with up to 60 layers of two-qubit gates and 2 880 CNOT gates, is out of reach for classical simulators. Once again, however, a remarkable follow-up work [156] showed for the same Ising model that a PEPS-based classical simulator not only can efficiently and accurately simulate IBM's Eagle processor but also IBM's Osprey and Condor newer quantum processors with 433 and 1 121 qubits respectively. Similar experiments demonstrating efficient simulations of TNs for the same transverse-field Ising model on 127 qubits have been conducted independently by other research groups [157], [158], [159].

By contrast, other experiments on practical quantum advantage conducted in 2024 suggest that state-of-the-art classical simulations of MPS and PEPS cannot match, by far, the same energy estimation accuracy or running time achieved by a quantum computer. This was shown by D-Wave [160] in quantum annealing experiments for a transverse-field Ising model (in two or more dimensions) [160] and, independently, by Fujitsu together with Osaka University [161] for quantum phase estimation using a 2D Hubbard model.

Nevertheless, the above unprecedented results provide evidence for the utility of TN-based simulators and refine the current benchmark baselines that future experiments will have to surpass to show quantum advantage.

C. QUANTUM ERROR CORRECTION

Performing quantum computations at an arbitrarily large scale beyond what classical computers can efficiently simulate is key to show a practical quantum advantage. Demonstrating this, however, is challenging: quantum noise easily destroys information encoded in quantum states and thus corrupts the result of quantum computations. Quantum error correction (QEC) methods enable fault-tolerant quantum computing at the expense of using many redundant physical qubits to implement a single, error-corrected, logical qubit [5], [8], [108], [149].

Recently, a general framework called Gleipnir [162] was proposed to analyze and quantify the presence of quantum errors in quantum circuits. The framework relies on MPS with truncated bond dimension to efficiently represent quantum states and to compute a certain distance metric, specifically a diamond norm, for estimating quantum errors in quantum states. To compute such diamond norm, Gleipnir solves an associated semi-definite programming (SDP) problem whose size scales exponentially with the maximum number of qubits used by quantum gates in a given circuit. However, all quantum gates are assumed to have two input qubits at most, as NISQ computers are unlikely to support more. Therefore, Gleipnir assumes such SDP is constant-sized so that the diamond norm can be computed efficiently. It also assumes that the noiseless quantum state (used as reference to compute said diamond norm) is known in advance. Under a simple bit-flip noise model, Gleipnir can provide error bounds 15% to 30% tighter than known diamond norm estimates, as shown by benchmarks [162] on: a quantum approximate optimization algorithm (i.e., a form of VQA, see Section V-A) and a Lenz-Ising model with up to 100 qubits and 2 265 quantum gates. Furthermore, using said diamond norm, Gleipnir can guide quantum program compilers on how to best map physical qubit to logical qubits for noise reduction given a specific quantum hardware architecture. An example of this is shown for three-qubit and five-qubit GHZ states on IBM's Boeblingen 20-qubit superconducting quantum computer [162].

TN simulators improved the scalability of QEC methods as well. In particular, exact and approximate PEPS simulations of error correction via surface codes, with more than 100 data qubits, have been demonstrated [163] under two realistic noise models: amplitude-damping and systematic-rotation noise models. QEC methods based on other TN topologies including MERA were already formulated in 2014 [164].

VIII. DISCUSSION

TNs can speed up and reduce memory usage of classical simulations for certain quantum circuits, while sacrificing accuracy by approximately rather than exactly representing quantum states. Their computational efficiency, the expressiveness to represent general quantum physical systems, and scalability via massively parallel hardware, are well-known advantages of TN methods that make them a viable alternative to full state-vector representations. This is reflected in the wide range of TN applications developed, particularly during the last decade, as reviewed in this paper.

In practice, however, whether a TN shows performance advantages depends on many different factors including: the specific TN structure, choice of TN contraction algorithm, critical TN parameters like bond dimension, performance

metrics benchmarked, and properties of the quantum circuit itself being simulated such as circuit depth or entanglement.

TABLE 2. Tensor network applications.

Research Area	Tensor Network	Application Description	Ref.
Machine learning	MPS	One-class linear classifier for anomaly detection in MNIST and Fashion-MNIST greyscale images	[89]
Machine learning	TTN	Binary classifier for MNIST greyscale and CIFAR-10 coloured images	[91]
Machine learning	MPS, TTN, PEPS	Multiclass classifier for MNIST and Fashion-MNIST greyscale images	[90]
Machine learning	MPS	Classification of quarks in calorimeter images generated at the Large Hadron Collider	[96]
Machine learning	MPS, TTN, MERA	Classification of quarks in calorimeter images generated at the Large Hadron Collider	[97]
Machine learning	MPS, PEPS	Multiclass classifier for Fashion-MNIST and COVID-19 X-ray chest images	[93]
Machine learning	MPS, TTN	Image generation of MNIST handwritten digits	[99]
Machine learning	PEPS	Generation of phase diagram images for a two-dimensional, frustrated, bilayer, Heisenberg Hamiltonian model	[100]
Optim.	MPS	Multiperiod mean-variance portfolio optimization	[115]
Optim.	TTN	Dose optimization in intensity-modulated radiation therapy for cancer treatment	[118]
Optim.	MPS	Analysis of a variational classical-quantum algorithm for solving QUBO problems	[116]
Optim.	MPS, TTN, MERA	Analysis of barren plateaus in cost functions in variational quantum optimization	[117], [123]
Materials science	MPS	Energy function minimization for a Hubbard Hamiltonian model of artificial graphene	[131]
Materials science	PEPS	Analysis of energy ground states for strontium copper borate, described by the Shastry-Sutherland model	[140], [145]
Materials science	MPS, 2D isoTNS	Computing thermal states for a 2D transverse-field Ising Hamiltonian model	[141]
Quantum chemistry	MPS	Analysis of hydrogen chains, torsional barrier of ethane and protein-ligand interactions in SARS-CoV-2	[134]
Quantum chemistry	MPS, TTN	Energy function minimization for a Hubbard Hamiltonian model of lithium fluoride	[135]
Quantum chemistry	MPS, TTN	Energy function minimization for a Hubbard model of tree-shaped molecules	[136]
Quantum simulation	MPS, TTN	Reducing simulation time for certain quantum circuits with a tree-shaped layout	[137]
Quantum simulation	PEPS	Approximate simulation of random quantum circuits including Google's Sycamore	[152], [153]

For example, compared with TTN and PEPS, image classification models based on MPS require fewer training parameters, especially if implemented on quantum computers instead of classically simulating them (see Section IV). Yet novel models based on MPS, TTN or PEPS, struggle to outperform or even match state-of-the-art convolutional neural networks in terms of classification test accuracy. Nevertheless, there exist applications in quantum many-body physics where certain TNs outperform others generally, for example: TTN for finding ground states of certain tree-shaped molecules (Section VI-C); and, PEPS for finding ground states of transverse-field Ising models with spins arranged on a 2D lattice as shown in quantum advantage experiments (Section VII-B). By contrast, TN applications based on MERA are scarce across all research domains reviewed, arguably due to: the lack of efficient contraction algorithms for MERA and the fact that already many high-dimensional quantum physical systems can be represented via TTN or PEPS at a lower computational cost.

Not surprisingly, one of the main current challenges is designing standardized benchmark suites and good practices to rigorously evaluate the performance of TN applications. This is especially important given the vast number of quantum circuit simulators available [23], [26]. In fact, this challenge is not specific to TN software but common to quantum-computing software in general [165]. Some pitfalls in experiment settings we found are, for example: benchmarking only one aspect of the application (e.g., test accuracy for applications in image classification); using unreliable metrics for classification like AUROC [103]; and, measuring wall-clock time but not number of cost/energy function evaluations, which is a more robust hardware-agnostic metric and often used in runtime algorithm analysis. We expect that future TN applications will benefit from recent developments [166], [167], [168], [169] in benchmark suites for quantum-computing applications and related software.

