

Improving success probability in Lechner-Hauke-Zoller parity embedding by computing with quantum walks

Jemma Bennett^{1,*}, Nicholas Chancellor², Viv Kendon³, and Wolfgang Lechner^{1,4}

¹*Institut für Theoretische Physik, Universität Innsbruck, Technikerstraße 21A, 6020 Innsbruck, Austria*

²*School of Computing, Newcastle University, 1 Science Square, Newcastle upon Tyne NE4 5TG, United Kingdom*

³*Department of Physics, University of Strathclyde, Glasgow G4 0NG, United Kingdom*

⁴*Parity Quantum Computing GmbH, Rennweg 1/Top 314, 6020 Innsbruck, Austria*

 (Received 22 April 2025; revised 29 August 2025; accepted 17 September 2025; published 23 October 2025)

Lechner-Hauke-Zoller (LHZ) parity embedding is one of the front-running methods for implementing difficult-to-engineer long-range interactions in quantum optimization problems. Continuous-time quantum walks are a leading approach for solving quantum optimization problems. Because they populate excited states, quantum walks can avoid the exponential gap-closing problems seen in other continuous-time techniques such as quantum annealing and adiabatic quantum computation. An important question, therefore, is how continuous-time quantum walks perform in combination with the LHZ parity embedding. By numerically simulating continuous-time quantum walks on four-, five-, and six-logical-qubit Sherrington-Kirkpatrick Ising spin glass instances embedded onto the LHZ parity embedding, we are able to verify the continued efficacy of heuristics used to estimate the optimal hopping rate and the numerical agreement with the theory behind the location of the lower bound of the LHZ parity constraint strength. In addition, by comparing several different LHZ-based decoding methods, we identified postreadout error correction techniques which improve the success probability of the quantum walk.

DOI: [10.1103/my7q-8tbj](https://doi.org/10.1103/my7q-8tbj)

I. INTRODUCTION

In the quantum optimization setting, we aim to solve useful optimization problems by mapping them to all-to-all connected Ising models. However, beyond a very small number of qubits, in current hardware this level of connectivity does not exist. For example, on the Zephyr topology of the current DWave machines, the largest complete graph that it is possible to directly embed is $n = 4$ qubits [1]. In comparison, IBM machines use the “heavy-hex” hardware graph, which favors low connectivity in order to reduce errors caused by frequency collisions and crosstalk. Here the largest complete graph that it is possible to directly embed is $n = 2$ [2]. In order to overcome the issues caused by this lack of connectivity in hardware, we have to use an embedding. Several methods have been proposed. The approach favored on DWave machines is minor embedding [3–5]. This involves connecting chains of qubits, such that several physical qubits represent a single one in the optimization problem. However, as the problem size scales, larger chain sizes are required. A larger chain means the qubits within it are more susceptible to flipping, and therefore increasing chain strength is required, which will eventually

lead to scaling problems. Another option is to use perturbative gadgets [6] to enact interactions; however, the strength of links between qubits for this technique does not scale favorably. In the Rydberg atom setting, all-to-all connectivity may be also be achieved using “quantum wires” or “crossing gadgets” [7,8].

Alternatively, embeddings which convert long-range couplings into locally connected structures have been proposed. In Ref. [9], configurations of three-qubit triangles within an Ising model are encoded onto pairs of coupled qubits, which are then connected and constrained such that the low-energy eigenspace of the original model is preserved. Another local encoding is the Lechner-Hauke-Zoller (LHZ) parity embedding [10], which is the focus of the research in this paper. Here, the parity of each of the couplings in an Ising model is represented by a single qubit. The strength of the coupling is enacted by a field on the qubit. Three- or four-body constraints are used to suppress states that do not correspond to states in the original Ising model, i.e., physical states that do not lie inside the logical code space. Due to these constraints, this embedding has some protection against error, as long as the strength of the constraints is strong enough that the eigenenergies of the physical states that lie outside the logical code space are well separated from those which are within the code space [11]. However, the high required strength of constraints leads to scalability concerns, as well as the possibility of a single bit-flip error cascading across several qubits.

The redundancy gained from using the LHZ embedding may also be used for error mitigation or correction. Several different decoding methods may be used. In Refs. [12,13],

*Contact author: jemma.bennett@uibk.ac.at

Published by the American Physical Society under the terms of the [Creative Commons Attribution 4.0 International](https://creativecommons.org/licenses/by/4.0/) license. Further distribution of this work must maintain attribution to the author(s) and the published article's title, journal citation, and DOI.

spanning trees were used to perform error mitigation for the quantum approximate optimization algorithm (QAOA) on the LHZ embedding. Spanning trees were previously suggested and utilized for error mitigation in the continuous-time setting in Refs. [10,14]. When spanning trees are used to decode from a physical to logical state, a number of trees are read out to give a set of “readout” states. Then, a choice must be made about how to select a single final logical ground state. Another possibility, instead of selecting a single final logical ground state, is to use the set of readout states as candidate solutions which are stored and then eventually utilized, as was described in Ref. [15]. Also, in certain situations, for example, where just an approximate solution is required, or to distinguish different influences on decoding methods, the mean energy of the readout states could be taken [13].

Where just a single final logical ground state needs to be selected, a choice must be made on how to select it. In most of the previous implementations, the final logical ground state was selected via a majority vote on the readout states given by the spanning trees. This approach is advantageous when the error rate is low and less excited states are populated [as you would expect in adiabatic quantum computation (AQC)], and it is more likely that a majority of correct states will be read out than not. Conversely, if the error rate is high, and more excited states are populated, an approach where you find or approximate the lowest energy state may be more beneficial. In Ref. [13], a method is presented where the spanning tree with the minimum energy is selected as the final logical ground state. For the research in this paper, when decoding with spanning-tree-based methods, we compare the performance of making the final logical ground state selection using the majority vote method versus the lowest-energy chosen method.

In Refs. [14,16–20], the performance of several belief-propagation-type decoders for error correction on the LHZ embedding were investigated. In Ref. [14], minimum weight and maximum likelihood decoders were also investigated. Where the logical state must be preserved, for example, during a continuing computation, a sequence of controlled-not (cnot) gates, or a combination of X measurements and Z rotations, may be used to preserve and act upon parity information [21–23].

Quantum walks (QWs) [24,25] are a form of continuous-time quantum computing (CTQC), which also includes AQC and quantum annealing. In general, all types of CTQC use a Hamiltonian of the form

$$\hat{H}(t) = A(t)\hat{H}_0 + B(t)\hat{H}_P, \quad (1)$$

where $\hat{H}_0$ is an easily implemented initial Hamiltonian and $\hat{H}_P$ is the problem Hamiltonian which has the solution to an optimization problem encoded into its ground state. $\hat{H}_0$ defines the graph of how the states in the Hilbert space are connected. There are multiple choices of $\hat{H}_0$, but in this paper we use

$$\hat{H}_0 = n\mathbb{I} - \sum_{j=0}^{n-1} \hat{\sigma}_j^x, \quad (2)$$

where n is the number of qubits and $\hat{\sigma}_j^x$ is a tensor product given by

$$\hat{\sigma}_j^x = \left(\bigotimes_{r=0}^{j-1} \mathbb{I}_2 \right) \otimes \hat{\sigma}^x \otimes \left(\bigotimes_{j+1}^n \mathbb{I}_2 \right), \quad (3)$$

where $\mathbb{I}_2$ is the identity operator and $\hat{\sigma}^x$ is the Pauli X operator acting on the j th qubit. In AQC and quantum annealing the functions $A(t)$ and $B(t)$ are time dependent and varied from 1 to 0 and from 0 to 1, respectively. By contrast, for quantum walks both $A(t)$ and $B(t)$ are time independent and set to $A(t) = \gamma$ and $B(t) = 1$, where γ is known as the “hopping rate” and determines the rate at which the quantum walk can pass through the graph of states in the Hilbert space. A more comprehensive introduction to optimization using continuous-time quantum walks may be found in Refs. [26,27]. In Ref. [26], it was shown that quantum walks can do better than Grover’s search algorithm when finding the ground states of Sherrington-Kirkpatrick (SK) spin glasses. More recently, multistage quantum walks have been shown to improve the performance of quantum walks and outperform QAOA in some cases [28–31].

In Ref. [32], it was suggested that the nonadiabaticity in quantum walks leads to possible superpositions of correct states, which can avoid the effects of first-order phase transitions caused by the closing of the gap in adiabatic techniques. Results in Ref. [33] suggest that continuous-time quantum walks may have some intrinsic robustness to error due to low-lying excited states populated during the quantum walk. These states have the potential to be used in error mitigation techniques. Furthermore, in Refs. [18–20], a combination of a Monte Carlo method [34] and belief-propagation and approximation of belief-propagation (bit-flipping) decoding algorithms (using syndromes defined in Ref. [16]) were used to find logical ground states of LHZ parity embedded SK Ising spin glass instances. When using the belief-propagation step of their decoding technique, they found that a substantial number of the physical states they were able to decode to the logical ground state lay outside the logical code space. In addition, they found the optimal constraint strength when using the belief-propagation-based decoding method was different than when it was not used. These results may indicate that different decoding methods may be better suited to different error conditions.

In this paper we performed numerical simulations of continuous-time quantum walks on four-, five-, and six-logical-qubit SK spin glass instances (first generated for research in Ref. [26] and can be found in Ref. [35]) embedded onto the LHZ parity embedding. Using our simulations, we were able to evaluate the efficacy of the heuristic method (developed in Ref. [28]) for estimating the optimal hopping rate of the quantum walk in the LHZ setting. Additionally, we numerically tested the theory on the lower bound of constraint strength in the LHZ parity embedding (developed in Ref. [11]) in the quantum walk setting. Utilizing the redundancy present in the LHZ parity embedding, we compared the performance on the success probability of several preexisting postreadout error correcting decoding methods: entire state decoding, random overlapping spanning trees, nonoverlapping spanning

trees, minimum weight decoding, and belief propagation. In addition, following the research in Ref. [13], we utilized the idea of decoding using lowest energy rather than majority votes, for decoding methods which give multiple choices of logical state. We then applied this innovation to the spanning tree decoding methods and compared the performance. As a sanity check, we compared the performance of all methods to the ideal but experimentally unrealistic directly embedded SK Ising spin glass models.

The paper is organized as follows. In Sec. II we introduce our SK Ising spin glass optimization problem setting and how to solve it using a continuous-time quantum walk. In Sec. II A we introduce and describe the calculation of the heuristic hopping rate γ_{heur} . In Sec. III we introduce and describe the LHZ parity embedding and how it is applied in our problem setting. In Sec. III A we analyze the effect of the constraint strength from the LHZ embedding on the performance of the quantum walk. In Sec. IV we introduce and describe the different decoding methods that we investigated in this research. In Sec. V we analyzed the performance of the different decoding methods on four different instances of five-qubit SK Ising spin glasses. In Sec. VI we show how the performance of the different decoding methods varies with problem size. Finally, in Sec. VII we make conclusions and suggestions of future directions.

II. ISING MODEL

The solution to an optimization problem may be encoded into the ground state of an Ising model. In order to simulate finding the solution to difficult optimization problems, we use a dataset of SK Ising spin glass models. Finding the ground state of these models has been shown to be NP-hard [36] and uniformly hard [37]. Using the original (direct) embedding, these models have a Hamiltonian of the form

$$\hat{H}_P = -\frac{1}{2} \sum_{j \neq k=0}^{n-1} J_{jk} \hat{\sigma}_j^z \hat{\sigma}_k^z - \sum_{j=0}^{n-1} h_j \hat{\sigma}_j^z, \quad (4)$$

where J_{jk} is the coupling strength between qubits i and j , h_j is the field strength acting on the j th qubit, and $\hat{\sigma}_j^z$ is [similarly to Eq. (3)] a tensor product of identity operators and the Pauli-Z operator acting on the j th qubit. In order for the quantum walk dynamics to occur, Eq. (4) is then combined with Eq. (1) to become

$$\hat{H} = \gamma \hat{H}_0 + \hat{H}_P, \quad (5)$$

which is then evolved according to the time-dependent Schrödinger equation.

