

## Limitations of tensor-network approaches for optimization and sampling: A comparison to quantum and classical Ising machines

Anna Maria Dziubyna<sup>1,2</sup> ,<sup>1,2</sup> Tomasz Śmierzchalski<sup>3</sup> ,<sup>3</sup> Bartłomiej Gardas<sup>3,\*</sup> ,<sup>3,\*</sup> Marek M. Rams<sup>1,†</sup> ,<sup>1,†</sup>  
and Masoud Mohseni<sup>4,‡</sup> 

<sup>1</sup>*Institute of Theoretical Physics, Jagiellonian University, Łojasiewicza 11, Kraków 30-348, Poland*

<sup>2</sup>*Doctoral School of Exact and Natural Sciences, Jagiellonian University, Łojasiewicza 11, Kraków 30-348, Poland*

<sup>3</sup>*Institute of Theoretical and Applied Informatics, Polish Academy of Sciences, Bałtycka 5, Gliwice 44-100, Poland*

<sup>4</sup>*Emergent Machine Intelligence, Hewlett Packard Labs, California, USA*



(Received 4 December 2024; accepted 25 April 2025; published 19 May 2025)

Optimization problems pose challenges across various fields. In recent years, quantum annealers have emerged as a promising platform for tackling such challenges. Here, to provide an additional perspective, we develop a heuristic tensor-network-based algorithm to reveal the low-energy spectra of Ising spin-glass systems with interaction graphs relevant to present-day quantum annealers. Our deterministic approach combines a branch-and-bound search strategy with an approximate calculation of marginals via tensor-network contractions. Its application to quasi-two-dimensional lattices with large unit cells of up to 24 spins, realized in current quantum annealing processors, requires a dedicated approach that uses sparse structures in the tensor-network representation and GPU hardware acceleration. We benchmark our approach on random problems defined on Pegasus and Zephyr graphs with up to a few thousand spins, comparing it against the D-Wave Advantage quantum annealer and the simulated bifurcation algorithm, with the latter representing an emerging class of classical Ising solvers. In addition to examining the quality of the best solutions, we compare the diversity of low-energy states sampled by all the solvers. For the largest considered independent and identically distributed problems with over 5000 spins, the state-of-the-art tensor-network approaches lead to solutions that are 0.1% to 1% worse than the best solutions obtained by Ising machines while being 2 orders of magnitude slower. We attribute these results to approximate contraction failures. For embedded tile planting instances, our approach reaches approximately 0.1% from the planted ground state, a factor of 3 better than the Ising solvers. While all three methods can output diverse low-energy solutions—e.g., differing by at least a quarter of spins with energy error below 1%—our deterministic branch-and-bound approach finds sets of a few such states at most. In contrast, both Ising machines prove capable of sampling sets of thousands of such solutions.

DOI: [10.1103/PhysRevApplied.23.054049](https://doi.org/10.1103/PhysRevApplied.23.054049)

### I. INTRODUCTION

Ising spin-glass ground-state identification is a prime example of an NP-hard problem [1] and a representative example of a discrete optimization problem [2], where even a two-dimensional Ising system with fields provides a universal computational model [3].

Many physics-inspired approaches have been developed to tackle this problem. A prominent and general family of methods, starting with the simulated annealing algorithm [4], employs thermal fluctuations introduced by Metropolis-Hastings updates [5,6] within a Markov-chain Monte Carlo (MCMC) paradigm. Rugged energy landscapes pose significant challenges for local Monte

Carlo updates, motivating elaborate schemes to navigate distinct local minima. Those include parallel tempering (PT) [7] to enhance thermal fluctuations; heuristic cluster moves, such as the isoenergetic or Houdayer cluster move [8,9]; and nonequilibrium Monte Carlo with inhomogeneous temperature distribution over identified surrogate problem backbones [10]. Alternative approaches to overcoming the limitations of local updates include subgraph-based methods, which leverage fixed-treewidth substructures to improve sampling efficiency [11]. Additionally, highly optimized implementations of simulated annealing for Ising spin glasses have been demonstrated to achieve significant performance gains through multispin coding, particularly targeting the previous graph geometry of D-Wave quantum annealers, i.e., Chimera graphs [12].

An alternative approach under development employs quantum fluctuations within the quantum annealing paradigm [13,14]. In particular, quantum annealers (QAs)

\*Contact author: [bartek.gardas@gmail.com](mailto:bartek.gardas@gmail.com)

†Contact author: [marek.rams@uj.edu.pl](mailto:marek.rams@uj.edu.pl)

‡Contact author: [masoud.mohseni@hpe.com](mailto:masoud.mohseni@hpe.com)

developed by D-Wave are a prime example of a quantum Ising machine, being a dedicated solver for Ising spin glasses on quasi-two-dimensional (quasi-2D) geometries. A scaling advantage of quantum annealing over standard Monte Carlo methods remains a possibility, based on analysis of some finite random problems [15–17]. In an idealistic QA, the time evolution is deterministic, with randomness entering while sampling from the final evolved state. Generally, there are no guarantees that this provides fair sampling from the low-energy manifold. For sufficiently fast quenches, the D-Wave QA was shown to give results consistent with the coherent Schrödinger dynamics [18,19]; however, for the slower quenches used for optimization in this article, decoherence affects the outputs of the quantum simulator [20], adding to the randomness of the results.

Classical Ising machines include a recently developed simulated bifurcation machine (SBM) [21–23]. This maps the discrete Ising problem to a continuous system undergoing dissipative and chaotic dynamics governed by ordinary Hamilton equations. Stationary states of this process, after the final discretization of continuous variables, correspond to local minima of the original problem. The system evolves from a set of random initial states leading to multiple trajectories, simulations of which can be parallelized efficiently.

Tensor networks (TNs) provide an alternative, typically deterministic, approach to studying the properties of many-body systems [24–28]. In particular, they give a compact representation of the partition function of a classical low-dimensional spin system at arbitrary temperature [29], with information about marginal configuration probabilities following from (approximate) tensor-network contractions. The underlying NP-hardness of the problem manifests itself in the difficulty of performing the contraction [30]. Nevertheless, we can turn a TN into an optimization solver. In particular, Ref. [31] combined approximate tensor network contraction with a branch-and-bound approach, testing this in the context of the previous quantum annealing processor geometry called the Chimera graph. The latter consists of unit cells of eight spins forming a square lattice, a natural geometry for established approximate tensor network contraction techniques. Therein, the TN-based approach showed an advantage over PT and QA in some considered Chimera-geometry-native examples [31].

At the same time, TNs allow one to perform sampling [32], which can be further enhanced in classical systems via combination with MCMC methods, where TN is used to provide nonlocal update candidates [33]. Very recently, promising results have been reported [34] for the application of hyperoptimized approximate contraction of tensor networks [35] to optimization problems with rugged energy landscapes in two and three dimensions (using a simplified approach where optimization problem is addressed via direct sampling).

A promising alternative to working at finite temperature is provided by tropical TN [36], considering the logarithm of the partition function in the zero-temperature limit. This is equivalent to a message-passing algorithm for ground-state energy identification [37]; however, automatic differentiation techniques applied in such TN contraction can also produce the corresponding ground-state configurations. Nevertheless, this approach is currently limited to the exact contraction of the network and optimization problems with small treewidth.

This article explores the application of TNs for graph geometries currently employed in D-Wave quantum anneals, which exhibit significantly enhanced connectivity over the previous generation processor and Chimera graph. For instance, such a graph can fit a cubic  $15 \times 15 \times 12$  lattice of dimers [15]. This poses significant challenges for employing approximate TN contraction techniques, which we tackle in this work. We aim to explore the limitations and gain insights into the sources of failure of our deterministic branch-and-bound TN approach, where increased problem complexity puts stringent constraints on the quality of the results produced by our solver. We compare this with representative examples of randomized algorithms, a class more commonly employed in the context of optimization.

The remainder of this article is organized as follows. In Sec. II, we discuss the Ising optimization problems, in particular, introducing the Pegasus and Zephyr graph geometries in which we are primarily interested here. Section III provides a comprehensive description of our algorithm, including branch-and-bound search and the construction and contraction of a projected entangled pair state (PEPS) tensor network for graphs with large unit cells. We comment on exploiting sparse tensor structures and the variational zipper algorithm used in boundary matrix product state (MPS) optimization. In Sec. IV, we discuss the optional concept of local dimensional reduction, employing loopy belief propagation in this context. In Sec. V, we collect the results, focusing on the ground-state search and diversity of low-energy solutions. We benchmark our algorithm against the D-Wave QA and the SBM. We end with a discussion on how and why the tensor network approach is breaking in Sec. VI, followed by concluding remarks in Sec. VII. In the Appendixes, we provide additional details on the stability of our approach, as well as other 2D graph geometries. Therein, we also discuss further details of the SBM algorithm.

## II. ISING PROBLEM

We focus on the problem of finding the low-energy states of a classical Ising Hamiltonian [2],

$$H(\underline{s}_N) = \sum_{\langle i,j \rangle \in \mathcal{E}} J_{ij} s_i s_j + \sum_{i=1}^N h_i s_i, \quad (1)$$

where  $\underline{s}_N$  denotes a particular configuration of  $N$  binary variables  $s_i = \pm 1$ . The problem instance is defined by the set of real couplings  $J_{ij}$  forming a graph  $\mathcal{E}$  and local fields  $h_i$ . We will be using a notation in which we denote a subconfiguration of the first  $n$  variables as

$$\underline{s}_n = (s_1, s_2, \dots, s_n). \quad (2)$$

Here, we are interested in quasi-2D graphs, particularly those of relevance for current quantum annealing processors. D-Wave Systems has introduced two notable topologies in the realm of quantum annealing processors, named Pegasus and Zephyr [see Figs. 1(a) and 1(c), respectively]. Pegasus and Zephyr represent a significant advancement over previous Chimera topology, featuring a quasi-2D lattice structure with improved connectivity and aiming to reduce noise [38–40]. In particular, the available Pegasus quantum processor, which we used as a reference, has 5616 qubits connected by over 40 000 couplers, where a typical qubit is connected to 15 different qubits. The Zephyr topology prototype we employed in this work has 563 spins, with each typical qubit connected to 20 different qubits, and plans to expand the size of the chip to approximately 7000 qubits. We provide further details on quantum annealing experiments in the Appendixes.

### III. TENSOR-NETWORK APPROACH

To address graph geometries with large unit cells, such as Pegasus and Zephyr, using the TN-based approach, we start by representing the Ising problem in Eq. (1) as a generalized Potts Hamiltonian with a reduced number of variables of larger dimension. To this end, we group binary variables that form natural unit cells into clusters, as depicted in Fig. 1 with red shapes. We group sets of 24 binary variables for Pegasus geometry and 16 variables for Zephyr geometry, resulting in the generalized Potts

Hamiltonian

$$H(\underline{x}_{\tilde{N}}) = \sum_{(m,n) \in \mathcal{F}} E_{x_m x_n} + \sum_{n=1}^{\tilde{N}} E_{x_n}. \quad (3)$$

Here,  $\mathcal{F}$  is a 2D square lattice with  $\tilde{N}$  nodes; see Figs. 1(b) and 1(d), where we indicate nearest-neighbor interactions with blue lines and diagonal connections with green lines. Each variable  $x_n$  takes up to  $d$  values, with  $d = 2^{24}$  for Pegasus and  $2^{16}$  for Zephyr geometry (in the maximal case when all qubits in the cluster are operational and are employed in a particular Ising Hamiltonian of interest). In Eq. (3),  $E_{x_n}$  indicates the intranode energy of the corresponding binary-variable configuration, and  $E_{x_m x_n}$  is the internode coupling. Our Potts Hamiltonian can also be considered as a factor graph [37], with interaction nodes limited to one-body and two-body terms, and with  $x_n$  constituting variable nodes.

We note that not all the spins in each red group in Figs. 1(a) and 1(c) interact with spins in neighboring groups. This allows us to further compress the internode energy matrices,

$$E_{x_m x_n} = \bar{E}_{\bar{x}_m \bar{x}_n}, \quad (4)$$

where we introduce the projected variable

$$\bar{x}_m = P^{mn}(x_m), \quad (5)$$

with  $P^{mn}$  projecting  $d$  configurations in the  $m$ th node onto  $d^{mn}$  unique subconfigurations for a subset of original binary variables that are actually coupled with spins in the  $n$ th node. Similarly, in the opposite direction,  $\bar{x}_n = P^{nm}(x_n)$ . This is depicted in Fig. 2, together with the resulting dimensions.

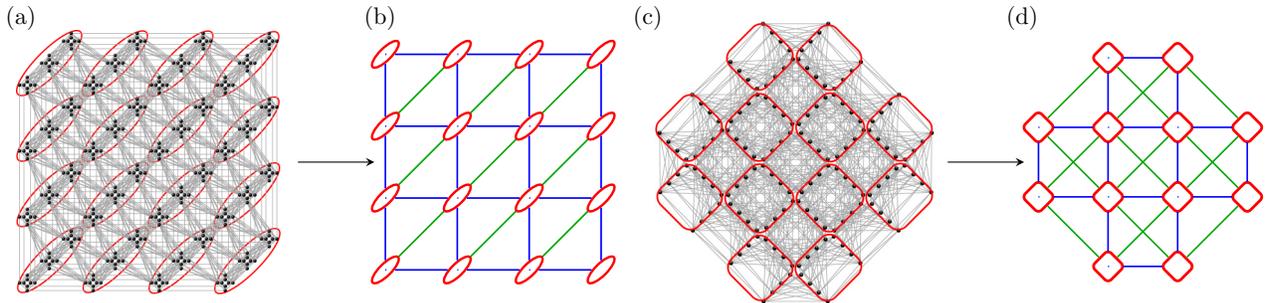


FIG. 1. Graph geometries appearing in the current D-Wave processors. We show the interaction structures of the so-called Pegasus graph in panel (a), here with 384 spins, and the Zephyr graph with 160 spins in panel (c). Black dots denote spins, and couplings between spins are represented by gray lines. To apply the tensor-network approach considered in this article, we employ quasi-2D structures for these graphs, with unit cells consisting of 24 spins for Pegasus and 16 spins for Zephyr, denoted by red lines in panels (a),(c), respectively. We group the spins in each unit cell to arrive at a generalized Potts Hamiltonian in Eq. (3), defined on a square lattice with the nearest neighbor (blue lines) and diagonal couplings (green lines) shown in panels (b),(d).

