

PAPER

A Fully-Connected Ising Model Embedding Method and Its Evaluation for CMOS Annealing Machines

Daisuke OKU^{†a)}, Kotaro TERADA^{†*}, Masato HAYASHI^{††}, Masanao YAMAOKA^{††},
Shu TANAKA^{†††,††††}, Nonmembers, and Nozomu TOGAWA^{†b)}, Senior Member

SUMMARY Combinatorial optimization problems with a large solution space are difficult to solve just using von Neumann computers. Ising machines or annealing machines have been developed to tackle these problems as a promising Non-von Neumann computer. In order to use these annealing machines, every combinatorial optimization problem is mapped onto the physical *Ising model*, which consists of spins, interactions between them, and their external magnetic fields. Then the annealing machines operate so as to search the ground state of the physical Ising model, which corresponds to the optimal solution of the original combinatorial optimization problem. A combinatorial optimization problem can be firstly described by an ideal fully-connected Ising model but it is very hard to embed it onto the physical Ising model topology of a particular annealing machine, which causes one of the largest issues in annealing machines. In this paper, we propose a fully-connected Ising model embedding method targeting for CMOS annealing machine. The key idea is that the proposed method replicates every logical spin in a fully-connected Ising model and embeds each logical spin onto the physical spins with the same chain length. Experimental results through an actual combinatorial problem show that the proposed method obtains spin embeddings superior to the conventional de facto standard method, in terms of the embedding time and the probability of obtaining a feasible solution.

key words: CMOS annealing, Ising model, Ising computing, graph embedding, combinatorial optimization

1. Introduction

A combinatorial optimization problem is to find an optimal solution from a combinatorial solution space, such as a graph partitioning problem, a traveling salesperson problem, a rectangle packing problem, and others. When these problems have a large solution space, solving them just using von Neumann computers may become very difficult. Non-von Neumann computers, such as CMOS annealing machines [1], [2], Digital Annealer [3], [4], quantum annealing machines [5]–[8], and coherent Ising machines [9], [10],

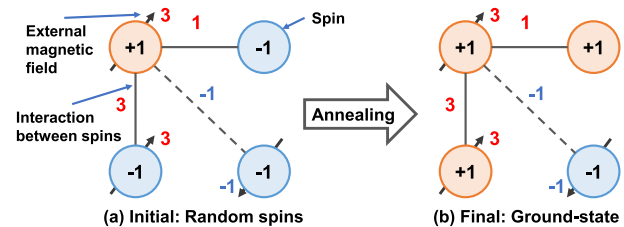


Fig. 1 Ising model and annealing.

have been developed to tackle these problems.

In those studies, a combinatorial optimization problem is mapped onto an ideal magnetic model in statistical mechanics called *Ising model* [11] and the annealing machines operate so as to search the ground-state of the Ising model, which equivalently corresponds to the optimal solution of the original combinatorial optimization problem.

The overview of the Ising model and the annealing process is shown in Fig. 1. The Ising model consists of spins, interactions between spins, and external magnetic fields of spins. In Fig. 1, every circle represents a spin. A spin has a value of (+1) or (−1). The edge between spins shows interaction between them. If the interaction has a positive value, the spins connected to it tend to have the same value. If it has a negative value, the spins connected to it tend to have different values. Every spin has an external magnetic field value. If it is positive, the spin tends to have (+1). If it is negative, the spin tends to have (−1). Figure 1 (a) shows an initial state of the Ising model before annealing process, which is not a ground-state. Figure 1 (b) shows the resultant ground-state of the Ising model after annealing process.

The overall flow of solving a combinatorial optimization problem using annealing machines is shown in Fig. 2. The flow consists of three phases.

Phase 1: A combinatorial optimization problem to be solved is mapped onto (formulated as) an Ising model including the objective function and the constraints.

Phase 2: The Ising model mapped (formulated) in Phase 1 is embedded onto the physical annealing machine through Ising model embedding. This embedding determines the required spins, interactions between spins, and external magnetic fields of spins.

Phase 3: An annealing machine operates so as to search the ground-state (at the point of the minimum energy) of the spins and the solution of the input combinatorial optimization problem is equivalently obtained.

Manuscript received December 6, 2018.

Manuscript revised March 1, 2019.

Manuscript publicized June 10, 2019.

[†]The authors are with the Department of Computer Science and Communications Engineering, Waseda University, Tokyo, 169–8555 Japan.

^{††}The authors are with the Hitachi, Ltd., Kokubunji-shi, 185–8601 Japan.

