

Practical hybrid decoding scheme for parity-encoded spin systems

Yoshihiro Nambu^{*}

NEC-AIST Quantum Technology Cooperative Research Laboratory, *National Institute of Advanced Industrial Science and Technology*, Tsukuba, Ibaraki 305-8568, Japan



(Received 23 July 2024; revised 14 May 2025; accepted 17 December 2025; published 20 January 2026)

We propose a practical hybrid decoding scheme for the parity-encoding architecture. This architecture was first introduced by N. Sourlas [Phys. Rev. Lett. **94**, 070601 (2005)] as a computational technique for tackling hard optimization problems, especially those modeled by spin systems such as the Ising model and spin glasses, and reinvented by W. Lechner, P. Hauke, and P. Zoller [Sci. Adv. **1**, e1500838 (2015)] to develop quantum annealing devices. We study the specific model, called the SLHZ model, aiming to achieve a near-term quantum annealing device implemented solely through geometrically local spin interactions. Taking account of the close connection between the SLHZ model and a classical low-density parity-check (LDPC) code, two approaches can be chosen for the decoding: (1) finding the ground state of a spin Hamiltonian derived from the SLHZ model, which can be achieved via stochastic decoders such as a quantum annealer or a classical Monte Carlo sampler; or (2) using deterministic decoding techniques for the classical LDPC code, such as belief propagation and a bit-flip decoder. The proposed hybrid approach combines the two approaches by applying bit-flip decoding to the readout of the stochastic decoder based on the SLHZ model. We present simulations demonstrating that this approach can reveal the latent potential of the SLHZ model, realizing the soft-annealing concept proposed by Sourlas.

DOI: [10.1103/drry-blqs](https://doi.org/10.1103/drry-blqs)

I. INTRODUCTION

The spin-glass model is a fundamental concept in physics and mathematics, primarily used to study magnetism and phase transitions in statistical mechanics. A close connection between spin-glass models and combinatorial-optimization problems (COPs) is widely recognized and used as a tool for solving COPs [1]. The key idea is that the energy of the random Ising Hamiltonian can be viewed as a cost function the ground state of which corresponds to a solution of the COP. Probabilistic simulation of Ising spin dynamics, such as simulated annealing (SA) [2] and quantum annealing (QA) [3], is often used to find the ground state. The spin-glass model has applications in various industrial fields, including routing, scheduling, planning, decision-making, transportation, and telecommunications. A similar connection has been established between the spin-glass model and the classical error-correcting codes (ECCs). Sourlas formulated the decoding of the classical ECCs in terms of

Bayesian inference [4]. He showed that the decoding can be expressed as a search for the ground state of the random Ising Hamiltonian. His study implies that decoding classical ECCs can be viewed as a COP, with the associated random Ising Hamiltonian corresponding to the cost function to be minimized (maximized).

The complexity class of the COPs is known to be NP-hard. When we use the spin-glass models to tackle the COPs, the presence of many local minima separated by large energy barriers is a primary problem. The simulation algorithm gets trapped in those minima. To tackle this problem, Sourlas noted that two mathematically equivalent Hamiltonians can be derived for an ECC: a Hamiltonian given in terms of the original information source and an extended Hamiltonian given in terms of the information source and additional parity information [5–7]. The extended Hamiltonian involves extra parity-constraint terms. He proposed the soft-annealing concept, which uses the extended Hamiltonian as a new approach to solve hard COPs [8]. The key idea is that one can circumvent barriers and accelerate the dynamics of the algorithm by enlarging the space in which the problem is defined.

On the other hand, the QA is expected to resolve the above problem through adiabatic quantum computing [9]. However, to apply a QA device to solve the COP universally, we need to simulate a fully connected graph model in a scalable manner using Ising spin hardware. The

^{*}Contact author: y-nambu@aist.go.jp

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.

soft-annealing approach is also helpful for this purpose. Lechner, Hauke, and Zoller (LHZ) proposed utilizing this approach to embed a random Ising model with all-to-all connectivity within an enlarged Ising system with geometrically local interactions only [10]. They proposed the parity-encoding (PE) architecture and its concrete realization, which we refer to as the SLHZ model in this paper. In this model, Ising spins are arranged on a plane. They are connected by four-body interactions in a latticelike manner, which serve as a penalty that constrains the parities of spins. Both the local fields and couplings in the original spin-glass model are mapped to local fields that act on each spin in the SLHZ model. This architecture is particularly attractive for realizing near-term QA devices based on superconducting quantum bits [11–14] because it is highly compatible with on-chip electronic circuitry [15–19]. LHZ also claimed that the SLHZ model is a programmable, scalable, and robust physical platform for QA. Notably, Sourlas was the first researcher to propose soft annealing using the SLHZ model and to recognize that this corresponds to the decoding of associated classical ECCs [8].

Later, Pastawski and Preskill [20] studied the error-correcting capability of the SLHZ model. They noted that the soft annealing using the SLHZ model can be viewed as decoding of the associated classical low-density parity-check (LDPC) code. Based on this observation, they applied the belief-propagation (BP) algorithm, a standard algorithm for LDPC codes [21], to decode the noisy readout of the SLHZ system [22,23]. They found that one can obtain an error-free state from the noisy readout of the SLHZ model with a high probability if errors are generated by independent and identically distributed (IID) random spin-flip noise.

To date, however, there has been little investigation focusing on whether the realistic classical decoding technique can fully exploit the potential of the SLHZ model. For example, Albash, Vincl, and Lidar have discussed whether several decoding algorithms may boost the performance of the QA device [24]. They have reported that simple majority-vote decoding (MVD) does not improve the performance of QA. Although they have found that more sophisticated minimum-weight decoding (MWD) can boost performance, it is NP-hard and an unrealistic approach for postreadout decoding of QA devices (for details, see Ref. [24] and Appendix A). In the context of developing near-term QA, it is preferable to devise a more practical decoding strategy tailored for the SLHZ model that can exploit its potential.

This paper demonstrates that realistic classical decoding techniques for LDPC codes can unveil potential in the SLHZ model that has not been recognized to date. We propose a practical and straightforward postreadout decoding algorithm tailored to the SLHZ model. The proposed algorithm is an iterative hard-decision decoding

algorithm based on majority voting of generalized syndromes, known as Gallager’s bit-flipping (BF) algorithm, which has been derived in the context of classical LDPC codes [22,23]. Although the MWD is NP-hard, the BF algorithm is P, similar to the BP algorithms. Moreover, the BF algorithm requires less computational effort and processing time than the BP algorithm and can be implemented in digital logic. To demonstrate the BF algorithm, we have performed two simulations. First, assuming IID spin-flip noise, we have demonstrated that our BF decoding algorithm can recover the noisy output of the SLHZ model, eliminating spin-flip errors with performance comparable to that of the BP algorithm. Second, we have performed classical simulations of stochastically sampled spin readouts from the SLHZ model using a Markov-chain Monte Carlo (MCMC) sampler. This analysis allows us to investigate tolerance to leakage errors arising from dynamical and thermal excitations during the sampling. We show evidence that the BF decoding algorithm can efficiently eliminate spin-flip errors if we make the four-body penalty constraint sufficiently weak. This simulation suggests that a hybrid decoding approach, combining two algorithms—MCMC sampling followed by BF decoding—can be applied to a broader range of spin-flip noise and can mitigate the computational overhead of SLHZ systems. Our results are consistent with earlier work by Albash *et al.*, who employed simulated quantum annealing to simulate the readout of the QA device based on the SLHZ model (hereafter referred to as the SLHZ-based QA device) when BP decoding was used. We try to understand this result by comparing our algorithm with several extended BF decoding algorithms. According to our analysis, this is a consequence of the combination of the Hamiltonian of the SLHZ system and the properties of BF or BP decoding. We believe that our BF decoding is promising for realizing a near-term SLHZ-based QA device.

This paper is organized as follows. Section II explains the PE architecture, the SLHZ model, and the close connection between classical LDPC codes and the SLHZ model. Section III describes classical decoding of the SLHZ model based on Bayesian inference. We propose the BF algorithm as a simple straightforward decoding algorithm tailored for the SLHZ model. Section IV demonstrates the proposed BF algorithm under the IID and non-IID spin-flip noise, and shows the performance. In Sec. V, we compare several BF decoding algorithms and discuss why our two-stage hybrid decoding strategy tolerates errors originating from non-IID spin-flip noise. We explain why two-stage hybrid decoding outperforms the two algorithms individually. Section VI concludes this paper. We also discuss the relevance and consistency of our study to the earlier work by Albash *et al.* [24] in Appendix A. Supplementary results are also provided in Appendix B.

II. ERROR-CORRECTING CODES AND PARITY-ENCODING ARCHITECTURE

We define the PE architecture and explain the SLHZ model from the viewpoint of classical ECCs, and its connection to classical LDPC codes.

A. Notion

In this paper, we assumed a discrete communication system (Fig. 1). Suppose that a binary source word $\tilde{M}$ is first encoded into a binary code word $\tilde{C}$ using some ECC. The code word is modulated into the physical signal $\tilde{S}$ and transmitted. The signal is affected by noise during transmission through a transmission channel. Let $\tilde{M}$ and $\tilde{C}$ be K bits and $N_v (> K)$ bits, respectively, and denote them by the binary vectors $\tilde{\mathbf{Z}} = (\tilde{Z}_1, \dots, \tilde{Z}_K) \in \{0, 1\}^K$ and $\tilde{\mathbf{z}} = (\tilde{z}_1, \dots, \tilde{z}_{N_v}) \in \{0, 1\}^{N_v}$, respectively. A linear code is defined by a one-to-one map from the set $\{\tilde{\mathbf{Z}}\}$ of 2^K source words $\tilde{M}$ of length K to the set $\{\tilde{\mathbf{z}}\}$ of 2^K code words $\tilde{C}$ of length N_v . They are specified by a generating matrix $\mathbf{G}_{K \times N_v}$ or a generalized parity-check matrix $\mathbf{H}_{N_v \times N_v}$ satisfying $\mathbf{G}\mathbf{H}^T = \mathbf{0}_{K \times N_c} \pmod{2}$. Here, $\mathbf{G}$ and $\mathbf{H}$ are binary matrices (i.e., their elements are 0 or 1) and “mod 2” denotes that the multiplication is modulo 2. The source word $\tilde{\mathbf{Z}}$ is mapped to the code word $\tilde{\mathbf{z}}$ by $\tilde{\mathbf{z}} = \tilde{\mathbf{Z}}\mathbf{G} \pmod{2}$. Note that any K linearly independent code words can be used to form the generating matrix. For an arbitrary N_v -dimensional binary vector $\tilde{\mathbf{x}} = (\tilde{x}_1, \dots, \tilde{x}_{N_v}) \in \{0, 1\}^{N_v}$, define the generalized syndrome vector $\tilde{\mathbf{s}}(\tilde{\mathbf{x}}) = (\tilde{s}_1(\tilde{\mathbf{x}}), \dots, \tilde{s}_{N_c}(\tilde{\mathbf{x}}))$ (generalized because it may not have $N_v - K$ bits) as $\tilde{\mathbf{s}}(\tilde{\mathbf{x}}) = \tilde{\mathbf{x}}\mathbf{H}^T \pmod{2}$. Then, $\tilde{\mathbf{x}}$ is a code word if and only if $\tilde{\mathbf{s}}(\tilde{\mathbf{x}}) = \mathbf{0}_{1 \times N_c}$. This vector equation defines a set of generalized parity-check equations, consisting of a set of N_c equations where only $N_v - K$ ones of them are linearly independent. It follows that we can choose the parity-check matrix for a given linear code in many ways and that we can define many syndrome vectors for the same code. The ratio of the length of the source word to that of the code word is called the rate: $r = K/N_v$. Later in this section, we will show specific examples of the matrices $\mathbf{G}$ and $\mathbf{H}$ for the PE architecture. The code word $\tilde{\mathbf{z}}$ is converted into a sequence

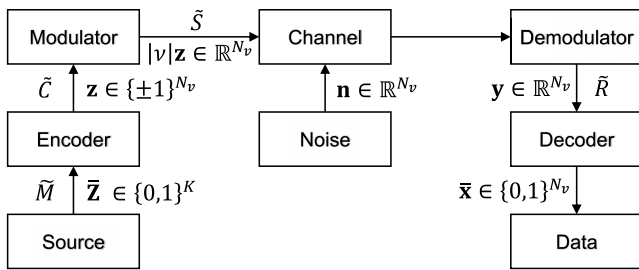


FIG. 1. A considered model for a communication system.

of bipolar variables $\tilde{\mathbf{z}} \rightarrow \mathbf{z} = (z_1, \dots, z_{N_v}) \in \{\pm 1\}^{N_v}$ (0 is mapped to +1 and 1 to -1) and assumed to be modulated to antipodal signals $|v|\mathbf{z}$ by a binary phase shift keying modulation, where $|v|$ is the signal amplitude. We assume that the signal $\tilde{S}$ is transmitted over a channel subject to additive white Gaussian noise (AWGN). At the end of the channel, the receiver obtains an observation $\tilde{R}$, denoted by an antipodal vector $\mathbf{y} = (y_1, \dots, y_{N_v}) = |v|\mathbf{z} + \mathbf{n} \in \mathbb{R}^{N_v}$, where $\mathbf{n} = (n_1, \dots, n_{N_v}) \in \mathbb{R}^{N_v}$ is a noise vector the elements of which are IID Gaussian random variables with zero mean and variance σ^2 . The purpose of ECC is to communicate reliably over a noisy channel. So, the goal of decoding is to reproduce the original source word $\tilde{M}$ or associated code word $\tilde{C}$ from the observation $\tilde{R}$.