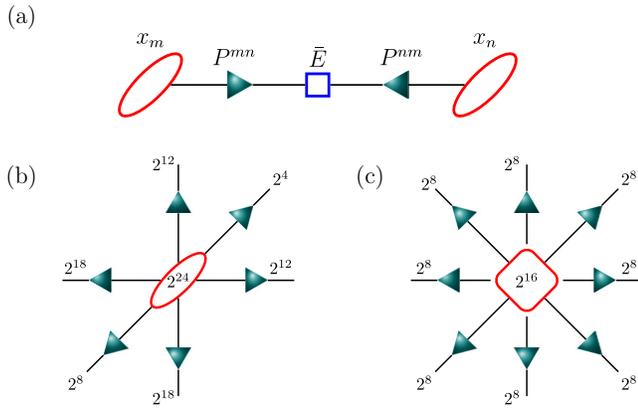


FIG. 2. Compressed interaction matrices in clustered Hamiltonian. The numbers of degrees of freedom in each cluster are  $2^{24}$  for Pegasus geometry and  $2^{16}$  for Zephyr geometry, which translates to linear dimensions of  $E_{x_m, x_n}$  matrices in Eq. (3); however, these can be significantly reduced by employing the fact that there are spins in a given cluster that are not coupled to any spin in a neighboring cluster. Projectors  $P^{mn}$  and  $P^{nm}$  project local degrees of freedom in each cluster to the relevant set of subconfigurations, limited to groups of interacting spins. Following Eqs. (4) and (5), this gives compressed interaction matrices  $\bar{E}_{\underline{x}_m, \underline{x}_n}$ , with linear dimensions for Pegasus and Zephyr geometries shown in panels (b),(c), respectively.

### A. Tensor networks for optimization problems on quasi-2D graphs

We approach the problem of identifying low-energy states—from  $2^N$  possible spin configurations—by translating it into the task of finding the most probable configurations according to a Boltzmann distribution at an inverse temperature  $\beta$ ,

$$p(\underline{x}_{\bar{N}}) = \frac{1}{Z} \exp(-\beta H(\underline{x}_{\bar{N}})), \quad (6)$$

where  $Z$  is a partition function. To effectively distinguish these low-energy states, we employ a branch-and-bound strategy, as illustrated in Fig. 3.

The strategy involves sweeping through the system to systematically build a set of high-probability configurations, adding one Potts variable at a time. At each step, we limit the number of partial configurations considered to at most  $M$ , keeping the computational complexity of the exploration process under control. This involves branching a set of  $M$  configurations supported on the first  $n$  clusters,  $\underline{x}_n$ , to include possible configurations in the  $(n+1)$ th cluster, resulting in  $d \times M$  trial configurations. For the subsequent step, we select  $M$  of those with the largest marginal probabilities,

$$p(\underline{x}_{n+1}) = p(x_{n+1} | \underline{x}_n) \times p(\underline{x}_n), \quad (7)$$

which are calculated employing the chain rule. Here, we assume that variable indexing is consistent with the order

in which the nodes are explored during the search (which can always be obtained with reindexing).

The efficiency of the exploration is greatly enhanced, keeping  $M$  fixed, by exploiting the locality of the interactions—namely, that the probabilities conditioned on partial configurations  $\underline{x}_n$  depend solely on the border part of  $\underline{x}_n$  directly interacting with the rest of the lattice,

$$\partial \underline{x}_n = \{P^{kl}(x_k) : \langle k, l \rangle \in \mathcal{F}, k \leq n, l > n\}. \quad (8)$$

As such, we identify equivalent partial configurations  $\underline{x}_n$ , which look the same from the point of view of the remaining part of the lattice. Specifically, for a given configuration  $\underline{x}_n$ , the conditional probability appearing in Eq. (7) relies solely on the subconfiguration at the border  $\partial \underline{x}_n$ ,

$$p(x_{n+1} | \underline{x}_n) = p(x_{n+1} | \partial \underline{x}_n). \quad (9)$$

This is used to merge configurations  $\underline{x}_n$  with identical boundary subconfigurations  $\partial \underline{x}_n$  [see Fig. 3(d)], which results in effective compression of the low-energy manifold and improved performance of the algorithm. At the same time, the merging process enables us to identify low-energy excitation in the system as it reveals the most probable subconfiguration (with the lowest energy for a fixed boundary subconfiguration), treated as the main branch, and low-energy excitations (spin-glass droplets) above these partial ground states [see Fig. 4].

The branch-and-bound procedure is applied iteratively until the last cluster, resulting in the ground-state candidate and, from the merging process, a set of excitations that gives an insight into the geometry of a low-energy manifold.

We also keep track of the largest discarded probability during the optimization process  $p_d$ , which gives an upper bound for the probability of any discarded configuration. Thus, if the highest retained probability, corresponding to the state with the lowest identified energy, is larger than  $p_d$ , this would be sufficient evidence for ground-state certification (for exact calculation of conditional probabilities; it remains heuristic evidence for approximate tensor-network contractions). In practice, reaching this condition proves to be infeasible in our examples, as the large inverse temperature  $\beta$  required to facilitate such a situation would make approximate tensor-network contraction numerically unstable. Nonetheless, intermediate  $\beta$  values, where approximate contraction still works reasonably well, are sufficient for obtaining good-quality solutions in the considered examples. We discuss these aspects in the following sections.

### B. Construction and contraction of PEPS for graphs with large unit cells

To employ the algorithm described in the previous section, it is crucial to effectively compute the conditional probabilities  $p(x_{n+1} | \partial \underline{x}_n)$  to apply the chain rule in Eq. (7).

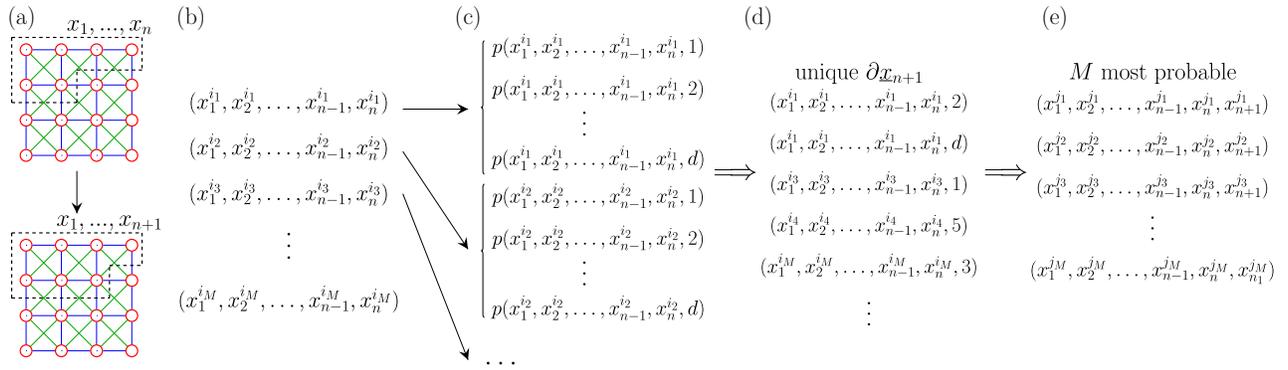


FIG. 3. Branch-and-bound algorithm. We translate the problem of identification of the low-energy configurations to the problem of finding the most probable configurations according to a Boltzmann distribution at some inverse temperature  $\beta$  [see Eq. (6)]. We sweep the 2D system, processing row after row. At each step, we identify a set of  $M$  configurations for  $n$  clusters,  $\underline{x}_n$ , with large marginal probabilities. In a single step of the sweep, depicted in panel (a), we use  $M$  configurations  $\underline{x}_n$  to identify  $M$  configurations  $\underline{x}_{n+1}$ , supported on one extra cluster. To this end, we branch  $\underline{x}_n$  configurations in panel (b) to include all possible values in the  $(n+1)$ th cluster [see panel (c)] and calculate their marginal probabilities  $p(\underline{x}_{n+1})$ . These marginal probabilities are determined from the approximate contraction of the PEPS tensor network representing the Boltzmann distribution, as illustrated in Fig. 6. Next, we bound the set of candidate configurations in two steps. First, in panel (d), we employ locality of interactions in a 2D lattice, grouping the states according to their subconfiguration,  $\partial \underline{x}_{n+1}$ , supported on a boundary with the remaining part of the lattice, i.e., with clusters  $n+2, n+3, \dots$ . Only the most probable configuration for each unique boundary is retained for further processing. Discarded configurations encode excitation in the system and can be recorded to map the low-energy spectrum of the system, as depicted in Fig. 4. Finally, in panel (e), we retain the  $M$  most probable configurations  $\underline{x}_{n+1}$ , completing a single step of the sweep.

This can be achieved by representing all probabilities using a two-dimensional PEPS tensor network [29,41]. For a classical system in Eq. (3), the construction of a thermal state is exact and equivalent to the representation of its

partition function [29,41],

$$Z = \sum_{\underline{x}_{\tilde{N}}} \exp(-\beta H(\underline{x}_{\tilde{N}})), \quad (10)$$

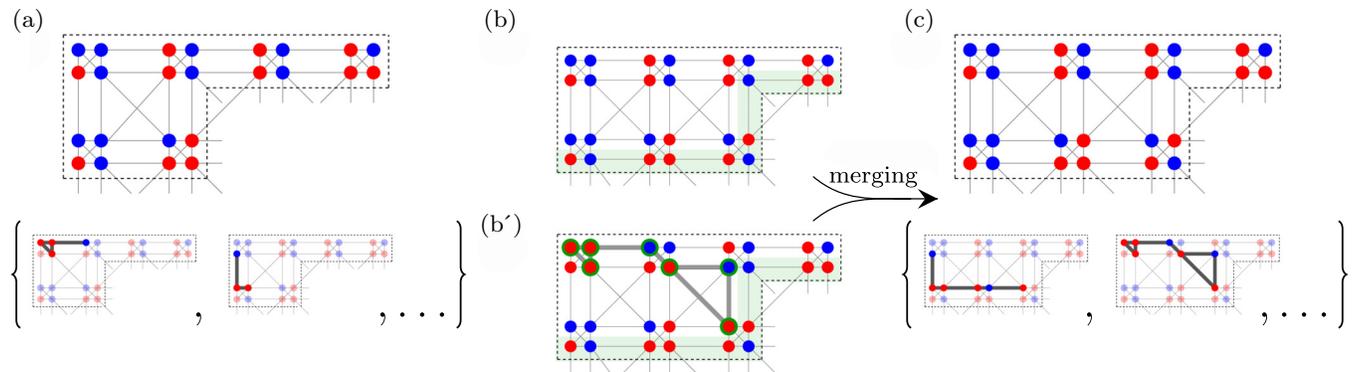


FIG. 4. Excited states in the branch-and-bound algorithm. In each step of the sweep, we bound the set of candidate partial configurations, keeping the most probable ones for each considered *unique* boundary. We depict a single such configuration in the upper part of panel (a), in which dots represent spins: a blue dot denotes a spin up and red a spin down. This is accompanied by a set of low-energy excitations, shown in the lower part of panel (a), that do not modify the boundary spins. In panels (b) and (b'), we show the merging process [step (d) in Fig. 3] of two spin configurations with the same boundary—highlighted in green. The one with a larger marginal probability becomes the main branch, depicted in the upper part of panel (c). The difference, indicated by bold gray lines in panel (b'), is added to the set of excitations above the main branch [lower part of panel (c)], together with excitations above (b) and (b') accumulated in the previous steps. To keep their number under control, we keep only those that are singly connected and with a Hamming distance between any pair of them above a specified threshold.

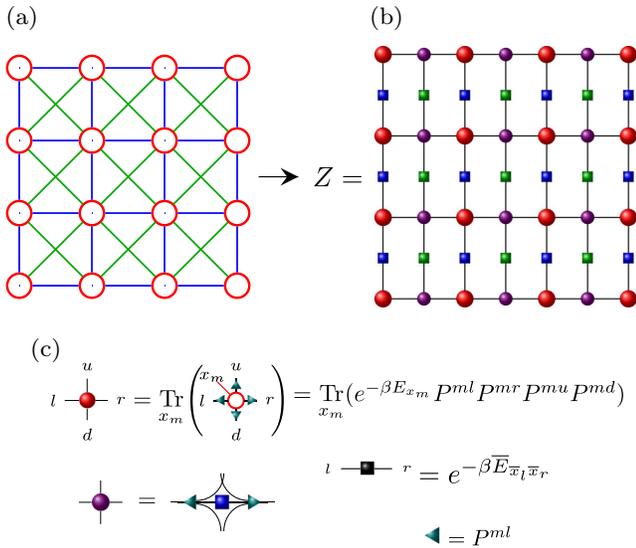


FIG. 5. Representation of the partition function as a tensor network. The partition function of a generalized Potts Hamiltonian in Eq. (3) on a square lattice with nearest-neighbor and diagonal interactions, panel (a), can be expressed as a contraction of a tensor network depicted in panel (b), with individual tensors defined in panel (c). Here, red dots denote site tensors, corresponding to clusters in panel (a), with four virtual legs mediating the interaction structure and traced-out Hamiltonian degrees of freedom. Square markers denote the exponents of the corresponding interaction matrices, purple dots show mediating tensors that carry diagonal interactions, and triangles depict the configuration projectors in Eq. (5). The result is a PEPS tensor network on a square grid [panel (b)] that is amenable to established tensor-network contraction strategies (see Fig. 6).

which is depicted in Fig. 5. In addition to *site tensors* carrying information about local energies and their interaction structure with neighboring sites (red circles in Fig. 5) and *bond tensors* encoding two-site interaction energies (squares), we introduce *mediating tensors* (purple circles). The latter encodes diagonal interactions in a tensor network defined on a square grid with nearest-neighbor connections only. This construction is compatible with standard approximate tensor-network contraction techniques.

Finding the ground state of a general Ising spin-glass problem is NP-hard [1]. In the tensor-network representation of spin-glass problems, this hardness manifests itself in the PEPS tensor network being  $\#P$ -hard to contract exactly [30] (i.e., to extract information from). The higher computational class than NP-hard reflects the fact that it contains information about the entire spectrum, not just the ground state. The exponential computational cost for exact contraction motivates the use of approximate (heuristic) contraction schemes, reducing the cost to polynomial in lattice parameters.