Figure 1 shows a directly embedded five-qubit all-to-all connected SK Ising spin glass. The black lines represent the couplings $\hat{\sigma}_j^z \hat{\sigma}_k^z$ between the qubits (in orange) and the qubits are labeled from 0 to 4. The numerical simulations in this paper utilized a dataset of SK Ising spin glass instances which were first generated for research reported in Ref. [26] and can be found in a data archive [35].

As a quantum walk is nonadiabatic, we expect in general that the probability of finding the correct ground state of $\hat{H}_P$ (the success probability) will be less than one. Therefore, we need to take repeats to improve the success probability. As

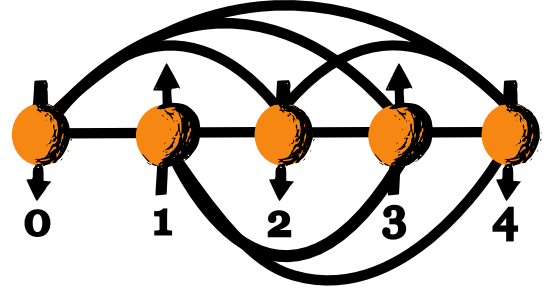


FIG. 1. Diagram of a five-qubit SK Ising spin glass.

the success probability varies over the course of the quantum walk, we calculate it at multiple times and take the average, which we call the average success probability $\bar{P}$. If we calculate the probability over the interval $[t, t + \Delta t]$, we can define the finite-time success probability as

$$\bar{P}(t, \Delta t) \equiv \frac{1}{\Delta t} \int_t^{t+\Delta t} dt_f P(t_f), \quad (6)$$

where t_f is the final time at which we calculate the probability.

An important variable in quantum walks is the hopping rate γ . This acts as a balancing term between the initial Hamiltonian $\hat{H}_0$ and the problem Hamiltonian $\hat{H}_P$ and must be optimized in order to maximize the success probability. Figure 2 shows $\bar{P}(t, \Delta t)$ where $t = 30$ and $\Delta t = 70$ versus γ for a single directly embedded five-qubit SK Ising spin glass instance plotted in dark blue. To embed the five-qubit SK Ising spin glass according to the LHZ parity embedding, 15 qubits are required (see Fig. 4). This is an equivalent number of qubits to three copies of the directly embedded five-qubit SK Ising spin glass. Therefore, we also plot $\bar{P}(t, \Delta t)$ versus γ for the three copies of the five-qubit SK Ising spin glass

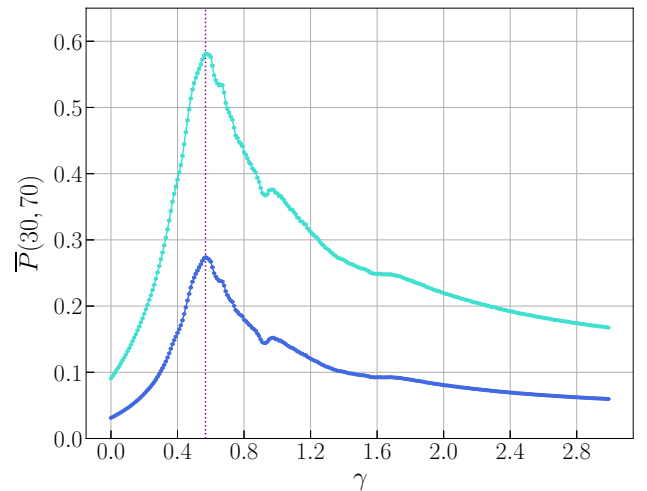


FIG. 2. Graph showing the average success probability $\bar{P}(t, \Delta t)$ where $t = 30$ and $\Delta t = 70$ vs the hopping rate γ for one copy of a five-qubit SK Ising spin glass instance with unique identifier (uid) “aagtzdpcgktnrzhnpvxqtvqql” in dark blue and for three separate copies of the same instance in light blue. Error bars for $\bar{P}(t, \Delta t)$ are plotted but are too small to be seen. The vertical dotted purple line indicates the location of the optimal hopping rate γ_{opt} where $\bar{P}(t, \Delta t)$ is maximized.

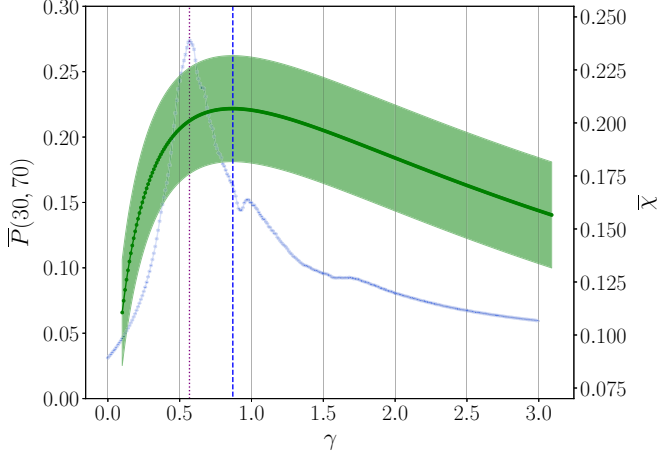


FIG. 3. Graph showing the average dynamic coefficient $\bar{\chi}$ (on the right-hand y axis) vs the hopping rate γ for a single directly embedded five-qubit SK Ising spin glass instance with uid “aagtzd-pchgtknrzihrnpvxqtvqiql” in green. Light green shading indicates the error in $\bar{\chi}$ calculated using Eq. (9). The vertical dashed blue line indicates the location of the heuristic hopping rate γ_{heur} where $\bar{\chi}$ is maximized. For comparison, $\bar{P}(30, 70)$ (on the left-hand y axis) vs γ is also plotted in faded blue, as well as a vertical dotted purple line which indicates the location of γ_{opt} .

instance in light blue. The optimal hopping rate γ_{opt} is defined as the value of γ where $\bar{P}(30, 70)$ is maximized. It is indicated in Fig. 2 by a vertical dotted purple line.

Heuristic hopping rate

It can be seen from Fig. 2 that $\bar{P}(t, \Delta t)$ is highly dependent on the hopping rate γ . For the SK Ising spin glass problem, the exact value of γ_{opt} cannot be calculated efficiently. However, in Ref. [28] a heuristic method for estimating the optimal hopping rate γ_{heur} was developed. There, they introduced the “dynamic coefficient” χ^{jk} , which is a measure of the dynamics experienced by a system. It can be defined as

$$\chi^{(jk)} = \frac{\zeta_{jk}/\Gamma(t)}{[1 + \zeta_{jk}/\Gamma(t)]^2}, \quad (7)$$

where $\zeta_{jk} = \frac{|\Delta_{jk}|}{2|\langle k|\hat{H}_0|j\rangle|}$ is a single scaled gap parameter, with $|\Delta_{jk}|$ the energy difference between states j and k . As $\hat{H}_0$ is not biased then $|\langle k|\hat{H}_0|j\rangle| = 1$ when j, k differ by one bit flip. If we average across all of the pairs of states j, k in a system we get the average dynamic coefficient,

$$\bar{\chi} = \langle \chi^{(jk)} \rangle_{jk}. \quad (8)$$

Due to the exponentially growing number of states, it is not efficient to calculate Eq. (8). However, we are able to approximate $\bar{\chi}$ by taking a sample of the available states j, k . The error in this approximation is

$$\delta\bar{\chi} \sim \frac{0.25}{N_{\text{samples}}^{1/2}}, \quad (9)$$

where N_{samples} is the number of samples.

We can estimate a heuristic value for the optimal hopping rate γ_{heur} by finding γ which maximizes $\bar{\chi}$. Figure 3 shows

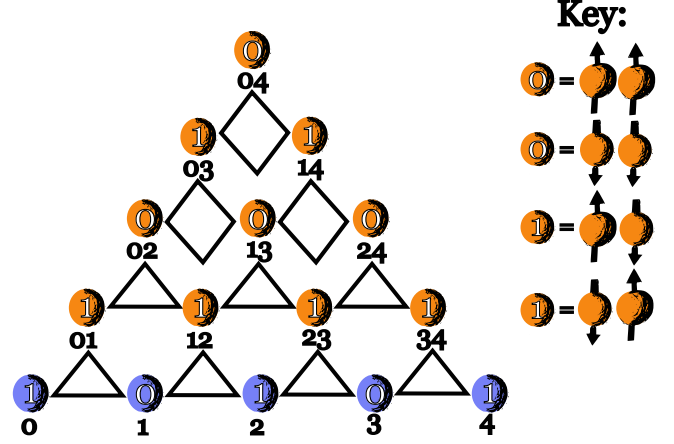


FIG. 4. Left: Diagram of a five-logical-qubit Sherrington-Kirkpatrick Ising spin glass embedded onto the LHZ embedding. The numbers below each of the orange qubits refer to which coupling in the original they represent. The numbers below the blue qubits represent the qubit on which each of the fields acts. Squares (triangles) represent four-body (three-body) constraints. Right: Key showing the parity conversion of couplings to LHZ qubits.

$\bar{\chi}$ (right-hand y axis) of a single directly embedded five-qubit SK Ising spin glass instance for values of γ between 0 and 3.2, plotted in dark green. Light green shading indicates the error in $\bar{\chi}$ calculated using Eq. (9), where $N_{\text{samples}} = 10^3$. A vertical dashed blue line indicates the value of γ at which $\bar{\chi}$ is maximized. For comparison we have also plotted $\bar{P}(t, \Delta t)$, where $t = 30$ and $\Delta t = 70$, versus γ for the same five-qubit directly embedded instance (same data as Fig. 2). A vertical dotted purple line indicates the location of γ_{opt} .

From Fig. 3, we can see that the value of γ_{heur} does not directly coincide with γ_{opt} . However, due to the width of the peak in $\bar{P}(t, \Delta t)$, for this five-qubit SK Ising spin glass instance, when using γ_{heur} as the hopping rate of the quantum walk, there will still be some improvement in the average success probability $\bar{P}$.

III. LHZ PARITY EMBEDDING

LHZ parity embedding was developed as a way to represent an all-to-all connected Ising model with n “logical” qubits on a locally connected model with K “physical” qubits. We refer to the original n -qubit Ising model as the *logical model* in its *directly embedded* configuration. As a demonstrative example, we first consider the n -qubit all-to-all connected SK spin glass Ising model we described in Sec. II. A five-qubit version of this model is depicted in Fig. 1. Figure 4 shows an LHZ embedding of the five-logical-qubit SK Ising spin glass model from Fig. 1.

In order to perform LHZ parity embedding on an n -logical-qubit Ising model, we start by mapping each coupling between qubits i and j ($J_{ij}\hat{\sigma}_i^z\hat{\sigma}_j^z$) in the logical model onto a local field acting on a single qubit $J_{ij}\tilde{\sigma}_{(ij)}^z$ in the LHZ model. For it to be a faithful representation of the logical Ising model, we want the value of each of the qubits $\{i, j\}$ in the LHZ model to take the value of the parity between the two qubits i and j in the logical model. Hence, we call these qubits *parity* qubits. A

diagrammatic key of this desired mapping can be seen in the right-hand side of Fig. 4. In the version of the LHZ embedding used in this research, we add an additional row of qubits. Each of these qubits represents the mapping of a local field on a single qubit $h_i \tilde{\sigma}_i^z$ in the logical model to a local field on a single qubit $h_i \tilde{\sigma}_i^z$ in the LHZ model. We refer to these qubits as *data* qubits. The number of physical qubits (parity plus data qubits) in the LHZ model is $K = \frac{n(n+1)}{2}$.