B. Parity-encoding architecture and LDPC codes

We illustrate a PE architecture with concrete examples. The PE architecture corresponds to the classical ECC according to the following map: $\tilde{z}_{ij} = \tilde{Z}_i \oplus \tilde{Z}_j$ for $1 \leq i \leq j \leq k$. As a simple example, consider the case $K = 4$ and assume that $\tilde{\mathbf{Z}} = (\tilde{Z}_1, \tilde{Z}_2, \tilde{Z}_3, \tilde{Z}_4) \in \{0, 1\}^K$ and $\tilde{\mathbf{z}} = (\tilde{z}_{12}, \tilde{z}_{13}, \tilde{z}_{14}, \tilde{z}_{23}, \tilde{z}_{24}, \tilde{z}_{34}) \in \{0, 1\}^{\binom{K}{2}}$. Because each element of $\tilde{\mathbf{z}}$ is the binary sum of two elements in $\tilde{\mathbf{Z}}$, the generating matrix associated with this map is given by the $K \times \binom{K}{2}$ matrix:

$$\mathbf{G} = \begin{pmatrix} 1 & 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 \end{pmatrix}. \quad (1)$$

Note that there are two instances of 1 in each column in $\mathbf{G}$. Then, we can choose the following two parity-check matrices,

$$\mathbf{H} = \begin{pmatrix} 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 \end{pmatrix} \quad (2)$$

and

$$\mathbf{H}' = \begin{pmatrix} 1 & 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 \end{pmatrix}, \quad (3)$$

satisfying the constraints $\mathbf{G}\mathbf{H}^T = \mathbf{G}\mathbf{H}'^T = \mathbf{0}$. From these matrices, two different syndrome vectors can be derived. One is weight-4 syndrome vector $\tilde{\mathbf{s}}^{(4)}(\tilde{\mathbf{x}}) = (\tilde{s}_{1223}^{(4)}, \tilde{s}_{1234}^{(4)}, \tilde{s}_{2334}^{(4)}) = \tilde{\mathbf{x}}\mathbf{H}^T \in \{0, 1\}^{\binom{K-1}{2}}$ and the other is weight-3 syndrome vector $\tilde{\mathbf{s}}^{(3)}(\tilde{\mathbf{x}}) = (\tilde{s}_{123}^{(3)}, \tilde{s}_{124}^{(3)}, \tilde{s}_{134}^{(3)}, \tilde{s}_{234}^{(3)}) = \tilde{\mathbf{x}}\mathbf{H}'^T \in$

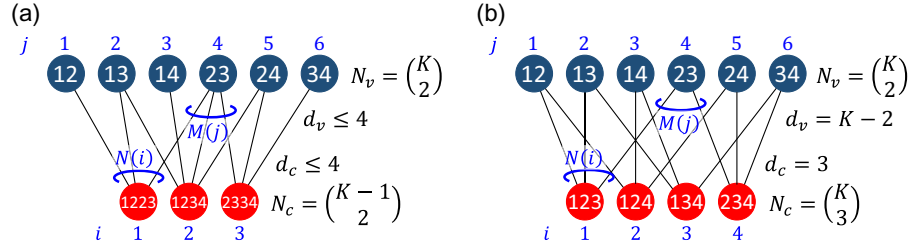


FIG. 2. Two examples of a bipartite graph for $K = 4$ logical spins. The dark blue circle with a label $\{k, l\}$ represents the spin variable x_{kl} , while the red circles with a label $\{k, l, m, n\}$ or $\{k, l, m\}$ represent the weight-4 syndrome $s_{klmn}^{(4)}$ and weight-3 syndrome $s_{klm}^{(3)}$, respectively. Let us relabel the variables with blue letters. An element of the code-word vector $\mathbf{x} = (x_1, \dots, x_{N_v}) \in \{\pm 1\}^{N_v}$ is called a variable node (VN). The i th syndrome of $\mathbf{x}$ is defined by $s_i(\mathbf{x}) = \prod_{k \in N(i)} x_k \in \{\pm 1\}$ and an element of vector $\mathbf{s}(\mathbf{x}) = (s_1(\mathbf{x}), \dots, s_{N_c}(\mathbf{x})) \in \{\pm 1\}^{N_c}$ is called a check node (CN), where the $N(i) = \{j : H_{ij}(H'_{ij}) = 1\}$ are the VNs adjacent to a CN i ($1 \leq i \leq N_c$) and the $M(j) = \{i : H_{ij}(H'_{ij}) = 1\}$ are the CNs adjacent to a VN j ($1 \leq j \leq N_v$). The column and row weights of the parity-check matrix are defined by $d_v(i) = |M(j)|$ and $d_c(i) = |N(i)|$, respectively. Table I shows the relevant parameters for your reference.

$\{0, 1\}^{\binom{K}{3}}$, where $\bar{\mathbf{x}} = (\bar{x}_{12}, \bar{x}_{13}, \bar{x}_{14}, \bar{x}_{23}, \bar{x}_{24}, \bar{x}_{34}) \in \{0, 1\}^{\binom{K}{2}}$ is an arbitrary binary vector. $\bar{\mathbf{s}}^{(3)}(\bar{\mathbf{x}})$ is weight-3 because each of its elements is written as a linear combination of three elements in $\bar{\mathbf{x}}$: $\bar{s}_{klm}^{(3)} = \bar{x}_{kl} \oplus \bar{x}_{lm} \oplus \bar{x}_{km}$, where $1 \leq k < l < m \leq K$. Similarly, $\bar{\mathbf{s}}^{(4)}(\bar{\mathbf{x}})$ is weight-4 because each of its elements is written as a linear combination of four elements in $\bar{\mathbf{x}}$ if $\bar{\mathbf{x}}$ is complemented by the fictitious components $\bar{x}_{ii}$ ($i = 2, \dots, K-1$) assigned by a fixed value 0: $\bar{s}_{klmn}^{(4)} = \bar{x}_{kl} \oplus \bar{x}_{lm} \oplus \bar{x}_{ln} \oplus \bar{x}_{kn}$. We note that any element in $\bar{\mathbf{s}}^{(4)}(\bar{\mathbf{x}})$ can be written as a linear combination of appropriate elements in $\bar{\mathbf{s}}^{(3)}(\bar{\mathbf{x}})$ and vice versa.

The connections between the variables $\bar{x}_{kl}$ and the weight-4 syndromes $\bar{s}_{klmn}^{(4)}$ are depicted by a sparse bipartite graph shown in Fig. 2(a). Similar connections between variables $\bar{x}_{kl}$ and the weight-3 syndromes $\bar{s}_{klm}^{(3)}$ are depicted in Fig. 2(b). The number of elements in $\bar{\mathbf{x}}$ is $N_v = \binom{K}{2}$, while the numbers of elements in $\bar{\mathbf{s}}^{(4)}(\bar{\mathbf{x}})$ and $\bar{\mathbf{s}}^{(3)}(\bar{\mathbf{x}})$ are $N_c = \binom{K-1}{2}$ and $N_c = \binom{K}{3}$, respectively. In the terminology of graph theory, the N_v elements in $\bar{\mathbf{x}}$ constitute variable nodes (VNs) and the N_c elements in $\bar{\mathbf{s}}^{(4)}(\bar{\mathbf{x}})$ or $\bar{\mathbf{s}}^{(3)}(\bar{\mathbf{x}})$ constitute check nodes (CNs). The matrix $\mathbf{H}$ or $\mathbf{H}'$ has N_c rows and N_v columns, where each row i represents a CN and each column j represents a VN; If the entry $H_{ij} = 1$, VN j is connected to CN i by an edge. Thus, each edge connecting a VN and a CN corresponds to an entry 1 in the row and column of $\mathbf{H}$ or $\mathbf{H}'$. Let d_c be the number of 1s in each row and d_v be the number of 1s in each column

of $\mathbf{H}$ or $\mathbf{H}'$. These numbers represent the number of edges connected to a VN and a CN, respectively, referred to as row and column weights. The matrix $\mathbf{H}'$ is regular, where the row weight $d_c = 3$ is common for all the CNs and the column weight $d_v = K - 2$ is common for all the VNs. In contrast, the matrix $\mathbf{H}$ is irregular and its weights depend on the associated nodes, which are at most 4.

If K is very large, we can always choose a sparse matrix $\mathbf{H}$ such that most of the entries are 0 and a few take value 1. Then, the PE architecture is closely connected to the LDPC codes. Only 2^K elements out of the $2^{\binom{K}{2}}$ possible $\bar{\mathbf{x}}$ are valid code words and satisfy the parity-check constraints $\bar{\mathbf{s}}^{(4)}(\bar{\mathbf{x}}) = \mathbf{0}_{1 \times \binom{K-1}{2}}$ or $\bar{\mathbf{s}}^{(3)}(\bar{\mathbf{x}}) = \mathbf{0}_{1 \times \binom{K}{3}}$. Conversely, the syndrome vector $\bar{\mathbf{s}}^{(n)}(\bar{\mathbf{x}})$ should be the all-zero vector, i.e., the vector the elements of which are all zero, for the associated vector $\bar{\mathbf{x}}$ to be a valid code word.

C. SLHZ model

In the remainder of this paper, we will discuss our arguments primarily in the language of spin glasses. Following Sourlas, our argument relies on isomorphism between the additive Boolean group $(\{0, 1\}, \oplus)$ and the multiplicative Ising group $(\{\pm 1\}, \cdot)$, where a binary variable $\bar{a}_i \in \{0, 1\}$ maps to the spin variable $a_i = (-1)^{\bar{a}_i} \in \{\pm 1\}$ and the binary sum maps to the product by $a_i a_j = (-1)^{\bar{a}_i \oplus \bar{a}_j} \in \{\pm 1\}$ [5–7]. Source word $\bar{\mathbf{Z}}$ and code word $\bar{\mathbf{z}}$ are mapped to vectors $\mathbf{Z} = (Z_1, \dots, Z_K) \in \{\pm 1\}^K$ and $\mathbf{z} = (z_1, \dots, z_{N_v}) \in \{\pm 1\}^{N_v}$ in the spin representation, respectively. Hereafter, we will refer to $\mathbf{Z}$ and $\mathbf{z}$ as a source state and a code state associated with source word $\bar{\mathbf{Z}}$ and code word $\bar{\mathbf{z}}$, respectively, and associated fictitious spins as logical and physical spins, respectively. Similarly, binary vectors $\bar{\mathbf{s}}$ and $\bar{\mathbf{x}}$ are mapped to the associated vectors $\mathbf{s}$ and $\mathbf{x}$ in the spin representation. In the following, we denote variables by symbols with and without an overbar in the

TABLE I. The parameters related to Fig. 2 for $4 \leq K \leq 7$.

K	$\binom{K}{2}$	$\binom{K-1}{2}$	$\binom{K}{3}$	$K - 2$
4	6	3	4	2
5	10	6	10	3
6	15	10	20	4
7	21	15	35	5

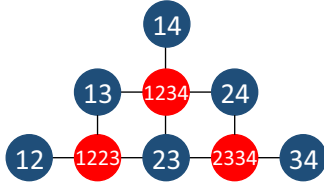


FIG. 3. The bipartite graph that is topologically equivalent to Fig. 2(a). This graph avoids any edge crossings.

binary and spin representations. It should be noted that by isomorphism, every addition of two binary variables corresponds to a unique product of spin variables and vice versa. For example, since $\bar{z} = \bar{\mathbf{Z}}\mathbf{G} \pmod{2}$ holds, $\mathbf{z}$ and $\mathbf{Z}$ are connected by the relation

$$z_i = (-1)^{\bigoplus_{j=1}^K \bar{z}_j G_{ji}} = \prod_{\{j:G_{ji}=1\}} Z_j \in \{\pm 1\}, \quad (4)$$

where $i = 1, \dots, N_v$. Similarly, $\mathbf{z}$ must satisfy the equation

$$(-1)^{\bigoplus_{j=1}^{N_v} \bar{z}_j H_{ij}} = \prod_{\{j:H_{ij}=1\}} z_j = (-1)^0 = +1, \quad (5)$$

for $i = 1, \dots, N_c$, if $\bar{\mathbf{z}}$ satisfies the parity-check equation $\bar{\mathbf{z}}\mathbf{H}^T \pmod{2} = \mathbf{0}_{1 \times N_c}$.

Now, let us consider the graph in Fig. 3, which is topologically equivalent to Fig. 2(a). This graph is simply the SLHZ model [10]. This implies that the SLHZ model is essentially a PE-based model that utilizes the weight-4 syndrome, as viewed from the perspective of the spin-glass model. The plaquette referred to by LHZ [10] and the stabilizer by Rocchetto, Benjamin, and Li [25] correspond to the weight-4 syndrome from the perspective of the classical ECCs. Each of the elements is given by a product of four elements in $\mathbf{x}$ if $\mathbf{x}$ is complemented by the fictitious spin variables x_{ii} ($i = 2, \dots, K-1$) with fixed value $+1$: $s_{klmn}^{(4)}(\mathbf{x}) = x_{km}x_{lm}x_{ln}x_{kn}$ [10]. The vector $\mathbf{s}^{(n)}(\mathbf{x})$ should be an all-one vector, i.e., the vector the elements of which are all one, for the associated state $\mathbf{x}$ of the physical spin to be a valid code state.

III. POSTREADOUT DECODING OF THE SLHZ MODEL

In this section, we consider postreadout decoding of the SLHZ model from the perspective of the classical ECCs. We first present a general probabilistic framework for classical decoding of ECCs. Next, we propose a simple straightforward decoding algorithm tailored for the SLHZ model.

A. Probabilistic decoding

Postreadout decoding can be performed according to Bayesian inference. Consider the conditional probability

$P(\mathbf{z}|\mathbf{y}) d\mathbf{y}$ that the prepared code state is $\mathbf{z}$ when the observation was between $\mathbf{y}$ and $\mathbf{y} + d\mathbf{y}$. According to the Bayes theorem,

$$P(\mathbf{z}|\mathbf{y}) = \frac{P(\mathbf{y}|\mathbf{z})P(\mathbf{z})}{\sum_{\mathbf{z}} P(\mathbf{y}|\mathbf{z})P(\mathbf{z})} = \kappa P(\mathbf{y}|\mathbf{z})P(\mathbf{z}) \quad (6)$$

holds, where κ is a constant independent of $\mathbf{z}$, which is determined by the normalization condition $\sum_{\mathbf{z}} P(\mathbf{z}|\mathbf{y}) = 1$, and $P(\mathbf{z})$ is the prior probability for the code state $\mathbf{z}$. According to Bayesian inference, we can consider the state that maximizes the conditional probability $P(\mathbf{z}|\mathbf{y})$, which offers the most probable word (“word maximum *a posteriori* probability” or “word-MAP” decoding). Alternatively, we can consider the state that maximizes its marginals, which offers the most probable symbol (“symbol maximum *a posteriori* probability” or “symbol-MAP” decoding).