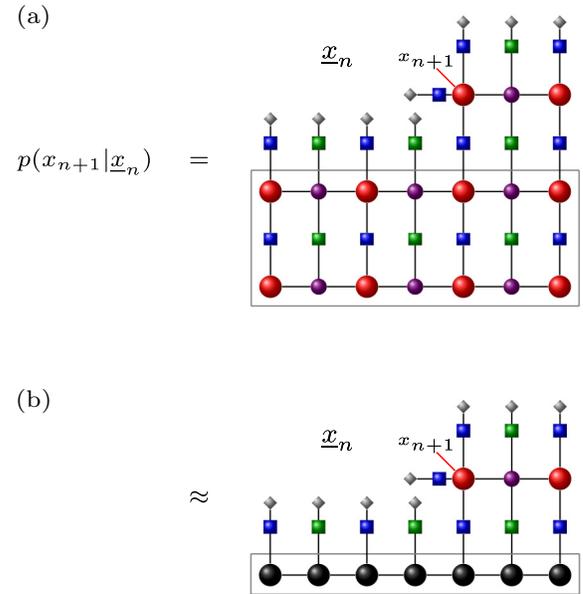


FIG. 6. Calculation of conditional probabilities via approximate contraction of a PEPS tensor network. The calculation of marginal probabilities of interest boils down to the calculation of conditional probabilities  $p(x_{n+1}|\underline{x}_n)$  which, up to normalization, are represented by a tensor network in panel (a)—closely related to the one in Fig. 5(b). Here, the gray diamonds project the virtual degrees of freedom on a boundary  $\partial \underline{x}_n$  of a given configuration  $\underline{x}_n$ . The  $(n+1)$ th site tensor has an extra leg representing (untraced) physical degrees of freedom. In panel (b), we systematically approximate the lower half of the network using the boundary MPS approach (see Fig. 8), where we mark the corresponding parts of (a),(b) with gray boxes. This preprocessing step is carried out once at the beginning of the calculation, being independent of a specific configuration  $\underline{x}_n$ . Finally, the approximate network in panel (b) is contracted numerically to exactly retrieve the desired conditional probabilities.

In this work, we adopt the boundary MPS approach [24,42], as illustrated in Fig. 6 in the context of a tensor network representation of the conditional probability in Eq. (7). Tensors in the gray box in Fig. 6(a) are approximated in Fig. 6(b) by a boundary MPS (black dots) with bond dimensions truncated to  $\chi$ . For lattices with multiple rows, boundary MPS approximations are obtained sequentially row by row. In each step, a product of the boundary MPS from the previous row and the transfer matrix formed by the next row of tensors, having the structure of matrix product operator (MPO), is approximated by a new boundary MPS of limited bond dimension  $\chi$ . We describe the procedure we use in detail in Sec. III B 2. This calculation of boundary MPSs is carried out once as a preprocessing step of the algorithm, as they are shared by all  $p(x_{n+1}|\partial \underline{x}_n)$  of interest. Finally, diagrams such as that in Fig. 6(b) are viable for numerically exact contraction. In this way, information from the entire lattice is included in the calculation of probabilities.

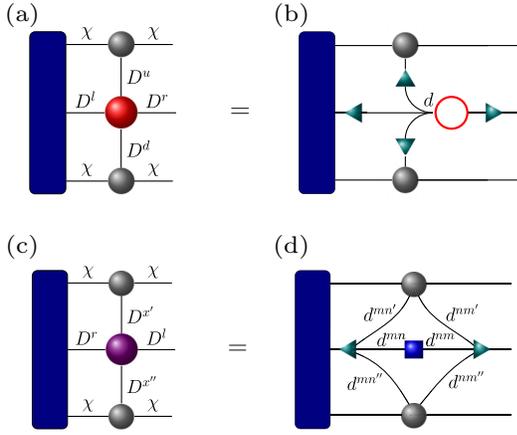


FIG. 7. Sparse structure of the PEPS tensor. The large bond dimensions of the site and mediating tensors, which appear in the representation of Pegasus and Zephyr geometries, make their direct construction infeasible. We exploit the fact that in all calculations—i.e., in the randomized singular value decomposition (SVD), variational optimization in Fig. 8, and contractions in Fig. 6—they appear as part of diagrams such as those in panels (a),(c). In these diagrams, the constituent tensor and the result of their contraction each have only one leg, corresponding to the (huge) legs of the PEPS tensors. For the bond dimensions resulting from the Pegasus and Zephyr graphs, these diagrams can still be contracted by employing the sparse structure of PEPS tensors and GPU hardware acceleration. For instance, in panel (b), the blue tensor and MPS gray tensors are projected [triangles are projectors in Fig. 2(a)] to a common physical space of a site tensor. Here, they can be efficiently contracted along the connecting legs of dimension  $\chi$ , and local Boltzmann factors (red circle) can be applied. Finally, the result is projected onto the remaining virtual leg of the PEPS tensor (outgoing triangle). Similar manipulations are performed in panel (d) for optimal contraction of diagrams involving mediating tensors.

Execution of the above techniques for the tensor sizes needed to encode the Ising Hamiltonian relevant to near-term quantum annealing technology requires a dedicated approach, which we describe below.

### 1. Employing sparsity of tensors

The sizes of tensors appearing in the TN representation of marginal probabilities for Pegasus and Zephyr geometries prevent their direct construction. Specifically, for a *site* tensor, the bond dimensions  $D^l$ ,  $D^u$ ,  $D^r$ , and  $D^d$  [see Fig. 7(a)] in a full Pegasus graph are  $2^{18}$ ,  $2^{14}$ ,  $2^{14}$ , and  $2^{18}$ , respectively, while in a full Zephyr graph they are  $2^{16}$ ,  $2^8$ ,  $2^8$ , and  $2^{16}$  (the values might be permuted depending on the direction from which the boundary MPS contraction of a 2D lattice is done). Calculation of a diagram in Fig. 7(a), which is the building block of all contraction algorithms that we use, would scale as  $\mathcal{O}(D^l D^u D^r D^d \chi^2)$  if the site tensor were constructed directly, making such an approach infeasible in practice.

A strategy to address this challenge is to leverage the sparse structure of this tensor, as shown in Fig. 5(c). We employ this in Fig. 7(b), which depicts an efficient strategy to contract the diagram. First, the legs of the tensors to be contracted with the legs of the site tensor are projected back to the physical space of the site tensor of size  $d$  (triangles). The virtual legs of those tensors are then contracted using batched matrix multiplication at a numerical cost of  $\mathcal{O}(\chi^3 d)$ . Additionally, this step greatly benefits from hardware acceleration on GPU. Nonetheless, it remains the computational bottleneck for the Pegasus graph with  $d = 2^{24}$ , which, together with available memory, limits the feasible boundary MPS bond dimensions to approximately  $\chi = 10$  [45]. Finally, the local energy term is applied, and the physical space is projected by the last right-pointing projector in Fig. 7(b), completing the contraction of the diagram.

A similar approach is needed for operations involving a mediating tensor, whose dimensions  $D^r$ ,  $D^{x'}$ ,  $D^l$ , and  $D^{x''}$  [see Fig. 7(c)] are  $2^{14}$ ,  $2^4$ ,  $2^{18}$ , and  $2^8$  for a full Pegasus graph, and are all equal to  $2^{16}$  for the Zephyr graph. The optimal contraction scheme employs the internal structure of this tensor, shown in Fig. 7(d), where the bond dimensions  $d^{mm'}$ ,  $d^{mn}$ ,  $d^{m'n'}$ ,  $d^{mm'}$ ,  $d^{nm}$ , and  $d^{m'n'}$  correspond to the dimensions in Figs. 2(b) and 2(c). For the Pegasus graph, they read  $2^4$ ,  $2^{12}$ ,  $2^0$ ,  $2^0$ ,  $2^{18}$ , and  $2^8$ , and for the Zephyr graph, all are equal to  $2^8$ . The specific order in which projectors, boundary tensors, and bond tensors get combined in Fig. 7(d) depends on the specific values of those dimensions and, hence, also on a graph or diagram rotation. For a full Pegasus graph, the number of numerical operations is  $\mathcal{O}(2^{26} \chi^3)$ . For a Zephyr graph, it scales as  $\mathcal{O}(2^{32} \chi^3)$ , being a computational bottleneck for this lattice. Those operations again benefit from GPU hardware acceleration, although the available memory on the GPU and the size of intermediary tensors appearing during the contraction again limit feasible  $\chi$  values to around 10 [45].

### 2. Variational zipper algorithm for boundary MPS

The construction of a boundary MPS, which we employ in an approximate PEPS contraction, requires systematic approximation of an MPO-MPS product by an MPS with a limited bond dimension  $\chi$ . A particular challenge here is the bond dimensions of the MPO tensors (*site* and *mediating* tensors); see the previous section.

To this end, we combine the zipper scheme of Ref. [46] with the standard variational optimization of the resulting MPS, which aims to iteratively maximize its overlap with the target MPO-MPS product [24]. The procedure is depicted in Fig. 8. The zipper scheme applies MPO tensors one at a time while gradually switching the canonical form of the resulting MPS. This makes each singular value decomposition (SVD) approximation of a tensor, represented by dashed boxes in Fig. 8(c), used to truncate the

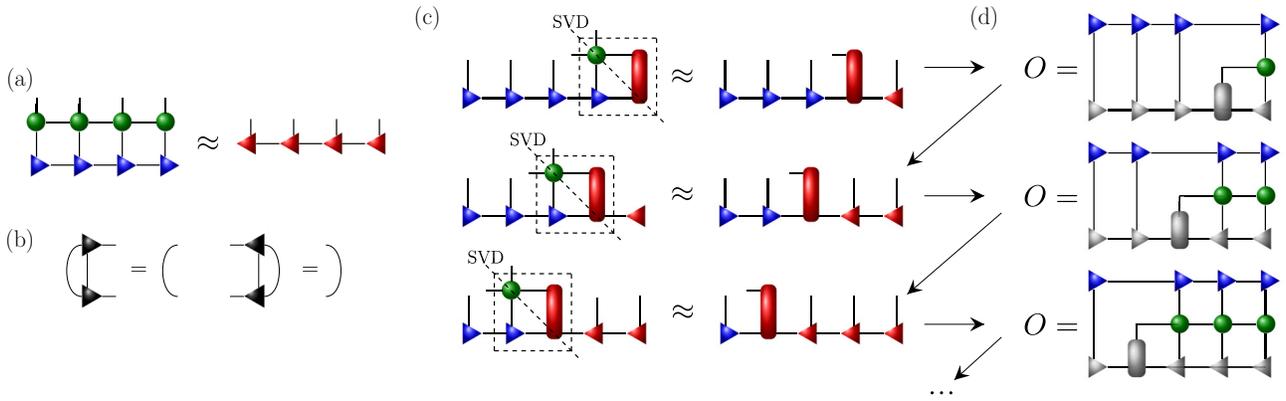


FIG. 8. Systematic approximation of the lower half of the PEPS lattice via boundary MPS. In panel (a), a boundary MPS for the  $(k + 1)$ th row with bond dimension  $\chi$  (blue triangles) is contracted with an MPO representing the  $k$ th row of the PEPS tensors (green dots) and approximated by a new boundary MPS of bond dimension  $\chi$  for the  $k$ th row (red triangles). To this end, we employ a zipper scheme, where the PEPS tensors are attached sequentially, one at a time, from right to left. Panel (c) starts with the initial MPS in the left canonical form [right(left)-pointing triangles represent left(right)-canonical MPS tensors; see panel (b)]. The first MPO tensor is attached, and a tensor in the dashed box is approximated via SVD, retaining  $\chi$  leading singular values. Here, the initial connecting tensor, represented by a red rectangle, is a trivial unity with all bond dimensions equal to one. The procedure is repeated in subsequent sites, with the MPS tensors to the left (right) of the connecting tensor in right (left) canonical form. This mixed canonical form makes an SVD approximation locally optimal. The problems we consider in this article have huge MPO bond dimensions, which requires resorting to randomized truncated SVD [43,44]. To increase the numerical stability, each step of a zipper procedure is followed by a variational optimization [24] of a new MPS, maximizing the overlaps in panel (d).

bond dimension, globally optimal at each step of the zipper. To handle the MPO tensors' bond dimensions related to Pegasus and Zephyr geometries, we use a truncated randomized SVD scheme [43,44] that directly targets the desired number of dominant singular values and vectors. This operates in a matrix-free way, meaning it does not require an explicit construction of the decomposed tensor, only a procedure that calculates its multiplication with a trial vector from the left and right sites. These take the form of diagrams such as in Fig. 7, allowing us to use the sparse structures of MPO tensors.

Randomized SVD algorithms targeting a subspace of singular values, while much faster and memory efficient than full-rank SVD algorithms based on Householder reflections [47], are numerically unstable and prone to getting stuck in local optima. To mitigate this issue, we conduct the truncation in two steps, targeting  $2\chi$  singular values with randomized SVD, which, after variational refinement, are further truncated to the target  $\chi$  singular values. More importantly, each step of the zipper procedure is followed by a sweep of variational optimization of the instantaneous stage of the zipper procedure, as shown in Fig. 8(d). To limit the computational effort, we can limit the range of the variational sweep to a few tensors in the vicinity of the instantaneous center of the zipper.

We have found the above procedure to be most stable, allowing us to operate with  $\chi$  values up to approximately 10 for the targeted lattices with the available resources [45]. We have also tested a strategy in which a full zipper sweep was performed first to give the initial state for

subsequent variational optimization, as well as a strategy in which only the variational optimization was used in combination with a gradual increase in inverse temperature  $\beta$ . Both those approaches, however, resulted in worse performance.

#### IV. LOCAL DIMENSIONAL REDUCTION

We additionally explore the combination of the algorithm in Sec. III with local dimension reduction of cluster degrees of freedom, which significantly reduces the numerical cost of subsequent tensor-network contractions. This is done by selecting the most probable states in each cluster, where we approximate the marginal probabilities of each Potts variable employing loopy belief propagation (LBP) algorithm [37,48,49].

We follow the standard LBP formulation [37], incorporating within it the projectors in Eq. (5) to significantly reduce the memory cost of the LBP representation for problems with large local dimensions. This reduction is essential for addressing the dimensions encountered in the Pegasus and Zephyr geometries after clustering.

Specifically, the LBP procedure seeks to find the marginal probabilities  $p(x_n)$  of configurations  $x_n$  in a cluster  $n$  by iteratively updating messages from clusters to edges and from edges to clusters. Such iterations converge to exact results for problems defined on a tree. Much worse performance is expected for lattices with loops, such as those considered here; however, the results only serve to select a fraction of the local degrees of freedom for each

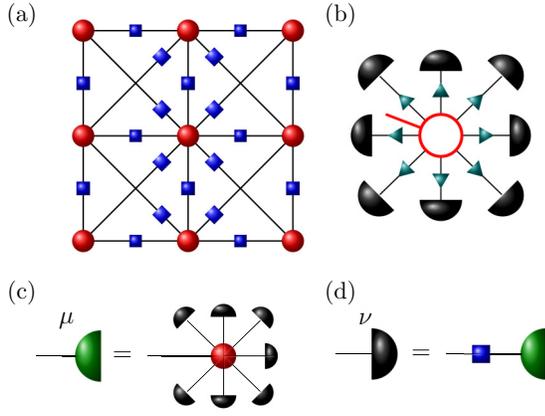


FIG. 9. Local dimension reduction. To decrease the numerical cost of tensor-network contraction, we employ a loopy belief propagation (LBP) algorithm to approximate the marginal probabilities of each cluster variable and retain a given number of the most probable (according to LBP) configurations in each cluster. This substantially reduces the cost of a subsequent global branch-and-bound search. An LBP can be rephrased as a simple approximate contraction scheme of a tensor-network representing the partition function, such as in panel (a) [see Fig. 5(a)], aiming at an approximation of local marginals in panel (b). These are obtained via an iterative procedure shown in panels (c),(d), where one sequentially updates, in panel (c), messages from clusters to edges [boundary tensors represented by green half-circles], and, in panel (d), messages from edges to clusters. In our numerical experiments, we perform a fixed number of such updates before the estimation of local marginal probabilities in panel (b).

cluster, and as such, this approach is viable for testing. The main steps of the procedure are outlined in Fig. 9 and can naturally be interpreted as a form of approximate tensor network contraction.