The left-hand side of Fig. 4 shows the triangular layout of the physical qubits in an LHZ model which represents the logical model of the five-qubit SK Ising spin glass shown in Fig. 1. The upper part of the triangle shows, in orange, the parity qubits which have been mapped from the couplings between qubits in the logical Ising model to local fields acting on single qubits in the LHZ model. The label of the logical coupling $\{ij\}$ each parity qubit represents is written below it in black. On the bottom row of the triangle, in blue, the data qubits which represent the mapping from a local field acting on a qubit in the logical Ising model to a local field acting on a qubit in the LHZ model are drawn. The label of the logical qubit i which each data qubit represents is written below it in black. The values that the physical qubits in the LHZ model should take in order to faithfully represent the values of the logical qubits in the five-qubit SK Ising spin glass in Fig. 1 are written on the qubits in white.

Due to the increased number of physical qubits increasing the size of the Hilbert space in the LHZ model and the lack of couplings between qubits meaning the physical qubits tend to follow their local field, it is possible for the LHZ model to end up in states which do not correspond to valid states in the logical Ising model; i.e., they lie outside the *logical code space*. We refer to these states as *invalid* states.

The left of Fig. 5 shows two examples of four parity qubits from an LHZ embedding model. These qubits are labeled 02, 03, 12, and 13. The parity qubits on the top left have the values 0, 1, 1, and 0 and those on the bottom left have values 0, 1, 1, and 1. Both of these states are possible within the Hilbert space of the LHZ model. The right-hand side of Fig. 5 shows the qubits and couplings in the corresponding logical models. We see in the top example all the values of the parity qubits may be satisfied on the couplings they correspond to in the logical model (satisfied couplings shown in green). On the other hand, in the bottom example, only the values of the parity qubits 02, 03, and 12 can be satisfied. The value of the parity qubit 13 may not be satisfied in the logical model (unsatisfied coupling shown in red). This indicates that the state 0, 1, 1, 1 in the LHZ model is invalid; i.e., it lies outside the logical code space.

In order to avoid these invalid states and ensure that the low-energy spectrum of the LHZ model matches that of the logical model, three- and four-body constraint terms of the form $C(\tilde{\sigma}_{(i,1)}^z \tilde{\sigma}_{(i,2)}^z \tilde{\sigma}_{(i,3)}^z (\tilde{\sigma}_{(i,4)}^z))$ are added to the LHZ Hamiltonian. These constraint terms add an energy penalty to configurations of three or four parity qubits which contain an odd number of qubits with the value 1 (and therefore correspond to an invalid (unsatisfied) coupling in the logical model, as seen in Fig. 5). The parity qubits are laid out in such a way that each of them participates in at least one constraint and they are only locally connected. Figure 4 shows a configuration of physical qubits which fulfill these criteria. The three- and four-body constraint terms are represented as

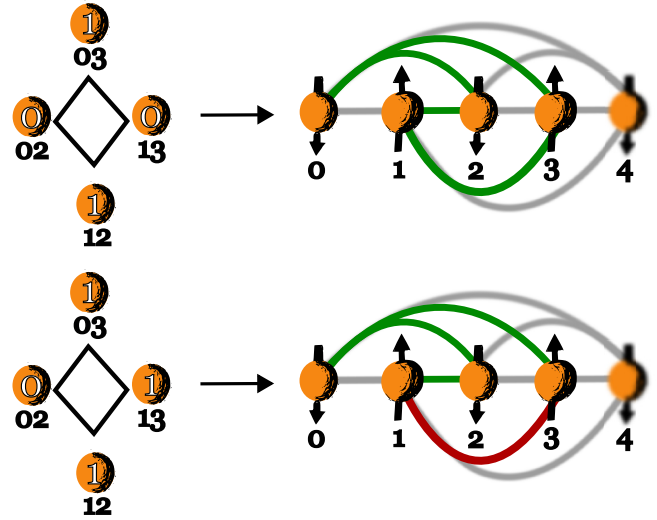


FIG. 5. The left hand side shows two examples of four LHZ embedded parity qubits with the labels 02, 03, 12, and 13 written in black underneath each qubit. The corresponding qubit values (written in white) are 0, 1, 1, and 0 in the top example and 0, 1, 1, and 1 in the bottom example. The right-hand side shows the couplings and qubits of the five-qubit SK Ising spin glass logical model which both examples correspond to. The labels of qubits 0, 1, 2, 3, and 4 are written in black under each qubit and their value is indicated by the direction of the arrow. The logical couplings and qubits which do not participate in the LHZ embedding section on the left have been blurred out. Couplings in the logical model which correspond to physical qubits in the LHZ embedding section are shown in green if the value of the parity qubit is satisfied and in red if it is not.

the black outlined triangles and squares. Each of the three- and/or four-body constraint terms may be referred to as a *plaque*.

Bringing all these elements together, the Hamiltonian of the LHZ embedding can be written as

$$\hat{H}_P^{\text{LHZ}} = \sum_{i=0}^{n-1} h_i \tilde{\sigma}_i^z + \sum_{k=0}^{K-(n-1)} J_k \tilde{\sigma}_k^z + \sum_{l=0}^{K-(n-1)} C(\tilde{\sigma}_{(l,1)}^z \tilde{\sigma}_{(l,2)}^z \tilde{\sigma}_{(l,3)}^z (\tilde{\sigma}_{(l,4)}^z)), \quad (10)$$

where i sums over all of the data qubits, k sums over all of the parity qubits, and l sums over all of the three- and four-body constraints.

Constraint strength

If we wish to fully suppress the invalid states using the constraint terms, we must ensure that the lower energy levels of the physical LHZ embedding Hamiltonian match those of the logical problem Hamiltonian. There is a minimal value of constraint strength C which can be set in LHZ Hamiltonian (10) such that these states are fully suppressed while minimizing the strength of the energy penalty (and energy scale of the system). A method to determine this minimal value was described in Ref. [11].

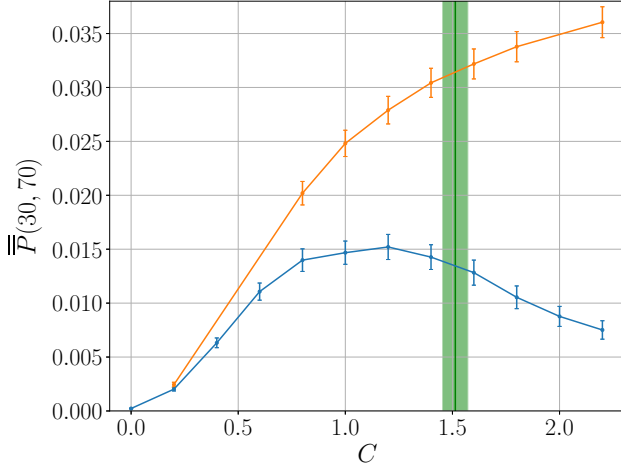


FIG. 6. Graph showing $\bar{P}(30, 70)$ at γ_{heur} (blue) and at γ_{opt} (orange) averaged over 100 instances of five-logical-qubit SK Ising spin glasses embedded onto the LHZ parity embedding versus the constraint strength C , which was varied from 0.0 to 2.2. The instances were decoded using the entire state decoding method (see Sec. IV). Error bars on each data point indicate the uncertainty in $\bar{P}(30, 70)$. Also plotted (vertical green line) is the lower bound of C averaged across the instances calculated using the method in Ref. [11]. Green shading indicates the error in this value.

In order to gain an understanding of how the constraint strength should be set for our problem set in the quantum walk setting, we next calculated the average finite-time success probability $\bar{P}(30, 70)$ of the quantum walk over a set of 100, 5-logical-qubit (15-physical-qubit) LHZ embedded SK Ising spin glasses versus LHZ constraint strength C . All instances were decoded using the entire state decoding method (see Sec. IV for more information).

Figure 6 shows the average finite-time success probability of the quantum walk averaged over 100 instances $\bar{P}(30, 70)$ of five-logical-qubit SK Ising spin glasses embedded onto the LHZ parity embedding versus the constraint strength C , which was varied from 0.0 to 2.2. Plotted in blue, $\bar{P}(30, 70)$ was calculated using the heuristic estimate of optimal hopping strength, γ_{heur} (calculated using the method described in Sec. II A from Ref. [28]). For comparison, plotted in orange, $\bar{P}(30, 70)$ was calculated using the optimal value of hopping strength, γ_{opt} . Also plotted (vertical green line) is the average value of the lower bound of C (calculated using the method in Ref. [11]) across the same 100 instances. The green shading indicates the standard error.

Due to the findings in Ref. [11], we might initially expect the value of constraint strength which provides the highest average success probability for the quantum walk $\bar{P}(30, 70)$ to fall above the average value of the lower bound of the constraint strength C . Indeed, looking at Fig. 6, for $\bar{P}(30, 70)$ calculated using γ_{opt} , this is true. However, we see that for $\bar{P}(30, 70)$ calculated using γ_{heur} , the optimal constraint strength for the quantum walk falls below the average lower bound.

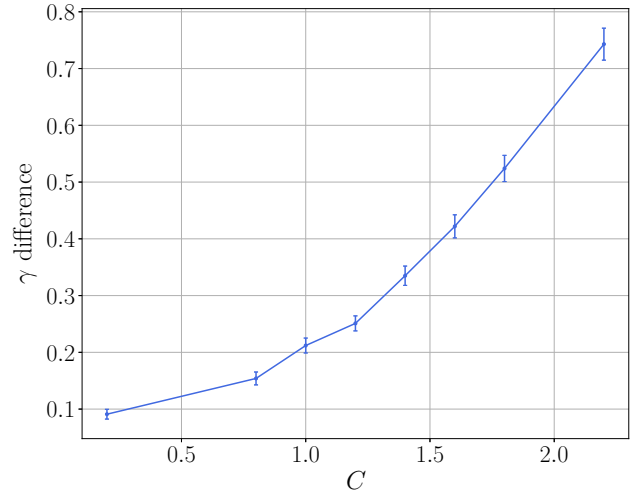


FIG. 7. Graph showing the average difference between γ_{opt} and γ_{heur} (γ difference) vs the constraint strength for 100 instances of five-logical-qubit LHZ embedded SK Ising spin glasses. Error bars on each point indicate the uncertainty in the γ difference.

Considering this counterintuitive finding, we note that the success probability of the quantum walk is also dependent on the proximity of the heuristic hopping rate γ_{heur} to the optimal hopping rate γ_{opt} (γ difference). In order to verify whether this was the cause of the discrepancy seen here, we calculated the average difference between γ_{heur} and γ_{opt} across the same 100 instances of five-logical-qubit LHZ embedded SK Ising spin glasses, for values of constraint strength between 0.2 and 2.2. Results are shown in Fig. 7.

Looking at Fig. 7, we see that in agreement with our expectations the γ difference increases with increasing constraint strength. In addition, we see that the difference increases with an accelerating rate. We argue that this trend is the cause of the peak $\bar{P}(30, 70)$ below the calculated average lower bound of the constraint strength, when it is calculated using γ_{heur} . This is further backed up by the fact that we do not see this peak when $\bar{P}(30, 70)$ is calculated using γ_{opt} .

Despite our finding (for five logical qubits) that when calculating $\bar{P}(30, 70)$ using γ_{heur} , the optimal value of C is on average below the lower bound; for the research in this paper we set a LHZ constraint strength of $C = 2$. The reasoning behind this was to try and avoid the potential effects of invalid LHZ states interacting with the quantum walk.

IV. DECODING

In this research we performed numerical simulations of continuous-time quantum walks (Sec. II) in order to find the ground states of n -logical-qubit SK Ising spin glass problem instances which had been embedded onto the LHZ parity embedding (Sec. III) with K physical qubits.