TABLE 4. Tensor network applications (continued).

Research Area	Tensor Network	Application Description	Ref.
Quantum simulation	MPS, 2D isoTNS	Benchmark the IBM Eagle 127-qubit processor for a 2D transverse-field Ising Hamiltonian model	[155]
Quantum simulation	MPS, PEPS, 2D isoTNS	Benchmark the IBM Eagle 127-qubit processor for a 2D transverse-field Ising Hamiltonian model	[156]
Comp. fluid dynamics	MPS	Solving the incompressible Navier-Stokes equations	[147], [148]
Quantum error correction	PEPS	Noise cancellation in quantum circuits made of non-Clifford gates via surface codes	[163]
Quantum error analysis	MPS	Error bounds estimation in noisy quantum programs	[162]

The rapid growth of TN applications and related software during the last decade has been enabled by the wealth of TN algorithms in the literatures [16], [17], [18], [19], [20], [21], [27], [28], and [29]. However, one notably less explored yet promising direction for future research is the application of hybrid methods based on TNs and other known approaches to quantum-circuit simulation. Two potential candidates that we found are tensor-based decision diagrams [52] and tensor-based circuit cutting [51]. Another topic unexplored in this review is TNs applications in open quantum systems [36]. TTN-based algorithms have been proposed, for example, for solving the "hierarchical equations of motion" [170] describing how quantum many-body systems interact with a surrounding environment and the effect of quantum errors (e.g., due to quantum decoherence). Since this paper focused on TN applications, a third valuable piece of future work is to carry a separate in-depth review focusing on TN performance based on those applications, in terms of their theoretical computational complexity as well as a comparative experimental analysis. Therefore, our survey serves as a starting point for said future work as well as a good opportunity for addressing the pitfalls that we identified in some experimental settings and incorporate the advances in benchmarking standards mentioned earlier.

IX. CONCLUSION

TNs constitute a well-established research area in both classical and quantum computing. However, it is challenging to keep track and have a general view of TNs' latest advances due to the wide range of existing TN methods and applications. Although some previous works review the theory of TN methods and certain applications, there has been a clear lack of reviews dedicated to TN applications across research domains. Overall, this review provides a representative and up-to-date account of many state-of-the-art TN applications across many fields of interest. We believe this work is useful not just to quickly grasp major trends and hot topics about TNs but also reach to readers from other research communities that may be unfamiliar with TNs.

The following Tables 2–4 summarize key TN applications in this review. Applications are separated by rows with the column fields: bibliographic reference that introduces the application; research domain where the application focuses as presented by the authors; brief description of the application; and, TN class (or classes) used in such application.

ACKNOWLEDGMENT

The authors are deeply grateful to Dr. Artur Garcia Sáez, a leading Researcher at Barcelona Supercomputing Center, Spain, for his helpful feedback and expertise on tensor networks that have positively contributed to the draft of this article. They also appreciate his suggestions of relevant articles and topics regarding TN applications in quantum computing. They thank Sergio Sánchez-Ramírez, a Researcher at Barcelona Supercomputing Center, for the helpful discussions about tensor networks.

REFERENCES

[1] R. P. Feynman, "Simulating physics with computers," *Int. J. Theor. Phys.*, vol. 21, nos. 6–7, pp. 467–488, Jun. 1982, doi: 10.1007/bf02650179.
[2] A. M. Turing, "On computable numbers, with an application to the entscheidungsproblem," *Proc. London Math. Soc.*, vols. s2–42, no. 1, pp. 230–265, 1937, doi: 10.1112/plms/s2-42.1.230.
[3] D. Deutsch, "Quantum theory, the Church–Turing principle and the universal quantum computer," *Proc. Roy. Soc. London A*, vol. 400, no. 1818, pp. 97–117, 1985, doi: 10.1098/rspa.1985.0070.
[4] E. Bernstein and U. Vazirani, "Quantum complexity theory," in *Proc. 25th Annu. ACM Symp. Theory Comput. (STOC)*, 1993, pp. 11–20, doi: 10.1145/167088.167097.
[5] M. A. Nielsen and I. L. Chuang, *Quantum Computation and Quantum Information*, 10th ed., Cambridge, U.K.: Cambridge Univ. Press, 2010, doi: 10.1017/CBO9780511976667.
[6] P. A. M. Dirac, *The Principles of Quantum Mechanics* (International Series of Monographs on Physics), 4th ed., Oxford, U.K.: Clarendon Press, 1982.
[7] S. Aaronson and L. Chen, "Complexity-theoretic foundations of quantum supremacy experiments," in *Proc. 32nd Comput. Complex. Conf. (CCC)*, in Leibniz International Proceedings in Informatics, vol. 79, Dagstuhl, Germany, R. O'Donnell, Ed., Jul. 2017, pp. 1–67, doi: 10.4230/LIPIcs.CCC.2017.22.
[8] A. J. Daley, I. Bloch, C. Kokail, S. Flannigan, N. Pearson, M. Troyer, and P. Zoller, "Practical quantum advantage in quantum simulation," *Nature*, vol. 607, no. 7920, pp. 667–676, Jul. 2022, doi: 10.1038/s41586-022-04940-6.
[9] D. E. Deutsch, "Quantum computational networks," *Proc. Roy. Soc. London A, Math., Phys. Eng. Sci.*, vol. 425, no. 1868, pp. 73–90, 1989. doi: 10.1098/rspa.1989.0099.
[10] D. Gottesman, "The Heisenberg representation of quantum computers," in *Proc. 22nd Int. Colloq. Group Theor. Methods Phys.*, 1998, pp. 32–43, doi: 10.48550/arXiv.quant-ph/9807006.
[11] L. G. Valiant, "Quantum circuits that can be simulated classically in polynomial time," *SIAM J. Comput.*, vol. 31, no. 4, pp. 1229–1254, Jan. 2002, doi: 10.1137/s0097539700377025.
[12] I. L. Markov and Y. Shi, "Simulating quantum computation by contracting tensor networks," *SIAM J. Comput.*, vol. 38, no. 3, pp. 963–981, Jan. 2008, doi: 10.1137/050644756.
[13] G. Vidal, "Efficient classical simulation of slightly entangled quantum computations," *Phys. Rev. Lett.*, vol. 91, no. 14, Oct. 2003, doi: 10.1103/physrevlett.91.147902.
[14] H. Guo, *What Are Tensors Exactly?*. Singapore: World Scientific, Jun. 2021, doi: 10.1142/12388.

[15] I. Affleck, T. Kennedy, E. H. Lieb, and H. Tasaki, "Rigorous results on valence-bond ground states in antiferromagnets," *Phys. Rev. Lett.*, vol. 59, no. 7, pp. 799–802, Aug. 1987, doi: 10.1103/physrevlett.59.799.

[16] R. Orús, "A practical introduction to tensor networks: Matrix product states and projected entangled pair states," *Ann. Phys.*, vol. 349, pp. 117–158, Oct. 2014, doi: 10.1016/j.aop.2014.06.013.

[17] M. C. Bañuls, "Tensor network algorithms: A route map," *Annu. Rev. Condens. Matter Phys.*, vol. 14, no. 1, pp. 173--191, Mar. 2023, doi: 10.1146/annurev-conmatphys-040721-022705.