^{†††}The author is with Green Computing Systems Research Organization, Waseda University, Tokyo, 162–0042 Japan.

^{††††}The author is with the Precursory Research for Embryonic Science and Technology, Japan Science and Technology Agency, Kawaguchi-shi, 332–0012 Japan.

*Presently, with Yahoo Japan Corporation.

a) E-mail: daisuke.oku@togawa.cs.waseda.ac.jp

b) E-mail: ntogawa@waseda.jp

DOI: 10.1587/transinf.2018EDP7411

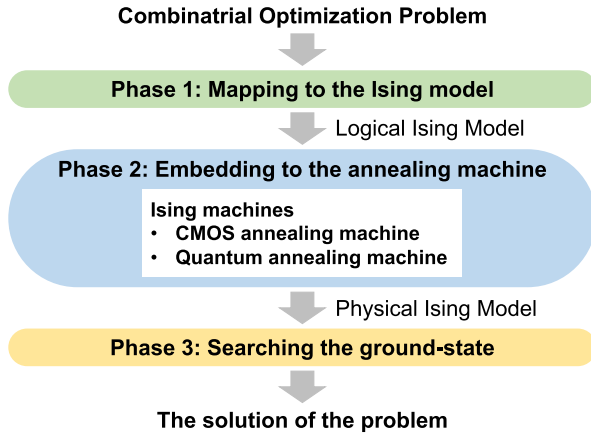


Fig. 2 Overall flow of solving a combinatorial optimization problem using annealing machines.

In Phase 1, when a combinatorial optimization problem is mapped onto (formulated as) a logical Ising model, the Ising model is assumed to have interactions between *any* pairs of spins, i.e., a fully-connected Ising model is assumed. After that, in Phase 2, a fully-connected Ising model is mapped onto the Ising model topology of a particular annealing machine. Phase 3 can be done by using the annealing machine targeted in Phase 2.

Phase 1 itself is a very difficult task but there have been proposed many Ising model mappings for practical combinatorial optimization problem [12]–[14]. However, Phase 2 must be effectively solved for every annealing machine architecture.

Based on this discussion, this paper proposes a fully-connected Ising model embedding method targeting for a newly developed annealing machine, called CMOS annealing machine [1]. The CMOS annealing machine is one of the annealing machines and expected to be used to solve practical problems efficiently.

Several Ising model embedding methods [6], [15]–[19] have been proposed targeting for the D-Wave quantum annealing machine (or D-Wave machine) [5], [6]. However, these methods cannot be efficiently applied to the CMOS annealing machine [1], since the Ising model topology of the D-Wave machine is very different from that of the CMOS annealing machine. Note that, [15] can be applied to an annealing machine with any Ising model topology but may lead to insufficient results (see Sect. 5).

The proposed Ising model embedding method embeds a fully-connected Ising model that represents a combinatorial optimization problem onto the Ising model on the CMOS annealing machine. The key idea is that the proposed method replicates every logical spin in a fully-connected Ising model and embeds each logic spin onto the physical spins in the CMOS annealing machine with the *same chain length* utilizing the regular structure of physical spins (see Sect. 3 for the detailed definition of the terminologies). Hence the number of required physical spins can be reduced and the annealing performance will be improved.

The main contributions of this paper are summarized into:

1. We propose a fully-connected Ising model embedding method targeting the CMOS annealing machine, which is expected to be one of the most practical annealing machines, *for the first time*, where the number of required physical spins becomes just $(n^2 + n)$ *theoretically*, when the number of logical spins is n .
2. Experimental results demonstrate that the number of required physical spins can be smaller in the practical problem size, compared to the de facto standard conventional method.
3. Experimental results through practical combinatorial optimization problems also demonstrate that our proposed embedding method can be superior, compared to the conventional de facto standard method, in terms of the solution quality and the probability of obtaining a feasible solution.

The remainder of this paper is organized as follows: Sect. 2 summarizes conventional spin embedding methods for annealing machines; Sect. 3 firstly describes the targeting Ising model and its spin topology in the CMOS annealing machine. After that, a spin embedding problem is defined; Sect. 4 proposes a spin embedding method targeting the CMOS annealing machine; Sect. 5 shows numerical simulation results and evaluates the efficiency of the proposed method; Sect. 6 discusses the results and evaluations of Sect. 5; Sect. 7 gives concluding remarks.

2. Related Works

There have been proposed many researches on solving combinatorial optimization problems recently using Ising models but almost all of them target the D-Wave quantum annealing machine [5], [6]. The D-Wave machine contains superconductor chips which work under very low temperature of milli-Kelvin order. The latest version of the D-Wave machine has 2,048-spin (or -qubit) Ising model [20]. The D-Wave machine has its specific Ising model topology called a Chimera graph [6], [20].