1. Word-MAP decoding

We follow the argument given by Sourlas, who first pointed out the link between the ECCs and the spin-glass model [4]. Sourlas showed that, based on Eqs. (4) and (5), two different formulations are possible for word-MAP decoding of the same ECC. The first formulation is given in terms of source state $\mathbf{Z}$. In this formulation, we assume the following prior probability for $\mathbf{z}$:

$$P(\mathbf{z}) = \mu \prod_{i=1}^{N_v} \delta\left(z_i, \prod_{\{j:G_{ji}=1\}} Z_j\right), \quad (7)$$

where μ is a normalization constant. The Kronecker δ ’s in Eq. (7) enforce the constraint that $\mathbf{z}$ obeys Eq. (4); it is a valid code state. Assuming that the noise is independent for each spin and that $P(\mathbf{y}|\mathbf{z}) = \prod_{i=1}^{N_v} P(y_i|z_i)$ holds (memoryless channel), we can derive the following equation:

$$\begin{aligned} -\ln P(\mathbf{z}|\mathbf{y}) &= \text{constant} - \sum_{i=1}^{N_v} B_i \prod_{\{j:G_{ji}=1\}} Z_j \\ &\equiv H^{\text{source}}(\mathbf{Z}), \end{aligned} \quad (8)$$

where

$$B_i = B_i(y_i) = \frac{1}{2} \log \frac{P(y_i|z_i = +1)}{P(y_i|z_i = -1)} \quad (9)$$

is the half log-likelihood ratio (LLR) for the channel observation y_i [5]. The vector $\mathbf{B} = (B_1, \dots, B_{N_v}) \in \mathbb{R}^{N_v}$ contains all the information about the observation $\mathbf{y}$. In this

paper, we focus on the communication through the AWGN channel. The likelihood is given as

$$P(y_i|z_i) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left[-\frac{(y_i - |v|z_i)^2}{2\sigma^2}\right], \quad (10)$$

where $|v|$ and σ^2 are the amplitude of the prepared signal and the variance of the common Gaussian noise. Then, the LLR is given by $B_i = \frac{1}{2}\beta y_i$, where $\beta = 2|v|/\sigma^2 > 0$ is called the channel reliability factor, and its inverse β^{-1} corresponds to the temperature in the language of spin glasses. Note that $\frac{1}{2}(|v|/\sigma)^2$ is the signal-to-noise ratio (SNR) of the AWGN channel. Thus, B_i is proportional to the magnitude of the channel observation y_i for the AWGN channel. It is evident that $H^{\text{source}}(\mathbf{Z})$ is in the form of a spin-glass Hamiltonian, where $\frac{1}{2}y_i$ is identified with the coupling constant J_i . In this formulation, the word-MAP decoding corresponds to finding the ground state of the Hamiltonian $H^{\text{source}}(\mathbf{Z})$.

Alternatively, the second formulation is given in terms of code state $\mathbf{z}$. In this formulation, according to Eq. (5), we assume the following prior probability for $\mathbf{z}$:

$$P(\mathbf{z}) = \mu \prod_{i=1}^{N_c} \delta(s_i(\mathbf{z}), +1), \quad (11)$$

where

$$s_i(\mathbf{z}) = \prod_{\{j: H_{ij}=1\}} z_j \quad (12)$$

is the i th syndrome for $\mathbf{z}$ in the spin representation. Then, we can derive the following equation:

$$\begin{aligned} -\ln P(\mathbf{z}|\mathbf{y}) &= -\sum_{i=1}^{N_b} B_i z_i + \lim_{\gamma \rightarrow \infty} \gamma \sum_{i=1}^{N_c} \frac{1 - s_i(\mathbf{z})}{2} \\ &\equiv H^{\text{code}}(\mathbf{z}). \end{aligned} \quad (13)$$

In Eq. (13), the δ 's in Eq. (11) are replaced by a soft constraint using the identity

$$\delta(x, +1) = \lim_{\gamma \rightarrow \infty} \exp\left[-\gamma \frac{1-x}{2}\right]. \quad (14)$$

The second term of Eq. (13) enforces the parity constraint that $\mathbf{z}$ should obey Eq. (5) to be a valid code state. In contrast, the first term captures the correlation between the observation $\mathbf{y}$ and the code state $\mathbf{z}$. The latter formulation is equivalent to the former one as long as $\mathbf{z}$ is a valid code state associated with a source state $\mathbf{Z}$, i.e., $s_i(\mathbf{z}) = 1$ for any i . We note that $H^{\text{code}}(\mathbf{z})$ can be considered a Hamiltonian

of an enlarged spin system. In this formulation, the word-MAP decoding corresponds to finding the most probable code state $\mathbf{z}$, namely,

$$\mathbf{z} = \arg \max_{\mathbf{x} \in C} P(\mathbf{x}|\mathbf{y}) = \arg \min_{\mathbf{x} \in C} H^{\text{code}}(\mathbf{x}), \quad (15)$$

where C denotes the set of all code states. Such decoded results can be obtained by, e.g., SA and/or QA.

Now, let us recall the SLHZ model. In this case, $s_i(\mathbf{z})$ is given by the weight-4 syndrome in the spin representation, which is written as a product of four elements in $\mathbf{z}$, i.e., $s_{klmn}^{(4)}(\mathbf{z}) = z_{km}z_{lm}z_{ln}z_{kn}$ [10]. Then, we see that $H^{\text{code}}(\mathbf{z})$ agrees with the Hamiltonian of the SLHZ model. Therefore, the word MAP decoding of the LDPC codes is equivalent to finding the ground state of the SLHZ model. Furthermore, since the two formulations, i.e., those based on $H^{\text{code}}(\mathbf{z})$ [Eq. (13)] and those based on $H^{\text{source}}(\mathbf{Z})$ [Eq. (8)] are mathematically equivalent, it follows that finding the ground code state of the SLHZ model is equivalent to finding the ground state of the spin glass.

2. Symbol-MAP decoding

The above discussion implies that we can solve the COPs through decoding the associated classical LDPC codes. Meanwhile, the word-MAP decoding is not the only strategy for decoding classical LDPC codes. There is another decoding strategy based on different principles. Instead of considering the most probable word, it is also allowed to be interested only in the most probable symbol, i.e., the most probable value z_i of the i th spin, ignoring the values of the other spin variables. In this strategy, we consider marginals,

$$\begin{aligned} P(x_i|\mathbf{y}) &:= \sum_{x_1=-1}^{+1} \cdots \sum_{x_{i-1}=-1}^{+1} \sum_{x_{i+1}=-1}^{+1} \cdots \sum_{x_{N_b}=-1}^{+1} P(\mathbf{x}|\mathbf{y}) \\ &= \sum_{x_k (k \neq i)} P(\mathbf{x}|\mathbf{y}). \end{aligned} \quad (16)$$

Decoding corresponds to finding the value of the spin variable that maximizes the marginals $P(x_i|\mathbf{y})$, i.e.,

$$\begin{aligned} z_i &= \arg \max_{x_i \in \{\pm 1\}} P(x_i|\mathbf{y}) \\ &= \text{sign}[P(x_i = +1|\mathbf{y}) - P(x_i = -1|\mathbf{y})] \\ &= \text{sign}[L_i(\mathbf{y})], \end{aligned} \quad (17)$$

where

$$L_i(\mathbf{y}) = \log \frac{P(x_i = +1|\mathbf{y})}{P(x_i = -1|\mathbf{y})} \quad (18)$$

is called the *a posteriori* LLR for the i th spin. Since the absolute value $|L_i(\mathbf{y})|$ gives the error probability of this

decision as

$$\epsilon_i = \frac{1}{1 + e^{|\tilde{L}_i(\mathbf{y})|}} = \begin{cases} P(x_i = -1|\mathbf{y}), & \lambda_i \geq 0, \\ P(x_i = +1|\mathbf{y}), & \lambda_i < 0, \end{cases} \quad (19)$$

it represents a metric to measure the reliability or uncertainty of a decision in Eq. (17), which indicates how likely x_i it is to be $+1$ or -1 . A value close to zero indicates a bad inference, while a larger value indicates a better inference [26].

A variety of algorithms can achieve symbol-MAP decoding. For example, the BP algorithm is widely regarded as the best-known algorithm for LDPC codes [21]. The BP algorithm is an iterative deterministic algorithm in which the *a posteriori* LLRs, initially given by the observation $\mathbf{y}$, gradually increase in absolute value as messages are iteratively exchanged between the VNs and CNs.

In the following section, we study an alternative, simple, and straightforward algorithm: the BF algorithm [22, 23]. This algorithm is a variety of the symbol-MAP decoding strategies. It uses the syndrome vector $\mathbf{s}(\mathbf{x}) = (s_1(\mathbf{x}), \dots, s_{N_c}(\mathbf{x}))$ as a key to decide the most probable value z_i of the i th spin, namely,

$$z_i = \arg \max_{x_i \in \{\pm 1\}} P(x_i|\mathbf{y}) = \text{sign}[\tilde{L}_i(\mathbf{x})], \quad (20)$$

where $\mathbf{x} = \text{sign}[\mathbf{y}] \in \{\pm 1\}^{N_v}$ is the hard decision of the observation $\mathbf{y}$ and

$$\tilde{L}_i(\mathbf{x}) = \log \frac{P(z_i = +1|\mathbf{B}_i(\mathbf{x}))}{P(z_i = -1|\mathbf{B}_i(\mathbf{x}))} \quad (21)$$

is the associated *a posteriori* LLR for the i th spin, where $\mathbf{B}_i(\mathbf{x})$ is the estimator of z_i , which depends on $\mathbf{s}(\mathbf{x})$, as shown later. The algorithm starts with an initial decision $\mathbf{x}^{(0)} = \text{sign}[\mathbf{y}]$ and iteratively updates the decision $x_i^{(m)} = \text{sign}[\tilde{L}_i(\mathbf{x}^{(m-1)})]$ by a spin-flip operation, which depends on the decision $\mathbf{x}^{(m-1)} = (x_1^{(m-1)}, \dots, x_{N_v}^{(m-1)})$ in the previous round. If the decision $\mathbf{x}$ converges after several rounds of iteration, we consider it the most probable code state $\mathbf{z}$.

In the BF algorithm, the reliability of the decision $|\tilde{L}_i(\mathbf{x})|$ gradually increases as the spin flipping is repeated. It is widely recognized that one of the key advantages of the BF algorithm over the BP algorithm is its simplicity and lower computational complexity. The BF algorithm is particularly useful in environments with limited computational resources because it requires fewer operations to decode the received message. This fact makes it a more efficient option when processing power is a concern. In contrast, although it provides better error-correcting performance in terms of the bit error rate (BER), the BP algorithm typically involves more complex calculations and higher computational requirements. Therefore, the BF decoding is often favored in communication applications where simplicity and low complexity are prioritized.

B. Readout decoding of the SLHZ model

1. Bit-flipping decoding algorithm

Now, let us analyze readout decoding of the SLHZ model from the perspective of the LDPC codes. In the following analysis, we assume that the readout $\mathbf{r} = (r_{12}, \dots, r_{K-1K}) \in \{\pm 1\}^{\binom{K}{2}}$ is available for the $\binom{K}{2}$ physical spins in the SLHZ model in accordance with the Ising model. We disregard the information about the observations $\mathbf{y}$ of the physical spins, even if it is available in the actual implementation. We will revisit this point later.

We first consider the simplest case, i.e., correcting errors caused by IID noise. This model is appropriate for an SLHZ-based QA device when measurement errors dominate other readout errors. We propose a simple decoding algorithm with very low decoding complexity, which is associated with Gallager's BF decoding algorithm in the spin representation [22, 23]. It is based on Massey's *a posteriori* probability (APP) algorithm [26]. Our algorithm only uses information about the syndrome vector $\mathbf{s}(\mathbf{r})$ to estimate errors $\mathbf{e}$ in the current decision $\mathbf{r}$ for the readout. Let us introduce the error pattern by $\mathbf{e} = \mathbf{r} \circ \mathbf{z} = (e_{12}, \dots, e_{K-1K}) \in \{\pm 1\}^{\binom{K}{2}}$, where $\circ$ denotes component-wise multiplication. The syndrome vector depends only on $\mathbf{e}$ because

$$\mathbf{s}(\mathbf{z}) = \left(\prod_{\{j:H_{1j}=1\}} z_j, \dots, \prod_{\{j:H_{N_c j}=1\}} z_j \right) = (+1, \dots, +1) \quad (22)$$

holds for any code state $\mathbf{z}$. Therefore, if we note that $z_{ij} \in \{\pm 1\}$ and $e_{ij} \in \{\pm 1\}$,

$$\mathbf{s}(\mathbf{r}) = \mathbf{s}(\mathbf{z} \circ \mathbf{e}) = \mathbf{s}(\mathbf{z}) \circ \mathbf{s}(\mathbf{e}) = \mathbf{s}(\mathbf{e}) \quad (23)$$

holds. It is important to note that although we never know $\mathbf{e}$, we can know $\mathbf{s}(\mathbf{e})$ from the knowledge of the decision $\mathbf{r}$ and utilize it to estimate errors $\mathbf{e}$ to be eliminated.