Once the physical ground state of the LHZ embedded model has been computed using a quantum walk, a decoding step then has to be taken, in order to recover a logical state of the SK Ising spin glass. If the quantum computation has proceeded without error, this state should correspond to the ground state of the logical SK Ising spin glass problem. If not, a carefully designed decoding step may be able to recover the

correct logical ground state. There are several options of how to decode the LHZ embedded physical state to a logical one. In this section, we discuss several of the possible options.

A. Entire state decoding

One option is to compare the entire physical ground state that has been computed with the physical ground state that corresponds exactly to the ground state of the directly embedded logical SK Ising spin glass. If the entire physical ground state does not correspond to the logical ground state, then we discard it as incorrect. This is most intuitive option as it is simply the reverse of embedding the logical problem onto the LHZ parity embedding. However, this method is sensitive to errors, as no constraints may be violated. Even just one bit flip can cause the state to be incorrect. This approach is problematic when a technique such as a quantum walk is used for computation. Unlike in AQC, quantum walks do not completely suppress the probability of the excited states of a system. It is therefore likely that a final “readout” state may contain some bit flips compared to the entirely correct state. This decoding method also does not make use of the redundancy available in the LHZ embedding.

B. Random overlapping spanning trees

Another option is to consider the state of one or more parts of the physical ground state which correspond to “spanning trees” in the logical SK Ising spin glass graph. A spanning tree is a subset of a graph where each node is connected to at least one other node and there are no cycles. The values of the physical qubits in the LHZ system which make up a spanning tree may be used to define a possible ground state of the logical SK Ising spin glass (up to a global bit or spin flip). Unlike entire state decoding, this decoding method allows us to still define a possible ground state even when constraints in the LHZ system have been violated, as is likely using a quantum walk. Also we are able to choose multiple spanning trees per LHZ triangle, thereby utilizing the redundancy of the embedding.

When decoding using spanning trees, there are multiple options of how to choose the trees. One option is to choose the trees randomly and allow them to overlap. Overlapping means that a physical qubit participates in more than one spanning tree. The top right of Fig. 8 shows three possibilities of overlapping spanning tree, in green, purple, and blue, on the directly embedded five-qubit SK Ising spin glass. Each of the couplings in each spanning tree on the directly embedded model corresponds to qubits (circled in the same color) in the LHZ embedding, shown on the left-hand side of the figure. Each of the blue data qubits is also circled in red, indicating their use as an extra “choice” of the possible ground state of the original directly embedded model, which is included in our implementation of this decoding method.

We see in Fig. 8 that qubits 01, 02, and 12 have two rings of more than one color surrounding them. This indicates that they participate in two overlapping spanning trees. If the spanning trees are chosen randomly, overlapping qubits become more likely as the number of spanning trees increases and are inevitable after the number of spanning trees exceeds

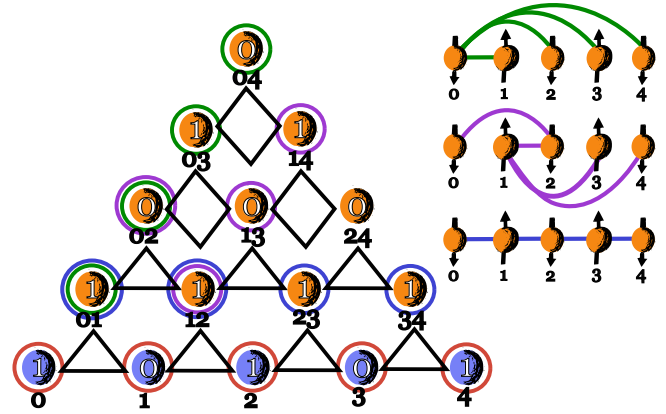


FIG. 8. Left: Diagram of the LHZ parity embedded five-logical-qubit SK Ising spin glass, with three possible choices of overlapping spanning trees. The qubits of each of the three trees are circled in green, purple, and blue. Also, the data qubits are circled in red to highlight their use as one “choice.” Right: Diagram highlighting the links of the respective green, purple, and blue spanning trees as they would appear on the original directly embedded SK Ising spin glass.

$K/(n - 1)$, where K is the number of physical qubits and n is the number of logical qubits. If the spanning trees are allowed to be overlapping, then the number of possible spanning trees is limited to Cayley’s number ($n^{(n-2)}$).

For most of the research carried out in this paper, the decoding method we call “random overlapping spanning trees” consists of selecting two random spanning trees (which are allowed to overlap) and then also measuring the values of the data qubits as a third choice. We denote this as “2 + 1” random overlapping spanning trees. Where we investigated the effect of an increasing number of randomly selected spanning trees which were allowed to overlap (see Appendix C), we continued to carry out a single readout of the data qubits which counted once toward the total number of trees.

Once the states of the spanning trees have been read out from the overall physical readout state, there must then be a selection process such that there is only a single state given as the computed logical ground state. In previous research [10,12,14], a majority vote on the spanning tree states was performed in order to determine which state was the computed logical ground state. In Ref. [12], the spanning tree states were also used to inform future steps of a QAOA approach. In Ref. [13], the performance of selecting the computed logical ground state from the spanning trees readout states using their mean or lowest energy was also investigated. In this paper we compare the performance of selecting the computed logical ground state using “majority vote” to “lowest energy.”

We also note the classical cost of this decoding method. Each time a further spanning tree is specified from the overall readout state a classical cost is levied. It is therefore important for the sake of fair testing that this classical cost is properly accounted for in comparisons. In Appendix A we provide analysis of the classical cost of measuring an increasing number of spanning trees. Then in Appendix B we define a semianalytical method of counting the total number of possible states introduced by measuring all the spanning trees. In Appendix C, we investigated the effect on the

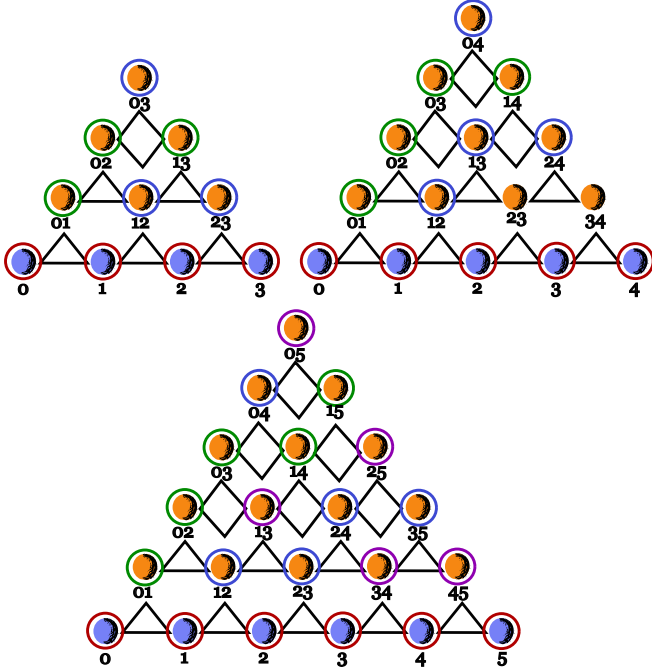


FIG. 9. Diagram of possible choices of readout qubits (in red, green, blue, and purple) for nonoverlapping spanning trees for four-, five-, and six-logical-qubit SK Ising spin glass problems embedded onto LHZ triangles.

quantum walk success probability of decoding using an increasing number of spanning trees and compared this to the success probability when decoding a randomly chosen state using the same number of spanning trees, for 100 instances of four- and five-logical-qubit SK Ising spin glasses.

C. Nonoverlapping spanning trees

Another issue with using random overlapping spanning trees to decode is that with increasing number of spanning trees the chances of overlapping qubits increases, thereby reducing the efficiency of the method. It is also not clear how to limit the number of spanning trees specified per readout state as the problem size increases. (The total number of spanning trees grows with $n^{(n-2)}$ so it is not feasible to specify them all at each size).

One way to avoid these issues is to use nonoverlapping spanning trees. Here, instead of choosing the spanning trees randomly, we choose the spanning trees methodically such that there are no “overlapping” qubits, avoiding the inefficiency. For a LHZ embedding with K physical qubits, the number of possible nonoverlapping spanning trees is $\lfloor K/(n-1) \rfloor$, where n is the number of logical qubits. This limits the number of possible spanning trees to a more feasible value.

Figure 9 shows two (four and five logical qubits) and three (six logical qubits) possible choices of nonoverlapping spanning trees chosen methodically. Again, we represent the spanning trees by surrounding each of the physical qubits (in the LHZ embedding) with circles of a specific color (in this case, blue, green, and purple). As for the random overlapping spanning trees, we have encircled the blue data qubits of each

of the LHZ embedded examples with red rings. This indicates that we again used the values of the data qubits as a “choice” of the possible ground state of the model.

To compare the performance of the nonoverlapping spanning trees fairly with the random overlapping spanning trees, we also fixed our method to reading out two spanning trees and the values of the data qubits as a third choice. We denote this as “2 + 1” nonoverlapping spanning trees. We note that there are multiple ways to choose nonoverlapping spanning trees, so the choices shown in Fig. 9 are not unique.

In order to systematically choose the nonoverlapping spanning trees used in this research, we developed the following method in order to define which spanning trees were used. For this method, we consider the LHZ triangle without the data qubits. First we find the total number of nonoverlapping spanning trees to be chosen: $s = \lfloor K/(n-1) \rfloor$. We then look at the “logical lines” of the LHZ triangle. These are the “lines” within the LHZ triangle which contain the same index [21,38]. Next, starting with logical line zero, we select qubits where their secondary index (i.e., the 1 in 01) does not yet appear in any of the secondary indices of the previously selected qubits until, in this logical line, we are left with a number of qubits equal to the remaining number of spanning trees to be chosen (after this one). We then move to the next logical line along and repeat this process until all possible secondary indices have appeared. If the set of qubits we have now selected corresponds to a spanning tree, we go back to the start of the method and start defining the next spanning tree. Conversely, if the set of qubits we have now selected does not correspond to a spanning tree, ensuring that we do not overfill a logical line, we select the qubit(s) with the lowest index that will complete the spanning tree. We then go back to the start of this method and continue to define spanning trees until we have defined the total number of nonoverlapping spanning trees to be chosen, s . In the research described here, we used two of the nonoverlapping spanning trees defined in this way at each problem size for our 2 + 1 nonoverlapping spanning trees decoding method.

Like for the random overlapping spanning trees, once the spanning trees had been specified from the overall readout state, there was a choice of how to select the computed logical ground state. Just as for the random overlapping spanning trees, we compared the performance of “majority vote” and “lowest energy” selection methods.

D. Minimum weight decoder

Another possible approach to decoding the ground state of the LHZ embedded SK Ising spin glasses is minimum weight decoding. Here, the physical state is read out and then a correction suggested which minimizes the number of bit flips in order to return a physical state outside the logical code space to the logical code space.

We can write this approach as its own optimization problem. If the number of plaquettes (three- and four-body terms) in the LHZ embedded model is given by $P = \frac{(N-1)(N-2)}{2} + (N-1)$, then we can loop through each of the possible $2^P - 1$ incorrect syndromes s^c (i.e., when at least one of the plaquettes $\sigma_{(l,1)}^z \tilde{\sigma}_{(l,2)}^z \tilde{\sigma}_{(l,3)}^z (\tilde{\sigma}_{(l,4)}^z) = -1$) in the set S , finding the

minimum number of bit flips in order to get back to being correct for each syndrome.

To set up the Hamiltonian H_{MW} which has the minimum bit-flip correction to the physical state, we first set all the local fields to one. This way all qubits prioritize being in the zero state. Then we connect these qubits with the same four-body terms as in the original problem, but with each of the terms' constraint strength set to its corresponding value in the current syndrome s^c multiplied by a constant λ which ensures that the plaquette values will all be $+1$ in its ground state. We can then write this Hamiltonian as

$$H_{MW} = - \sum_{j=0}^{N-1} \tilde{\sigma}_j^z - \sum_{v=0}^{P-1} s_v^c \lambda (\tilde{\sigma}_{(v,1)}^z \tilde{\sigma}_{(v,2)}^z \tilde{\sigma}_{(v,3)}^z \tilde{\sigma}_{(v,4)}^z). \quad (11)$$

We first note that the solution to H_{MW} is likely to be degenerate; i.e., a single syndrome has multiple correction options. Therefore, in this research we pick one random correction for each syndrome. Looking at the performance of the minimum weight decoder with multiple solutions could be an interesting avenue for future research.