Fully-connected Ising model embedding methods have been proposed targeting for the Ising model topology of the D-Wave machine [6], [16], [18]. For example, in [6], Bunyk et al. proposed a fully-connected Ising model embedding method onto the Chimera graph topology. In this method, the number of required spins on the D-Wave machine becomes in proportion to the square of the number of spins in the input Ising model. Although the methods [6], [16], [18] can embed a fully-connected Ising model onto the Chimera graph topology, the methods cannot be used for other annealing machines. Hence, we cannot use these methods for our targeting CMOS annealing machine.

The target of the embedding method proposed in [15] is also the D-Wave machine but it can embed any Ising model onto an arbitrary Ising model topology. Since such an embedding problem is known as NP-hard [21], the method [15]

is based on heuristic shortest-path search. The method is currently a de facto standard embedding method for the D-Wave machine. Since both input Ising model topology and the targeting annealing machine topology can be defined by a user in this method, it can be also applied to the CMOS annealing machine. However, embedding results for the CMOS annealing machine may be inefficient, since the method [15] does not take into account the regular structure of the CMOS annealing machine.

3. Problem Formulation

In this section, we define Ising models, the target CMOS annealing machine, and an Ising model embedding problem.

3.1 Ising Model

An Ising model is a theoretical magnetic model in statistical mechanics, which consists of microscopic variables called *spins*, an interaction between them, and a field on each spin called an *external magnetic field*. An Ising model is represented by an undirected graph $\mathcal{M} = (\mathcal{V}, \mathcal{E})$ where \mathcal{V} is a set of vertexes and \mathcal{E} is a set of edges describing interactions between spins. Spin σ_i is associated with vertex $i \in \mathcal{V}$ and has either of the two values $+1/-1$ (or states up/down). For each pair of vertexes $i, j \in \mathcal{V}$ ($i \neq j$), $(i, j) \in \mathcal{E}$ denotes a connection between i and j . In other words, σ_i and σ_j are connected. The energy (or Hamiltonian) H of an Ising model is calculated by

$$H = - \sum_{(i,j) \in \mathcal{E}} J_{ij} \sigma_i \sigma_j - \sum_{i \in \mathcal{V}} h_i \sigma_i \quad (1)$$

where J_{ij} is the weight of connection of vertexes $i, j \in \mathcal{V}$, in other words, the interaction between the spin σ_i and spin σ_j and h_i is the external magnetic field of the spin σ_i . J_{ij} and h_i are given by real numbers. $|X|$ represents the number of elements of the set X . In the Ising model, H is minimized when the spin states (or values) reach the ground-state. CMOS annealing machines operate so as to search for the ground-state energy by updating the states (values) of spins.

In this paper, we define two types of Ising models; logical Ising model and physical Ising model.

3.1.1 Logical Ising Model

A logical Ising model $M_L = (V_L, E_L)$ is an Ising model where interactions can be defined between *any pairs* of spins. Any graph topologies can be represented by logical Ising models. Combinatorial optimization problems are also mapped onto (formulated as) logical Ising models.

Spins on a logical Ising model are called *logical spins*. An interaction between logical spins σ_i and σ_j is denoted by J_{Lij} , and an external magnetic field on a logical spin σ_i is denoted by h_{L_i} .

3.1.2 Physical Ising Model

A physical Ising model denoted by $M_P = (V_P, E_P)$ is an

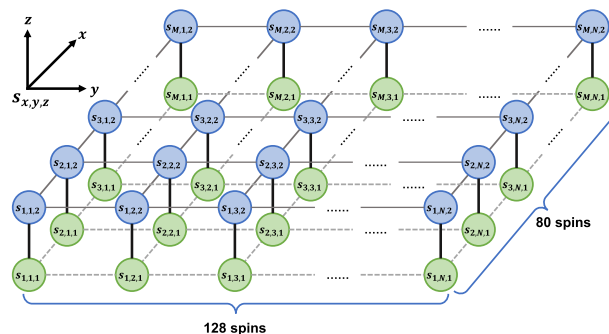


Fig. 3 128 × 80 × 2-lattice Ising model topology of the CMOS annealing machine [1].

Ising model where interactions can only be defined between pairs of spins which are physically connected on the topology of a particular annealing machine. If we assume an annealing machine whose topology is fully-connected, we can directly and easily embed a logical Ising model which has interactions between any pairs of spins onto the annealing machine. Typical annealing machines have physical Ising model topology where only physically adjacent spins are connected, for example. A physical Ising model is equivalent to the physical annealing machine and it can be directly embedded onto the target annealing machine.