Estimating the correct value z_{ij} is equivalent to estimating error e_{ij} , as $z_{ij} = r_{ij} e_{ij}$ holds. The algorithm uses the weight-3 syndrome $s_{ijk}^{(3)}(\mathbf{r}) = r_{ij} r_{jk} r_{ik} = e_{ij} e_{jk} e_{ik}$ rather than the weight-4 syndrome to decode the current decision $\mathbf{r}$ because it has the advantage of treating all variables (VN and CN) symmetrically [20]. The trade-off is that more constraints may increase the decoding complexity. However, as will be discussed later, this is not a fatal problem, as we can leverage symmetry to reduce computational costs. An optimal estimate e_{ij}^* can be obtained from $N_v = \binom{K}{2}$ syndromes $s_{ijk}^{(3)}(\mathbf{r})$ for $k \neq i, j$. They are considered $K - 2$ parity checks orthogonal on e_{ij} [26] and their values suggest whether the parity-check equation is satisfied (being $+1$) or violated (being -1). Based on the majority voting, we decide an error variable e_{ij}^* whether

the spin ij should be flipped ($e_{ij}^* = -1$) or not ($e_{ij}^* = +1$). The optimal estimate e_{ij}^* can be obtained according to the APP decoding by [26]

$$e_{ij}^* = \text{sign}[\Delta_{ij}(\mathbf{r})], \quad (24)$$

where

$$\Delta_{ij}(\mathbf{r}) = w_0 + \sum_{k \neq i,j}^K w_k s_{ijk}^{(3)}(\mathbf{r}) \quad (25)$$

is called the inversion function [27,28], w_0 and w_k are given by

$$w_0 = \log \frac{1 - \gamma_{ij}}{\gamma_{ij}}, \quad (26)$$

$$w_k = \log \frac{1 - p_k}{p_k}, \quad (27)$$

with

$$\gamma_{ij} = P(1 \neq e_{ij}) = P(e_{ij} \neq 1), \quad (28)$$

$$p_k = P(s_{ijk}^{(3)}(\mathbf{r}) \neq \hat{e}_{ij}) = P(e_{jk}e_{ik} \neq 1). \quad (29)$$

The decision r_{ij}^* can be written as

$$r_{ij}^* = r_{ij} e_{ij}^* = \text{sign}[L_{ij}^*(\mathbf{r})], \quad (30)$$

where

$$L_{ij}^*(\mathbf{r}) = r_{ij} \Delta_{ij}(\mathbf{r}) = \log \frac{P(r_{ij}^* = +1 | \mathbf{B}_{ij}(\mathbf{r}))}{P(r_{ij}^* = -1 | \mathbf{B}_{ij}(\mathbf{r}))}, \quad (31)$$

$$\begin{aligned} \mathbf{B}_{ij}(\mathbf{r}) &= (B_{ij0}(\mathbf{r}), B_{ij1}(\mathbf{r}), \dots, B_{ijK}(\mathbf{r})) \\ &= r_{ij} (1, s_{ij1}^{(3)}(\mathbf{r}), \dots, s_{ijK}^{(3)}(\mathbf{r})), \end{aligned} \quad (32)$$

since

$$\gamma_{ij} = P(r_{ij} \neq r_{ij}^*) = P(B_{ij0}(\mathbf{r}) \neq r_{ij}^*), \quad (33)$$

$$p_k = P(r_{ij} s_{ijk}^{(3)}(\mathbf{r}) \neq r_{ij}^*) = P(B_{ijk}(\mathbf{r}) \neq r_{ij}^*) \quad (34)$$

holds [26,29]. The function $L_{ij}^*(\mathbf{r})$ can be considered as the LLR for the decision r_{ij}^* based on the estimators $\mathbf{B}_{ij}(\mathbf{r})$, which depends on the weight-3 syndrome vector $\mathbf{s}^{(3)}(\mathbf{r})$. In Eq. (24), $e_{ij}^* = -1$ indicates that we should invert the sign of $r_{ij} \rightarrow r_{ij}^* = r_{ij} e_{ij}^* = -r_{ij}$. Parameters γ_{ij} and p_k are crosstalk parameters that characterize the binary symmetric channels shown in Fig. 4. We should note that the probability p_k is given by the probability of an odd number of -1 s among the errors exclusive of e_{ij} that are checked by $s_{ijk}^{(3)}(\mathbf{r}) = e_{ij} e_{jk} e_{ik}$, so that it is given by [26]

$$p_k = P(e_{jk}e_{ik} = -1) = \frac{1}{2} (1 - (1 - 2\gamma_{jk})(1 - 2\gamma_{ik})). \quad (35)$$

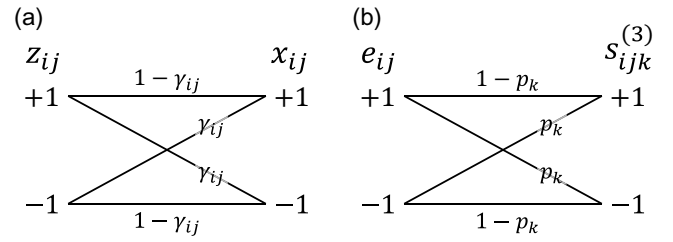


FIG. 4. The binary symmetric channel associated with the parameters γ_{ij} and p_k .

If we assume IID noise, such that $\gamma_{ij} = \gamma_0$ holds for every $\{i,j\}$ where $0 < \gamma_0 < \frac{1}{2}$, we can confirm that w_0 can be approximated $w_0 = w_k > 0$ for any k . Then, we obtain the following simple algorithm: we calculate the best estimate e_{ij}^* by

$$e_{ij}^* = \text{sign} \left(1 + \sum_{k \neq i,j}^K s_{ijk}^{(3)}(\mathbf{r}) \right). \quad (36)$$

This equation means that if the majority vote of the estimators given by $\mathbf{A}_{ij}(\mathbf{r}) = (1, s_{ij1}^{(3)}(\mathbf{r}), \dots, s_{ijK}^{(3)}(\mathbf{r}))$ is negative, r_{ij} should be flipped to increase the value of $\sum_{k \neq i,j}^K s_{ijk}^{(3)}(\mathbf{r})$. Thus, Eq. (36) is reduced to the following best estimate $r_{ij}^* \in \{\pm 1\}$:

$$r_{ij}^* = r_{ij} e_{ij}^* = \text{sign} \left(r_{ij} + \sum_{k \neq i,j}^K r_{jk} r_{ki} \right). \quad (37)$$

Equation (37) can be interpreted as Gallager's BF decoding [22,23]. Note that the best estimate r_{ij}^* is represented in terms of the values of the current variables in VNs. In Fig. 5, we show the relevant bipartite graph for $K = 5$. Note that the graph associated with the SLHZ system is

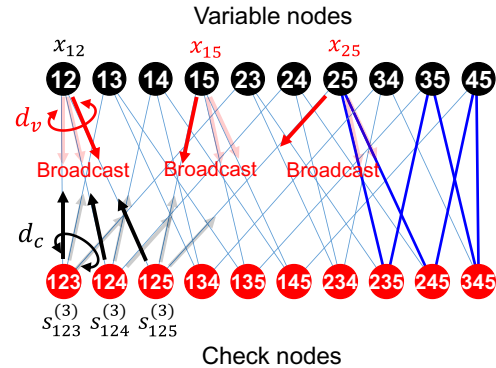


FIG. 5. The bipartite graph for $K = 5$ logical spins. The solid blue lines show an example of the shortest loop of edges connecting VNs and CNs, which has a length of 6.

more loopy, with a minimum length of 4, whereas this graph is less loopy, with a minimum length of 6. The initial decision $\mathbf{r}$ on the VNs is broadcast to its adjacent CNs connected by edges. Each CN then reports the syndrome value to its adjacent VNs connected by the edge. All VNs update their values simultaneously according to Eq. (36), representing a majority vote of 1 and the values of adjacent CNs.

2. Application to the SLHZ system

Next, let us discuss how to apply the BF decoding to the SLHZ model. We introduce a matrix representation of the state of the SLHZ model. Consider the $K \times K$ symmetrized matrix $\hat{\mathbf{z}}$, the elements of which are given by spin variable $z_{ij} = z_{ji} \in \{\pm 1\}$ and have unit diagonal elements, namely,

$$\hat{\mathbf{z}} = \begin{bmatrix} 1 & z_{12} & z_{13} & \cdots & z_{1K-1} & z_{1K} \\ z_{21} & 1 & z_{23} & \cdots & z_{2K-1} & z_{2K} \\ z_{31} & z_{32} & 1 & \cdots & z_{3K-1} & z_{3K} \\ \vdots & \vdots & \vdots & \ddots & \cdots & \vdots \\ z_{K-11} & z_{K-12} & z_{K-13} & \cdots & 1 & z_{K-1K} \\ z_{K1} & z_{K2} & z_{K3} & \cdots & z_{KK-1} & 1 \end{bmatrix}. \quad (38)$$

We use similar notations, $\hat{\mathbf{r}}$ and $\hat{\mathbf{e}}$, for the symmetrized matrices associated with the current readout $\mathbf{r}$ and error pattern $\mathbf{e}$. They satisfy $\hat{\mathbf{r}} = \hat{\mathbf{z}} \circ \hat{\mathbf{e}}$. Hereafter, we will refer to $\hat{\mathbf{e}}$ as the error matrix. Then, Eq. (37) can be conveniently written as

$$\hat{\mathbf{r}}^* = \mathcal{F}(\hat{\mathbf{r}}) = \text{sign}[\hat{\mathbf{r}}(\hat{\mathbf{r}} - \mathbf{I}_{K \times K})], \quad (39)$$

where we assume that the sign function is component-wise. This operation updates the $\binom{K}{2}$ elements in $\hat{\mathbf{r}}$ simultaneously. Therefore, Eq. (39) represents a parallel BF algorithm. We further consider the iterative operation of $\mathcal{F}$, i.e.,

$$\hat{\mathbf{r}}^{(n)} = \hat{\mathbf{r}} \circ \hat{\mathbf{e}}^{(n)} = \mathcal{F}^{(n)}(\hat{\mathbf{r}}). \quad (40)$$

If $\hat{\mathbf{r}}^{(n)} \rightarrow \hat{\mathbf{z}}$, or in other words, $\hat{\mathbf{e}}^{(n)} \rightarrow \hat{\mathbf{1}}_{K \times K}$ at $n = n_0$, where $\hat{\mathbf{1}}_{K \times K}$ is $K \times K$ matrix the entries of which are all one (all-one matrix), we say that the decoding has succeeded after n_0 iterations of the BF decoding. In the case of success, it follows that $s_{ijk}^{(3)}(\hat{\mathbf{r}}^{(n)}) = 1$ for a set of possible $\{i, j, k\}$ at $n = n_0$. Our algorithm can be viewed as an iterative process that gradually increases the number of syndromes with unit values by flipping spins according to the majority vote of the syndromes associated with each spin.

IV. EXPERIMENTAL DEMONSTRATION

We demonstrate that the BF algorithm is valid for eliminating readout errors in the SLHZ model arising not only from IID noise but also from non-IID noise.

A. In the case of IID noise

The performance of the BF algorithm has been investigated under the assumption of IID noise, the simplest model of noisy readout in QA. Pastawski and Preskill have previously studied this model using the BP algorithm [20]. Note that the BF decoding in Eq. (36) depends only on the weight-3 syndromes, which treat all variables (VN and CN) symmetrically [20]. Furthermore, if we note that the LDPC codes are linear codes and assume that the AWGN channel is symmetric, i.e., $P(y_i|z_i) = P(-y_i|-z_i)$ holds, the probability of successful decoding is independent of the input code state $\mathbf{z}$ [30]. In this case, without loss of generality, the performance of decoding can be analyzed using the assumption that the all-one code state $\mathbf{z} = (+1, \dots, +1)$ has been transmitted [30,31]. In physics, this assumption corresponds to choosing the ferromagnetic gauge and identifying the current decision $\mathbf{r}$ with an error pattern $\mathbf{e}$. Thus, we assume that the input is the all-one code state, i.e., $\hat{\mathbf{z}} = \hat{\mathbf{1}}_{K \times K}$. All the demonstrations have been performed using the *Mathematica*® Ver.14 platform on the Windows 11 operating system. We have generated 5000 symmetric matrices $\hat{\mathbf{r}} = \hat{\mathbf{e}}$ with unit diagonal elements and other elements randomly assigned -1 with probability $\varepsilon < \frac{1}{2}$ and $+1$ otherwise. After $n = 5$ iterations, we have checked whether the errors have been eliminated. In Fig. 6(a), we show the performance of the BF algorithm, plotting the probability of decoding failure as a function of the number K of logical spins ranging from 2 to 40 (the associated SLHZ model consists of $\binom{K}{2}$ physical spins) for seven values of common bit error rate ε ($= 0.05, 0.07, 0.1, 0.15, 0.2, 0.3, 0.4$). The MATLAB code implementing the described BF decoder and producing the figure is included as Supplemental Material [32]. The failure probability falls steeply as N increases if ε is not too close to the threshold value $1/2$. A similar performance calculation has been reproduced for the decoding using the BP algorithm after being iterated 5 times, as shown in Fig. 6(b) [20]. Comparing these figures shows that the performance of the BF algorithm is comparable to that of the BP algorithm. Note that the BP algorithm updates all marginal probabilities $P(x_i|\mathbf{y})$ associated with the $\binom{K}{2}$ spins sequentially by passing a real-valued message between the associated VNs and CNs per a single iteration. In contrast, in the BF algorithm, all the $\binom{K}{2}$ variables r_{ij} in the matrix $\hat{\mathbf{r}}$ are updated in parallel per a single iteration. Thus, each spin variable was updated 5 times in these algorithms. Figure 7 is an typical example for a successfully decoded result when $K = 40$ and $\varepsilon = 0.3$.

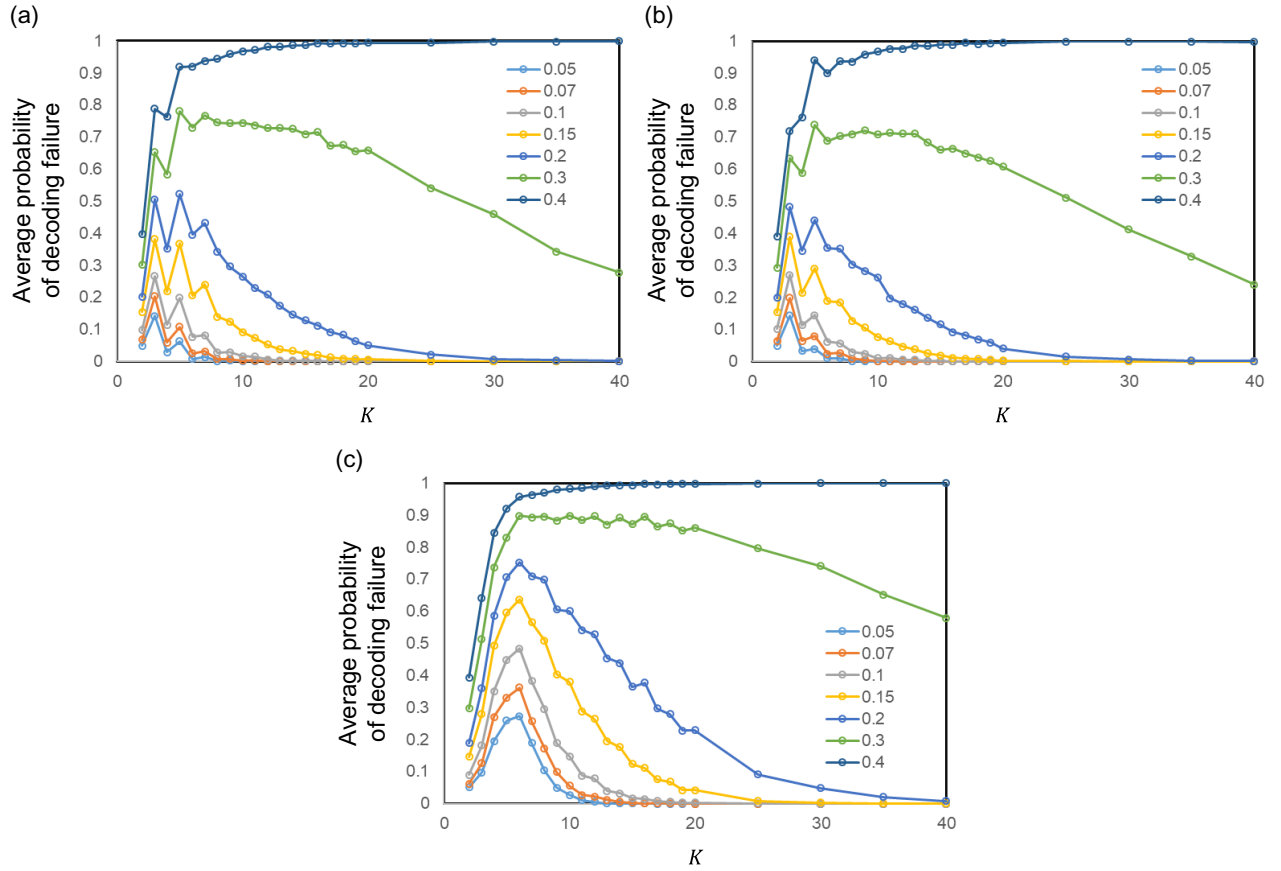


FIG. 6. A comparison of the performance of the (a) bit-flipping (BF), (b) belief-propagation (BP), and (c) Markov-chain Monte Carlo (MCMC) decodings. Assuming that the error probability ε is common to all physical spins, the average probabilities of decoding failure are plotted as a function of the number K of logical spins (associated with $\binom{K}{2}$ physical spins) for seven values of ε . Each data point has been obtained by averaging over 5000 error-matrix realizations. The BF and BP algorithms have been iterated 5 times for each realization and the MCMC sampling has been iterated $\binom{K}{2}$ times. We have considered a tie in the majority voting to be a failure in the BF decoding, although we can resolve it by introducing a tie-breaking rule, such as coin tossing.