In Ref. [14], Eq. (11) was shown to be equivalent to MAX-2-SAT and therefore NP-hard, which means its computational cost will grow exponentially with problem size. On the other hand, since this method is problem instance independent, it is perhaps viable for problems where the same size of LHZ parity embedding will be used multiple times.

E. Belief-propagation decoder

The final decoding technique researched here was the belief-propagation decoder, which was first described and researched (for the purposes of LHZ decoding) in Ref. [16]. Here, instead of using the plaquette constraints, separate three-parity-qubit constraints are defined which themselves imply the values of three logical qubits. This is repeated such that the value of each logical qubit is implied by three of these constraints.

In Ref. [16] it was found that this form of decoding had good results under independent and identically distributed noise. However, it was consequently shown in Ref. [14], using a simulated annealing quantum Monte Carlo method, that this was not a realistic noise model for quantum annealing. Nevertheless, this decoding technique has had success when used in decoding LHZ for gate-based applications [17]. More recently, investigations using the belief decoder and an adjusted form of it with a rejection-free Markov chain Monte Carlo method for simulated annealing have shown that it can be effective especially under nonoptimal values of the LHZ constraint strengths [18–20].

V. PERFORMANCE COMPARISON OF DECODING TECHNIQUES

We next investigated, using numerical simulations, the performance of ground-state computation using a continuous-time quantum walk on four different instances of LHZ parity embedded five-logical-qubit SK Ising spin glasses, decoded using six different methods (described in Sec. IV), compared to their directly embedded counterparts. The four instances of five-logical qubit SK Ising spin glass were first generated for

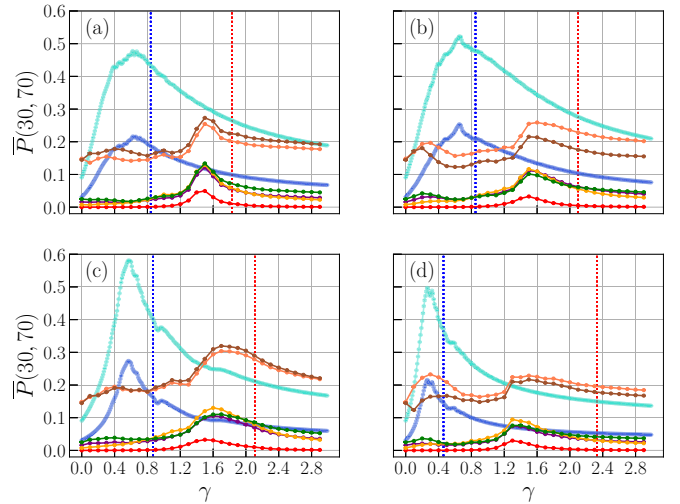


FIG. 10. Graph showing the success probability $\bar{P}(30, 70)$ vs γ for four different instances of LHZ embedded five-qubit SK Ising spin glasses decoded (from physical to logical state) in six ways: entire state decoding (red), minimum weight decoding (green), belief propagation decoding (purple), 2 + 1 nonoverlapping spanning trees with majority vote (gold), 2 + 1 nonoverlapping spanning trees with lowest energy chosen (salmon), and 2 + 1 random (overlapping) spanning trees with lowest energy chosen (brown). For comparison, $\bar{P}(30, 70)$ vs γ for the same instances directly embedded are plotted: single copy (dark blue), three copies (light blue). The γ_{heur} for the directly embedded instances (dotted vertical blue line) and LHZ embedded instances (dotted vertical red line) are also plotted. These were calculated using the heuristic method described in Sec. II A.

research in Ref. [26] and can be found in a data archive [35], where they may be identified using their uids: “aaavmaiqliol-nplcovmzxjzakyvyayz,” “aacrcjsbugeteaageltzcpnpovkcm,” “aagtzdpchgtnkrzilhnpvxqvtvqiql,” and “aakxejqunlcpqhmn-fnrckailrczyp.” We carried out the parity embedding as explained in Sec. III, setting the constraint strength to $C = 2$.

Figures 10(a)–10(d) shows the finite-time success probability $\bar{P}(t, \Delta t)$ [calculated using Eq. (6)], where $t = 30$ and $\Delta t = 70$, versus the hopping rate γ for four instances of five-logical-qubit SK Ising spin glass embedded onto the LHZ parity embedding. By considering their $\bar{P}(t, \Delta t)$ versus γ , we could compare the performance of six decoding methods: entire state decoding (red), minimum weight decoding (green), belief propagation decoding (purple), 2 + 1 nonoverlapping spanning trees with majority vote (gold), 2 + 1 nonoverlapping spanning trees with lowest energy chosen (salmon), and 2 + 1 random overlapping spanning trees with lowest energy chosen (brown).

For comparison, $\bar{P}(30, 70)$ versus γ was also plotted for one (dark blue) and three (light blue) copies of the same instances but directly embedded. The reason the $\bar{P}(30, 70)$ of three copies of the directly embedded instances was plotted is because the three copies contain 15 qubits, which is equivalent to the number of qubits used in the LHZ embedding of five logical qubits. Although we see that, for the single directly embedded copy, the performance of $\bar{P}(30, 70)$ at the respective γ_{opt} values of the LHZ embedded instances decoded using 2 + 1 nonoverlapping spanning trees with lowest energy

chosen (salmon), and $2 + 1$ random overlapping spanning trees with lowest energy chosen (brown) is close or slightly better, their performance does not approach that of the three copies. We note, however, that, as mentioned in the Introduction, this level of connectivity is not present in current hardware and therefore in practice some form of (nondirect) embedding would be required.

We observe that, for all four instances, $\bar{P}(30, 70)$ is lowest across all values of γ when they were decoded using entire state decoding (red). This is likely because the LHZ embedding has increased the number of physical qubits to $K = \frac{n(n+1)}{2}$ (n is the number of logical qubits), thereby increasing the number of states in the Hilbert space from 2^n to 2^K qubits. However, still only 2^n of the physical states correspond to valid states which can be decoded to a logical state by the entire state decoder, of which one would correspond to the logical ground state. The rest of the states are *invalid* states which cannot be decoded by the entire state decoder. This, alongside the higher probability of reading out an excited state when computing using a quantum walk (see Sec. IV A), is the likely cause of the lower performance in $\bar{P}(30, 70)$ when using this decoding method.

On the other hand, we see that the performance of $\bar{P}(30, 70)$ versus γ , for the four LHZ embedded instances when they were decoded with the minimum weight decoder (green), the belief propagation decoder (purple), the $2 + 1$ nonoverlapping spanning trees with majority vote (gold), the $2 + 1$ nonoverlapping spanning trees with lowest energy chosen (salmon), and the $2 + 1$ random overlapping spanning trees with lowest energy chosen (brown), is improved compared to the entire state decoding. The likely reason for this improvement is that, unlike the entire state decoder, these decoders are able to decode some *invalid* physical states into logical states. They do this by utilizing the redundancy present in the LHZ parity embedding to gain information on where an error in the physical state may have occurred.

We rationalize this theory in the following way: In the direct embedding there are 2^n possible states in the Hilbert space, which means if a state is chosen randomly there is a $1/2^n$ chance of finding the ground state. By using the LHZ embedding, the number of possible states in the Hilbert space is increased to 2^K . However, there are still only 2^n valid states; the rest of the $2^K - 2^n$ states are *invalid* states which cannot be decoded by the entire state decoder. If we again choose a state randomly and use the entire state decoder, the chance of finding the ground state is now reduced to $1/2^K$.

If we instead consider a decoder which is able to utilize the redundant information in the LHZ embedding in order to decode each of the invalid physical states evenly toward each of the 2^n valid states, there would be 2^{K-n} physical states per valid (logical) state. This would mean if a physical state was chosen randomly, there would be a $2^{K-n}/2^K = 1/2^n$ chance of finding the ground state. This is the same probability as for the directly embedded model. In reality, the decoder most likely will not be able to decode states evenly to each logical state (or even be able to decode every physical state to a logical one); therefore, this probability will be reduced. This is likely why we see an improvement in the $\bar{P}(30, 70)$ by using these decoders.

We see the highest performance of $\bar{P}(30, 70)$ versus γ for the four LHZ embedded instances when they were decoded using the $2 + 1$ nonoverlapping spanning trees with lowest energy chosen (salmon) or the $2 + 1$ random overlapping spanning trees with lowest energy chosen (brown). As these two methods perform better than the $2 + 1$ nonoverlapping spanning trees with majority vote (gold), it is likely that the use of the lowest energy rather than majority vote is providing the performance improvement for these two decoding methods. We analyze further the reasons for this difference in Sec. V A.

As mentioned in Sec. II, the hopping rate γ is an important parameter for computation with quantum walks. γ_{opt} is the value of γ where $\bar{P}(30, 70)$ is maximized. Looking at Fig. 10, we can see maxima in $\bar{P}(30, 70)$ for both the directly and LHZ embedded instances. Qualitatively, as we can see the displacement of the maxima in $\bar{P}(30, 70)$ to the right, the effect of the LHZ embedding on the value of γ_{opt} is to increase its value. This increase is likely due to the energy introduced by the constraints in the LHZ embedding, which shifts the location of the “balancing point” between the problem Hamiltonian $\hat{H}_P$ and initial Hamiltonian $\hat{H}_0$. The method used to decode the LHZ embedding from physical to logical state does not seem to strongly affect the location of the optimal γ - (for the maxima on the righthand side).

However, we also note that clearly for two of the four instances [Figs. 10(b) and 10(d)] there are secondary maxima in $\bar{P}(30, 70)$ on the left-hand side which appear most clearly when the LHZ embedded instance is decoded using a spanning-tree-based method with lowest energy chosen. We discuss why these secondary maxima may occur in Sec. V A.

In Sec. II A we noted that, for the SK Ising spin glass, the value of γ_{opt} cannot be calculated efficiently. (For Fig. 10, $\bar{P}(30, 70)$ was inefficiently calculated for several values of γ .) However, by using the heuristic method (described in Sec. II A), we calculated the heuristic optimal hopping rate for the directly embedded instances $\gamma_{\text{heur}}^{\text{direct}}$ and for the LHZ embedded instances $\gamma_{\text{heur}}^{\text{LHZ}}$. These are plotted in Fig. 10 as dotted vertical lines in blue and red, respectively. Similarly to the γ_{opt} maxima, we see that the value of $\gamma_{\text{heur}}^{\text{LHZ}}$ is increased compared to $\gamma_{\text{heur}}^{\text{direct}}$. This increase is likely due to the fact that the heuristic calculation uses the difference in energy between states, $|\Delta_{jk}|$, in its calculation. The constraint terms added in the LHZ embedding have higher energies leading to larger values of $|\Delta_{jk}|$, thereby increasing the value of $\gamma_{\text{heur}}^{\text{LHZ}}$. The fact that the increase in γ_{heur} for the LHZ embedded instances mirrors a similar increase in γ_{opt} indicates to us that the heuristic method is still satisfactorily functioning in this setting.