Spins on a physical Ising model are called *physical spins*. An interaction between physical spins σ'_i and σ'_j is denoted by J_{Pij} and an external magnetic field on a physical spin σ'_i is denoted by h_{P_i} .

3.2 CMOS Annealing Machine

In this paper, our target annealing machine is the CMOS annealing machine [1]. The CMOS annealing machine is implemented by using CMOS technology (65nm technology node) and works at room temperature. Hence, it does not need specific cooling equipment or power because CMOS circuits are used to realize annealing process of Ising model.

Figure 3 shows 128 × 80 × 2-lattice Ising model topology of the CMOS annealing machine where every spin is just connected to its adjacent spins. The axes are defined as shown in Fig. 3. In this paper, a physical vertex located at a coordinate (x, y, z) is denoted by $s_{x,y,z}$. For example, the bottom-left vertex in Fig. 3 is denoted by $s_{1,1,1}$. A physical spin $\sigma'_{x,y,z}$ exists on a physical vertex $s_{x,y,z}$.

3.3 Embedding a Logical Ising Model to a Physical Ising Model

As we described in Sect. 3.1.1, we cannot usually embed a logical Ising model onto the annealing machine directly, while we can easily do a physical Ising model. In order to embed a logical Ising model onto the annealing machine, we need to convert a logical Ising model into an equivalent physical Ising model. To achieve this, we replicate a single logical spin and prepare more than one physical spin corresponding to it. By preparing redundant physical spins and

embedding a single logical spins to one or more physical spins, every interaction between logical spins can be defined on the interactions between physical spins. These physical spins which represent a single logical spin are called a *spin-chain*. We can convert a logical Ising model into an equivalent physical Ising model using spin-chains. Physical spins in every spin-chain are connected with the interaction of the strong ferromagnetic interactions ($J_F > 0$), so that all the physical spins in a spin-chain will have the same value (or state). Additionally, it is known that the uniform length of every spin-chain results in good annealing result, i.e., near-optimal solution, throughout the annealing process [16], [22][†]. The uniform length of spin-chains is strongly required.

Based on these discussions, we define an embedding problem of logical Ising models onto physical Ising models targeting the CMOS annealing machine as follows:

Definition 1. *The embedding problem of a logical Ising model $M_L = (V_L, E_L)$ to a physical Ising model $M_P = (V_P, E_P)$ is, to define a mapping function $\varphi : V_L \rightarrow 2^{V_P}$ which satisfies the following conditions:*

(Condition 1) *For any logical vertex $i \in V_L$, the physical vertexes in $\varphi(i)$ are all connected,*

(Condition 2) *For any pair of logical vertexes i and $j \in V_L (i \neq j)$, $\varphi(i) \cap \varphi(j) = \emptyset$,*

(Condition 3) *If two logical vertexes i and $j \in V_L$, are connected on M_L , one physical vertex in $\varphi(i)$ and one physical vertex in $\varphi(j)$ are connected on M_P ,*

(Condition 4) *For any pair of logical vertexes i and $j \in V_L (i \neq j)$, $|\varphi(i)| = |\varphi(j)|$.*

□

By Definition 1, a logical spin $\sigma_i (i \in V_L)$ is embedded to physical spins $\sigma'_{\varphi(i)} (\varphi(i) \in V_P)$. If two logical spins σ_i and $\sigma_j (i, j \in V_L)$ are connected on M_L , one physical spins in $\sigma'_{\varphi(i)} (\varphi(i) \in V_P)$ and one physical spins in $\sigma'_{\varphi(j)} (\varphi(j) \in V_P)$ are connected on M_P .

4. Proposed Ising Model Embedding Method

In this section, we propose an embedding method which embeds a fully-connected logical Ising models onto the physical Ising model on the CMOS annealing machine.

[†]According to [22], the embedding method with uniform length of spin-chain can lead to better results in annealing than the embedding method with non-uniform length of spin-chain.

There is no direct comparison of the both embedding methods (uniform v.s. non-uniform) in [16]. However, in [22], it comes to the conclusion above based on the many annealing results shown in [16].

Actually in Fig. 6 and Fig. 7 of Sect. 5, our proposed method (which is the embedding method with uniform length of spin-chain) is superior to the embedding method with non-uniform length of spin-chain [15] in terms of the probabilities of obtaining feasible solutions and the expected quality of solutions.