In this figure, each entry of the error matrix $\hat{\mathbf{e}}$ is plotted after $n = 1, \dots, 4$ rounds of iteration of the operation in Eq. (39). The blue pixels correspond to spins with error ($e_{ij} = -1$), the number of which gradually decreases as rounds of iteration are added. In this example, an error-free matrix has been obtained after $n = 3$ iterations. Such

results have been observed for more than 70 % of the error matrices generated.

In addition to the BF and BP algorithms, let us focus on the word-MAP decoding introduced in Sec. III A and compare its performance with the above two algorithms. The word-MAP decoding says that any code state is the

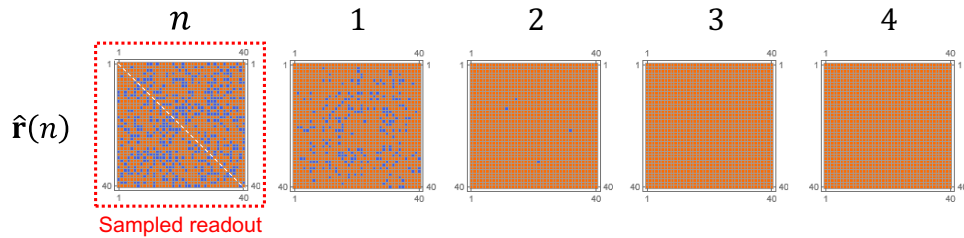


FIG. 7. A typical example of successfully decoded results by the BF decoding when $K = 40$ and $\varepsilon = 0.3$. The initial readout has been generated according to the IID-noise model. The estimated matrix $\hat{\mathbf{r}}$ is plotted from left to right in increasing order of iterations n . The blue pixels represent spins with error.

ground state of the Hamiltonian, as shown in Eq. (13),

$$H^{\text{code}}(\hat{\mathbf{x}}) \equiv \gamma \sum_{\{i,j,k\}} \frac{1 - s_{ijk}^{(3)}(\hat{\mathbf{x}})}{2}, \quad (41)$$

where $\hat{\mathbf{x}} \in \{\pm 1\}^{K \times K}$ is the matrix representing the general state of the SLHZ system. In general, $H^{\text{code}}(\hat{\mathbf{x}}) \geq 0$ and $H^{\text{code}}(\hat{\mathbf{x}}) = 0$ if and only if $\hat{\mathbf{x}}$ is a code state. In Eq. (41), the correlation term (the first term in Eq. (13) has been omitted. This is because the all-one code-state assumption uses no information about the observation $\mathbf{y}$. Under this assumption, the ground state of $H^{\text{code}}(\hat{\mathbf{x}})$ is given by

$$\hat{\mathbf{z}} = \arg \min_{\hat{\mathbf{x}} \in \{\pm 1\}^{K \times K}} H^{\text{code}}(\hat{\mathbf{x}}) = \hat{\mathbf{1}}_{K \times K}. \quad (42)$$

We have searched the ground state $H^{\text{code}}(\hat{\mathbf{x}})$ using the classical MCMC sampler, which is equivalent to a word-MAP decoding. We refer to it as the MCMC decoding hereafter. The performance of MCMC decoding has been evaluated based on $H^{\text{code}}(\hat{\mathbf{x}})$ in Eq. (41). We have generated 5000 random symmetric error matrices $\hat{\mathbf{e}}$ with a bit error rate ε . We have used rejection-free MCMC sampling, in which all self-loop transitions are removed from the standard MCMC [33]. We have sampled a sequence $\{\hat{\mathbf{x}}\}$, each of which was of a size $\binom{K}{2}$, starting from the initial state $\hat{\mathbf{x}} = \hat{\mathbf{e}}$. The hyperparameter γ has been assigned $\gamma \approx 1$, which has been found to be experimentally optimal. We have evaluated the average probability that the ground state $\hat{\mathbf{z}} = \hat{\mathbf{1}}_{K \times K}$ was found in the sequence of samples $\{\hat{\mathbf{x}}\}$, which gives the success probability of the MCMC decoding. In Fig. 6(c), we indicate the success probability as a function of K for seven values of ε . Although its performance is not as good as that of the BF and BP algorithms, it shows a similar dependence on K and ε . This result is quite reasonable and suggestive. Later, we will discuss the reason.

B. In the case of non-IID noise

Next, we consider the problem of primary interest in our study: what is expected when applying the BF algorithm to the readout of the SLHZ-based QA device. As the development of QA devices is ongoing and classical simulation of a quantum mechanical many-body system is computationally challenging, it is difficult to investigate the potential of our BF decoding algorithm when applied to the readouts of an SLHZ-based QA device, both in actual QA devices and in classical computer simulations. In this study, we simulate spin readout in the SLHZ system using data stochastically sampled by a classical MCMC sampler. This analysis enables us to investigate the tolerance of the SLHZ model to leakage errors arising from dynamical and thermal excitations during the sampling.

We have used the following Hamiltonian to sample data from the SLHZ system stochastically:

$$H^{\text{code}}(\hat{\mathbf{x}}) \equiv -\beta \sum_{\{i,j\}} J_{ij} x_{ij} + \gamma \sum_{\{i,j,k,l\}} \frac{1 - s_{ijkl}^{(4)}(\hat{\mathbf{x}})}{2}, \quad (43)$$

where $J_{ij} \in \mathbb{R}$ is a coupling constant that is identified with the channel observation $y_{ij} \in \mathbb{R}$ in the AWGN channel model. The first term in Eq. (43) corresponds to the distance metric between a code word $\mathbf{x}$ and the received information $\mathbf{J}$, while the second term corresponds to the constraint satisfaction metric [34]. Parameters $\{\beta, \gamma\}$ are the Lagrange weights for these two terms.

If the weight-4 syndrome is defined by

$$s_{ijkl}^{(4)}(\hat{\mathbf{x}}) = x_{ik}x_{jk}x_{jl}x_{il}, \quad (44)$$

with the assumption $x_{ii} = 1$ for $i = 1, \dots, K$, Eq. (43) is just the Hamiltonian of the SLHZ system given in Fig. 3. The $(K-1)$ -degenerated ground state of the second term in Eq. (43) defines the code states. Sampling from the low-temperature equilibrium state of $H^{\text{code}}(\hat{\mathbf{x}})$, we can see which is most likely to be the code state given by the matrix $\hat{\mathbf{z}} = \mathbf{Z}^T \mathbf{Z}$ that minimizes $H^{\text{code}}(\hat{\mathbf{x}})$, as shown in Eq. (42), where $\mathbf{Z} = (Z_1, \dots, Z_K)$ is the source state. Therefore, optimization using this Hamiltonian simply amounts to word-MAP decoding. However, in contrast to Eq. (42), $\hat{\mathbf{z}} = \hat{\mathbf{1}}_{K \times K}$ cannot be assumed in the evaluation of performance in this case. This is because both the correlation term (the first term) as well as the penalty term (the second term) in Eq. (43) violate the symmetry conditions required for the all-one code-state assumption to hold. As a result, the performance of the word-MAP decoding based on Eq. (43) depends not only on the error distribution $\hat{\mathbf{e}}$ but also on $\hat{\mathbf{z}}$. This implies that the performance is dependent on a set of coupling constants $\{J_{ij}\}$ and, thus, is problem dependent.

Here, we must draw attention to coupling constants $\{J_{ij}\}$. Recall that the primary purpose of this study is to propose and demonstrate a practical method for readout decoding of the SLHZ-based QA devices. In this context, the $\{J_{ij}\}$ are given by the problem we want to tackle. For example, they are given by an externally defined COP. In other cases, we can consider minimization of the Hamiltonian in Eq. (43) as a decoding of the observation $\mathbf{J} = (J_{12}, \dots, J_{K-1,K}) \in \mathbb{R}^{\binom{K}{2}}$ in the AWGN channel model. To avoid duplicative consideration in analyzing the potential of the BF algorithm for readout decoding of the SLHZ model, we have disregarded the reliability information for each spin readout of the SLHZ model itself, even if it was available in the actual QA device. Thus, the following results can be considered valid if readout errors are solely due to leakage errors arising from dynamical and thermal excitations during the sampling process, while neglecting

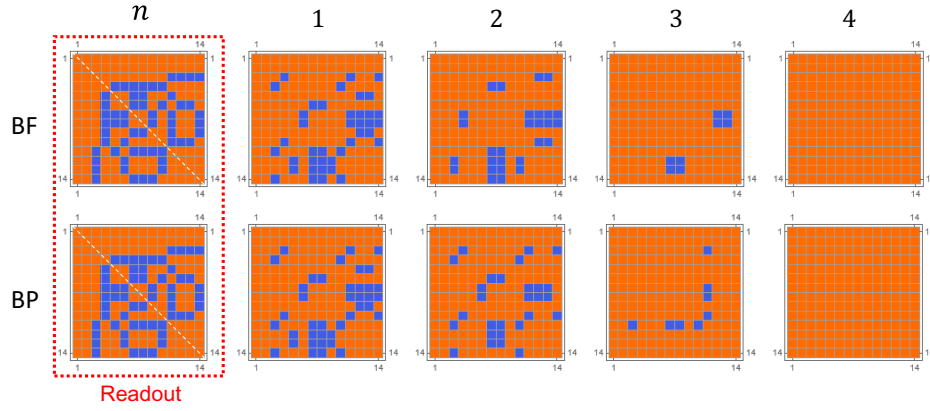


FIG. 8. Examples of successful BF and BP decoding for $K = 14$ logical spins. An MCMC-sampler sampled readout based on the Hamiltonian $H^{\text{code}}(\hat{\mathbf{x}})$ in Eq. (43). The readout matrix $\hat{\mathbf{r}}$ is plotted from left to right in order of increasing number of iterations n of the two algorithms. The blue pixels represent spins in error.

other errors, such as measurement errors. In Appendix B, we present a supplementary result when IID noise is introduced during the readout stage. This demonstrates that the above simplified approach also exhibits reasonable robustness against IID readout errors.

As a starting point, we provide typical examples demonstrating that both our BF and BP algorithms successfully eliminate errors in the readout of the SLHZ model generated by a classical MCMC sampler. In this example, we have simulated readouts $\hat{\mathbf{r}} = \hat{\mathbf{e}} \circ \hat{\mathbf{z}} \in \{\pm 1\}^{K \times K}$ of the SLHZ system associated with a spin-glass problem for $K = 14$ (K_{14}) as a toy model. We have generated a set of 12 logical random instances on complete graphs K_{14} with couplings $J_{ij} \in [-\frac{1}{4}, \frac{1}{4}]$ chosen uniformly at random and all logical local fields set to zero, where the code state $\hat{\mathbf{z}}$ for a given $\mathbf{J}$ has been precomputed by brute force. The left-most matrix plot in Fig. 8 visualizes a typical example of the error matrix $\hat{\mathbf{e}}$ associated with a sampled readout $\hat{\mathbf{r}}$ of the SLHZ system. In this example, the error distribution is quite different from the expected one for an IID-noise model in Fig. 7. In Fig. 8, we show the results where $\hat{\mathbf{e}}$ has been decoded by the BF algorithm (upper) and the BP algorithm (lower) when the number of iterations was limited to five rounds. These results suggest that our BF algorithm, as well as the BP algorithm, can eliminate errors in the stochastically sampled readouts of the SLHZ system, which does not follow the IID-noise model. Here, although we cannot know the error rate for each spin because the correct ground state is generally unknown *a priori*, an initial error rate of 0.25 has been assumed for every spin when running the BP algorithm.

The readout in the above examples has been obtained during the evaluation of the following two decoding methods based on $H^{\text{code}}(\hat{\mathbf{x}})$ in Eq. (43). First, we have evaluated the MCMC decoding. Starting from a random state, we have sampled $\hat{\mathbf{x}}$ using a rejection-free MCMC sampler

and stored a sequence $\{\hat{\mathbf{x}}\}$ by executing MCMC loops many times. The performance has been evaluated by independently repeating the same simulation, obtaining many sampled sequences $\{\hat{\mathbf{x}}\}$. We have evaluated the average probability that $\{\hat{\mathbf{x}}\}$ involves the error-free state $\hat{\mathbf{z}}$, which gives the success probability of decoding. In addition to MCMC decoding, we have performed a two-stage decoding, which we refer to as MCMC-BF hybrid decoding. In this method, we have sampled the sequence $\{\hat{\mathbf{x}}\}$ using the same rejection-free MCMC sampler in the first stage. In the second stage, each element of the sequence $\{\hat{\mathbf{x}}\}$ has been decoded using the BF algorithm to eliminate the errors in $\hat{\mathbf{x}}$ and update $\{\hat{\mathbf{r}}\}$. Then, we have evaluated the average probability that $\{\hat{\mathbf{r}}\}$ involves $\hat{\mathbf{z}}$, which gives the success probability of this hybrid decoding. The MCMC decoding can be identified with simulated annealing. Similarly, the MCMC-BF hybrid decoding can be viewed as a combination of simulated annealing and subsequent classical error correction.