[18] J. C. Bridgeman and C. T. Chubb, "Hand-waving and interpretive dance: An introductory course on tensor networks," *J. Phys. A, Math. Theor.*, vol. 50, no. 22, Jun. 2017, Art. no. 223001, doi: 10.1088/1751-8121/aa6dc3.

[19] K. Okunishi, T. Nishino, and H. Ueda, "Developments in the tensor network—From statistical mechanics to quantum entanglement," *J. Phys. Soc. Jpn.*, vol. 91, no. 6, Jun. 2022, doi: 10.7566/jpsj.91.062001.

[20] R. Orús, "Tensor networks for complex quantum systems," *Nature Rev. Phys.*, vol. 1, no. 9, pp. 538–550, Aug. 2019, doi: 10.1038/s42254-019-0086-7.

[21] U. Schollwöck, "The density-matrix renormalization group in the age of matrix product states," *Ann. Phys.*, vol. 326, no. 1, pp. 96--192, Jan. 2011, doi: 10.1016/j.aop.2010.09.012.

[22] S. R. White, "Density-matrix algorithms for quantum renormalization groups," *Phys. Rev. B, Condens. Matter*, vol. 48, no. 14, pp. 10345–10356, Oct. 1993, doi: 10.1103/physrevb.48.10345.

[23] C. Psarras, L. Karlsson, J. Li, and P. Bientinesi, "The landscape of software for tensor computations," 2021, *arXiv:2103.13756*.

[24] J. Provazza, K. Gunst, H. Zhai, G. K.-L. Chan, T. Shiozaki, N. C. Rubin, and A. F. White, "Fast emulation of fermionic circuits with matrix product states," *J. Chem. Theory Comput.*, vol. 20, no. 9, pp. 3719–3728, May 2024, doi: 10.1021/acs.jctc.4c00200.

[25] C. D. Sherrill, D. E. Manolopoulos, T. J. Martínez, and A. Michaelides, "Electronic structure software," *J. Chem. Phys.*, vol. 153, no. 7, Aug. 2020, doi: 10.1063/5.0023185.

[26] K. Young, M. Scese, and A. Ebnenasir, "Simulating quantum computations on classical machines: A survey," 2023, *arXiv:2311.16505*.

[27] G. Evenbly, "A practical guide to the numerical implementation of tensor networks I: Contractions, decompositions, and gauge freedom," *Frontiers Appl. Math. Statist.*, vol. 8, pp. 1–14, Jun. 2022, doi: 10.3389/fams.2022.806549.

[28] L. Grasedyck, D. Kressner, and C. Tobler, "A literature survey of low-rank tensor approximation techniques," *GAMM-Mitteilungen*, vol. 36, no. 1, pp. 53–78, Aug. 2013, doi: 10.1002/gamm.201310004.

[29] T. G. Kolda and B. W. Bader, "Tensor decompositions and applications," *SIAM Rev.*, vol. 51, no. 3, pp. 455–500, Aug. 2009, doi: 10.1137/07070111x.

[30] A. Cichocki, N. Lee, I. Oseledets, A.-H. Phan, Q. Zhao, M. Sugiyama, and D. P. Mandic, "Tensor networks for dimensionality reduction and large-scale optimization: Part 2 applications and future perspectives," *Found. Trends Mach. Learn.*, vol. 9, no. 6, pp. 431–673, 2017, doi: 10.1561/2200000067.

[31] Y. Ji, Q. Wang, X. Li, and J. Liu, "A survey on tensor techniques and applications in machine learning," *IEEE Access*, vol. 7, pp. 162950–162990, 2019, doi: 10.1109/ACCESS.2019.2949814.

[32] J. J. García-Ripoll, "Quantum-inspired algorithms for multivariate analysis: From interpolation to partial differential equations," *Quantum*, vol. 5, p. 431, Apr. 2021, doi: 10.22331/q-2021-04-15-431.

[33] A. Baiardi and M. Reiher, "The density matrix renormalization group in chemistry and molecular physics: Recent developments and new challenges," *J. Chem. Phys.*, vol. 152, no. 4, Jan. 2020, doi: 10.1063/1.5129672.

[34] S. Szalay, M. Pfeffer, V. Murg, G. Barcza, F. Verstraete, R. Schneider, and Ö. Legeza, "Tensor product methods and entanglement optimization for *ab initio* quantum chemistry," *Int. J. Quantum Chem.*, vol. 115, no. 19, pp. 1342–1391, Oct. 2015, doi: 10.1002/qua.24898.

[35] D. Jaksch, P. Givi, A. J. Daley, and T. Rung, "Variational quantum algorithms for computational fluid dynamics," *AIAA J.*, vol. 61, no. 5, pp. 1885–1894, May 2023, doi: 10.2514/1.j062426.

[36] D. Jaschke, S. Montangero, and L. D. Carr, "One-dimensional many-body entangled open quantum systems with tensor network methods," *Quantum Sci. Technol.*, vol. 4, no. 1, Nov. 2018, Art. no. 013001, doi: 10.1088/2058-9565/aae724.

[37] I. M. Georgescu, S. Ashhab, and F. Nori, "Quantum simulation," *Rev. Mod. Phys.*, vol. 86, no. 1, pp. 153–185, Mar. 2014, doi: 10.1103/revmodphys.86.153.

[38] L. Gyongyosi and S. Imre, "A survey on quantum computing technology," *Comput. Sci. Rev.*, vol. 31, pp. 51–71, Feb. 2019, doi: 10.1016/j.cosrev.2018.11.002.

[39] R. P. Feynman, "Space-time approach to non-relativistic quantum mechanics," *Rev. Mod. Phys.*, vol. 20, no. 2, pp. 367–387, Apr. 1948, doi: 10.1103/revmodphys.20.367.

[40] Z.-Y. Chen, Q. Zhou, C. Xue, X. Yang, G.-C. Guo, and G.-P. Guo, "64-qubit quantum circuit simulation," *Sci. Bull.*, vol. 63, no. 15, pp. 964–971, Aug. 2018, doi: 10.1016/j.scib.2018.06.007.

[41] K. De Raedt, K. Michielsen, H. De Raedt, B. Trieu, G. Arnold, M. Richter, T. Lippert, H. Watanabe, and N. Ito, "Massively parallel quantum computer simulator," *Comput. Phys. Commun.*, vol. 176, no. 2, pp. 121–136, Jan. 2007, doi: 10.1016/j.cpc.2006.08.007.

[42] X.-C. Wu, S. Di, E. M. Dasgupta, F. Cappello, H. Finkel, Y. Alexeev, and F. T. Chong, "Full-state quantum circuit simulation by using data compression," in *Proc. Int. Conf. High Perform. Comput., Netw., Storage Anal.*, New York, NY, USA, Nov. 2019, pp. 1–24, doi: 10.1145/3295500.3356155.

[43] G. G. Guerreschi, J. Hogaboam, F. Baruffa, and N. P. D. Sawaya, "Intel quantum simulator: A cloud-ready high-performance simulator of quantum circuits," *Quantum Sci. Technol.*, vol. 5, no. 3, May 2020, Art. no. 034007, doi: 10.1088/2058-9565/ab8505.

[44] L. Burgholzer, A. Ploier, and R. Wille, "Tensor networks or decision diagrams? Guidelines for classical quantum circuit simulation," 2023, *arXiv:2302.06616*.

[45] R. E. Bryant, "Graph-based algorithms for Boolean function manipulation," *IEEE Trans. Comput.*, vol. C35, no. 8, pp. 677–691, Aug. 1986, doi: 10.1109/TC.1986.1676819.

[46] D. M. Miller and M. A. Thornton, "QMDD: A decision diagram structure for reversible and quantum circuits," in *Proc. 36th Int. Symp. Multiple-Valued Log. (ISMVL)*, Singapore, Jun. 2006, p. 30, doi: 10.1109/ISMVL.2006.35.