Below, we simplify the LBP formulation for general factor graphs [37], which can contain interactions between arbitrary numbers of variables, to the case of at most two-site terms appearing in Eq. (3). The messages from edges to clusters,  $v_{mn}(\bar{x}_n)$ , where the cluster indices  $m$  and  $n$  uniquely identify directional edges for two-site interactions, are depicted as black semicircles in Fig. 9. These are initialized as a uniform distribution,  $v_{mn}(\bar{x}_n) \sim 1$ , implying equal probabilities for all states. The messages from the clusters to the edges of the graph, denoted by  $\mu_{mn}(\bar{x}_m)$  and represented as green semicircles in Fig. 9, follow the update

$$\mu_{mn}(\bar{x}_m) = \sum_{p^{mn}(x_m)=\bar{x}_m} \exp(-\beta E_{x_m}) \prod_{m' \in N(m) \setminus n} v_{m'm}(x_m), \quad (11)$$

as shown in Fig. 9(c). Here,  $N(m)$  indicates sites interacting with site  $m$ , and we use projectors from Eq. (5), where  $v_{m'm}(x_m) = v_{m'm}(\bar{x}_m)$ , corresponding to the definition of the *site tensor* represented by red circles in Fig. 5. Subsequently, one updates the message from the edges to clusters as

$$v_{mn}(\bar{x}_n) = \sum_{\bar{x}_m} \exp(-\beta \bar{E}_{\bar{x}_n \bar{x}_m}) \mu_{mn}(\bar{x}_m), \quad (12)$$

as depicted in Fig. 9(d).

After several such iterations (a fixed number or after reaching convergence), we can approximate the marginal probabilities of the spin configurations associated with each cluster within the graph. In particular, the probability of configuration  $x_v$  is proportional to the product of all messages originating from neighboring edges linked to the respective cluster [see Fig. 9(b)],

$$p(x_n) \propto \exp(-\beta E_{x_n}) \prod_{m \in N(n)} v_{mn}(x_n). \quad (13)$$

For dimensional reduction, we select the desired number of most probable states in each cluster according to such a procedure.

We note that within the temperature range considered in this study, belief propagation (BP) messages do not fully converge, with the maximal change in messages during iteration saturating at around  $10^{-3}$ . Hence, after reaching saturation in convergence, messages from a predefined number of iterations are used to select representatives of each cluster. At sufficiently high temperatures (small  $\beta$ ), BP is expected to converge; however, such temperature regimes are not practically relevant to the optimization problems studied here. Finally, we note that the selection based on local marginals may not identify local configurations contributing to a globally optimal state.

## V. RESULTS

In this section, we present three classes of instances that we used for testing the algorithms of Secs. III and IV and discuss the performance metrics we adopt. Finally, we show the results for problems defined on the Pegasus and Zephyr geometries. In addition to the TN-based algorithm described in Secs. III and IV, all instances were executed on the D-Wave quantum annealer and optimized using the SBM algorithm [21–23]. In addition, we have collected further evidence in the Appendixes, including problems defined on a simple square lattice and a square lattice with diagonals. Therein, we also show the dependence of our algorithm's performance on various parameters, most notably the inverse temperature,  $\beta$ .

TABLE I. Classes of problem instances considered in this article.

Class	Description
I	Random couplings in $[-1, 1]$ .
II	Class I + random local fields in $[-0.1, 0.1]$ .
III	CBFM-P instances of Ref. [51] for Pegasus graph.
IV	Tile planting instances of Ref. [52,53] for Pegasus graph.

### A. Problem instances

We employ random instances with all available couplings drawn from the same distribution, primarily focusing on the Pegasus and Zephyr coupling graphs [50] (see Fig. 1):

(a) Class I. We employ a uniform distribution in  $[-1, 1]$  for  $J_{ij}$  and set local fields  $h_i = 0$  [see Eq. (1)]. These instances have a global reflection symmetry.

(b) Class II: Similar to Class I, instances are defined by couplings drawn from a uniform distribution in  $[-1, 1]$ . We include small local fields with  $h_i$  drawn from a uniform distribution in  $[-0.1, 0.1]$ , breaking reflection symmetry.

(c) Class III: We adopt the corrupted biased ferromagnets instances formulated on a Pegasus graph (CBFM-P) of Ref. [51]. These are defined by discrete distributions with  $P(J_{ij} = 0) = 0.35$ ,  $P(J_{ij} = -1) = 0.10$ ,  $P(J_{ij} = 1) = 0.55$ ,  $P(h_i = 0) = 0.15$ , and  $P(h_i = -1) = 0.85$ .

(d) Class IV: We use tile planting instances on a square spin lattice, following the method outlined in Ref. [52,53]. The instances defined on a square lattices with system sizes  $L = 10$ ,  $L = 20$ , and  $L = 40$  are subsequently embedded into Pegasus graphs. After embedding, the numbers of spins increase to 164, 721, and 3065, respectively. The instances are constructed with a planted solution, meaning that the ground state is known *a priori*. The complexity of the planted instances is controlled by the probabilities  $p_1$ ,  $p_2$ , and  $p_3$ , such that  $p_1 + p_2 + p_3 \leq 1$ . We set  $p_2 = 1.0$ , for which the instances are expected to be the hardest to solve.

We summarize the considered problem classes in Table I.

The probability distribution defining instances in class III was selected for Pegasus geometry through a trial-and-error method to ensure that problems from this class are challenging for classical algorithms. Specifically, Ref. [51] presents extensive benchmarks comparing the time to solution between the D-Wave Advantage quantum annealer and a selection of classical algorithms, including simulated annealing, parallel tempering with isoenergetic cluster moves, integer quadratic programming, and message passing (but not SBM). The study demonstrated that, for this class and certain target approximation ratios, the quantum annealer provides run-time benefits over a tested collection

of established classical methods. A recent comparison with the SBM algorithm was performed in Ref. [54] on max-cut problems, revealing the competitive performance of SBM.

### B. Metrics: best energy, diversity of solutions

We use two fundamental performance metrics: the best energy found and the diversity of identified low energy configurations, both within the framework of time to solution.

In the first metric, we focus on the lowest energy obtained with given resources, which we quantify using solver runtime. We will be plotting a relative energy difference above the best energy found,

$$d_E = (E - E_{\text{best}})/2|E_{\text{best}}|, \quad (14)$$

where the reference energy  $E_{\text{best}}$  is established by collecting the results of all tested solvers when the ground truth is unknown.

In the second metric, we probe the quality of sampling of diverse approximate solutions of a spin-glass problem [55,56]. We summarize the idea behind the diversity measure below. We focus on good-quality solutions with energies within a given *approximation ratio*  $a_r$ , that is, the solutions for which  $d_E \leq a_r$ . The goal is to select the largest set of independent solutions from the available set of such solutions. The size of the set provides a diversity of solutions. We refer to two spin configurations,  $\underline{s}_N$  and  $\underline{s}'_N$ , using the original spin variables, as independent if their Hamming distance  $d$  is above a given threshold

$$d(\underline{s}_N, \underline{s}'_N) \geq RN, \quad (15)$$

where  $N$  is the number of spins and  $R \in [0, 1]$  is the relative distance threshold. Indeed, a large threshold value makes it unlikely to transition between two such solutions by relying on local updates. Below, we set  $R = 1/4$  and  $R = 1/2$ . For class I instances, we do not distinguish solutions differing by a global reflection; i.e., we first flip some solutions to ensure that they have the same value for the first spin in all samples.

To compare the performance of the considered solvers, we proceed as follows. We start with the estimation of the ground-truth diversity  $\mathcal{D}_{\text{total}}$  of a given problem instance. To this end, we collect the solutions from all solvers and search for the largest set of independent solutions within given approximation ratio  $a_r$ . This is equivalent to finding a maximum clique (i.e., a subgraph of the analyzed graph in which each pair of nodes is adjacent) in the graph for which low-energy solutions form the graph's nodes, and any two low-energy states with a Hamming distance greater than the threshold are adjacent. The maximum clique problem is known to be NP-complete [57], and we approximate its solution using a randomized heuristic approach [58]. Here,

$\mathcal{D}_{\text{total}}$  is equal to the size of the set obtained in the above procedure.

The obtained reference set of independent solutions is then used to calculate the diversity of solutions  $\mathcal{D}$  found by each solver for a given problem instance. To this end, we iterate over all low-energy solutions obtained by a given solver and mark the reference independent solution to which they are the closest. The diversity for the solver is equal to the number of marked states.

### C. Results of the ground-state search

Figure 10 shows the time to median approximation ratio for 10 instances of class I for Pegasus geometries with 216, 1176, and 5376 spins, and Zephyr graphs with 332 and 563 spins. Analogous data for other instance classes are collected in the Appendixes, where we observe similar performance for instances in classes I–III. For the largest considered problems in class IV, the TN algorithm outperforms QA and SBM in terms of the best energy found, improving  $d_E$  by a factor of 3 to 4.

In addition to full-space TN simulations, we additionally conduct simulations with local dimensional reduction of cluster degrees of freedom, as described in Sec. IV. Here, we reduce the  $2^{24}$  cluster states to  $2^{20}$  and  $2^{16}$  states for Pegasus instances, and from  $2^{16}$  states to  $2^{14}$  and  $2^{12}$  states

for Zephyr instances. Due to memory limitations, for the Pegasus instances with 5376 spins, we only performed calculations with local dimensional reduction to  $2^{16}$  states. The parameter selection was performed separately for each type of graph: an inverse temperature  $\beta = 0.5$  was used for the Pegasus instances, while  $\beta = 1.0$  was used for the Zephyr instances. We further show the influence of  $\beta$  on the quality of the results in the next section and in Appendix C.

For the smallest Pegasus graph in Fig. 10(a), TN simulations with no dimensional reduction find the ground state (for median instance). Employing dimensional reduction to  $2^{16}$  prevents us from identifying the ground state in some cases—we have checked that this is due to dimension reduction that no longer supports the ground state configuration, showing the limitations of the LBP preprocessing. For larger Pegasus lattices in Figs. 10(b) and 10(c), the best energies reached are systematically below the approximation ratio of  $10^{-2}$ , both with and without dimensional reduction. We are not, however, typically able to find the ground state in those cases. The outcome of the TN procedure depends on the edge of the quasi-2D lattice from which the branch-and-bound is initiated, as indicated by the improvement in the best results after collecting data from various trials. This suggests deficiencies in the stability of the approach for feasible execution parameters.

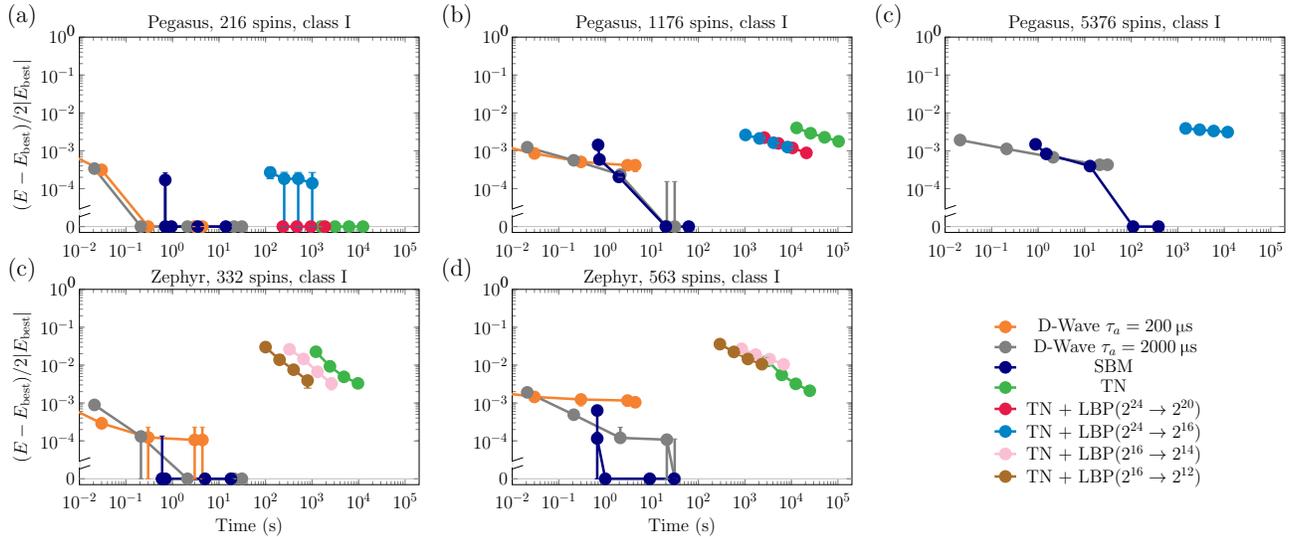


FIG. 10. Time to approximation ratio. We focus on class I instances for Pegasus and Zephyr for a selection of system sizes. We compare the considered solvers in terms of the time needed to reach a given quality of solution, quantified by energy above the best identified one, selected among D-Wave QA (orange and gray), SBM (dark blue), and TNs. Here, we show the median based on 20 instances. In addition to full-space tensor-network results (green), we plot results obtained with dimensional reduction of local cluster degrees of freedom: truncation from  $2^{24}$  states for Pegasus to  $2^{16}$  states (blue) and  $2^{20}$  states (red), and truncation from  $2^{16}$  states for Zephyr to  $2^{12}$  states (brown) and  $2^{14}$  states (pink). The points for QA and SBM are from 10, 100, 1000, 10 000, and all available samples (trajectories). Here, we estimate QA time as annealing time plus  $100 \mu\text{s}$ . SBM and TN times are for a single GPU [45]. While the TN algorithm is in principle deterministic, it can be executed starting from different corners of a quasi-2D lattice (8 lattice transformations in total), and the points for TN correspond to trying 1, 2, 4, and 8 such transformations. Here, we used  $\beta = 0.5$  for Pegasus and  $\beta = 1.0$  for Zephyr instances, MPS bond dimension  $\chi = 8$ , and the maximal number of branches in the search  $M = 1024$ . The error bars of the median follow from bootstrapping.