The performance of the two methods differs significantly. Columns (a) and (b) in Fig. 9 indicate landscapes of the success probability for (a) the MCMC decoding and (b) the MCMC-BF hybrid decoding, respectively. In Fig. 9, the top row of figures shows the success probability of finding the error-free code state $\hat{\mathbf{z}}$ and the bottom row of figures shows the probability of finding any code state $\hat{\mathbf{x}} = \mathbf{X}^T \mathbf{X}$, where $\mathbf{X} \in \{\pm 1\}^K$ is any logical state. It is very important to note that the sizes of the sample $\{\hat{\mathbf{x}}\}$ for the MCMC decoding and the decoded sample $\{\hat{\mathbf{r}}\}$ for the MCMC-BF hybrid decoding differ significantly; they are $1200 \binom{K}{2}$ in the MCMC decoding and $4 \binom{K}{2}$ in the MCMC-BF hybrid decoding. This indicates that the number of iterations required for MCMC sampling is 300 times smaller for the MCMC-BF hybrid decoding than for the MCMC decoding. Therefore, the MCMC-BF hybrid decoding provides a better trade-off between error performance and decoding

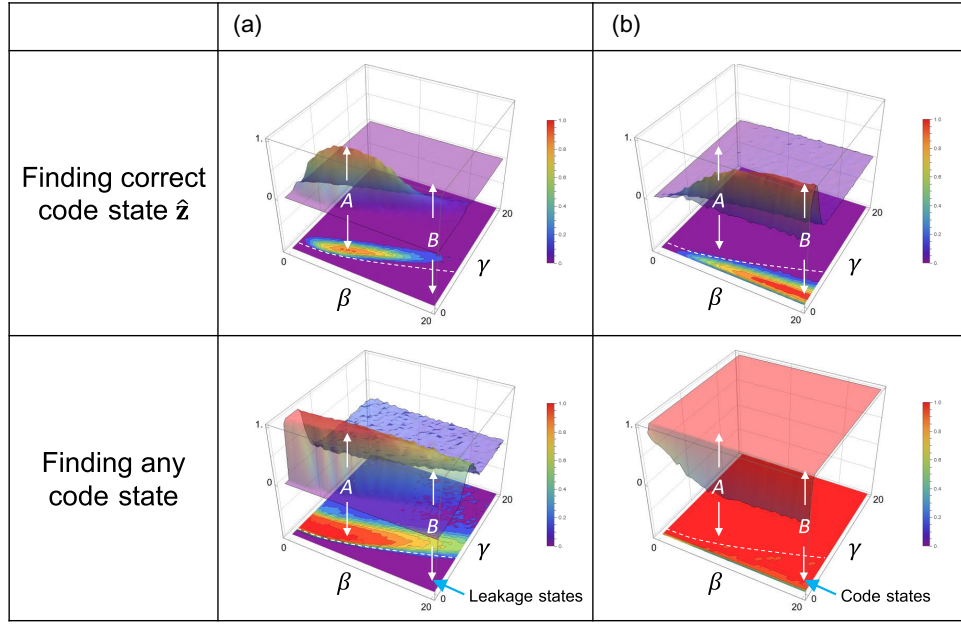


FIG. 9. Landscapes of the probability distribution for successful decoding. They are plotted as functions of the Lagrange weights $\{\beta, \gamma\}$ in Eq. (43). The left and right columns are the results of (a) MCMC decoding and (b) MCMC-BF hybrid decoding, respectively. The target states in the upper and lower rows of these figures are different: they are the correct code state $\hat{\mathbf{z}}$ for the upper row and any code states for the lower row.

complexity if the decoding complexity of the BF decoding is negligible. We will revisit the validity of this assumption later.

Let us note two arrows A and B shown in Fig. 9. They correspond to the sets of Lagrange weights $A = \{\beta_A, \gamma_A\}$ and $B = \{\beta_B, \gamma_B\}$ for which the success probability for decoding is maximized in the MCMC decoding (A) and the MCMC-BF hybrid decoding (B), respectively. The figure clearly indicates that the parameter sets suitable

for use differ between the respective decoding methods. Figures 10(a) and 10(b) are matrix plots showing the average error matrix $\langle \hat{\mathbf{e}} \rangle = \langle \hat{\mathbf{x}} \rangle \circ \hat{\mathbf{z}}$ over the set of samples $\{\hat{\mathbf{x}}\}$ obtained from the MCMC sampler under the parameter sets A and B , respectively. The marginal probability P_c that $\hat{x}_{ij}$ gives the correct value, i.e., $\hat{x}_{ij} = \hat{z}_{ij}$, is given by $P_c = (1 + \langle \hat{e}_{ij} \rangle)/2$. So, a negative entry in $\langle \hat{\mathbf{e}} \rangle$ indicates that it is likely that there is an error in the corresponding entry in $\hat{\mathbf{x}}$. We can see that $\langle \hat{\mathbf{e}} \rangle$ depends on the chosen parameter sets. The average error matrix $\langle \hat{\mathbf{e}} \rangle$ in Fig. 10(b) is consistent with the error matrix $\hat{\mathbf{e}}$ in Fig. 8, because they have been sampled with the same parameter set B . Note that it is highly likely to fail sampling a code state by the MCMC sampler for the set B (see the lower left plot in Fig. 9), while it is highly likely to succeed sampling a code state by the MCMC sampler for the set A . This implies that the state $\hat{\mathbf{x}}$ sampled in the first stage of the MCMC-BF decoding is not a code state, although the state $\hat{\mathbf{x}}$ sampled in the MCMC decoding is a code state. This is quite reasonable, because the subsequent BF decoding can correct errors only if $\hat{\mathbf{x}}$ sampled at the first stage is a leakage state in the MCMC-BF hybrid decoding. In contrast, the sampled state must be a code state if it is the correct state in the MCMC decoding. This is why the optimal parameter sets $\{\beta_{\text{opt}}, \gamma_{\text{opt}}\}$ differ between the MCMC decoding and MCMC-BF hybrid decoding. It is important to note that this is a consequence of our BF decoding algorithm and that it is independent of the sampling algorithm in the first stage. Our results suggest that optimal Lagrange weights should be carefully chosen when applying our BF

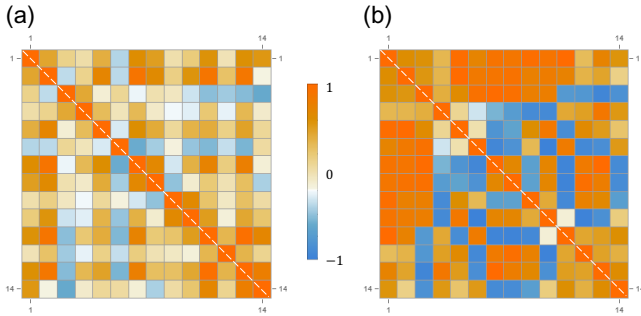


FIG. 10. Matrix plots representing the averaged error matrix $\langle \hat{\mathbf{e}} \rangle = \langle \hat{\mathbf{x}} \rangle \circ \hat{\mathbf{z}}$ for the sampled states by the MCMC sampling, where each entry reflects the marginal error probability in inferring the correct sign of the associated entry. Warm- (cold-) colored pixels indicate that there is likely no error (error) in the inference of the associated spin. (a),(b) The plots correspond to $\langle \hat{\mathbf{e}} \rangle$ when $\{\hat{\mathbf{x}}\}$ has been sampled under the parameter sets (a) A and (b) B in Fig. 9, respectively.

decoding for readouts of QA, as they may differ from the optimal parameters for QA when used alone.

V. DISCUSSION

A. Relationship between MCMC-BF hybrid algorithm and other known BF algorithms

Both the MCMC and BF decoding algorithms are hard-decision algorithms, in which only spin variables are of concern. The MCMC-BF decoding algorithm is considered an extended BF algorithm that incorporates MCMC decoding as a preprocessing step. We discuss the limitations of BF decoding and how they have been resolved in the existing extended BF algorithms.

There are two obvious limitations in the simple BF decoding algorithm. If we introduce the MCMC decoding as a preprocessing step, these limitations can be overcome. The first limitation is that the BF algorithm neglects the soft information contained in $\mathbf{J}$. As mentioned, the BF algorithm assumes a symmetric channel and treats all the spin variables (VNs) and syndromes (CNs) symmetrically, justifying the all-one code-state assumption. This is the result of neglecting the correlation term depending on $\mathbf{J}$, which breaks the necessary symmetry in the Hamiltonian $H^{\text{code}}(\hat{\mathbf{x}})$ defined by Eq. (13) [see also Eq. (41)]. Soft information should be considered to improve decoding, which is frequently fulfilled by taking the reliability information into account in the algorithm. Unfortunately, the BF decoding loses soft information, since it approximates the weighted majority vote in Eqs. (24) and (25) to the majority vote in Eq. (36) by assuming that $\gamma_{ij} = \gamma_0$ for the error probability of every decision r_{ij} . In general, γ_{ij} may depend on the spin $\{i, j\}$. For example, in the AWGN channel model, the LLR for the channel observation J_{ij} is given by $2B_{ij} = \beta J_{ij}$ [see Eq. (9)]. It follows that γ_{ij} is small (large) if $|J_{ij}|$ is small (large) and that the hard decision $x_{ij} = \text{sign}[J_{ij}]$ is less (more) reliable.

Many researchers have developed sophisticated versions of the weighted BF (WBF) to incorporate reliability information of the hard decisions into the BF algorithm (see details in Refs. [35,36] and references therein). Alternatively, there is another approach to incorporating soft information. For example, Wadayama *et al.* have proposed the gradient-descent bit-flipping (GDBF) algorithm [27]. They are based on modifications to the inversion function that may improve decoding performance, though this comes at the cost of increased decoding complexity. For example, the inversion functions employed in the BF, WBF, and GDBF algorithms are formally written as

$$\Delta_k^{(\text{BF})}(\mathbf{x}) = 1 + \sum_{i \in M(k)} s_i(\mathbf{x}), \quad (45)$$

$$\Delta_k^{(\text{WBF})}(\mathbf{x}) = \beta |J_k| + \sum_{i \in M(k)} w_k s_i(\mathbf{x}), \quad (46)$$

and

$$\Delta_k^{(\text{GDBF})}(\mathbf{x}) = J_k x_k + \sum_{i \in M(k)} s_i(\mathbf{x}), \quad (47)$$

respectively. Here, $\mathbf{x} = (x_1, \dots, x_{N_v}) \in \{\pm 1\}^{N_v}$ is the current decision for the spin variables (VNs), β is a positive real parameter to be adjusted,

$$s_i(\mathbf{x}) = \prod_{j \in N(i)} x_j \in \{\pm 1\} \quad (48)$$

is an i th syndrome (CN) for the decision $\mathbf{x}$ in the spin representation, and J_k is identified with channel observation y_k in the AWGN channel model. Here, the set $N(i) = \{j : H_{ij} = 1\}$ are the VNs adjacent to an i th CN ($1 \leq i \leq N_c$) and the set $M(j) = \{i : H_{ij} = 1\}$ are the CNs adjacent to a j th VN ($1 \leq j \leq N_v$). For the definition of these sets, see Fig. 2. Note that as long as $s_i(\mathbf{x})$ is a weight-3 syndrome, $\Delta_k^{(\text{BF})}(\mathbf{x})$ is symmetric for the permutation of the elements of $\mathbf{x}$. In other words, $\Delta_k^{(\text{BF})}(\mathbf{x})$ is invariant under exchanging $i \longleftrightarrow j$ for any pair of indices of VN other than k , justifying the all-one code-state assumption. In contrast, $\Delta_k^{(\text{WBF})}(\mathbf{x})$ is not symmetric if $s_i(\mathbf{x})$ is a weight-4 syndrome. Similarly, because $\Delta_k^{(\text{WBF})}(\mathbf{x})$ involves weight w_k in the second term and $\Delta_k^{(\text{GDBF})}(\mathbf{x})$ involves J_k in the first term, both depend on $\mathbf{J}$. Thus, $\Delta_k^{(\text{WBF})}(\mathbf{x})$ and $\Delta_k^{(\text{GDBF})}(\mathbf{x})$ are not symmetric even if $s_i(\mathbf{x})$ is a weight-3 syndrome. It is interesting to note that the inversion functions in Eq. (45)–(47) can be formally derived from the following Hamiltonians:

$$H^{(\text{BF})}(\mathbf{x}) = - \sum_{i=1}^{N_v} x_i - \sum_{i=1}^{N_c} s_i(\mathbf{x}), \quad (49)$$

$$H^{(\text{WBF})}(\mathbf{x}) = -\beta \sum_{i=1}^{N_v} |J_i| x_i - \sum_{i=1}^{N_c} w_i s_i(\mathbf{x}), \quad (50)$$

$$H^{(\text{GDBF})}(\mathbf{x}) = -\frac{1}{2} \sum_{i=1}^{N_v} J_i x_i - \sum_{i=1}^{N_c} s_i(\mathbf{x}), \quad (51)$$

respectively. If we note that when we invert the sign of x_k , i.e., $x_k \rightarrow -x_k$, the increase in energy $\Delta H_k^{(X)}(\mathbf{x})$ is given by

$$\Delta H_k^{(X)}(\mathbf{x}) = 2\Delta_k^{(X)}(\mathbf{x}), \quad (52)$$

where $X = \text{BF}, \text{WBF}, \text{or GDBF}$, it follows that if $\Delta_k^{(X)}(\mathbf{x}) < 0$, flipping the k th spin reduces the total energy of the spins. Therefore, the BF, WBF, and GDBF decoding algorithms can be considered deterministic algorithms that determine the most suitable spins to be flipped to

reduce the energy $H_k^{(X)}(\mathbf{x})$ based on the inverse function $\Delta_k^{(X)}(\mathbf{x})$. In addition, in contrast to the BF algorithm, the BP algorithm essentially considers soft information, as it calculates a consistent marginal probability $P(x_i|y)$ by exchanging real-valued messages between VNs and CNs. On the other hand, the inversion function and the associated Hamiltonian for the MCMC decoding are formally written as

$$\Delta_k^{(\text{MCMC})}(\mathbf{x}) = \beta J_k x_k + \frac{\gamma}{2} \sum_{i \in M(k)} s_i(\mathbf{x}) \quad (53)$$

and

$$H^{(\text{MCMC})}(\mathbf{x}) = -\beta \sum_{i=1}^{N_v} J_i x_i + \gamma \sum_{i=1}^{N_c} \frac{1 - s_i(\mathbf{x})}{2}. \quad (54)$$

The MCMC decoding algorithm uses MCMC sampling to find $\mathbf{x}$ that reduces the energy $H^{(\text{MCMC})}(\mathbf{x})$ based on the inversion function $\Delta_k^{(\text{MCMC})}(\mathbf{x})$. Therefore, the MCMC decoding at the first stage inherently incorporates soft information into the correlation term, thereby resolving the limitation of BF decoding.