Success probability by number of spanning trees correct

In order to further analyze the reasons for the difference in finite-time success probability between the $2 + 1$ nonoverlapping spanning tree decoder which selected the computed logical ground state using lowest energy and the $2 + 1$ nonoverlapping spanning tree decoder which selected the computed logical ground state using majority vote, in Fig. 11 we reanalyzed the $\bar{P}(30, 70)$ versus γ data of the same

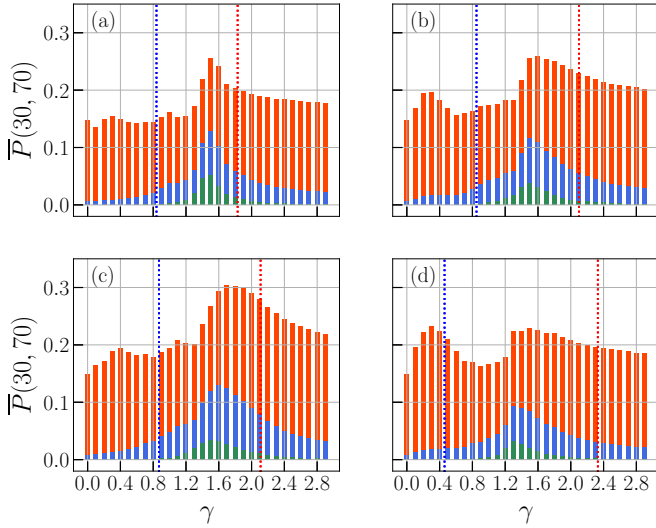


FIG. 11. Graph showing the success probability $\bar{P}(30, 70)$ vs γ for the same (as in Fig. 10) five-qubit LHZ embedded SK Ising spin glass instances decoded using 2 + 1 nonoverlapping spanning trees with lowest energy chosen. The total success probability is split up into three (2 + 1) spanning trees correct (green), two spanning trees correct (blue), and one spanning tree correct (orange-red). $\gamma_{\text{opt}}^{\text{direct}}$ and $\gamma_{\text{heur}}^{\text{LHZ}}$ are again plotted as dotted vertical lines in blue and red, respectively.

four LHZ embedded five-qubit SK Ising spin glass instances decoded using the 2 + 1 nonoverlapping spanning trees with lowest energy chosen. In this section, we still use the same 2 + 1 nonoverlapping spanning trees decoding method (where “+1” indicates that one of the spanning trees is in fact the readout of the data qubits). However, for simplicity, we refer to 2 + 1 spanning trees as *three* spanning trees.

For the reanalysis, we plotted $\bar{P}(30, 70)$ as bars split into cases when three spanning trees were correct (green), two spanning trees were correct (blue), or just one spanning tree was correct (orange-red). We verify that the height of the total bars is the same as the height of the nonoverlapping spanning trees with lowest energy chosen points (salmon) in Fig. 10 (note the change in y-axis scale).

We first observe that, for all of the instances, the fraction of the bar where just one spanning tree was correct (red-orange) makes up over half of the total bar for most values of γ . This indicates that the lowest energy method of selection which allows the “correct” ground state to be found, even if it occurs just once in the spanning trees readout, can strongly improve the performance of a decoding method. We also observe that the height of the fraction of the bars where three (green) or two (blue) spanning trees were correct correlates with the height of the points of the 2 + 1 nonoverlapping spanning trees with majority vote in Fig. 10. This indicates that the performance of the “lowest energy” selection method when combined with the limitation of needing at least two of the spanning trees correct is similar to the performance when using the “majority votes” selection method.

We hypothesize the reason for these two observations below. Like the minimum weight and belief-propagation decoders, the spanning tree decoders are able to decode some

invalid states back to valid logical states. The difference in performance occurs when there is a choice of possible logical states from which the decoder needs to select the computed final logical state. For the minimum weight, belief-propagation, and spanning trees with majority vote decoders, a single quantum walk readout physical state will give, once the decoder is applied, a single logical output state. However, for the spanning tree decoder with lowest energy chosen, a single quantum walk readout physical state could lead to two or three different logical output states depending on the evaluation of the energy. We suspect this is the reason behind why this form of the spanning tree decoding method outperforms the others.

As noted in Sec. V about Figs. 10(b) and 10(d), we also observe secondary maxima in $\bar{P}(30, 70)$ at $\gamma \ll \gamma_{\text{opt}}^{\text{LHZ}}$ in Fig. 11 clearly for instances in Figs. 11(b) and 11(d). In addition, we observe that these maxima consist almost entirely of the fraction of the bar where only a single spanning tree is correct and is not clearly present in the two or three spanning tree correct fractions.

We hypothesize the reasons for these secondary maxima below. Normally, when solving optimization problem using quantum walks, the aim is to find the ground state of the optimization problem. γ_{opt} represents the value of γ where the probability of finding this ground state is maximized. On the other hand, when using the LHZ embedding, decoding methods allow us to find the logical ground when we are in an “other state” that is not the physical ground state of the LHZ model. The value of γ which maximizes the probability of finding this other state may be different than $\gamma_{\text{opt}}^{\text{LHZ}}$. We suggest that the secondary maxima in Figs. 10 and 11 represents a value of γ which is optimal for a state that is not the physical ground state, but one that can be decoded to the logical ground state using the decoding method of spanning trees with lowest energy chosen.

VI. SUCCESS PROBABILITY VERSUS PROBLEM SIZE: VARIOUS DECODING METHODS

We next looked at how the performance of each of the decoding methods for the quantum walk scaled with the logical number of qubits of the LHZ parity embedded SK Ising spin glasses. Figure 12 shows the average finite-time success probability $\bar{P}(30, 70)$ of LHZ parity embedded SK Ising spin glass instances after decoding by different methods versus the logical problem size of 4 (10 physical), 5 (15 physical), and 6 (21 physical) qubits. For four and five logical qubits $\bar{P}(30, 70)$ was an average over 100 instances, and for six logical qubits, 48 instances. The methods of decoding used were entire state decoding (solid red), belief-propagation decoding (solid purple), minimum weight decoding (solid green), 2 + 1 random spanning trees with lowest energy chosen (solid salmon), and 2 + 1 nonoverlapping spanning trees with lowest energy chosen (solid brown). The 2 + 1 label of spanning trees signifies that two spanning trees plus the set of data qubits were read out from the physical state computed by the quantum walk.

Although (as noted in the Introduction) the level of connectivity required for direct embedding is not present in current hardware, as a benchmark for the $\bar{P}(30, 70)$ performance of our LHZ embedded instances, we also plotted the average

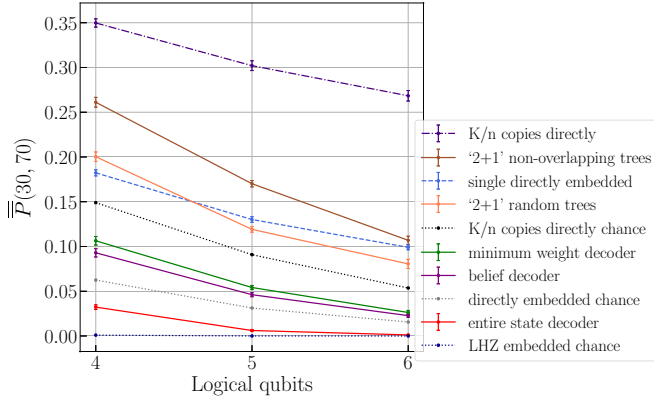


FIG. 12. Graph showing $\bar{P}(30, 70)$ averaged over 100 instances of four- and five-logical-qubit and 48 instances of 6-logical-qubit SK Ising spin glasses embedded onto the LHZ parity embedding versus the logical qubit number. The results are shown from five different decoding methods (solid lines): entire state (red), belief propagation (purple), minimum weight (green), 2 + 1 random spanning trees (salmon), and 2 + 1 nonoverlapping spanning trees (brown). Error bars on each of the data points indicate the uncertainty in $\bar{P}(30, 70)$. For comparison, the average success probability of K/n copies and a single copy of the directly embedded instances are plotted as dot-dashed indigo and dashed blue lines, respectively. The random chance probabilities (dotted) are shown for an LHZ embedded instance (dark blue) and a single directly embedded instance (grey); K/n copies of a directly embedded instance (black) are also plotted.

finite-time success probability $\bar{P}(30, 70)$ of K/n copies (dot-dashed purple line) and a single copy (dashed blue line) of the directly embedded instances. K/n directly embedded copies have a number of qubits equivalent to the number of physical qubits required for an LHZ embedding of n logical qubits. Note that K/n is not an integer number of copies for 4 (10/4) and 6 (21/6) logical qubits.

Furthermore, to check whether the $\bar{P}(30, 70)$ performance was better than the probability of finding the correct ground state when selecting a state randomly (before performing quantum walk), we also plotted several random chance success probabilities. We found the random chance success probabilities of K/n copies of the directly embedded model ($p_{K/n} = 1 - (1 - p_n)^{K/n}$) (dotted black line), a single copy of the directly embedded model ($p_n = 1/2^n$) (dotted gray line), and a single copy of the LHZ embedded model ($p_K = 1/2^K$) (dotted dark blue line).

Looking at Fig. 12, we see that all decoding methods outperform the random chance of finding the correct ground state on a LHZ embedded model p_K (dotted dark blue line). However, the LHZ embedded instances decoded using the entire state decoding method (solid red line) do not outperform the random chance of finding the correct ground state on a single copy of the directly embedded model p_n (dotted gray line). This indicates that, in order to achieve better than random guessing performances when computing using quantum walks in the LHZ embedding, a decoding method other than entire state decoding should be used.

On the other hand, both the belief-propagation (solid purple line) and minimum weight decoders (solid green line)

outperform the random chance of finding the correct ground state on the directly embedded model p_n (dotted gray), but not the random chance of finding the correct ground state in K/n copies of the directly embedded model $p_{K/n}$ (dotted black). This indicates that using the belief propagation or minimum weight decoding improves the success probability of the quantum walk on the LHZ embedded instance, such that it outperforms random guessing on the original number of qubits. However, if the number of qubits is scaled up to a number equivalent to that required for the LHZ embedding, this improvement in performance is lost.

Conversely, we see that both spanning tree decoding methods with lowest energy chosen (solid salmon and solid brown lines) outperform the random chance of finding the correct ground state in K/n copies of the directly embedded model $p_{K/n}$ (dotted black line). This indicates that, by decoding with these spanning tree decoding methods, the computation via quantum walk on the LHZ embedded instances can outperform random guessing on an equivalent number of qubits.

The 2 + 1 nonoverlapping spanning trees method (solid brown line) also outperforms the single copies of directly embedded instances (dashed blue line) at the problem sizes shown. This indicates that at these sizes the 2 + 1 nonoverlapping spanning trees decoding method outperforms the best ideal (but not achievable on current hardware) way of embedding for the same number of logical qubits (at the cost of having more physical qubits). However, the scaling in Fig. 12 suggests that the single copies of directly embedded models would outperform both 2 + 1 spanning tree decoding methods at larger numbers of logical qubits. We suggest that increasing the number of spanning trees read out as the number of logical qubits increases could potentially improve this scaling. We also predict the advantage of the 2 + 1 nonoverlapping trees over the 2 + 1 random trees would trend to zero as the size of the model increased due to the reducing likelihood of spanning trees overlapping.

All LHZ parity embedded methods perform worse than K/n copies of the directly embedded instances (dot-dashed indigo line). However, it is currently experimentally infeasible to directly embed all-to-all connected models, even with low numbers of qubits (such as the models simulated in this paper). As the number of qubits increases to amounts that could potentially show quantum advantage, the chances of being able to use a direct embedding become slimmer. Therefore, some form of embedding will be required. The results of this paper indicate that by using the LHZ parity embedding with the appropriate decoding techniques, we may be able to recover the success probability to a level close to what would be achievable if a direct embedding were possible.

VII. CONCLUSION

In this work we have analyzed and compared the performance of continuous-time quantum walks in solving LHZ parity embedded optimization problems while using several different decoding methods. We have found that the average finite-time success probability $\bar{P}(30, 70)$ of the quantum walk on the LHZ parity embedded problems is improved when a

decoding method is selected which utilizes the redundancy present in the embedding in order to decode *invalid* physical states into valid logical states.