The Zephyr geometry proves to be more challenging to our approach [see Figs. 10(d) and 10(e)]. In some cases, we observe a trade-off between TN contraction stability and the degree of dimension reduction. While overall, the best results are obtained for no reduction, there are instances where the best result, even the ground state, is identified only after a preceding TN-based branch-and-bound search with a local dimensional reduction. We expect that the drop in performance of our TN-based approach for the Zephyr geometry, as compared to Pegasus, is due to the larger fraction of diagonal interactions in the former and its impact on contraction stability. In Appendix B, we show the results for an Ising model defined on a plain square lattice and a square lattice with additional diagonal terms, where TN produces the best solutions.

Finally, Fig. 10 includes the results of QA and SBM. Both approaches operate on a timescale that is orders of magnitude smaller than TN and typically allow us to identify the ground state. The probabilistic nature of these algorithms makes the operation time roughly proportional to the number of samples, with  $10^4$  samples typically sufficient to identify the ground state in our examples. In contrast, the TN algorithm is deterministic and requires completing a systematic sweep through the system to return all identified solutions in one go.

### D. Diversity of solutions

Figure 11 shows the time to median diversity ratio calculated for the same set of instances as in Fig. 10. Here, we focus on a large distance threshold  $R = 1/2$ , where distinct

solutions differ by at least half of the spins, and approximation ratio  $a_r = 0.01$ . Other classes of instances, as well as the data for distance threshold  $R = 1/4$ , are collected in the Appendixes.

For  $R = 1/2$ , all solvers provide a comparable level of sampling fairness, cross-checking that they successfully explore distinct regions of the low-energy problem manifold; however, for  $R = 1/4$  (see Appendix A), QA and SBM obtained a significantly larger number of diverse states than the TN approach, which outputs similar numbers as for  $R = 1/2$  (operating with  $M = 1024$  in the branch-and-bound search). This may be related to the fact that the D-Wave machine and SBM generally yield orders of magnitude more states in their results (equal to the number of samples) than the TN approach, which is limited by the branch-and-bound search width  $M$ . As we relax the criterion for diverse states by decreasing  $R$ , this quantitative advantage naturally translates to more diverse states being found.

## VI. CONTRACTION STABILITY

To diagnose the sources of limitation in the TN approach, we now focus on the smallest problem defined on the Pegasus graph with 216 spins in class I. We test a range of inverse temperatures  $\beta$ , boundary MPS bond dimensions  $D$ , and adopted numerical precisions. We employ local dimension reduction to  $2^{20}$  to expand the available range of  $D$  (we verify that ground states remain supported in reduced spaces). For clarity, in this section, we adopt a metric that tests whether a given algorithm run

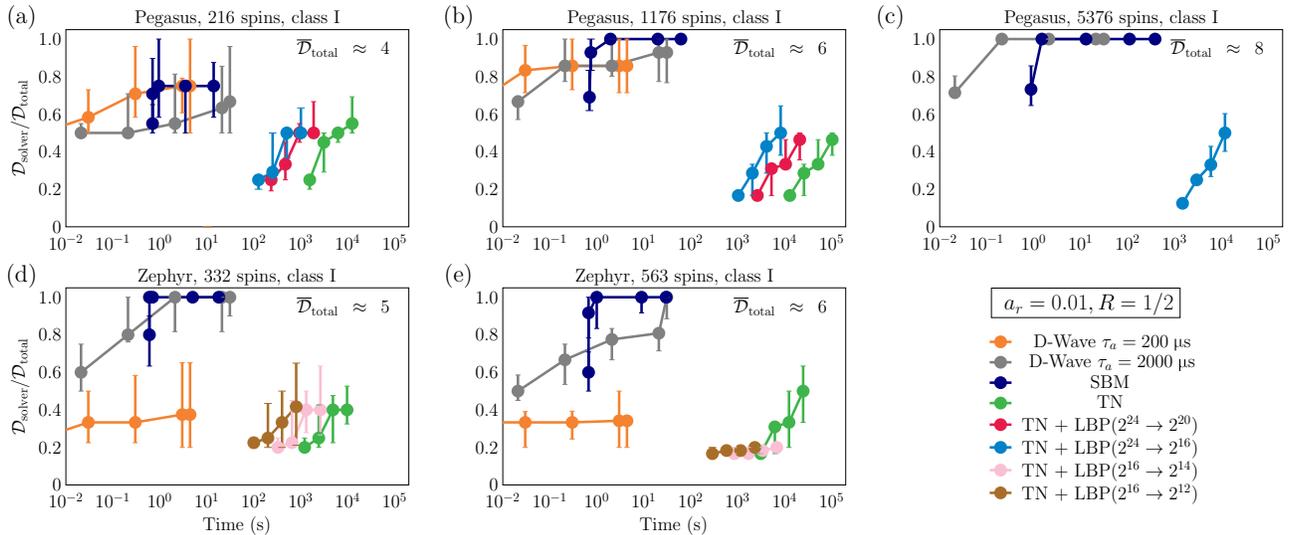


FIG. 11. Time to diversity ratio. We show the time needed to reach a given median fraction of diverse solutions within  $a_r = 0.01$  for class I instances of the Pegasus and Zephyr graphs. The total diversity of each instance is obtained by combining the outputs of all considered solvers. The median among 20 instances is annotated as  $\bar{D}_{\text{total}}$  in each panel. We focus on the relative distance threshold  $R = 1/2$ , which gives sets of independent solutions consisting of only a few spin configurations, and all three solvers are able to identify such varying states. The simulation parameters are the same as in Fig. 10.

has found the ground state. The resulting success rate is shown in Fig. 12(a), collecting data for 20 instances and all possible rotations of each instance.

A general observation is that there is a trade-off between the resolution offered by the branch-and-bound procedure (where higher  $\beta$  is desired) and the stability of TN contraction (where smaller  $\beta$  allows for more precise calculation of marginals). Indeed, for the smallest values of  $\beta < 0.25$ , the procedure cannot identify the ground state, as the branch-and-bound procedure (here, with  $M = 1024$ ) gets stuck in local minima—independent of contraction parameters.

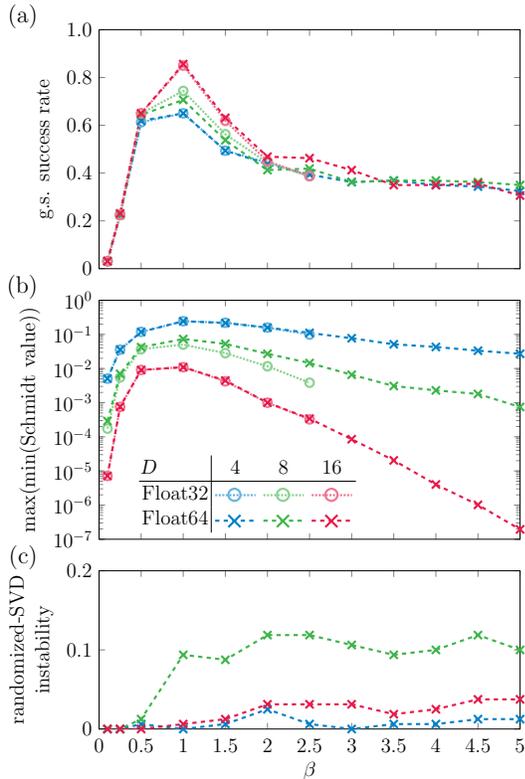


FIG. 12. Contraction stability. We focus on a minimal Pegasus example with 216 spins. In panel (a), we show a percentage of algorithm runs where the ground state has been identified, calculated from 20 problem instances and all right rotations (contraction directions) for each instance. We identify the existence of optimal inverse temperature  $\beta$ , giving a trade-off between the branch-and-bound resolution and TN contraction failures. We test the influence of the boundary MPS bond dimension  $D$  and numerical precision on the latter. In panel (b), we illustrate the truncation error, estimated from the smallest retained singular values at the worst MPS cut (the median over all instances is plotted). In panel (c), we show a percentage of runs where repeated execution gave different outcomes (with respect to finding the ground state). This can be attributed to randomized SVD being employed in the algorithm and the inability of variational MPS optimization to correct the resulting fluctuations.

Those local minima can be avoided by employing larger  $\beta$  values, where, however, the failure of the procedure can be attributed to TN contraction failure. From this perspective, the metric we adopt here provides a direct probe of the stability of the contraction at larger  $\beta$ . We note that the PEPS tensors we use are built from non-negative elements. In general, this improves contraction stability [59] (at the very least, this minimizes numerical-precision errors related to subtracting numbers of similar magnitude). Negative tensor elements, however, do appear in approximate boundary MPS.

Interestingly, in the considered range of parameters, the numerical precision—i.e., a default 64-bit precision or a lowered 32-bit precision used during contraction—does not appreciably affect the results (although for 32-bit precision, some underlying numerical operations break for  $\beta \geq 3$ ). This can be understood from Fig. 12(b), where we show the truncation accuracy, measured by the smallest retained singular value of boundary MPSs at the worst cut (the cut with the largest such singular value). For  $\beta < 3$ , 32-bit precision is sufficient to represent those values for the tested  $D$ . We can also see, from the plots at fixed  $D$ , that with increased  $\beta$ , the singular values vanish more quickly. This indicates that numerical precision should ultimately become an issue for the largest  $\beta$ , desired from the perspective of branch and bound.

Finally, we identify that the procedure failure for the largest  $\beta$  value is closely tied to randomized truncated SVD [43,44], which we employ in the calculation of approximate boundary MPSs. Indeed, while the branch-and-bound procedure, in principle, is deterministic, we see that employing randomized SVD can lead to different outcomes for repeated algorithm runs (with all parameters otherwise fixed). In Fig. 12(c), we plot a percentage of cases where two repeated executions of the algorithm gave different outcomes (in terms of our metric focusing on ground-state identification; the instability would be even more pronounced if we compared obtained states). Such an instability is marginal for the smallest values of  $\beta$  and the smallest  $D = 4$ ; however, the percentage of unstable runs becomes significant for larger values of  $\beta$  and  $D$ .

This indicates that the randomized SVD employed in boundary MPS compression results in local optima that the one-site variational overlap maximization procedure cannot escape (see Sec. III B 2). This issue becomes particularly severe when singular values vanish quickly, with a few dominant ones differing by orders of magnitude—as for  $D \geq 8$  at the largest  $\beta$  in Fig. 12. This shows that a stable procedure to optimally identify boundary MPSs (or, more broadly, contract the network) plays a critical role beyond basic parameters like the bond dimension  $D$ .

The adoption of randomized SVD and variational optimization of boundary MPSs limited to one-site updates is a consequence of the huge dimensions of local spaces and the resulting PEPS bond dimensions. These appear difficult

to avoid in the optimization problems we focus on in this article. Any improvement in the randomized SVD performance should translate to improved quality of the results from our TN procedure. This also partially explains the good performance of the algorithm for the Chimera graph in Ref. [31], where standard SVD—and more reliable initialization of boundary MPS optimization—are available, allowing for larger  $\beta$  values. Nevertheless, we observe that the hardness of contraction grows with lattice connectivity. Typically, the TN approach performs reasonably well for a square lattice, but its performance declines with increased connectivity (including diagonal couplings in a quasi-2D graph). Zephyr seems more complicated than Pegasus, which, in turn, is harder than Chimera or a square lattice.

## VII. DISCUSSION

In this work, we implemented a TN-based branch-and-bound search algorithm to identify low-energy states of quasi-2D spin-glass problems. We focused on optimization problems defined on Pegasus and Zephyr graphs, which are currently realized in the D-Wave QA.

We compare the quality of solutions output from our approach with those from the QA and SBM algorithms. The latter two, which can be considered examples of randomized algorithms, operate on approximately 2 orders of magnitude smaller timescales and, via sampling, provide denser coverage of the low-energy solutions manifold for the considered random problems. The deterministic nature of our branch-and-bound approach likely contributes to the reduced expressiveness of sampling, suggesting a possible route for future improvements of the method.

TN operates best in low-dimensional systems, and in this article, we execute them for problems on the verge of their practical applicability. The limitation to low-dimensional systems is shared by the QA, as we focus here on optimization problems that use the entire interaction graph of the existing hardware. SBM, however, is not limited by such considerations and can be executed on a fully connected graph.

TN algorithms, including our approach, typically have a sequential nature, making them hard to parallelize and affecting the total execution time for a single problem instance. This aspect is shared by a QA, which produces one classical sample at a time. Nonetheless, a single annealing execution time of the order of a millisecond makes it competitive in outputting thousands of samples (though the access time to the online system might add significant overhead). In comparison, the SBM algorithm, like MCMC approaches, is easily parallelizable, allowing for multiple trajectories to be calculated simultaneously, efficiently using hardware accelerators such as GPUs.

Nevertheless, our TN approach is not limited to Ising problems, like the two latter solvers. It can be directly

employed for generalized Potts problems (or equivalently, Random Markov Fields models) defined on king's graphs [see Fig. 1(d)].

We identify the limitations of our approach related to the huge bond dimensions appearing in the TN representation of marginals for the families of problems we focus on and instabilities in approximate boundary MPS contraction schemes. The performance of our approach might be further improved by introducing additional heuristic TN gauge transformations to stabilize the contraction, which we do not explore in this work.

Our results also illustrate that optimization problems can be employed as stringent tests of the performance of various approximate TN contraction schemes, where the quality of obtained low-energy states directly probes the quality of the contraction, which is a #P-hard problem in general.

One emerging area is the development of a heterogeneous high-performance quantum-classical computing architecture with adaptive graph partitioning subroutines [60]. This framework could consist of different kinds of coprocessors working in concert to accelerate the sampling of different subgraphs or complementary areas of configuration space. This paradigm of hybrid computing could include quantum accelerators (analog quantum annealers or digital quantum approximate optimization algorithms), conventional classical computing accelerators (CPU and GPU), and unconventional classical processors (in-memory [61] and/or probabilistic accelerators [62]). We envision that TN approaches can also be incorporated as specialized deterministic quantum-inspired solvers for sampling low-dimensional subgraphs within such heterogeneous frameworks.

## ACKNOWLEDGMENTS

This project was supported by the National Science Center (NCN), Poland, under Projects No. 2020/38/E/ST3/00269 (T.S.), 2022/47/B/ST6/02380 (B.G.), and 2020/38/E/ST3/00150 (A.D.), and Project No. 2021/03/Y/ST2/00184 within the QuantERA II Programme, which has received funding from the European Union Horizon 2020 research and innovation program under Grant Agreement No. 101017733 (M.M.R.). This work was in part supported by the Defense Advanced Research Projects Agency (DARPA) under Air Force Research Laboratory (AFRL) Contract No. FA8650-23-3-7313 (M.M.).

## DATA AVAILABILITY

The data that support the findings of this article are openly available [63,64].