In addition, MCMC decoding in the first stage also solves another limitation of the BF algorithm in the second stage. The MCMC sampling is stochastic in contrast to our deterministic BF algorithm and the gradient-descent algorithm used in the GDBF algorithm. This stochasticity introduces randomness into the spin-flip selection. It allows spins to flip even when $\Delta_k^{(\text{MCMC})}(\mathbf{x}) > 0$, which provides an escape from spurious local minima and makes it more likely to arrive at the neighborhood of the global minimum of $H^{(\text{MCMC})}(\mathbf{x})$. In this way, the stochastic spin-flip selection helps find the global minimum of $H^{(\text{MCMC})}(\mathbf{x})$. A similar stochasticity can be incorporated directly into the GDBF algorithm, either by adding a noise term to the inversion function $\Delta_k^{(\text{GDBF})}(\mathbf{x})$ (Noisy GDBF [28]) or by taking stochastic spin-flip selection into account in the algorithm (probabilistic GDBF [37]). Briefly speaking, our MCMC-BF hybrid decoding algorithm can be considered an alternative to the BP algorithm and other existing BF decoding algorithms. Our simulation suggests that it is important to properly control the fluctuations of $\mathbf{x}$ during MCMC sampling by adjusting the two Lagrange weights $\{\beta, \gamma\}$ to optimize the MCMC-BF decoding. It also suggests that the contribution of the penalty term to the Hamiltonian must be small enough not to force $\mathbf{x}$ into a valid code state. Controlling Lagrange weights is important for switching between the two decoding modes. In Fig. 11, we show schematic diagrams illustrating two decoding modes: (a) MCMC decoding and (b) MCMC-BF hybrid decoding, respectively. Decoding is formulated as a constrained COP. The large and small ellipses indicate the search space and the code-state space that minimizes the penalty term

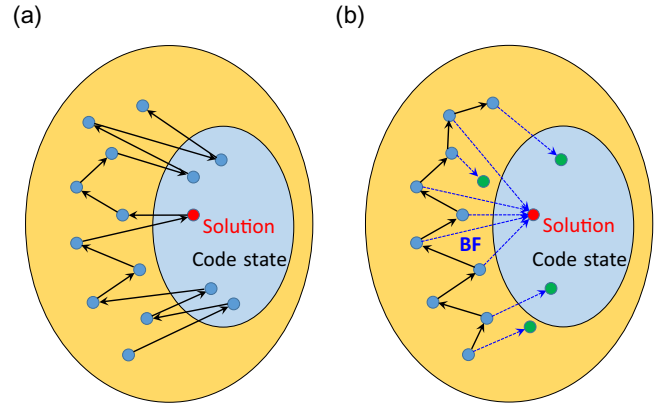


FIG. 11. Schematic diagrams illustrating two operation modes of the MCMC sampling that optimize (a) MCMC decoding and (b) MCMC-BF hybrid decoding. The large and small ellipses indicate the search space and the code-state space, respectively. The blue circles indicate the sampled state by the MCMC sampler. The green circles indicate the decoded states by the BF decoding. The red circle indicates the correct code state $\mathbf{z}$. The solid black arrows indicate a sequence of MCMC sampling. The broken blue arrows indicate the map due to the BF decoding.

in the Hamiltonian. In MCMC decoding, the MCMC samples both the code states (feasible solution) and leakage states (infeasible solution) [Fig. 11(a)]. On some occasions, the correct code state (solution) may be sampled. In MCMC-BF hybrid decoding, on the other hand, the first-stage MCMC sampler samples only the leakage state and the second-stage BF decoding maps the leakage state to the correct code state occasionally [Fig. 11(b)]. Our simulations have shown that the MCMC-BF hybrid decoding [Fig. 11(b)] is over 2 orders of magnitude more efficient than the MCMC decoding [Fig. 11(a)] if we ignore the decoding cost of the postreadout BF decoding and compare performance simply by the number of required MCMC samples needed to obtain at least one correct code state.

B. Spin-flipping mechanism and decoding cost

We briefly discuss the mechanism for selecting which spins to flip at each iteration and the decoding cost of the BF algorithm. Spin-flipping mechanisms are free to choose from a range of strategies. For example, we can flip only the most suitable spin chosen based on the inversion function $\Delta_k^{(X)}(\mathbf{x})$ values estimated by the current decision $\mathbf{x}$. Alternatively, we can flip several spins chosen based on the $\Delta_k^{(X)}(\mathbf{x})$ values at once. These are called the single-spin-flipping and multi-spin-flipping strategies, respectively. In this classification, the MCMC decoding algorithm belongs to the single-spin flipping strategy, while our BF decoding algorithm belongs to the multi-spin-flipping strategy. Note that the BF decoding algorithm consists of two operations: evaluating the inversion function $\Delta_k^{(\text{BF})}(\mathbf{x})$ and determining the spins to be flipped. The inversion function $\Delta_k^{(\text{BF})}(\mathbf{x})$

for every spin k is computed at once from the current $\mathbf{x}$ by matrix multiplication. Subsequent sign evaluation for each element of $\Delta_k^{(\text{BF})}(\mathbf{x})$ determines which spins are flipped at once.

We can see from Fig. 2 that more edges are connected to VNs for the weight-3 syndromes ($d_v = K - 2$) than for the weight-4 syndromes ($d_v \leq 4$). Thus, each spin variable affects more syndromes when using the weight-3 syndrome than the weight-4 syndrome. Conversely, each spin variable is determined from more syndromes when using the weight-3 syndrome than the weight-4 syndrome. Consequently, when using the weight-3 syndrome, one must compute the sum of products of data distributed globally across many spins to decide whether to invert each spin. In contrast, when using the weight-4 syndrome, it is sufficient to compute the sum of products of data distributed over at most eight adjacent spins to determine whether to invert each spin. The reader may fear that a global reduction operation to compute the inversion function when using weight-3 syndrome will require more effort at the expense of accelerating error correction at each spin flip, but this is not the case, as discussed below.

We can perform the matrix multiplication required for calculating $\Delta_k^{(\text{BF})}(\mathbf{x})$ easily and gain significant benefits from parallel-computing techniques and hardware engines that compute matrix sum products, such as GPUs and vector processing engines developed for machine learning. This fact provides the practical merit of choosing the weight-3 syndrome in $\Delta_k^{(\text{BF})}(\mathbf{x})$ [Eq. (45)]. In fact, comparing the computation times of the MCMC and MCMC-BF hybrid decoding presented in Fig. 9 performed on the *Mathematica*[®] platform, the computation time of MCMC-BF hybrid decoding was one fifth of that of the MCMC decoding. Although the computational cost depends on the algorithm used and the software or hardware platforms on which we perform the calculations, this result suggests that the MCMC-BF hybrid decoding is more efficient than MCMC decoding alone. Furthermore, we can confirm that our BF algorithm was much less computationally expensive than the BP algorithm. For example, the computation time of the performance evaluation for the BF decoding shown in Fig. 6(a) was about 1/40 of that for the BP decoding shown in Fig. 6(b).

This paper does not discuss decoding cost in any further detail. Let us recall that the purpose of this paper is to propose the BF decoding and demonstrate its potential as a practical postprocessing algorithm for a preprocessing stochastic algorithm, such as a QA. Since the development of QA devices is ongoing, it is not easy to demonstrate our BF decoding algorithm as a postprocessing step for an actual QA device. Instead, the potential of the BF decoding has been demonstrated using a classical MCMC sampler as a preprocessor in this study. We believe that the present result depends primarily on the high compatibility between the SLHZ model and the BF decoding algorithm, rather

than on the MCMC sampler. Furthermore, our result is consistent with the general belief that two decoding algorithms based on different mechanisms can be combined to solve problems that neither can solve alone. We believe that our BF decoding algorithm is promising for correcting readout errors of the QA devices due to measurement errors as well as dynamic errors that may be encountered during QA.

VI. CONCLUSIONS

This paper has proposed a practical decoding algorithm to correct readout errors tailored for the SLHZ model. Given the close connection between the SLHZ model and classical LDPC codes, classical decoding techniques for LDPC codes are expected to help exploit the latent potential of the SLHZ model. We have proposed a simple BF algorithm based on iterative majority voting, a decoding algorithm for LDPC codes, which is much simpler than the standard BP algorithm. We have demonstrated that the BF algorithm provides strong protection against IID noise for spin readouts of the SLHZ system and is as efficient as the BP algorithm. To study the tolerance to errors caused by a broader range of noise, we have conducted a classical simulation of spin readouts on the SLHZ system using the MCMC sampler. We have found that BF decoding can correct leakage errors caused by dynamical and thermal excitations during MCMC sampling. Efficient decoding is possible if we reduce energy penalties, allowing thermal excitations to populate correctable leakage states more frequently. This is quite reasonable, given that stochastic sampling followed by classical error correction can be regarded as a two-stage hybrid decoding algorithm. Our observation suggests that the SLHZ model exhibits error correction against a broader range of noise models than the IID-noise model when combined with postreadout BF decoding. Note that the BF algorithm does not suffer from the energy-barrier issue inherent to stochastic sampling. Thus, the two-stage hybrid decoding algorithm can be regarded as a practical implementation of the soft-annealing proposed by Sourlas [8]. Controlling the Lagrange weights in the first-stage sampler is crucial for efficiently obtaining the correct state. Special attention should be paid to the fact that the optimal Lagrange weights depend heavily on whether postreadout decoding is performed.

Since our demonstration has used readouts sampled stochastically by a classical method, specifically MCMC sampling, we are not necessarily convinced that the BF decoding is valid for readouts obtained through an SLHZ-based QA device. Further research is needed to assess how effectively the QA device can collaborate with BF decoding under realistic conditions. Nevertheless, we believe that most of our insights stem from the intrinsic nature of the SLHZ model and BF decoding

and are applicable regardless of the stochastic sampling mechanism employed in the first stage. For example, if measurement error is the primary source of readout error of the SLHZ system, our BF decoding algorithm offers a straightforward solution to mitigate it. In addition, it would be reasonable to expect that a two-stage hybrid computation combining QA and postreadout BF decoding may address issues that neither method can solve independently. Our research also emphasizes the importance of selecting appropriate decoding algorithms to exploit the potential of SLHZ systems.

In this study, we have tested only a small number of instances K_{14} to investigate the performance and characteristics of the BF decoding; however, this is insufficient to draw a general conclusion. Of particular importance for future study is to investigate the size dependence of the SLHZ model on the performance. In our discussion, we have suggested that postreadout BF decoding may partially compensate for the poor spin-update properties of the SLHZ system (for details, see Appendix A). In particular, it will be interesting to see whether this compensation works well for a problem of arbitrary size. Albash *et al.* have also pointed out another significant limitation of the SLHZ system: the strength of the energy penalty must grow with the problem size [24]. Introducing postreadout BF decoding may alleviate this limitation by reducing the penalty strength. Both these issues are important for achieving scalability. More research is needed to determine the efficacy of postreadout decoding.

ACKNOWLEDGMENTS

I would like to thank Dr. T. Kadowaki at the Global Research and Development Center for Business by Quantum-AI technology (G-QuAT) and Professor H. Nishimori at the Institute of Science Tokyo for their valuable comments and discussions. I also thank Dr. Masayuki Shirane of NEC Corporation and the National Institute of Advanced Industrial Science and Technology for his continuous support. This paper is partly based on results obtained from a project, Grant No. JPNP16007, commissioned by the New Energy and Industrial Technology Development Organization (NEDO), Japan.

DATA AVAILABILITY

The data that support the findings of this article are not publicly available upon publication. The data are available from the authors upon reasonable request.

APPENDIX A: RELEVANCE AND CONSISTENCY TO THE EARLIER WORK BY ALBASH *et al.*

In Ref. [24], the authors have presented minimum-weight decoding (MWD), which aims to find an error

pattern $\mathbf{e}$ with minimum Hamming distance from the hard-decided readouts $\mathbf{r}$ and with the same syndrome vector as $\mathbf{s}^{(4)}(\mathbf{r})$. They have directly estimated the optimal error pattern $\mathbf{e}^* \in \{\pm 1\}^{N_v}$ from the global ground state of the following Hamiltonian:

$$H^{(\text{MWD})}(\mathbf{x}) = -\sum_{i=1}^{N_v} x_i - \lambda \sum_{i=1}^{N_c} s_i^{(4)}(\mathbf{r}) s_i^{(4)}(\mathbf{x}), \quad (\text{A1})$$

where $\mathbf{x} = (x_1, \dots, x_{N_v})$ is an arbitrary spin state and $\mathbf{s}^{(4)}(\mathbf{x}) = (s_1^{(4)}(\mathbf{x}), \dots, s_{N_c}^{(4)}(\mathbf{x}))$ is the associated weight-4 syndrome vector. Here, λ is assumed to be sufficiently large such that $\mathbf{x}$ minimizes all constraint terms. The second term in Eq. (A1) forces $\mathbf{x}$ to satisfy $\mathbf{s}^{(4)}(\mathbf{x}) = \mathbf{s}^{(4)}(\mathbf{r})$ and the first term minimizes the number of spins with $x_i = -1$. Let C denote the set of $\mathbf{x}$ satisfying $\mathbf{s}^{(4)}(\mathbf{x}) = \mathbf{s}^{(4)}(\mathbf{r})$. Then, the optimal estimate $\mathbf{e}^*$ is given by

$$\mathbf{e}^* = \arg \min_{\mathbf{x} \in C} H^{(\text{MWD})}(\mathbf{x}). \quad (\text{A2})$$

Thus, $\mathbf{e}^*$ is the error pattern with the same syndrome vector and the smallest number of errors (i.e., the smallest number of elements e_k with a value -1). Consequently, the state $\mathbf{z}^*$ estimated by the MWD is given by $\mathbf{z}^* = \mathbf{r} \circ \mathbf{e}^*$. The authors have also related the MWD to finding the ground state of the following spin-glass Hamiltonian:

$$H_{\text{spin}}^{(\text{MWD})}(\mathbf{s}) = -\sum_{i < j} g_{ij} s_i s_j, \quad (\text{A3})$$

where a set of spin variables $\mathbf{g} = (g_{12}, \dots, g_{K-1K}) \in \{\pm 1\}^{\binom{K}{2}}$ can be regarded as a gauge transformation. Once the ground state of Eq. (A3) is found, one can deduce the optimal estimate $\mathbf{e}^*$ and obtain the estimate $\mathbf{z}^*$. The problem here is that the MWD requires the solution that minimizes the Hamiltonian defined in Eq. (A3). This is equivalent to solving an instance of maximum 2-satisfiability, which is as hard as the original COP to be solved. Therefore, the MWD does not necessarily resolve the performance bottleneck for the SLHZ system. We think that MWD is challenging to apply in realistic scenarios when considering implementation in the near-term QA devices. In contrast, our BF decoding algorithm is an approximation, in which the current estimate $\mathbf{r}$ is iteratively updated to approach a better estimate $\mathbf{r}'$. If the estimate $\mathbf{r}$ converges to a certain state within finite iterations, it gives a correct estimate $\mathbf{z}^*$ with a finite probability. Our BF algorithm can be executed on a deterministic Turing machine in polynomial time, since it completes with a finite number of iterations of matrix multiplication and the associated sign evaluation of every entry of the resultant matrix. Therefore, we believe that our BF decoding algorithm is practical and realistic for implementation in near-term QA devices.