Although all-to-all directly embedded models are experimentally unfeasible, for the research in this paper, as a baseline, we compared the $\bar{P}(30, 70)$ performance of the LHZ embedded models against the $\bar{P}(30, 70)$ performance of a number of copies of the directly embedded models which matched the number qubits required for the LHZ embedding.

We confirmed (in Sec. V) that the heuristic method developed in Ref. [28] continues to be an effective way of estimating the value of the optimal hopping rate γ_{opt} in the LHZ setting. In addition, we showed (in Sec. III A) that our results are in agreement with the theory in Ref. [11] outlining the lower bound of the constraint strength C , if we take into account the increasing difference between the true value of the optimal hopping rate γ_{opt} and the heuristically calculated hopping rate γ_{heur} as the strength of C increases. We found that, due to the interplay between the values of constraint strength C and γ , the value of C which optimizes the average finite-time success probability $\bar{P}(30, 70)$ when using $\gamma_{\text{heur}}^{\text{LHZ}}$ lies below the calculated average lower bound of C . This counter-intuitive result raises the question, following results in Refs. [18–20], whether, under alternative continuous-time quantum computing techniques and decoding methods, different constraint strength strategies could be utilized more effectively.

We found that the finite-time success probability $\bar{P}(30, 70)$, when using the decoding methods belief propagation, minimum weight, and $2 + 1$ nonoverlapping spanning trees with majority vote, showed modest improvements over entire state decoding. We suggest that this improvement is due to these methods’ ability to utilize the redundancy of the LHZ embedding in order to find valid logical states from invalid physical states, i.e., states which lie outside the logical code space.

Furthermore, when using the decoding methods $2 + 1$ random spanning trees with lowest energy chosen and $2 + 1$ nonoverlapping spanning trees with lowest energy chosen, we saw further improvements in $\bar{P}(30, 70)$ performance. We suggest that this further improvement is due to these decoding methods’ (computed logical ground state) selection process, which considers the energy of the possible options rather than a majority vote, allowing a single physical state to be decoded to multiple “choices” of logical states (rather than a single physical state to a single logical state as with the other decoding methods). This suggestion is backed up by the fact that this further improvement is not seen when the majority votes selection method is used on the same decoder. Also, many of the correct ground states found using the “lowest energy” selection method were read out from just one of the spanning trees specified.

We found that although the method of $2 + 1$ nonoverlapping spanning trees with lowest energy chosen outperformed the single directly embedded models for the sizes simulated, its scaling with logical qubit number was worse. We suggest this scaling could be improved if the number of spanning trees was increased with logical qubit number (as was done in Ref. [13]).

The results presented in this paper suggest that decoding methods for the LHZ parity embedding which consider the energies of a selection of logical states when making the selection of final computed logical ground state, and which are scaled with problem size, may provide improvements in success probability, when using nonadiabatic continuous-time quantum computing techniques, such as quantum walks. Therefore, we suggest that future decoding methods for the LHZ parity embedding in this setting should consider the above criteria as a guide to improving their performance.

ACKNOWLEDGMENTS

We thank Anita Weidinger for interesting discussions and Berend Klaver for providing error syndrome to belief-propagation correction mappings. This work was supported by the Austrian Research Promotion Agency (FFG Project No. FO999896208). This research was funded in whole, or in part, by the Austrian Science Fund (FWF) SFB Beyond C Project No. F7108-N38, through a START grant under Projects No. Y1067-N27 and No. I 6011. This project was funded within the QuantERA II Programme that has received funding from the European Union’s Horizon 2020 research and innovation programme under Grant Agreement No. 101017733. V.K. and N.C. were funded by UKRI EPSRC UK Quantum Technology Hub in Computing and Simulation (Grant No. EP/T001062/1). We acknowledge funding from the UKRI EPSRC International Network on Quantum Annealing (Grant No. EP/W027003/1) was useful in the preparation of this work. The computational results presented here have been achieved in part using the LEO HPC infrastructure of the University of Innsbruck.

DATA AVAILABILITY

The data that support the findings of this article are openly available [35].

APPENDIX A: THE CLASSICAL COST OF MEASURING SPANNING TREES

An instructive way to consider the cost of measuring spanning trees is to compare it to the cost of taking random guesses.

We consider a four-qubit logical SK Ising spin glass problem, embedded onto an LHZ embedding with six qubits (here we exclude the contribution from the data qubits), prior to performing a (quantum) computation and assuming a uniform probability distribution. If we decode using the entire state decoding method, it can be seen that we have a 1 in $2^6 = 64$ chance of getting the correct answer: the same chance as a random guess. The effect of the quantum computation should be to at least increase the probability of measuring the correct ground state of the Ising model, even when using a simplistic decoding method such as entire state decoding.

If we continue to look at the uniform probability distribution, and continue to decode using entire state decoding, we can see that each time we take a new guess, the probability

that we find the correct ground state increases with $g \times 1/64$, with g being the number of guesses.

However, if we instead use a random overlapping spanning trees method, we change how this probability behaves. If we just specify a single spanning tree, we already increase the number of physical states that we find which decode to the correct ground state. This is because we only read out three qubits for the spanning tree, meaning the other three are free to be in whichever state, meaning that there are 8 possible correct states out of the total of 64, meaning that we now have an 8-in-64 (or 1/8) chance of getting the correct answer.

If we were to specify one additional spanning tree, we might naively expect to add another $2^3 - 1 = 7$ possible states (2^3 minus the state where both spanning trees occur, which was already counted). Indeed this would be the case if we were to ensure that the additional spanning tree did not overlap with the original. In the uniform probability case this would mean that the chance of measuring the correct ground state when using two spanning trees is $(8 + 7)/64 = 15/64$.

However, it is likely that if we were picking spanning trees (that are allowed to overlap) randomly, we would see an overlap of at least one qubit between spanning trees. This likelihood would increase with each additional spanning tree. (It is also impossible in the four-logical-qubit, six-physical-qubit case for the third spanning tree in the two nonoverlapping trees scenario, to not overlap.) When a spanning tree overlaps with another one, this means that the number of possible states that are added is reduced. For instance, if the second spanning tree were to have overlapped with the original with just one qubit, then the number of additional possible states would have been $2^3 - 2^1 = 6$. If the overlap was two then it would have been $2^3 - 2^2 = 4$. Generally we could say that the number of additional states is $2^{(K-N)} - 2^{O_{AB}}$, where O_{AB} is the overlap (in number of qubits) between the first two spanning trees A and B . If we were to add a third spanning tree C then the additional number of states would be $2^{(K-N)} - 2^{O_{AC}} - (2^{O_{BC}} - 2^{O_{ABC}})$. As the number of spanning trees increases, so does the number of ways they can overlap and so the number of terms in the above equations.

However, following a reasoning in Table I, we are able to count the growing number of additional states in a simpler way. First we write out (for $K = 6$, $n = 4$) the $n^{(n-2)} = 16$ possible spanning trees. To generalize, we denote the two possible qubit values as X (for correct as part of the spanning tree) and Y (for incorrect but not part of the current spanning tree). We denote qubits that may be in either state as “_”. We then count the additional possible states by a process of elimination. For example, the first tree XXX _ _ _ adds 2^3 possible states as there are three qubits that can be in either state. Then when we add the second tree XX _ X _ , we notice that the states where the third qubit was X were already counted by the first spanning tree, so there is only one choice for the third qubit (Y), and thus we have XXY _ X _ and so $2^2 = 4$ additional states. This happens again if we add the third spanning tree XX _ _ X. Then we have XXY _ YX, and so $2^1 = 2$ additional states.

If we continue this process of counting the additional states for each possible spanning tree (16 for 4 logical qubits) then

TABLE I. Table showing the 16 possible spanning trees for the four-logical-qubit SK Ising spin glass on the six physical qubits of the LHZ parity encoded model in the second column. (The columns are counted left to right). The third column indicates the possible additional states (that have not been counted yet), where “_” indicates the qubit may be in either 0 or 1 and “Y” indicates that it must be the opposite state to that which has appeared in a previous spanning tree (only one choice). The fourth column indicates the number of new states that the corresponding spanning tree has added and the fifth column shows the running total.

m	Spanning tree 01 02 03 12 13 23	New states	No. of new states	Total
1	XXX _ _ _	XXX _ _ _	$2^3 = 8$	8
2	XX _ X _	XXY _ X _	$2^2 = 4$	12
3	XX _ _ X	XXY _ YX	$2^1 = 2$	14
4	X _ X X _	XYX X _ _	$2^2 = 4$	18
5	X _ X _ X	XYX Y _ X	$2^1 = 2$	20
6	X _ _ XX _	XYX XX _	$2^1 = 2$	22
7	X _ _ X _ X	XYX XYX	$2^0 = 1$	23
8	X _ _ _ XX	XYX YXX	$2^0 = 1$	24
9	_ XX X _ _	YXX X _ _	$2^2 = 4$	28
10	_ XX _ X _	YXX YX _	$2^1 = 2$	30
11	_ X _ XX _	YXY XX _	$2^1 = 2$	32
12	_ X _ X _ X	YXY XYX	$2^0 = 1$	33
13	_ X _ _ XX	YXY YXX	$2^0 = 1$	34
14	_ _ X XX _	YYX XX _	$2^1 = 2$	36
15	_ _ X X _ X	YYX XYX	$2^0 = 1$	37
16	_ _ X _ XX	YYX YXX	$2^0 = 1$	38

we find that we end up with 38 possible states. In terms of probability, this represents for the uniform distribution a 38/64 chance of finding the correct ground state, if we were to specify all the possible spanning trees. This shows the difference between the spanning tree decoding and repeated guesses of the entire state. If we were to continue guessing random entire states we would eventually find the ground state (success probability of 1). However, even using every single spanning tree this approach will not find the ground state with certainty. This is because when measuring spanning trees, we are not altering the values of each of the physical qubits, but using them to decode from a physical to a logical state.

We also note that a different order of spanning trees would add additional states at a different rate; i.e., two spanning trees with less overlap would add more states to the total. However, since there is a fixed total number of spanning trees, this means that the total number of states that can be added if all spanning trees are specified is fixed, even if the order in which the spanning trees are specified changes.

APPENDIX B: SEMIANALYTICAL METHOD FOR COUNTING THE TOTAL NUMBER OF POSSIBLE STATES INTRODUCED BY MEASURING ALL SPANNING TREES

We start off by noting that a valid possible state must have (in the X notation of the previous section) at least $(n - 1)$ X's.

If we look at the previous six-qubit examples we can say that we can find the number of combinations of three X's using the combination equation as follows:

$$C = \frac{K!}{(K - n_X)!n_X!}. \quad (\text{B1})$$

When $K = 6$ and $n_X = 3$, we find there are 20 possible combinations.

However, we know that there exist a few invalid combinations, i.e., those that do not correspond to spanning trees in the original model. Therefore, we must subtract the number of invalid combinations from 20, in order to get the number of possible states with exactly three X's. We know that for this example, there are four invalid states: $XX_X_$, $X_X_X_$, $_XX_X$, and $_XXX$. Therefore, there are 16 valid combinations of three X's which correspond to possible states. As expected, this corresponds to the number of possible spanning trees on the four-qubit SK Ising spin glass. However, we also need to know about the possible states with more than three X's.

If we next look at the number of combinations of four X's, using Eq. (B1), we can find that there are 15 possible combinations. In general we would normally need to now remove the combinations that correspond to invalid states; however, in this example every combination containing four X's contains at least one valid spanning tree. Continuing the same steps for the permutations containing five and six qubits, we find that they add six and one possible state(s), respectively. If we sum all the valid possible states with three or more X's together, we find that this gives us a total of 38, which equals the total found using the method in Table I.