## APPENDIX A: AUXILIARY RESULTS

In Fig. 13, we show the time to approximation ratios for class II, III, and IV instances achieved by all examined solvers: D-Wave QA, SBM, and the TN approach. In contrast to classes I, II, and III, which exhibit similar performance trends, class IV instances demonstrate an advantage in energy results achieved by the TN approach. Although TN outperforms both D-Wave QA and SBM in terms of energy for instances with 721 and 3065 spins, it fails to identify their ground states.

The time to diversity ratios for classes I, II, III, and IV with the distance thresholds  $R = 1/2$  and  $R = 1/4$  are shown in Figs. 14 and 15, respectively.

## APPENDIX B: SQUARE LATTICE

To illustrate our motivations for using TN in optimization problems, we present results for class I instances on a

square lattice with one spin per lattice site and a king's lattice generated by adding diagonals to the square lattice. For reference, we conducted simulations using SBM. The results, focusing on best energies, are presented in Fig. 16. For these geometries, the TN-based technique consistently finds states with lower energies than SBM. The king's lattice proves harder for TN than the square lattice. Indeed, for the square lattice, nearly all runs for all instances and lattice transformations for  $\beta = 4$  and  $\beta = 8$  result in the ground state (beyond the median for  $\beta = 4$  shown in Fig. 16). For the king's lattice, the success rate in identifying the ground state for a  $50 \times 50$  lattice falls to around 90% for  $\beta = 4$ , and around 50% for  $\beta = 8$ .

The diversity of solutions (vastly different, with  $R = 1/2$ ) found by both solvers is comparable, as shown in Fig. 17, indicating that both solvers are capable of sampling from the same distant parts of the low-energy

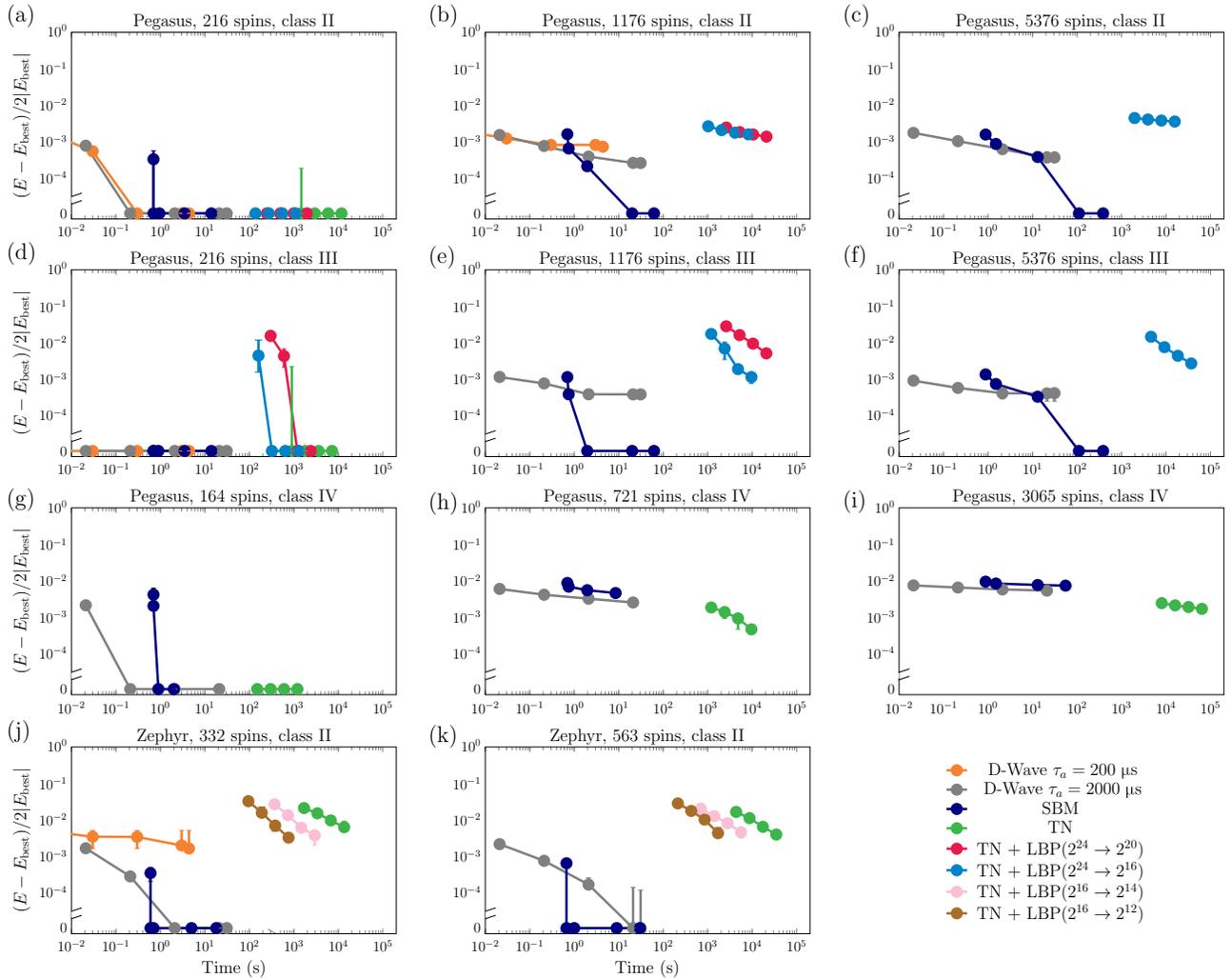


FIG. 13. Time to approximation ratio. Here, we show the results for instances in classes II, III, and VI for Pegasus and Zephyr geometries for a selection of system sizes. The data complement the results in Fig. 10 and were obtained using the same simulation parameters. We show the median based on 20 instances.

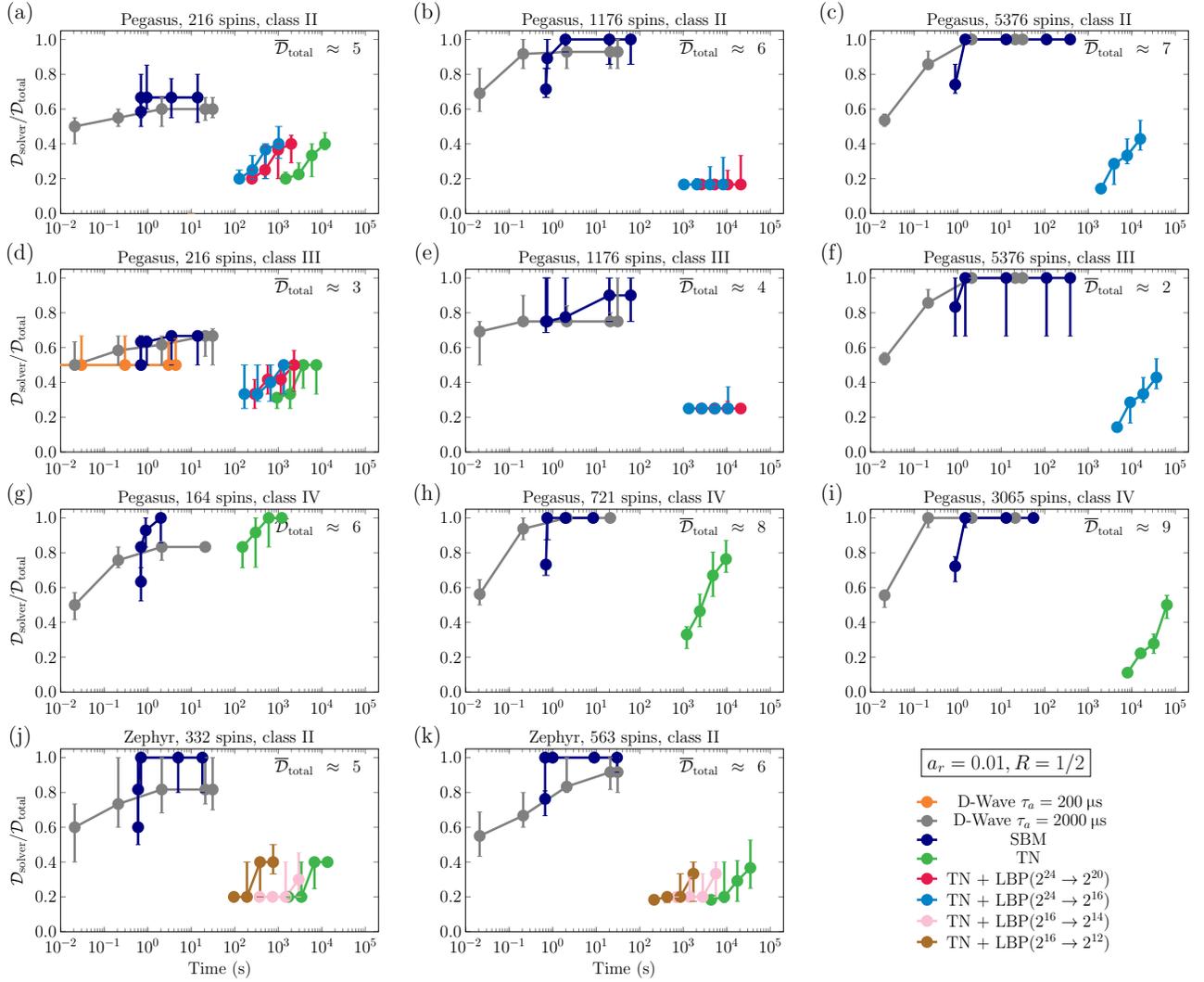


FIG. 14. Time to diversity ratio for  $R = 1/2$ . We show the time needed to reach a given fraction of diverse solutions within  $a_r = 0.01$  and  $R = 1/2$  for instances in classes II, III, and IV, complementing the data in Fig. 11.

manifolds; however, TN no longer has an advantage in that metric.

Finally, we also tested the zipper boundary MPS contraction procedure using randomized SVD (see Sec. III B 2) against a more standard procedure where the entire MPO (related to a PEPS row) is applied first to the previous boundary MPS, the result is truncated using full-rank SVD, and fine-tuned variationally. The quality of the results obtained from both procedures turns out to be similar in these problems.

### APPENDIX C: DECOMPOSITION OF PEPS INTO MPOS, OPTIMAL $\beta$ , AND STABILITY TEST

We can consider various types of decomposition of the PEPS network into MPOs, as shown in Figs. 18(a) and 18(b) with pale-blue boxes. All the results presented

in this article employ the decomposition in Fig. 18(a). Here, we compare the influence of such decomposition using a single instance in class I on the Pegasus graph with 1176 spins as an example. At the same time, we probe various inverse temperatures  $\beta$ . In Fig. 18(c), we show the relative energy above the ground state for runs using all eight lattice rotations. We observe weak sensitivity of the results to both the decomposition and  $\beta$ . All our results in the main text use  $\beta = 0.5$  for the Pegasus graph. Based on a similar analysis, we decided on  $\beta = 1.0$  for instances defined on the Zephyr graph. In Fig. 18(d), we show another metric indicating contraction stability. Specifically, we compare the value of the partition function  $Z$  following eight lattice rotations (thus, performing boundary-MPS contraction of the PEPS network from various edges of the lattice). Figure 18(d) shows the standard deviation among the eight obtained values for each

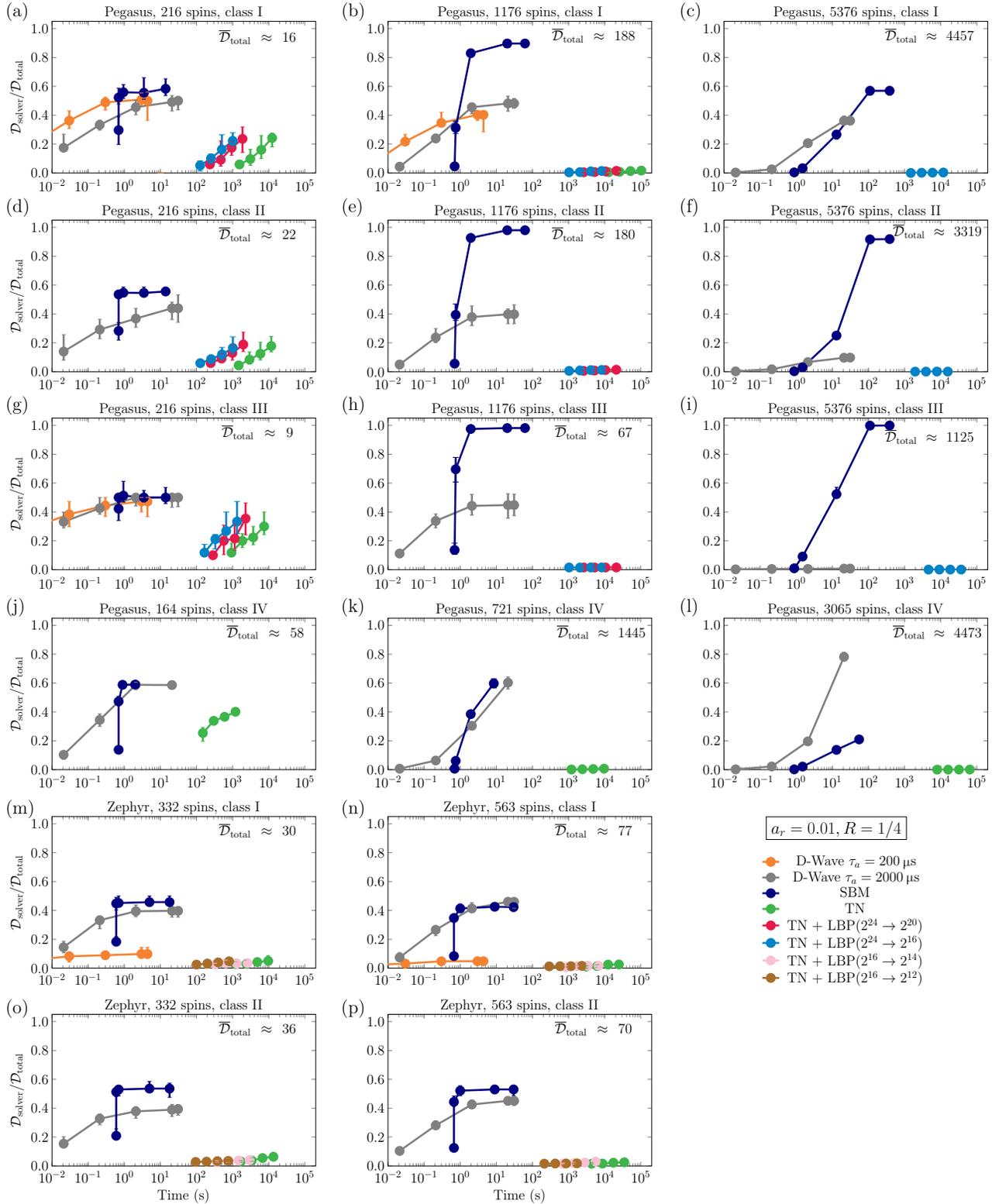


FIG. 15. Time to diversity ratio for  $R = 1/4$ . We show the data for all four instance classes and approximation ratio  $a_r = 0.01$ , complementing similar results in Figs. 11 and 14 for  $R = 1/2$ . For large system sizes, the fraction of diverse solutions identified by TN becomes negligible compared to the two other solvers, where the limited available width of the branch-and-bound procedure ( $M = 1024$ ) apparently allows identifying only a small fraction of independent solutions for  $\mathcal{D}_{\text{total}}$  in the hundreds. The simulation parameters are the same as in Fig. 10.