[47] P. Niemann, R. Wille, D. M. Miller, M. A. Thornton, and R. Drechsler, "QMDDs: Efficient quantum function representation and manipulation," *IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst.*, vol. 35, no. 1, pp. 86–99, Jan. 2016, doi: 10.1109/TCAD.2015.2459034.

[48] A. Zulehner and R. Wille, "Advanced simulation of quantum computations," *IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst.*, vol. 38, no. 5, pp. 848–859, May 2019, doi: 10.1109/TCAD.2018.2834427.

[49] I. L. Markov, A. Fatima, S. V. Isakov, and S. Boixo, "Quantum supremacy is both closer and farther than it appears," 2018, *arXiv:1807.10749*.

[50] L. Burgholzer, H. Bauer, and R. Wille, "Hybrid Schrödinger–Feynman simulation of quantum circuits with decision diagrams," in *Proc. IEEE Int. Conf. Quantum Comput. Eng. (QCE)*, Broomfield, CO, USA, Oct. 2021, pp. 199–206, doi: 10.1109/QCE52317.2021.00037.

[51] D. Guala, S. Zhang, E. Cruz, C. A. Riofrío, J. Klepsch, and J. M. Arrazola, "Practical overview of image classification with tensor-network quantum circuits," *Sci. Rep.*, vol. 13, no. 1, p. 4427, Mar. 2023, doi: 10.1038/s41598-023-30258-y.

[52] X. Hong, X. Zhou, S. Li, Y. Feng, and M. Ying, "A tensor network based decision diagram for representation of quantum circuits," *ACM Trans. Des. Autom. Electron. Syst.*, vol. 27, no. 6, pp. 1–30, Nov. 2022, doi: 10.1145/3514355.

[53] T. Häner, D. S. Steiger, M. Smelyanskiy, and M. Troyer, "High performance emulation of quantum circuits," in *Proc. Int. Conf. High Perform. Comput., Netw., Storage Anal.*, Salt Lake City, UT, USA, Nov. 2016, pp. 866–874, doi: 10.1109/SC.2016.73.

[54] M. Soltanolkotabi and E. J. Candés, "A geometric analysis of subspace clustering with outliers," *Ann. Statist.*, vol. 40, no. 4, pp. 2195–2238, Aug. 2012, doi: 10.1214/12-aos1034.

[55] H. Gao, F. Nie, X. Li, and H. Huang, "Multi-view subspace clustering," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Santiago, Chile, Dec. 2015, pp. 4238–4246, doi: 10.1109/ICCV.2015.482.

[56] Y. Liu, J. Chen, Y. Lu, W. Ou, Z. Long, and C. Zhu, "Adaptively topological tensor network for multi-view subspace clustering," *IEEE Trans. Knowl. Data Eng.*, vol. 36, no. 11, pp. 5562–5575, Nov. 2024, doi: 10.1109/TKDE.2024.3391627.

[57] Z. Long, C. Zhu, J. Chen, Z. Li, Y. Ren, and Y. Liu, "Multi-view MERA subspace clustering," *IEEE Trans. Multimedia*, vol. 26, pp. 3102–3112, 2024, doi: 10.1109/TMM.2023.3307239.