To describe this method more generally we could write the number of possible states S as

$$S = \sum_{i=l}^K \left(\frac{K!}{i!(K-i)!} - I_i \right), \quad (\text{B2})$$

where l is the length of the spanning trees ($n-1$) and I_i is the number of invalid states for that number of X's. When the number of X's, n_X , is equal to l , we are able to find I_l by subtracting the number of possible spanning trees (found using Cayley's formula) from the number of combinations found using Eq. (B1). However, for $n_X > l$, how to find the number of invalid states I_i remains an open question.

APPENDIX C: RANDOM OVERLAPPING SPANNING TREE DECODER: PERFORMANCE VERSUS NUMBER OF TREES

We next investigated the average performance when decoding using random overlapping spanning trees across 100 instances of LHZ parity embedded SK Ising spin glasses versus the number of random overlapping spanning trees specified for logical problem sizes of 4 (10 physical) and 5 (15 physical) qubits.

Figure 13 shows the average finite-time success probability $\bar{P}(30, 70)$ versus number of spanning trees, over 100 instances of four-logical-qubit (red) and five-logical-qubit (blue) LHZ parity embedded SK Ising spin glass instances. The standard error in $\bar{P}(30, 70)$ is shown as error bars on each data point. For comparison we also plotted the chance of finding the

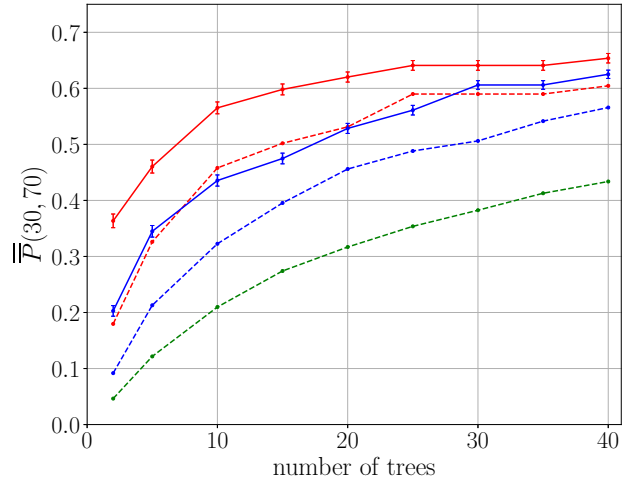


FIG. 13. Graph showing the average finite-time success probability $\bar{P}(30, 70)$ averaged over 100 instances of four- (solid red line) and five-logical qubit (solid blue line) SK Ising spin glasses embedded onto the LHZ parity embedding, vs the number of random overlapping spanning trees. Error bars on each data point indicate the uncertainty in the average finite-time success probability. Plotted in the corresponding color at each problem size, the dotted lines represent the number of states containing at least one correct spanning tree divided by the total number of states (2^K), i.e., the probability of finding the correct ground state when decoding a randomly chosen state using random spanning trees. These data points do not have error bars as they are counted directly from the number of states determined by the order and number of spanning trees which is fixed at each problem size.

correct ground state from decoding a random physical state using random overlapping spanning trees versus the number of specified trees (dashed lines) for both four and five logical qubits in the corresponding colors and for six logical qubits in green.

We calculated this random chance by first finding the number of possible physical states containing at least one correct spanning tree at each number of spanning trees specified, and then dividing by the total number of possible physical states (2^K). We found the number of possible physical states containing at least one correct spanning tree using the method outlined in Appendix B.

As the number of possible physical states which contain at least one correct spanning tree is determined solely by the number and position of the spanning trees, there are no error bars on these data points. We note, however, that a different choice of spanning trees would change slightly the location and rate of increase of the data points. However, as the total number of spanning trees is fixed, so is the total number of possible physical states containing a correct spanning tree, as we saw in Appendix B.

Looking at Fig. 13, we see that, as expected for both logical sizes—four- and five-logical qubits—the success probability of the quantum walk outperforms random chance. In both the quantum walk and random chance cases we see a reduction in the rate of increase in the success probability as the number of spanning trees increase. This is due to the increased chance of a new spanning tree having an overlapping qubit with another

spanning tree. An overlapping qubit causes a reduction in the number of new possible states and therefore the likelihood that a new correct spanning tree is found. We see no increase in

the success probability if a new spanning tree provides no new correct states or if a new spanning tree reads out no new qubits (totally overlapping).

-
- [1] K. Boothby, A. D. King, and J. Raymond, Zephyr topology of D-wave quantum processors, Technical Report Number: 14-1056A-A, Company: D-Wave (Canada, 2021).
 - [2] C. Chamberland, G. Zhu, T. J. Yoder, J. B. Hertzberg, and A. W. Cross, Topological and subsystem codes on low-degree graphs with flag qubits, *Phys. Rev. X* **10**, 011022 (2020).
 - [3] V. Choi, Minor-embedding in adiabatic quantum computation: I. The parameter setting problem, *Quantum Inf. Process.* **7**, 193 (2008).
 - [4] V. Choi, Minor-embedding in adiabatic quantum computation: II. Minor-universal graph design, *Quantum Inf. Process.* **10**, 343 (2011).
 - [5] minorminer, <https://github.com/dwavesystems/minorminer> (2018).
 - [6] E. Mozgunov, Precision of quantum simulation of all-to-all coupling in a local architecture, [arXiv:2302.02458](https://arxiv.org/abs/2302.02458).
 - [7] X. Qiu, P. Zoller, and X. Li, Programmable quantum annealing architectures with Ising quantum wires, *PRX Quantum* **1**, 020311 (2020).
 - [8] M. T. Nguyen, J. G. Liu, J. Wurtz, M. D. Lukin, S. T. Wang, and H. Pichler, Quantum optimization with arbitrary connectivity using Rydberg atom arrays, *PRX Quantum* **4**, 010316 (2023).
 - [9] A. Palacios, A. Garcia-Saez, and M. P. Estarellas, Scalable 2-local architecture for quantum annealing of Ising models with arbitrary dimensions, *Phys. Rev. Applied* **23**, 054070 (2025).
 - [10] W. Lechner, P. Hauke, and P. Zoller, A quantum annealing architecture with all-to-all connectivity from local interactions, *Sci. Adv.* **1**, e1500838 (2015).
 - [11] M. Lanthaler and W. Lechner, Minimal constraints in the parity formulation of optimization problems, *New J. Phys.* **23**, 083039 (2021).
 - [12] A. Weidinger, G. B. Mbeng, and W. Lechner, Error mitigation for quantum approximate optimization, *Phys. Rev. A* **108**, 032408 (2023).
 - [13] A. Weidinger, G. B. Mbeng, M. Fellner, D. Khachatryan, and W. Lechner, Performance of parity QAOA for the signed max-cut problem, [arXiv:2409.14786](https://arxiv.org/abs/2409.14786).
 - [14] T. Albash, W. Vinci, and D. A. Lidar, Simulated-quantum-annealing comparison between all-to-all connectivity schemes, *Phys. Rev. A* **94**, 022327 (2016).
 - [15] J. King, M. Mohseni, W. Bernoudy, A. Fréchette, H. Sadeghi, S. V. Isakov, H. Neven, and M. H. Amin, Quantum-assisted genetic algorithm, [arXiv:1907.00707](https://arxiv.org/abs/1907.00707).
 - [16] F. Pastawski and J. Preskill, Error correction for encoded quantum annealing, *Phys. Rev. A* **93**, 052325 (2016).
 - [17] A. Messinger, V. Torggler, B. Klaver, M. Fellner, and W. Lechner, Fault-tolerant quantum computing with the parity code and biased-noise qubits, *Phys. Rev. Applied* **23**, 044032 (2025).
 - [18] Y. Nambu, Error correction of parity-encoding-based annealing through post-readout decoding, [arXiv:2402.08839](https://arxiv.org/abs/2402.08839).
 - [19] Y. Nambu, Error correction for encoded quantum annealing revisited, [arXiv:2407.15480](https://arxiv.org/abs/2407.15480).
 - [20] Y. Nambu, Efficient classical error correction for parity encoded spin systems, [arXiv:2502.07170](https://arxiv.org/abs/2502.07170).
 - [21] M. Fellner, A. Messinger, K. Ender, and W. Lechner, Universal parity quantum computing, *Phys. Rev. Lett.* **129**, 180503 (2022).
 - [22] A. Messinger, M. Fellner, and W. Lechner, Constant depth code deformations in the parity architecture, in *Proceedings of the 2023 IEEE International Conference on Quantum Computing and Engineering (QCE)* (IEEE, New York, 2023), pp. 120–130.
 - [23] I. D. Smith, H. P. Nautrup, and H. J. Briegel, Parity quantum computing as YZ-plane measurement-based quantum computing, *Phys. Rev. Lett.* **132**, 220602 (2024).
 - [24] E. Farhi and S. Gutmann, Quantum computation and decision trees, *Phys. Rev. A* **58**, 915 (1998).
 - [25] A. M. Childs and J. Goldstone, Spatial search by quantum walk, *Phys. Rev. A* **70**, 022314 (2004).
 - [26] A. Callison, N. Chancellor, F. Mintert, and V. Kendon, Finding spin glass ground states using quantum walks, *New J. Phys.* **21**, 123022 (2019).
 - [27] J. G. Morley, N. Chancellor, S. Bose, and V. Kendon, Quantum search with hybrid adiabatic-quantum-walk algorithms and realistic noise, *Phys. Rev. A* **99**, 022339 (2019).
 - [28] A. Callison, M. Festenstein, J. Chen, L. Nita, V. Kendon, and N. Chancellor, An energetic perspective on rapid quenches in quantum annealing, *Phys. Rev. X Quantum* **2**, 010338 (2021).
 - [29] R. J. Banks, E. Haque, F. Nazef, F. Fethallah, F. Ruqaya, H. Ahsan, H. Vora, H. Tahir, I. Ahmad, I. Hewins, I. Shah, K. Baranwal, M. Arora, M. Asad, M. Khan, N. Hasan, N. Azad, S. Fedaiee, S. Majeed, S. Bhuyan *et al.*, Continuous-time quantum walks for max-cut are hot, *Quantum* **8**, 1254 (2024).
 - [30] S. Schulz, D. Willsch, and K. Michielsen, Guided quantum walk, *Phys. Rev. Res.* **6**, 013312 (2024).
 - [31] L. Gerblich, T. Dasanjh, H. Q. X. Wong, D. Ross, L. Novo, N. Chancellor, and V. Kendon, Advantages of multistage quantum walks over QAOA, [arXiv:2407.06663](https://arxiv.org/abs/2407.06663).
 - [32] R. Ghosh, L. A. Nutricati, N. Feinstein, P. A. Warburton, and S. Bose, Exponential speed-up of quantum annealing via n-local catalysts, [arXiv:2409.13029](https://arxiv.org/abs/2409.13029).
 - [33] J. Bennett, Suppressing precision errors by connecting copies of Ising models for continuous-time quantum computing, Ph.D. thesis, Durham University, 2023.
 - [34] Y. Nambu, Rejection-free Monte Carlo simulation of QUBO and Lechner-Hauke-Zoller optimization problems, *IEEE Acc.* **10**, 84279 (2022).
 - [35] N. Chancellor, A. Callison, V. Kendon, and F. Mintert, *Finding Spin-Glass Ground States Using Quantum Walks*, data archive

- at Durham University, UK (2019), for spin glass instances used in [26], doi: [10.15128/r21544bp097](https://doi.org/10.15128/r21544bp097).
- [36] V. Choi, Adiabatic quantum algorithms for the NP-complete maximum-weight independent set, exact cover and 3SAT problems, [arXiv:1004.2226](https://arxiv.org/abs/1004.2226).
- [37] D. Sherrington and S. Kirkpatrick, Solvable model of a spin-glass, *Phys. Rev. Lett.* **35**, 1792 (1975).
- [38] A. Rocchetto, S. C. Benjamin, and Y. Li, Stabilizers as a design tool for new forms of the Lechner-Hauke-Zoller annealer, *Sci. Adv.* **2**, e1601246 (2016).