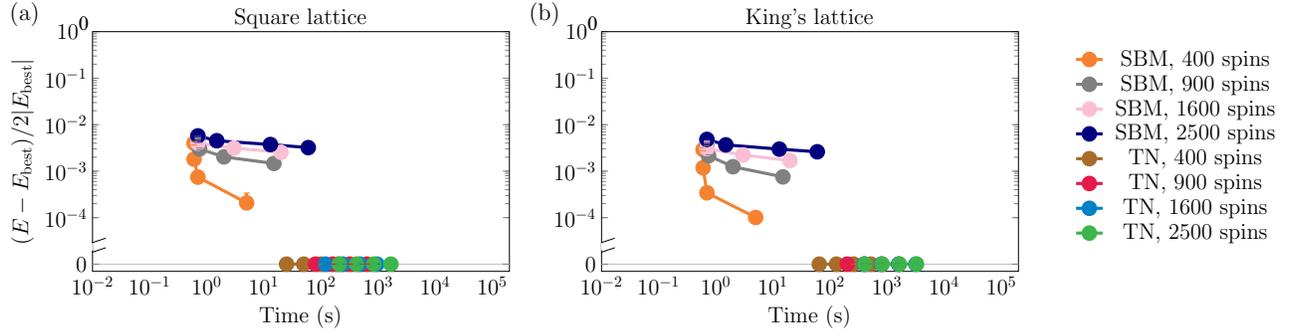


FIG. 16. Time to approximation ratio for (a) class I instances on a square lattice and (b) a king's lattice (square lattice with diagonals) for a selection of system sizes. As in Fig. 10, we show the median based on 20 problem instances. The best energies  $E_{\text{best}}$  have been chosen from the results of the SBM and TN algorithms. For TN, we show data for bond dimension  $\chi = 16$ , the maximum number of states considered during the search set to  $M = 256$ , and inverse temperature  $\beta = 4.0$ .

$\beta$  and decomposition. The decomposition in Fig. 18(a) appears more stable within this metric for the lowest values of  $\beta$ , with the deviation growing systematically as  $\beta$  increases.

Finally, we focus on dimensional reduction based on LBP and how the results depend on  $\beta$ . The data are presented in Fig. 19, where we perform simulations for the same single instance as in Fig. 18 above. In principle, the LBP procedure and TN branch-and-bound algorithm do not have to employ the same  $\beta$ , and here we probe various combinations of  $\beta_{\text{BP}}$  and  $\beta_{\text{algorithm}}$  (in the main article, they are set to the same value). The overall consistency of the results seems mostly influenced by  $\beta_{\text{algorithm}}$ .

#### APPENDIX D: SIMULATED BIFURCATION MACHINE

Building upon the concept of quantum adiabatic optimization using a network of nonlinear quantum oscillators, the simulated bifurcation machine (SBM) leverages the phenomenon of bifurcation within a classical nonlinear system governed by Hamilton's equations of motion [22]:

$$\begin{aligned} \dot{q}_i &= \frac{\partial H_{\text{cl}}}{\partial p_i} = a_0 p_i, \\ \dot{p}_i &= -\frac{\partial H_{\text{cl}}}{\partial q_i} = -[q_i^2 + a_0 - a(t)] q_i + c_0 \Phi(q_i), \end{aligned} \quad (\text{D1})$$

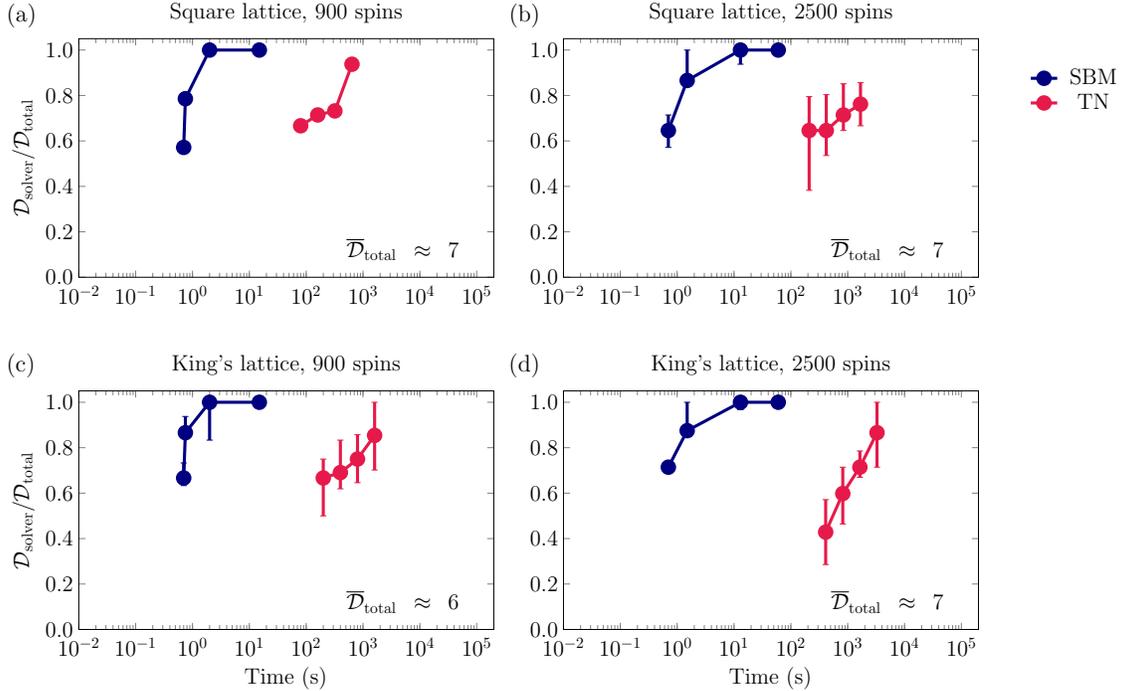


FIG. 17. Time to diversity ratio for (a),(b) class I instances on a square lattice and (c),(d) a king's lattice for the distance threshold  $R = 1/2$  and approximation ratio  $a_r = 0.005$ . The simulation parameters are the same as in Fig. 17.

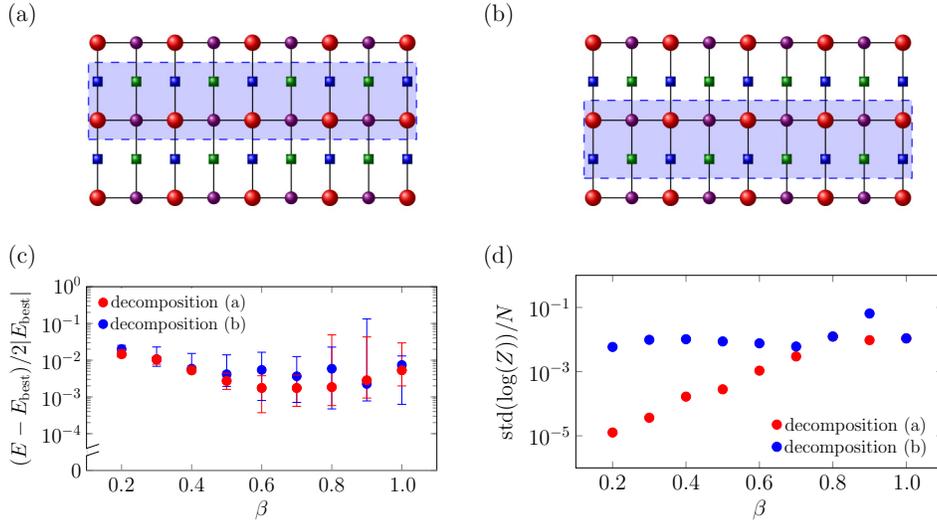


FIG. 18. The PEPS tensor network can be decomposed into MPOs in many ways, as shown in panels (a),(b) with pale-blue boxes. In panels (c),(d), we focus on a single instance in class I for the Pegasus graph with 1176 spins. Panel (c) displays the energy obtained by the TN solver above the best energy identified for this instance. For each  $\beta$  and decomposition, we show the median from eight runs of the algorithm using different rotations of the lattice (effectively performing the search and contraction from various corners of the lattice). The best and worst results are indicated by error bars. Panel (d) focuses on the stability of contraction, using the partition function as the metric, and compares the results obtained for eight lattice rotations.

where  $\Phi(q_i) = \sum_{j=1}^N J_{ij} q_j + h_i$ , and

$$H_{\text{cl}} = \frac{a_0}{2} \sum_{i=1}^N p_i^2 + \sum_{i=1}^N \left( \frac{q_i^4}{4} + \frac{a_0 - a(t)}{2} q_i^2 \right) - c_0 \sum_{i=1}^N \left( h_i q_i + \frac{1}{2} \sum_{j=1}^N J_{ij} q_i q_j \right), \quad (\text{D2})$$

in which  $J$  and  $h$  are defined in Eq. (1).

This system can be interpreted as particles with mass  $1/a_0$ , positions  $q_i$ , and momenta  $p_i$ , moving within a time-dependent potential and interacting through Ising-like couplings. The constants  $a_0$  and  $c_0$  are hyperparameters (in this work, we set  $a_0 = 1$  and  $c_0 = 1/\lambda_{\text{max}}$ , where  $\lambda_{\text{max}}$  denotes the largest eigenvalue of  $J$ ), while  $a(t)$  is a continuous (here linear) increasing function of time that drives the

system across a bifurcation point occurring approximately when  $a(t) = a_0$ . Beyond that point, the energy landscape governing the system's evolution approximately encodes the *local* minima of the Ising term, driving particles to move toward low-energy solutions. The final (discrete) solutions are then obtained as  $s_i = f(q_i^s)$ , where  $q_i^s$  is the steady state of (D), and  $f$  is taken as a sign function, for simplicity.

To mitigate errors resulting from the relaxation of binary variables to continuous ones, a modified version of the SBM (which was used in this work) was proposed in Ref. [23]. Therein, the nonlinear term,  $q_i^4$ , was replaced with perfectly inelastic walls at  $|q_i| = 1$ , and the Ising contribution to the dynamics is discretized. Specifically, the term  $\sum_{j=1}^N J_{ij} q_j$  in Eq. (D) is substituted with  $\sum_{j=1}^N J_{ij} \text{sign}(q_j)$ . These adjustments ensure that the dynamics can be halted when  $a(t) = a_0$ , with the system

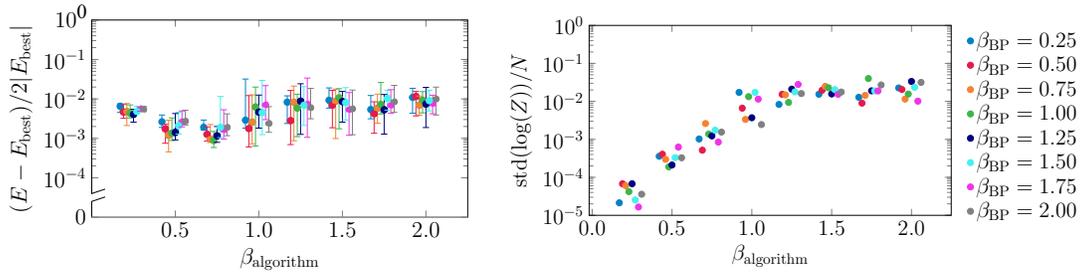


FIG. 19. Influence of inverse temperature on the stability of the results using LBP for dimensional reduction, where we independently change  $\beta_{\text{BP}}$  in LBP and  $\beta_{\text{algorithm}}$  in the TN branch-and-bound algorithm. We consider the same single instance and the same performance metrics as in Figs. 18(c) and 18(d). We use LBP to truncate the number of states in each node from  $2^{24}$  to  $2^{16}$ . For better visibility of the data on the graph, we spaced out the data for different temperatures  $\beta_{\text{BP}}$ .

TABLE II. Physical properties of the D-Wave Advantage\_system6.1 machine and Advantage2\_prototype1.1.

Parameter	Advantage_system6.1	Advantage2_prototype1.1
Qubits	5616	563
Couplers	40135	4790
Qubit temperature (mK)	$16.0 \pm 0.1$	$13.9 \pm 1.0$
$M_{\text{AFM}}$ (pH) <sup>a</sup>	1.554	0.582
$L_q$ (pH) <sup>b</sup>	382.180	142.920
$C_q$ (fF) <sup>c</sup>	118.638	169.388
$I_c$ ( $\mu\text{A}$ ) <sup>d</sup>	1.994	4.083
Average single-qubit thermal width (Ising units)	0.221	0.117
FM problem freezeout (scaled time)	0.073	0.015
Single qubit freezeout (scaled time)	0.616	0.619
$\Phi_{\text{CCJ}}^i$ ( $\Phi_0$ ) <sup>e</sup>	-0.624	-0.686
$\Phi_{\text{CCJ}}^f$ ( $\Phi_0$ ) <sup>f</sup>	-0.723	-0.766
Readout time range ( $\mu\text{s}$ ) <sup>g</sup>	18.0 to 173.0	15.0 to 48.0
Programming time <sup>h</sup> ( $\mu\text{s}$ )	$\sim 14200$	$\sim 5500$
QPU delay time per sample ( $\mu\text{s}$ )	20.5	21.0
Readout error rate <sup>i</sup>	$\leq 0.001$	$\leq 0.001$

<sup>a</sup>Maximum available mutual inductance achievable between pairs of flux-qubit bodies.

<sup>b</sup>Qubit inductance.

<sup>c</sup>Qubit capacitance.

<sup>d</sup>Qubit critical current.

<sup>e</sup>Initial value of the external flux applied to qubit compound Josephson-junction structures at the start of an anneal ( $s = 0$ ).

<sup>f</sup>Final value at the end of an anneal ( $s = 1$ ).

<sup>g</sup>Typical readout times for reading between one qubit and the full QPU.

<sup>h</sup>Typical for problems run on this QPU. Actual problem programming times may vary slightly depending on the nature of the problem.

<sup>i</sup>Error rate when reading the full system.

residing in a local minimum. In general, there is no guarantee that  $f$  maps  $q_i^s$  onto the ground state of (1),  $s_i^g$ . Finding a function  $f$  (if one exists) that ensures this mapping remains an open problem.