[58] G. Liu, Z. Lin, S. Yan, J. Sun, Y. Yu, and Y. Ma, "Robust recovery of subspace structures by low-rank representation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 1, pp. 171–184, Jan. 2013, doi: 10.1109/TPAMI.2012.88.
[59] Y. Xie, D. Tao, W. Zhang, Y. Liu, L. Zhang, and Y. Qu, "On unifying multi-view self-representations for clustering by tensor multi-rank minimization," *Int. J. Comput. Vis.*, vol. 126, no. 11, pp. 1157–1179, Nov. 2018, doi: 10.1007/s11263-018-1086-2.
[60] X. Yang, H. Che, and M.-F. Leung, "Tensor-based unsupervised feature selection for error-robust handling of unbalanced incomplete multi-view data," *Inf. Fusion*, vol. 114, Feb. 2025, Art. no. 102693, doi: 10.1016/j.inffus.2024.102693.
[61] J. J. Gerbrands, "On the relationships between SVD, KLT and PCA," *Pattern Recognit.*, vol. 14, nos. 1–6, pp. 375–381, Jan. 1981, doi: 10.1016/0031-3203(81)90082-0.
[62] X. Ma, P. Zhang, S. Zhang, N. Duan, Y. Hou, M. Zhou, and D. Song, "A tensorized transformer for language modeling," in *Proc. 33rd Int. Conf. Neural Inf. Process. Syst.*, vol. 32, H. M. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché Buc, and E. B. Fox, Eds., Vancouver, BC, Canada: Curran Associates, Jun. 2019, pp. 2229–2239. [Online]. Available: https://dl.acm.org/doi/abs/10.5555/3454287.3454487
[63] I. Balazevic, C. Allen, and T. Hospedales, "TuckER: Tensor factorization for knowledge graph completion," in *Proc. Conf. Empirical Methods Natural Lang. Process., 9th Int. Joint Conf. Natural Lang. Process. (EMNLP-IJCNLP)*. Hong Kong, China: Association for Computational Linguistics, 2019, pp. 5185–5194, doi: 10.18653/v1/d19-1522.
[64] R. Socher, D. Chen, C. D. Manning, and A. Y. Ng, "Reasoning with neural tensor networks for knowledge base completion," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 26, C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Weinberger, Eds., Red Hook, NY, USA: Curran Associates, Dec. 2013, pp. 926–934. [Online]. Available: https://dl.acm.org/doi/10.5555/2999611.2999715
[65] A. Novikov, D. Podoprikhin, A. Osokin, and D. Vetrov, "Tensorizing neural networks," in *Proc. 28th Int. Conf. Neural Inf. Process. Syst.*, vol. 1. Cambridge, MA, USA: MIT Press, Dec. 2015, pp. 442–450. [Online]. Available: https://dl.acm.org/doi/10.5555/2969239.2969289
[66] K. Hegde, H. Asghari-Moghaddam, M. Pellauer, N. Crago, A. Jaleel, E. Solomonik, J. Emer, and C. W. Fletcher, "ExTensor: An accelerator for sparse tensor algebra," in *Proc. 52nd Annu. IEEE/ACM Int. Symp. Microarchit.*, Columbus, OH, USA, Oct. 2019, pp. 319–333, doi: 10.1145/3352460.3358275.
[67] F. Kjolstad, S. Kamil, S. Chou, D. Lugato, and S. Amarasinghe, "The tensor algebra compiler," *Proc. ACM Program. Lang.*, vol. 1, pp. 1–29, Oct. 2017, doi: 10.1145/3133901.
[68] T. Zhou, R. Tian, R. A. Ashraf, R. Gioiosa, G. Kestor, and V. Sarkar, "ReACT: Redundancy-aware code generation for tensor expressions," in *Proc. Int. Conf. Parallel Archit. Compilation Techn.*, Chicago, IL, USA, Oct. 2022, pp. 1–13, doi: 10.1145/3559009.3569685.
[69] A. Auddy, D. Xia, and M. Yuan, "Tensor methods in high dimensional data analysis: Opportunities and challenges," 2024, *arXiv:2405.18412*.
[70] A. Cichocki, N. Lee, I. Oseledets, A.-H. Phan, Q. Zhao, and D. P. Mandic, "Tensor networks for dimensionality reduction and large-scale optimization: Part 1 low-rank tensor decompositions," *Found. Trends Mach. Learn.*, vol. 9, nos. 4–5, pp. 249–429, 2016, doi: 10.1561/2200000059.
[71] Y.-Y. Shi, L.-M. Duan, and G. Vidal, "Classical simulation of quantum many-body systems with a tree tensor network," *Phys. Rev. A, Gen. Phys.*, vol. 74, no. 2, Aug. 2006, doi: 10.1103/physreva.74.022320.
[72] 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, Condens. Matter*, vol. 80, no. 23, Dec. 2009, doi: 10.1103/physrevb.80.235127.
[73] F. Verstraete and J. I. Cirac, "Renormalization algorithms for quantum-many body systems in two and higher dimensions," 2004, *arXiv:cond-mat/0407066*.
[74] M. P. Zaletel and F. Pollmann, "Isometric tensor network states in two dimensions," *Phys. Rev. Lett.*, vol. 124, no. 3, Jan. 2020, doi: 10.1103/physrevlett.124.037201.
[75] J. Gray and S. Kourtis, "Hyper-optimized tensor network contraction," *Quantum*, vol. 5, p. 410, Mar. 2021, doi: 10.22331/q-2021-03-15-410.
[76] E. Amir, "Approximation algorithms for treewidth," *Algorithmica*, vol. 56, no. 4, pp. 448–479, Apr. 2010, doi: 10.1007/s00453-008-9180-4.
[77] S. Arnborg, D. G. Corneil, and A. Proskurowski, "Complexity of finding embeddings in a k-tree," *SIAM J. Algebr. Discrete Methods*, vol. 8, no. 2, pp. 277–284, Apr. 1987, doi: 10.1137/0608024.
[78] L. Chi-Chung, P. Sadayappan, and R. Wenger, "On optimizing a class of multi-dimensional loops with reduction for parallel execution," *Parallel Process. Lett.*, vol. 7, no. 2, pp. 157–168, Jun. 1997, doi: 10.1142/s0129626497000176.
[79] V. de Silva and L.-H. Lim, "Tensor rank and the ill-posedness of the best low-rank approximation problem," *SIAM J. Matrix Anal. Appl.*, vol. 30, no. 3, pp. 1084–1127, Jan. 2008, doi: 10.1137/06066518x.
[80] N. Schuch, M. M. Wolf, F. Verstraete, and J. I. Cirac, "Computational complexity of projected entangled pair states," *Phys. Rev. Lett.*, vol. 98, no. 14, Apr. 2007, doi: 10.1103/physrevlett.98.140506.
[81] D. I. Lyakh, T. Nguyen, D. Claudino, E. Dumitrescu, and A. J. McCaskey, "ExaTN: Scalable GPU-accelerated high-performance processing of general tensor networks at exascale," *Frontiers Appl. Math. Statist.*, vol. 8, pp. 1–13, Jul. 2022, doi: 10.3389/fams.2022.838601.
[82] S. Orgler and M. Blacher, "Optimizing tensor contraction paths: A greedy algorithm approach with improved cost functions," 2024, *arXiv:2405.09644*.
[83] S. Sánchez-Ramírez, J. Vallès-Muns, and A. Garcia-Saez, "EinExprs: Contraction paths of tensor networks as symbolic expressions," 2024, *arXiv:2403.18030*.
[84] D. G. A. Smith and J. Gray, "Opt_einsum—A Python package for optimizing contraction order for einsum-like expressions," *J. Open Source Softw.*, vol. 3, no. 26, p. 753, Jun. 2018, doi: 10.21105/joss.00753.
[85] U. Schollwöck, "Matrix product state algorithms: DMRG, TEBD and relatives," in *Strongly Correlated Systems: Numerical Methods*, vol. 176, A. Avella and F. Mancini, Eds., Cham, Switzerland: Springer, 2013, pp. 67–98, doi: 10.1007/978-3-642-35106-8_3.
[86] R. Orús, A. C. Doherty, and G. Vidal, "First order phase transition in the anisotropic quantum orbital compass model," *Phys. Rev. Lett.*, vol. 102, no. 7, Feb. 2009, Art. no. 077203, doi: 10.1103/physrevlett.102.077203.
[87] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proc. IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998, doi: 10.1109/5.726791.
[88] P. Grother and K. Hanaoka, "NIST special database 19: Handprinted forms and characters 2nd edition," Nat. Inst. Standards Technol., Gaithersburg, MD, USA, Tech. Rep., 2016, doi: 10.18434/T4H01C. [Online]. Available: https://s3.amazonaws.com/nist-srd/SD19/sd19_users_guide_edition_2.pdf
[89] J. Wang, C. Roberts, G. Vidal, and S. Leichenauer, "Anomaly detection with tensor networks," in *Proc. 34th Conf. Neural Inf. Process. Syst. (NeurIPS)*, in First Workshop on Quantum Tensor Networks in Machine Learning, Jan. 2020, pp. 1–9. [Online]. Available: https://tensorworkshop.github.io/NeurIPS2020/accepted_papers/combined.pdf
[90] H. Chen and T. Barthel, "Machine learning with tree tensor networks, CP rank constraints, and tensor dropout," 2023, *arXiv:2305.19440*.
[91] D. Liu, S.-J. Ran, P. Wittek, C. Peng, R. B. García, G. Su, and M. Lewenstein, "Machine learning by unitary tensor network of hierarchical tree structure," *New J. Phys.*, vol. 21, no. 7, Jul. 2019, Art. no. 073059, doi: 10.1088/1367-2630/ab31ef.
[92] H. Xiao, K. Rasul, and R. Vollgraf, "Fashion-MNIST: A novel image dataset for benchmarking machine learning algorithms," 2017, *arXiv:1708.07747*.
[93] L. Li and H. Lai, "Multi-layered projected entangled pair states for image classification," *Sustainability*, vol. 15, no. 6, p. 5120, Mar. 2023, doi: 10.3390/su15065120.
[94] A. Krizhevsky, "Learning multiple layers of features from tiny images," Univ. Toronto, Toronto, ON, Canada, Tech. Rep. 2009. [Online]. Available: https://www.cs.toronto.edu/~kriz/learning-features-2009-TR.pdf
[95] M. E. H. Chowdhury, T. Rahman, A. Khandakar, R. Mazhar, M. A. Kadir, Z. B. Mahbub, K. R. Islam, M. S. Khan, A. Iqbal, N. A. Emadi, M. B. I. Reaz, and M. T. Islam, "Can AI help in screening viral and COVID-19 pneumonia?" *IEEE Access*, vol. 8, pp. 132665–132676, 2020, doi: 10.1109/ACCESS.2020.3010287.
[96] J. Y. Araz and M. Spannowsky, "Quantum-inspired event reconstruction with tensor networks: Matrix product states," *J. High Energy Phys.*, vol. 2021, no. 112, Aug. 2021, doi: 10.1007/jhep08(2021)112.
[97] J. Y. Araz and M. Spannowsky, "Classical versus quantum: Comparing tensor-network-based quantum circuits on large hadron collider data," *Phys. Rev. A, Gen. Phys.*, vol. 106, no. 6, Dec. 2022, doi: 10.1103/physreva.106.062423.
[98] E. M. Stoudenmire and D. J. Schwab, "Supervised learning with tensor networks," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 29, D. Lee, M. Sugiyama, U. Luxburg, I. Guyon, and R. Garnett, Eds., Red Hook, NY, USA: Curran Associates, Jan. 2016, pp. 4799–4807.

[99] S. Cheng, L. Wang, T. Xiang, and P. Zhang, "Tree tensor networks for generative modeling," *Phys. Rev. B, Condens. Matter*, vol. 99, no. 15, Apr. 2019, doi: 10.1103/physrevb.99.155131.