The authors have also pointed out that the SLHZ system faces more challenges with single spin updates than the other embedding method, i.e., the minor-embedding (ME) model [38,39]. We expect that such challenges can be alleviated by introducing the BF decoding. Recall that, in the ME and SLHZ models, there is an overhead resulting from the embedding, where the chain of short-range physical interactions simulates the effect of long-range physical interactions. A chain consists of local two-body interactions in the ME model, while it consists of local four-body interactions in the SLHZ model. On the other hand, the spin update in our BF algorithm relies on numerous weight-3 syndromes, including long-range three-body interactions. As shown in Fig. 2, the column weight d_v for the weight-4 syndrome is always less than 4, while that for the weight-3 syndrome depends on the size K of the original logical problem and increases with increasing K . For $K > 6$, the use of the weight-3 syndrome instead of the weight-4 syndrome can make spin updates more efficient, since it increases the connectivity between spins. This has actually been confirmed in the following observation. Let us recall that MCMC decoding using the Hamiltonian in Eq. (41) results in the performance curves shown in Fig. 6(c). However, when a similar evaluation has been performed by replacing the weight-3 syndrome $s_{ijk}^{(3)}(\hat{\mathbf{x}})$ with the weight-4 syndrome $s_{ijk}^{(4)}(\hat{\mathbf{x}})$, the performance has been found to be much worse. This is quite reasonable, considering that the mixing property of the MCMC depends on connectivity between the spins; the more connectivity each spin has, the better the mixing property should be. In other words, embedding is not possible without sacrificing mixing properties. Since we calculate BF decoding on a digital computer, we do not have to worry about connectivity issues. Therefore, there is no reason to avoid using weight-3 syndromes in the postreadout BF decoding. Even if the spin update of the SLHZ system is less efficient, the postreadout BF decoding can vastly recover its unfavorable spin-update properties by using weight-3 syndromes.

Our simple BF algorithm fails to account for the soft information incorporated in the logical coupling constants $\mathbf{J} = (J_{12}, \dots, J_{K-1,K}) \in \mathbb{R}^{\binom{K}{2}}$. Thus, it is required to increase the relative importance of the correlation terms to the four-body penalty terms to incorporate the soft information contained in the Hamiltonian $H^{\text{code}}(\hat{\mathbf{x}})$ of the SLHZ model. Previously, Albash *et al.* had tested using distributions sampled by simulated quantum annealing (SQA) and parallel tempering (PT). They found that BP, as well as MWD, offers a substantial performance boost over MVD when the penalty strength is brought close to zero. However, they concluded that this boost originated from MWD itself, because it was typically observed in cases where the ground state of the original logical problem and the same problem approximated by $g_{ij} = \text{sign}(J_{ij})$ happen to coincide. In our simulation, a similar performance

boost has been observed for the BF decoding when the penalty strength was almost zero, as shown in Fig. 9. Moreover, a performance boost has been observed in all instances studied, regardless of whether they belonged to the above specific case. We believe that this performance boost comes from the mechanism schematically depicted by Fig. 11 in Sec. V A. The readout state just before decoding by our BF algorithm must be a state with correctable leakage errors. By reducing the penalty strength, the soft information contained in the correlation term of the Hamiltonian $H^{\text{code}}(\hat{\mathbf{x}})$ of the SLHZ system can be better reflected, thereby boosting the sampling probability of states with correctable leakage errors. As shown above, our study suggests that the BF and BP algorithms can cope not only with IID errors but also with a broader range of errors.

We recognize that the performance boost of the MCMC-BF hybrid decoding at a vanishing penalty strength provides evidence supporting the inherent error-correcting capability of the SLHZ model. Our study has suggested that controlling the penalty strength is crucial to exploiting the potential of the SLHZ model. Nonetheless, we do not claim that this performance boost is sufficient for the SLHZ model to outperform the ME model. Rather, we consider it essential that the SLHZ system be competitive with the ME model. For supplementary information, we have confirmed in our preliminary experiments that the ME model shows no performance boost after postreadout MVD.

APPENDIX B: SUPPLEMENTARY RESULT ON EXPERIMENTAL DEMONSTRATION

In Sec. IV, we have demonstrated that the postreadout BF decoding algorithm can eliminate not only IID errors but also leakage errors due to dynamic and thermal excitations during the MCMC sampling process. However, when we imagine that the BF decoding is applied to the readout of actual QA devices rather than MCMC simulation, several types of errors may occur simultaneously. In this appendix, we assume IID errors in the readouts of the MCMC sampler, which might be a realistic model of readout errors for imperfect annealing devices, and investigate their impact on the performance of the MCMC-BF hybrid decoding. We present a result showing that the MCMC-BF decoder is robust to such readout errors to some extent.

We simulate noisy readout $\hat{\mathbf{r}}'$ by multiplying the error matrix $\hat{\mathbf{e}}$ generated by a pseudorandom number generator and the readout matrix $\hat{\mathbf{r}}$, i.e., $\hat{\mathbf{r}}' = \hat{\mathbf{r}} \circ \hat{\mathbf{e}}$, where $\hat{\mathbf{r}}$ has been obtained from the same spin-glass problem (K_{14}) as in the main text. Subsequently, we have applied the BF decoding to $\hat{\mathbf{r}}'$, and investigated how the performance of the MCMC-BF hybrid decoder changes. To perform a statistical evaluation, we have run 1000 independent experiments. In Fig. 12, we show the success probabilities for obtaining the correct state when incorporating IID errors into the readouts of the MCMC sampler with four sample sizes.

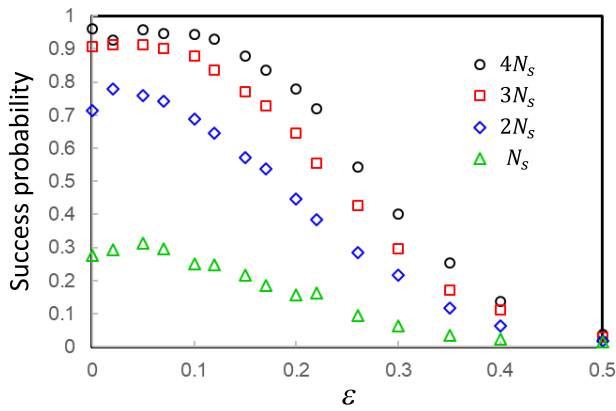


FIG. 12. The success probability for the MCMC-BF decoder as a function of the bit error rate ε for the readouts of the MCMC sampler with four sample sizes. The sample size is expressed in a unit $N_s = \binom{K}{2}$, where $K = 14$.

In this figure, we have assumed that each spin flips with the bit error rate ε ($0 \leq \varepsilon \leq 0.5$), and that the Lagrange weights are the values at which the MCMC-BF hybrid decoder achieves maximum efficiency (corresponding to arrow B in Fig. 9). As expected, the probabilities reduce as ε increases and reach almost zero at $\varepsilon = 0.5$. It has been confirmed that decoding succeeds even when $\varepsilon \neq 0$ and, in particular, that the probability remains nearly unchanged in the region in which $\varepsilon \leq 0.1$. This result suggests that the postreadout BF decoder can accommodate realistic noise in imperfect annealing devices.

- [1] D. L. Stein and C. M. Newman, *Spin Glasses and Complexity* (Princeton University Press, Princeton, 2013).
- [2] S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi, Optimization by simulated annealing, *Science* **220**, 671 (1983).
- [3] T. Kadowaki and H. Nishimori, Quantum annealing in the transverse Ising model, *Phys. Rev. E* **58**, 5355 (1998).
- [4] N. Sourlas, Spin-glass models as error-correcting codes, *Nature* **339**, 693 (1989).
- [5] N. Sourlas, Spin glasses, error-correcting codes and finite-temperature decoding, *Europhys. Lett. (EPL)* **25**, 159 (1994).
- [6] N. Sourlas, Statistical mechanics and error-correction codes, [arXiv:cond-mat/9811406](https://arxiv.org/abs/cond-mat/9811406).
- [7] N. Sourlas, Statistical mechanics and capacity-approaching error-correcting codes, *Physica A* **302**, 14 (2001).
- [8] N. Sourlas, Soft annealing: A new approach to difficult computational problems, *Phys. Rev. Lett.* **94**, 070601 (2005).
- [9] T. Albash and D. A. Lidar, Adiabatic quantum computation, *Rev. Mod. Phys.* **90**, 015002 (2018).
- [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. H. Devoret, J. M. Martinis, D. Esteve, and J. Clarke, Resonant activation from the zero-voltage state of a

- current-biased Josephson junction, *Phys. Rev. Lett.* **53**, 1260 (1984).
- [12] J. M. Martinis, M. H. Devoret, and J. Clarke, Energy-level quantization in the zero-voltage state of a current-biased Josephson junction, *Phys. Rev. Lett.* **55**, 1543 (1985).
- [13] M. H. Devoret, J. M. Martinis, and J. Clarke, Measurements of macroscopic quantum tunneling out of the zero-voltage state of a current-biased Josephson junction, *Phys. Rev. Lett.* **55**, 1908 (1985).
- [14] Y. Nakamura, Yu. A. Pashkin, and J. S. Tsai, Coherent control of macroscopic quantum states in a single-Cooper-pair box, *Nature* **398**, 786 (1999).
- [15] S. Puri, C. K. Andersen, A. L. Grimsmo, and A. Blais, Quantum annealing with all-to-all connected nonlinear oscillators, *Nat. Commun.* **8**, 15785 (2017).
- [16] S. E. Nigg, N. Lörch, and R. P. Tiwari, Robust quantum optimizer with full connectivity, *Sci. Adv.* **3**, e1602273 (2017).
- [17] P. Zhao, Z. Jin, P. Xu, X. Tan, H. Yu, and Y. Yu, Two-photon driven Kerr resonator for quantum annealing with three-dimensional circuit QED, *Phys. Rev. Appl.* **10**, 024019 (2018).
- [18] T. Onodera, E. Ng, and P. L. McMahon, A quantum annealer with fully programmable all-to-all coupling via Floquet engineering, *npj Quantum Inf.* **6**, 1 (2020).
- [19] T. Yamaji, S. Masuda, A. Yamaguchi, T. Satoh, A. Morioka, Y. Igarashi, M. Shirane, and T. Yamamoto, Correlated oscillations in Kerr parametric oscillators with tunable effective coupling, *Phys. Rev. Appl.* **20**, 014057 (2023).
- [20] F. Pastawski and J. Preskill, Error correction for encoded quantum annealing, *Phys. Rev. A* **93**, 052325 (2016).
- [21] J. Pearl, in *Probabilistic and Causal Inference*, edited by H. Geffner, R. Dechter, and J. Y. Halpern (Association for Computing Machinery, New York, 1982) 1st ed., p. 129.
- [22] R. G. Gallager, Low-density parity-check codes, *IRE Trans. Inf. Theory* **8**, 21 (1962).
- [23] R. G. Gallager, *Low-Density Parity-Check Codes* (MIT Press, Cambridge, 1963).
- [24] T. Albash, W. Vinci, and D. A. Lidar, Simulated-quantum-annealing comparison between all-to-all connectivity schemes, *Phys. Rev. A* **94**, 022327 (2016).
- [25] 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).
- [26] J. L. Massey, Ph.D. thesis, M.I.T. Dept. of Electrical Engineering, 1962.
- [27] T. Wadayama, K. Nakamura, M. Yagita, Y. Funahashi, S. Usami, and I. Takumi, Gradient descent bit flipping algorithms for decoding LDPC codes, *IEEE Trans. Commun.* **58**, 1610 (2010).
- [28] G. Sundararajan, C. Winstead, and E. Boutillon, Noisy gradient descent bit-flip decoding for LDPC codes, *IEEE Trans. Commun.* **62**, 3385 (2014).
- [29] G. J. C. Clerk and B. J. Cain, *Error-Correction Coding for Digital Communications* (Plenum Press, New York, 1981).
- [30] T. Richardson and R. Urbanke, The capacity of low-density parity-check codes under message-passing decoding, *IEEE Trans. Inf. Theory* **47**, 599 (2001).
- [31] M. Vuffray, Ph.D. thesis, École Polytechnique Fédérale de Lausanne, 2014.

- [32] See the Supplemental Material at <http://link.aps.org/supplemental/10.1103/drry-blqs> for the MATLAB code implementation of the described BF decoding.
- [33] Y. Nambu, Rejection-free Monte Carlo simulation of QUBO and Lechner-Hauke-Zoller optimization problems, *IEEE Access* **10**, 84279 (2022).
- [34] A. D. Sarma, U. Majumder, V. Vaidya, M. G. Chandra, A. A. Kumar, and S. Pramanik, in *Parallel Processing and Applied Mathematics. PPAM 2022*, edited by R. Wyrzykowski, J. Dongarra, E. Deelman, and K. Karczewski, Lecture Notes in Computer Science, Vol. 13827 (Springer, Cham, 2023).
- [35] Khoa Le Trung, Ph.D. thesis, École Nationale Supérieure de l'Électronique et de ses Applications, 2017.
- [36] Kennedy Masunda, Ph.D. thesis, University of the Witwatersrand, 2017.
- [37] O. A. Rasheed, P. Ivaniš, and B. Vasić, Fault-tolerant probabilistic gradient-descent bit flipping decoder, *IEEE Commun. Lett.* **18**, 1487 (2014).
- [38] V. Choi, Minor-embedding in adiabatic quantum computation: I. The parameter setting problem, *Quantum Inf. Process.* **7**, 193 (2008).
- [39] V. Choi, Minor-embedding in adiabatic quantum computation: II. Minor-universal graph design, *Quantum Inf. Process.* **10**, 343 (2011).