The differential Eq. (D) are separable with respect to positions  $q_i$  and momenta  $p_i$ . This allows for numerical integration using the simple yet stable symplectic Euler method, which can also be efficiently implemented on GPUs. Furthermore, the chaotic nature of the SBM equations implies sensitivity to initial conditions. Since each simulation is independent, multiple (random) starting points can be evolved simultaneously, enhancing efficiency and enabling extensive parallelization.

## APPENDIX E: QUANTUM ANNEALING DEVICES

D-Wave reference results for the Pegasus instances were obtained using Advantage\_system6.1, while Zephyr instances were solved on Advantage2\_prototype1.1. The properties of both machines are detailed in Table II.

- [1] F. Barahona, On the computational complexity of Ising spin glass models, *J. Phys. A: Math. Gen.* **15**, 3241 (1982).  
 [2] A. Lucas, Ising formulations of many NP problems, *Front. Phys.* **2**, 5 (2014).

- [3] G. De las Cuevas and T. S. Cubitt, Simple universal models capture all classical spin physics, *Science* **351**, 1180 (2016).  
 [4] S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi, Optimization by simulated annealing, *Science* **220**, 671 (1983).  
 [5] N. Metropolis and S. Ulam, The Monte Carlo method, *J. Amer. Statist. Assoc.* **44**, 335 (1949).  
 [6] W. K. Hastings, Monte Carlo sampling methods using Markov chains and their applications, *Biometrika* **57**, 97 (1970).  
 [7] D. J. Earl and M. W. Deem, Parallel tempering: Theory, applications, and new perspectives, *Phys. Chem. Chem. Phys.* **7**, 3910 (2005).  
 [8] J. Houdayer, A cluster Monte Carlo algorithm for 2-dimensional spin glasses, *Eur. Phys. J. B* **22**, 479 (2001).  
 [9] Z. Zhu, A. J. Ochoa, and H. G. Katzgraber, Efficient cluster algorithm for spin glasses in any space dimension, *Phys. Rev. Lett.* **115**, 077201 (2015).  
 [10] M. Mohseni, D. Eppens, J. Strumpfer, R. Marino, V. Denchev, A. K. Ho, S. V. Isakov, S. Boixo, F. Ricci-Tersenghi, and H. Neven, Nonequilibrium Monte Carlo for unfreezing variables in hard combinatorial optimization, [arXiv:2111.13628](https://arxiv.org/abs/2111.13628).  
 [11] A. Selby, Efficient subgraph-based sampling of Ising-type models with frustration, [arXiv:1409.3934](https://arxiv.org/abs/1409.3934).  
 [12] S. Isakov, I. Zintchenko, T. Rønnow, and M. Troyer, Optimised simulated annealing for Ising spin glasses, *Comput. Phys. Commun.* **192**, 265 (2015).  
 [13] T. Kadowaki and H. Nishimori, Quantum annealing in the transverse Ising model, *Phys. Rev. E* **58**, 5355 (1998).

- [14] G. E. Santoro, R. Martoňák, E. Tosatti, and R. Car, Theory of quantum annealing of an Ising spin glass, *Science* **295**, 2427 (2002).
- [15] A. D. King, J. Raymond, T. Lanting, R. Harris, A. Zucca, F. Altomare, A. J. Berkley, K. Boothby, S. Ejtemaee, and C. Enderud *et al.*, Quantum critical dynamics in a 5,000-qubit programmable spin glass, *Nature* **617**, 61 (2023).
- [16] M. Bernaschi, I. González-Adalid Pemartín, V. Martín-Mayor, and G. Parisi, The quantum transition of the two-dimensional Ising spin glass, *Nature* **631**, 749 (2024).
- [17] H. M. Bauza and D. A. Lidar, Scaling advantage in approximate optimization with quantum annealing, *Phys. Rev. Lett.* **134**, 160601 (2025).
- [18] A. D. King, S. Suzuki, J. Raymond, A. Zucca, T. Lanting, F. Altomare, A. J. Berkley, S. Ejtemaee, E. Hoskinson, and S. Huang *et al.*, Coherent quantum annealing in a programmable 2000-qubit Ising chain, *Nat. Phys.* **18**, 1324 (2022).
- [19] A. D. King, A. Nocera, M. M. Rams, J. Dziarmaga, R. Wiersema, W. Bernoudy, J. Raymond, N. Kaushal, N. Heinsdorf, and R. Harris *et al.*, Beyond-classical computation in quantum simulation, *Science* **388**, 199 (2025).
- [20] Y. Bando, Y. Susa, H. Oshiyama, N. Shibata, M. Ohzeki, F. J. Gómez-Ruiz, D. A. Lidar, S. Suzuki, A. del Campo, and H. Nishimori, Probing the universality of topological defect formation in a quantum annealer: Kibble-Zurek mechanism and beyond, *Phys. Rev. Res.* **2**, 033369 (2020).
- [21] T. Kashimata, M. Yamasaki, R. Hidaka, and K. Tatsumura, Efficient and scalable architecture for multiple-chip implementation of simulated bifurcation machines, *IEEE Access* **12**, 36606 (2024).
- [22] H. Goto, K. Tatsumura, and A. R. Dixon, Combinatorial optimization by simulating adiabatic bifurcations in nonlinear Hamiltonian systems, *Sci. Adv.* **5**, eaav2372 (2019).
- [23] H. Goto, K. Endo, M. Suzuki, Y. Sakai, T. Kanao, Y. Hamakawa, R. Hidaka, M. Yamasaki, and K. Tatsumura, High-performance combinatorial optimization based on classical mechanics, *Sci. Adv.* **7**, eabe7953 (2021).
- [24] F. Verstraete, J. I. Cirac, and V. Murg, Matrix product states, projected entangled pair states, and variational renormalization group methods for quantum spin systems, *Adv. Phys.* **57**, 143 (2008).
- [25] R. Orús, A practical introduction to tensor networks: Matrix product states and projected entangled pair states, *Ann. Phys.* **349**, 117 (2014).
- [26] J. Biamonte and V. Bergholm, Tensor networks in a nutshell, [arXiv:1708.00006](https://arxiv.org/abs/1708.00006).
- [27] S.-J. Ran, E. Tirrito, C. Peng, X. Chen, G. Su, and M. Lewenstein, *Tensor Network Contractions* (Springer, Cham, 2020).
- [28] M. C. Bañuls, Tensor network algorithms: A route map, *Annu. Rev. Condens. Matter Phys.* **14**, 173 (2023).
- [29] T. Nishino, Y. Hieida, K. Okunishi, N. Maeshima, Y. Akutsu, and A. Gendiar, Two-dimensional tensor product variational formulation, *Progr. Theor. Phys.* **105**, 409 (2001).
- [30] N. Schuch, M. M. Wolf, F. Verstraete, and J. I. Cirac, Computational complexity of projected entangled pair states, *Phys. Rev. Lett.* **98**, 140506 (2007).
- [31] M. M. Rams, M. Mohseni, D. Eppens, K. Jałowiecki, and B. Gardas, Approximate optimization, sampling, and spin-glass droplet discovery with tensor networks, *Phys. Rev. E* **104**, 025308 (2021).
- [32] K. Ueda, R. Otani, Y. Nishio, A. Gendiar, and T. Nishino, Snapshot observation for 2D classical lattice models by corner transfer matrix renormalization group, *J. Phys. Soc. Jap.* **74**, 111 (2005).
- [33] M. Frias-Perez, M. Marien, D. P. Garcia, M. C. Banuls, and S. Iblisdir, Collective Monte Carlo updates through tensor network renormalization, *SciPost Phys.* **14**, 123 (2023).
- [34] A. A. Gangat and J. Gray, Hyperoptimized approximate contraction of tensor networks for rugged-energy-landscape spin glasses on periodic square and cubic lattices, *Phys. Rev. E* **110**, 065306 (2024).
- [35] J. Gray and G. K.-L. Chan, Hyperoptimized approximate contraction of tensor networks with arbitrary geometry, *Phys. Rev. X* **14**, 011009 (2024).
- [36] J.-G. Liu, L. Wang, and P. Zhang, Tropical tensor network for ground states of spin glasses, *Phys. Rev. Lett.* **126**, 090506 (2021).
- [37] M. Mezard and A. Montanari, *Information, Physics, and Computation*, Oxford graduate texts (Oxford University Press, Oxford, New York, 2009).
- [38] N. Dattani, S. Szalay, and N. Chancellor, Pegasus: The second connectivity graph for large-scale quantum annealing hardware, [arXiv:1901.07636](https://arxiv.org/abs/1901.07636).
- [39] K. Boothby, P. Bunyk, J. Raymond, and A. Roy, Next-generation topology of D-Wave quantum processors, [arXiv:2003.00133](https://arxiv.org/abs/2003.00133).
- [40] K. Boothby, A. D. King, and J. Raymond, Zephyr topology of D-Wave quantum processors, [https://www.dwavesys.com/media/2uznec4s/14-1056a-a\\_zephyr\\_topology\\_of\\_d-wave\\_quantum\\_processors.pdf](https://www.dwavesys.com/media/2uznec4s/14-1056a-a_zephyr_topology_of_d-wave_quantum_processors.pdf) Tech. Rep. D-Wave (2021).
- [41] F. Verstraete, M. M. Wolf, D. Perez-Garcia, and J. I. Cirac, Criticality, the area law, and the computational power of projected entangled pair states, *Phys. Rev. Lett.* **96**, 220601 (2006).
- [42] M. Lubasch, J. I. Cirac, and M.-C. Bañuls, Unifying projected entangled pair state contractions, *New J. Phys.* **16**, 033014 (2014).
- [43] N. Halko, P.-G. Martinsson, and J. A. Tropp, Finding structure with randomness: Probabilistic algorithms for constructing approximate matrix decompositions, *SIAM Rev.* **53**, 217 (2009).
- [44] P.-G. Martinsson and J. A. Tropp, Randomized numerical linear algebra: Foundations and algorithms, *Acta Numer.* **29**, 403 (2020).
- [45] Our numerical tests were performed on Nvidia GeForce RTX 3090 or 4090 GPUs with 24 GB of memory.
- [46] A. Sinha, M. M. Rams, and J. Dziarmaga, Efficient representation of minimally entangled typical thermal states in two dimensions via projected entangled pair states, *Phys. Rev. B* **109**, 045136 (2024).
- [47] G. H. Golub and C. F. Van Loan, *Matrix Computations* (The Johns Hopkins Press Ltd., London, 1996).
- [48] J. Pearl, in *Proceedings American Association of Artificial Intelligence National Conference on AI* (Pittsburgh, 1982), p. 133.

- [49] J. Yedidia, W. Freeman, and Y. Weiss, in *Exploring Artificial Intelligence in the New Millennium*, edited by G. Lakemeyer and B. Nebel (Morgan Kaufmann Publishers, San Francisco, 2003), Chap. 8, p. 239.
- [50] As some of the qubits and couplers that would form an ideal Pegasus or Zephyr lattice are not available in the quantum annealing hardware, we set the corresponding local fields and couplings to zero so that all solvers use the same instances.
- [51] B. Tasseff, T. Albash, Z. Morrell, M. Vuffray, A. Y. Lokhov, S. Misra, and C. Coffrin, On the emerging potential of quantum annealing hardware for combinatorial optimization, *J. Heuristics* **30**, 325 (2024).
- [52] D. Perera, F. Hamze, J. Raymond, M. Weigel, and H. G. Katzgraber, Computational hardness of spin-glass problems with tile-planted solutions, *Phys. Rev. E* **101**, 023316 (2020).
- [53] D. Perera, I. Akpabio, F. Hamze, S. Mandra, N. Rose, M. Aramon, and H. G. Katzgraber, Chook – A comprehensive suite for generating binary optimization problems with planted solutions, [arXiv:2005.14344](https://arxiv.org/abs/2005.14344).
- [54] J. Vodeb, V. Eržen, T. Hrga, and J. Povh, Accuracy and performance evaluation of quantum, classical and hybrid solvers for the max-cut problem, [arXiv:2412.07460](https://arxiv.org/abs/2412.07460).
- [55] M. Mohseni, M. M. Rams, S. V. Isakov, D. Eppens, S. Pielawa, J. Strumpfer, S. Boixo, and H. Neven, Sampling diverse near-optimal solutions via algorithmic quantum annealing, *Phys. Rev. E* **108**, 065303 (2023).
- [56] A. Zucca, H. Sadeghi, M. Mohseni, and M. H. Amin, Diversity metric for evaluation of quantum annealing, [arXiv:2110.10196](https://arxiv.org/abs/2110.10196).
- [57] I. M. Bomze, M. Budinich, P. M. Pardalos, and M. Pelillo, in *Handbook of Combinatorial Optimization: Supplement Volume A*, edited by D.-Z. Du and P. M. Pardalos (Springer US, Boston, MA, 1999), p. 1.
- [58] We iterate over the list of states in a random order, building, in the process, a set of independent states. At each step of the iteration, a configuration is added to the list of independent states if it is independent of all configurations already in the list. We select the largest set obtained across all the restarts of that procedure. Typically, we observe saturation in the obtained clique sizes after performing of the order of 100 restarts.
- [59] J. Chen, J. Jiang, D. Hangleiter, and N. Schuch, Sign problem in tensor network contraction, *PRX Quantum* **6**, 010312 (2025).
- [60] M. Mohseni *et al.*, How to build a quantum supercomputer: Scaling from hundreds to millions of qubits, [arXiv:2411.10406](https://arxiv.org/abs/2411.10406).
- [61] G. Pedretti, F. Böhm, M. Hizzani, T. Bhattacharya, P. Bruel, J. Moon, S. Serebryakov, D. Strukov, J. Strachan, and J. Ignowski *et al.*, in *2023 International Electron Devices Meeting (IEDM)* (IEEE, San Francisco, 2023), p. 1.
- [62] S. Chowdhury, A. Grimaldi, N. A. Aadit, S. Niazi, M. Mohseni, S. Kanai, H. Ohno, S. Fukami, L. Theogarajan, G. Finocchio, S. Datta, and K. Y. Camsari, A full-stack view of probabilistic computing with p-bits: Devices, architectures, and algorithms, *IEEE J. Explor. Solid-State Comput. Devices Circuits* **9**, 1 (2023).
- [63] T. Śmiechrzalski, A. M. Dziubyna, K. Jałowiecki, Z. Mzaouali, Ł. Pawela, B. Gardas, and M. M. Rams, SpinGlassPEPS.jl: Tensor-network package for Ising-like optimization on quasi-two-dimensional graphs, [arXiv:2502.02317](https://arxiv.org/abs/2502.02317) (2025), <https://github.com/euro-hpc-pl/SpinGlassPEPS.jl>.
- [64] <https://doi.org/10.18150/OUGFIO>.