[100] K. Kottmann, P. Corboz, M. Lewenstein, and A. Acín, "Unsupervised mapping of phase diagrams of 2D systems from infinite projected entangled-pair states via deep anomaly detection," *SciPost Phys.*, vol. 11, no. 2, p. 25, Aug. 2021, doi: 10.21468/scipostphys.11.2.025.

[101] X. Gao, Z.-Y. Zhang, and L.-M. Duan, "A quantum machine learning algorithm based on generative models," *Sci. Adv.*, vol. 4, no. 12, p. 9004, Dec. 2018, doi: 10.1126/sciadv.aat9004.

[102] M. Schwarz, K. Temme, and F. Verstraete, "Preparing projected entangled pair states on a quantum computer," *Phys. Rev. Lett.*, vol. 108, no. 11, Mar. 2012, Art. no. 110502, doi: 10.1103/physrevlett.108.110502.

[103] D. Chicco and G. Jurman, "The Matthews correlation coefficient (MCC) should replace the ROC AUC as the standard metric for assessing binary classification," *BioData Mining*, vol. 16, no. 4, pp. 1–23, Feb. 2023, doi: 10.1186/s13040-023-00322-4.

[104] Y. Pang, T. Hao, A. Dugad, Y. Zhou, and E. Solomonik, "Efficient 2D tensor network simulation of quantum systems," in *Proc. Int. Conf. High Perform. Comput., Netw., Storage Anal.*, Atlanta, GA, USA, Nov. 2020, pp. 1–14, doi: 10.1109/SC41405.2020.00018.

[105] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Los Alamitos, CA, USA, Jun. 2015, pp. 1–9, doi: 10.1109/CVPR.2015.7298594.

[106] L. Chen, S. Li, Q. Bai, J. Yang, S. Jiang, and Y. Miao, "Review of image classification algorithms based on convolutional neural networks," *Remote Sens.*, vol. 13, no. 22, pp. 1–51, Nov. 2021, doi: 10.3390/rs13224712.

[107] A. Abbas, D. Sutter, C. Zoufal, A. Lucchi, A. Figalli, and S. Woerner, "The power of quantum neural networks," *Nature Comput. Sci.*, vol. 1, no. 6, pp. 403–409, Jun. 2021, doi: 10.1038/s43588-021-00084-1.

[108] K. Bharti, A. Cervera-Lierta, T. H. Kyaw, T. Haug, S. Alperin-Lea, A. Anand, M. Degroote, H. Heimonen, J. S. Kottmann, T. Menke, W.-K. Mok, S. Sim, L.-C. Kwek, and A. Aspuru-Guzik, "Noisy intermediate-scale quantum algorithms," *Rev. Mod. Phys.*, vol. 94, no. 1, Feb. 2022, Art. no. 015004, doi: 10.1103/revmodphys.94.015004.

[109] J. Kempe, A. Kitaev, and O. Regev, "The complexity of the local Hamiltonian problem," *SIAM J. Comput.*, vol. 35, no. 5, pp. 1070–1097, Jan. 2006, doi: 10.1137/s0097539704445226.

[110] T. Albash and D. A. Lidar, "Adiabatic quantum computation," *Rev. Mod. Phys.*, vol. 90, no. 1, Jan. 2018, Art. no. 015002, doi: 10.1103/revmodphys.90.015002.

[111] D. Sherrington and S. Kirkpatrick, "Solvable model of a spin-glass," *Phys. Rev. Lett.*, vol. 35, no. 26, pp. 1792–1796, Dec. 1975, doi: 10.1103/physrevlett.35.1792.

[112] P. F. Stadler, "Towards a theory of landscapes," in *Complex Systems and Binary Networks* (Lecture Notes in Physics), vol. 461, R. López-Peña, H. Waelbroeck, R. Capovilla, R. García-Pelayo, and F. Zertuche, Eds., Berlin, Germany: Springer, 1995, pp. 76–163, doi: 10.1007/BFb0103571.

[113] F. Glover, G. Kochenberger, and Y. Du, "Quantum bridge analytics I: A tutorial on formulating and using QUBO models," *4OR*, vol. 17, no. 4, pp. 335–371, Dec. 2019, doi: 10.1007/s10288-019-00424-y.

[114] A. Lucas, "Ising formulations of many NP problems," *Frontiers Phys.*, vol. 2, pp. 1–15, Jun. 2014, doi: 10.3389/fphy.2014.00005.

[115] S. Mugel, C. Kuchkovsky, E. Sánchez, S. Fernández-Lorenzo, J. Luis-Hita, E. Lizaso, and R. Orús, "Dynamic portfolio optimization with real datasets using quantum processors and quantum-inspired tensor networks," *Phys. Rev. Res.*, vol. 4, no. 1, Jan. 2022, doi: 10.1103/physrevresearch.4.013006.

[116] M. R. Perelshtein, A. I. Pakhomchik, A. A. Melnikov, M. Podobrii, A. Termanova, I. Kreidich, B. Nuriev, S. Iudin, C. W. Mansell, and V. M. Vinokur, "NISQ-compatible approximate quantum algorithm for unconstrained and constrained discrete optimization," *Quantum*, vol. 7, p. 1186, Nov. 2023, doi: 10.22331/q-2023-11-21-1186.

[117] Q. Miao and T. Barthel, "Isometric tensor network optimization for extensive Hamiltonians is free of barren plateaus," *Phys. Rev. A, Gen. Phys.*, vol. 109, no. 5, May 2024, doi: 10.1103/physreva.109.l050402.

[118] S. Cavinato, T. Felser, M. Fusella, M. Paiusco, and S. Montangero, "Optimizing radiotherapy plans for cancer treatment with tensor networks," *Phys. Med. Biol.*, vol. 66, no. 12, Jun. 2021, Art. no. 125015, doi: 10.1088/1361-6560/ac01f2.

[119] J. Tilly, H. Chen, S. Cao, D. Picozzi, K. Setia, Y. Li, E. Grant, L. Wossnig, I. Rungger, G. H. Booth, and J. Tennyson, "The variational quantum eigensolver: A review of methods and best practices," *Phys. Rep.*, vol. 986, pp. 1–128, Nov. 2022, doi: 10.1016/j.physrep.2022.08.003.

[120] J. R. McClean, S. Boixo, V. N. Smelyanskiy, R. Babbush, and H. Neven, "Barren plateaus in quantum neural network training landscapes," *Nature Commun.*, vol. 9, no. 1, Nov. 2018, doi: 10.1038/s41467-018-07090-4.

[121] G. Scriva, N. Astrakhantsev, S. Pilati, and G. Mazzola, "Challenges of variational quantum optimization with measurement shot noise," *Phys. Rev. A, Gen. Phys.*, vol. 109, no. 3, Mar. 2024, doi: 10.1103/physreva.109.032408.

[122] M. Cerezo, A. Sone, T. Volkoff, L. Cincio, and P. J. Coles, "Cost function dependent barren plateaus in shallow parametrized quantum circuits," *Nature Commun.*, vol. 12, no. 1, p. 1791, Mar. 2021, doi: 10.1038/s41467-021-21728-w.

[123] E. Cervero Martín, K. Plekhanov, and M. Lubasch, "Barren plateaus in quantum tensor network optimization," *Quantum*, vol. 7, p. 974, Apr. 2023, doi: 10.22331/q-2023-04-13-974.

[124] L. Leone, S. F. E. Oliviero, L. Cincio, and M. Cerezo, "On the practical usefulness of the hardware efficient ansatz," *Quantum*, vol. 8, p. 1395, Jul. 2024, doi: 10.22331/q-2024-07-03-1395.

[125] H. Markowitz, "Portfolio selection," *J. Finance*, vol. 7, no. 1, pp. 77–99, Mar. 1952, doi: 10.2307/2975974.

[126] D. Bienstock, "Computational study of a family of mixed-integer quadratic programming problems," *Math. Program.*, vol. 74, no. 2, pp. 121–140, Aug. 1996, doi: 10.1007/bf02592208.

[127] Y. Jin, R. Qu, and J. Atkin, "Constrained portfolio optimisation: The state-of-the-art Markowitz models," in *Proc. 5th Int. Conf. Oper. Res. Enterprise Syst. (ICORES)*. Setúbal, Portugal: SciTePress, 2016, pp. 388–395, doi: 10.5220/0005758303880395.

[128] W. Choi, I. Lahiri, R. Seelaboyina, and Y. S. Kang, "Synthesis of graphene and its applications: A review," *Crit. Rev. Solid State Mater. Sci.*, vol. 35, no. 1, pp. 52–71, Feb. 2010, doi: 10.1080/10408430903505036.

[129] A. B. Seabra, A. J. Paula, R. de Lima, O. L. Alves, and N. Durán, "Nanotoxicity of graphene and graphene oxide," *Chem. Res. Toxicol.*, vol. 27, no. 2, pp. 159–168, Feb. 2014, doi: 10.1021/tx400385x.

[130] M. Gibertini, A. Singha, V. Pellegrini, M. Polini, G. Vignale, A. Pinczuk, L. N. Pfeiffer, and K. W. West, "Engineering artificial graphene in a two-dimensional electron gas," *Phys. Rev. B, Condens. Matter*, vol. 79, no. 24, Jun. 2009, doi: 10.1103/physrevb.79.241406.

[131] A. Pérez-Obiol, A. Pérez-Salinas, S. Sánchez-Ramírez, B. G. M. Araújo, and A. Garcia-Saez, "Adiabatic quantum algorithm for artificial graphene," *Phys. Rev. A, Gen. Phys.*, vol. 106, no. 5, Nov. 2022, Art. no. 052408, doi: 10.1103/physreva.106.052408.

[132] J. Hubbard, "Electron correlations in narrow energy bands," *Proc. Roy. Soc. London A*, vol. 276, no. 1365, pp. 238–257, 1963, doi: 10.1098/rspa.1963.0204.

[133] C. Cade, L. Mineh, A. Montanaro, and S. Stanisic, "Strategies for solving the Fermi–Hubbard model on near-term quantum computers," *Phys. Rev. B, Condens. Matter*, vol. 102, no. 23, Dec. 2020, Art. no. 235122, doi: 10.1103/physrevb.102.235122.

[134] H. Shang, Y. Fan, L. Shen, C. Guo, J. Liu, X. Duan, F. Li, and Z. Li, "Towards practical and massively parallel quantum computing emulation for quantum chemistry," *NPJ Quantum Inf.*, vol. 9, no. 1, p. 33, Apr. 2023, doi: 10.1038/s41534-023-00696-7.

[135] 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.*, vol. 11, no. 3, pp. 1027–1036, Mar. 2015, doi: 10.1021/ct501187j.

[136] 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.*, vol. 138, no. 13, pp. 1–21, Apr. 2013, doi: 10.1063/1.4798639.

[137] P. Seitz, I. Medina, E. Cruz, Q. Huang, and C. B. Mendl, "Simulating quantum circuits using tree tensor networks," *Quantum*, vol. 7, p. 964, Mar. 2023, doi: 10.22331/q-2023-03-30-964.

[138] X.-G. Wen, "*Colloquium*: Zoo of quantum-topological phases of matter," *Rev. Mod. Phys.*, vol. 89, no. 4, Dec. 2017, Art. no. 041004, doi: 10.1103/revmodphys.89.041004.

[139] B. Field and T. Simula, "Introduction to topological quantum computation with non-abelian anyons," *Quantum Sci. Technol.*, vol. 3, no. 4, Oct. 2018, Art. no. 045004, doi: 10.1088/2058-9565/aacad2.

[140] P. Corboz and F. Mila, "Crystals of bound states in the magnetization plateaus of the shastry-sutherland model," *Phys. Rev. Lett.*, vol. 112, no. 14, Apr. 2014, Art. no. 147203, doi: 10.1103/physrevlett.112.147203.
[141] W. Kadow, F. Pollmann, and M. Knap, "Isometric tensor network representations of two-dimensional thermal states," *Phys. Rev. B, Condens. Matter*, vol. 107, no. 20, May 2023, Art. no. 205106, doi: 10.1103/physrevb.107.205106.
[142] B. Sriram Shastry and B. Sutherland, "Exact ground state of a quantum mechanical antiferromagnet," *Phys. B+C*, vol. 108, nos. 1–3, pp. 1069–1070, Aug. 1981, doi: 10.1016/0378-4363(81)90838-x.
[143] H. Kageyama, K. Yoshimura, R. Stern, N. V. Mushnikov, K. Onizuka, M. Kato, K. Kosuge, C. P. Slichter, T. Goto, and Y. Ueda, "Exact dimer ground state and quantized magnetization plateaus in the two-dimensional spin system $SrCu_2(BO_3)_2$," *Phys. Rev. Lett.*, vol. 82, no. 15, pp. 3168–3171, 1999, doi: 10.1103/physrevlett.82.3168.
[144] B. S. Shastry and B. Kumar, "$SrCu_2(BO_3)_2$: A unique Mott Hubbard insulator," in *Progress of Theoretical Physics Supplement*, vol. 145. London, U.K.: Oxford Univ. Press, 2002, pp. 1–16, doi: 10.1143/PTPS.145.1.
[145] Z. Shi, S. Dissanayake, P. Corboz, W. Steinhardt, D. Graf, D. M. Silevitch, H. A. Dabkowska, T. F. Rosenbaum, F. Mila, and S. Haravifard, "Discovery of quantum phases in the Shastry-Sutherland compound $SrCu_2(BO_3)_2$ under extreme conditions of field and pressure," *Nature Commun.*, vol. 13, no. 1, p. 2301, Apr. 2022, doi: 10.1038/s41467-022-30036-w.
[146] P. Moin and K. Mahesh, "Direct numerical simulation: A tool in turbulence research," *Annu. Rev. Fluid Mech.*, vol. 30, no. 1, pp. 539–578, Jan. 1998, doi: 10.1146/annurev.fluid.30.1.539.
[147] N. Gourianov, M. Lubasch, S. Dolgov, Q. Y. van den Berg, H. Babaee, P. Givi, M. Kiffner, and D. Jaksch, "A quantum-inspired approach to exploit turbulence structures," *Nature Comput. Sci.*, vol. 2, no. 1, pp. 30–37, Jan. 2022, doi: 10.1038/s43588-021-00181-1.
[148] M. Kiffner and D. Jaksch, "Tensor network reduced order models for wall-bounded flows," *Phys. Rev. Fluids*, vol. 8, no. 12, Dec. 2023, Art. no. 124101, doi: 10.1103/physrevfluids.8.124101.
[149] P. W. Shor, "Fault-tolerant quantum computation," in *Proc. 37th Conf. Found. Comput. Sci.*, Burlington, VT, USA, Jul. 1996, pp. 56–65, doi: 10.1109/SFCS.1996.548464.
[150] F. Arute et al., "Quantum supremacy using a programmable superconducting processor," *Nature*, vol. 574, no. 7779, pp. 505–510, Oct. 2019, doi: 10.1038/s41586-019-1666-5.
[151] F. Pan, K. Chen, and P. Zhang, "Solving the sampling problem of the Sycamore quantum circuits," *Phys. Rev. Lett.*, vol. 129, no. 9, Aug. 2022, Art. no. 090502, doi: 10.1103/physrevlett.129.090502.
[152] C. Guo, Y. Liu, M. Xiong, S. Xue, X. Fu, A. Huang, X. Qiang, P. Xu, J. Liu, S. Zheng, H.-L. Huang, M. Deng, D. Poletti, W.-S. Bao, and J. Wu, "General-purpose quantum circuit simulator with projected entangled-pair states and the quantum supremacy frontier," *Phys. Rev. Lett.*, vol. 123, no. 19, Nov. 2019, Art. no. 190501, doi: 10.1103/physrevlett.123.190501.
[153] Y. Liu, X. Liu, F. Li, H. Fu, Y. Yang, J. Song, P. Zhao, Z. Wang, D. Peng, H. Chen, C. Guo, H. Huang, W. Wu, and D. Chen, "Closing the 'quantum supremacy' gap: Achieving real-time simulation of a random quantum circuit using a new sunway supercomputer," in *Proc. Int. Conf. High Perform. Comput., Netw., Storage Anal.*, Los Alamitos, CA, USA, Nov. 2021, pp. 1–12, doi: 10.1145/3458817.3487399.
[154] D. Aharonov, X. Gao, Z. Landau, Y. Liu, and U. Vazirani, "A polynomial-time classical algorithm for noisy random circuit sampling," in *Proc. 55th Annu. ACM Symp. Theory Comput.*, Orlando, FL, USA, Jun. 2023, pp. 945–957, doi: 10.1145/3564246.3585234.
[155] Y. Kim, A. Eddins, S. Anand, K. X. Wei, E. van den Berg, S. Rosenblatt, H. Nayfeh, Y. Wu, M. Zaletel, K. Temme, and A. Kandala, "Evidence for the utility of quantum computing before fault tolerance," *Nature*, vol. 618, no. 7965, pp. 500–505, Jun. 2023, doi: 10.1038/s41586-023-06096-3.
[156] S. Patra, S. S. Jahromi, S. Singh, and R. Orús, "Efficient tensor network simulation of IBM's largest quantum processors," *Phys. Rev. Res.*, vol. 6, no. 1, Mar. 2024, Art. no. 013326, doi: 10.1103/physrevresearch.6.013326.
[157] T. Begušić, J. Gray, and G. K.-L. Chan, "Fast and converged classical simulations of evidence for the utility of quantum computing before fault tolerance," *Sci. Adv.*, vol. 10, no. 3, p. 4321, Jan. 2024, doi: 10.1126/sciadv.adk4321.
[158] H.-J. Liao, K. Wang, Z.-S. Zhou, P. Zhang, and T. Xiang, "Simulation of IBM's kicked Ising experiment with projected entangled pair operator," 2023, *arXiv:2308.03082*.
[159] J. Tindall, M. Fishman, E. M. Stoudenmire, and D. Sels, "Efficient tensor network simulation of IBM's eagle kicked ising experiment," *PRX Quantum*, vol. 5, no. 1, Jan. 2024, Art. no. 010308, doi: 10.1103/prxquantum.5.010308.
[160] A. D. King et al., "Computational supremacy in quantum simulation," 2024, *arXiv:2403.00910*.
[161] R. Toshio, Y. Akahoshi, J. Fujisaki, H. Oshima, S. Sato, and K. Fujii, "Practical quantum advantage on partially fault-tolerant quantum computer," 2024, *arXiv:2408.14848*.
[162] R. Tao, Y. Shi, J. Yao, J. Hui, F. T. Chong, and R. Gu, "Gleipnir: Toward practical error analysis for quantum programs," in *Proc. 42nd ACM SIGPLAN Int. Conf. Program. Lang. Design Implement.*, Jun. 2021, pp. 48–64, doi: 10.1145/3453483.3454029.
[163] A. S. Darmawan and D. Poulin, "Tensor-network simulations of the surface code under realistic noise," *Phys. Rev. Lett.*, vol. 119, no. 4, Jul. 2017, Art. no. 040502, doi: 10.1103/physrevlett.119.040502.
[164] A. J. Ferris and D. Poulin, "Tensor networks and quantum error correction," *Phys. Rev. Lett.*, vol. 113, no. 3, Jul. 2014, Art. no. 030501, doi: 10.1103/physrevlett.113.030501.
[165] M. Fingerhuth, T. Babej, and P. Wittek, "Open source software in quantum computing," *PLoS ONE*, vol. 13, no. 12, Dec. 2018, Art. no. e0208561, doi: 10.1371/journal.pone.0208561.
[166] E. F. Dumitrescu, A. L. Fisher, T. D. Goodrich, T. S. Humble, B. D. Sullivan, and A. L. Wright, "Benchmarking treewidth as a practical component of tensor network simulations," *PLoS ONE*, vol. 13, no. 12, Dec. 2018, Art. no. e0207827, doi: 10.1371/journal.pone.0207827.
[167] J. R. Finžgar, P. Ross, L. Hölscher, J. Klepsch, and A. Luckow, "QUARK: A framework for quantum computing application benchmarking," in *Proc. IEEE Int. Conf. Quantum Comput. Eng. (QCE)*, Broomfield, CO, USA, Sep. 2022, pp. 226–237, doi: 10.1109/QCE53715.2022.00042.
[168] A. Jamadagni, A. M. Läuchli, and C. Hempel, "Benchmarking quantum computer simulation software packages: State vector simulators," 2024, *arXiv:2401.09076*.
[169] T. Lubinski, S. Johri, P. Varosy, J. Coleman, L. Zhao, J. Necaise, C. H. Baldwin, K. Mayer, and T. Proctor, "Application-oriented performance benchmarks for quantum computing," *IEEE Trans. Quantum Eng.*, vol. 4, pp. 1–32, 2023, doi: 10.1109/TQE.2023.3253761.
[170] Y. Ke, "Tree tensor network state approach for solving hierarchical equations of motion," *J. Chem. Phys.*, vol. 158, no. 21, Jun. 2023, Art. no. 211102, doi: 10.1063/5.0153870.

MARCOS DÍEZ GARCÍA received the B.S. and M.S. degrees (Hons.) in computer engineering from the University of Seville, Seville, Spain, in 2016, and the Ph.D. degree in computer science from the University of Exeter, Exeter, U.K., in 2021.

He was a One-Semester Intern from the Research Group on Natural Computing, University of Seville. From 2018 to 2019, he was a Postgraduate Teaching Assistant for workshops on artificial intelligence and applications with the University of Exeter. From 2021 to 2022, he completed a Postgraduate Research Associate Program with the University of Exeter, where he developed constraint-handling optimization methods for the digital annealer in collaboration with Fujitsu Research of Europe Ltd., U.K., and presented at top international venues like the Genetic and Evolutionary Computation Conference. He officially joined Fujitsu Research of Europe Ltd., in 2022, where he is currently a Senior Researcher. His research interests include tensor networks, quantum circuit simulation, quadratic unconstrained binary optimization, and quantum-inspired and nature-inspired optimization methods.

ANTONIO MÁRQUEZ ROMERO received the B.Sc. degree in physics from the University of Seville, Seville, Spain, in 2014, the M.Sc. degree in physics and mathematics (FisyMat) from the University of Granada, Granada, Spain, in 2015, and the Ph.D. degree in physics from the University of York, York, U.K., in 2020.

From 2020 to 2022, he was a Postdoctoral Researcher with the University of North Carolina at Chapel Hill, Chapel Hill, NC, USA. From 2022 to 2023, he was a Postdoctoral Researcher wth the University of Barcelona, Barcelona, Spain. He officially joined the Quantum Application Research Group, Fujitsu Research of Europe Ltd., as a Senior Researcher, in 2023.