In a fully-connected topology, there must be connections between any pairs of logical spins. To represent these connections onto the physical Ising model, we introduce the following ideas:

(Idea i) In the bottom layer ($z = 1$ in Fig. 3) of the CMOS annealing machine topology, the physical spins which correspond to one logical spin are arranged vertically (x -axis direction) and connected with strong ferromagnetic interactions (J_F) each other to form a spin-chain.

(Idea ii) In the top layer ($z = 2$ in Fig. 3) of the CMOS annealing machine topology, the physical spins which correspond to one logical spin are arranged horizontally (y -axis direction) and connected with strong ferromagnetic interactions (J_F) each other to form a spin-chain.

(Idea iii) Based on the ideas i and ii above, all the combinations of any two logical spins can appear in the connections between the bottom and the top layers by effectively using the physical spins. Hence interactions J_{ij} in a given logical Ising model are set to the corresponding connections between the bottom and the top layers.

As above, any fully-connected logical Ising models can be expressed onto the physical Ising models.

Based on our ideas above, our proposed method is described below (see also Fig. 4, for example):

Input

A logical Ising model $M_L = (V_L, E_L)$ composed of n logical vertex spins.

Output

A physical Ising model $M_P = (V_P, E_P)$ on the CMOS annealing machine as shown in Fig. 3.

Step 1: Spins mapping

For each logical vertex $i (1 \leq i \leq n = |V_L|)$, the mapping function $\varphi(i)$ is defined as follows:

$$\varphi(i) = \bigcup_{x=1}^i \{s_{x,n-i+1,1}\} \cup \bigcup_{y=1}^{n-i+1} \{s_{i,y,2}\}. \quad (2)$$

Step 2: Interactions setting

Step 2.1

For each logical spin $\sigma_i (i \in V_L)$, the interactions between the adjacent physical spins existed on $\varphi(i)$ (obtained in Step 1) are set to J_F (very large value).

Step 2.2

For $1 \leq i, j \leq n = |V_L|$ (where $i < j$), if there is a connection edge between the logical spins σ_i and σ_j , then the interaction between the two physical spins $\sigma'_{i,n-j+1,1}$ and $\sigma'_{i,n-j+1,2}$ is set to be J_{ij} . Otherwise, it is set to be zero.

Step 2.3

For all other connections on M_P whose interactions have not been set in Step 2.1 nor Step 2.2,

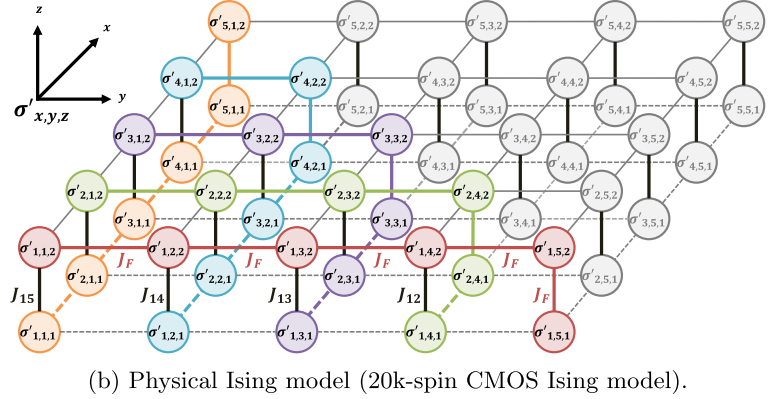
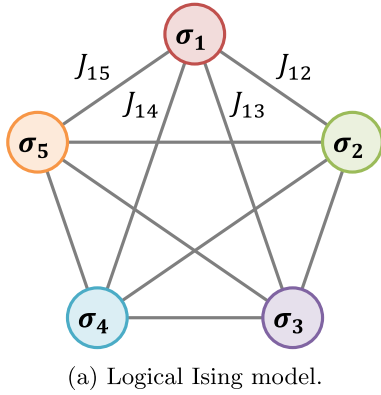


Fig. 4 Example of our proposed embedding method which embeds (a) Input: Fully-connected logical Ising model (K_5) to (b) Output: Physical Ising model on the CMOS annealing machine.

the interaction is set to be zero.

Step 3: External magnetic field setting

For each logical spin σ_i , the external magnetic fields of the physical spins $\sigma'_{x,y,z}$ existed on $s_{x,y,z} \in \varphi(i)$ ($i \in V_L$) are set to be $h_{p_i} = h_i/(n+1)$ to distribute the external magnetic field of σ_i onto the corresponding physical spins equivalently.

Figure 4 shows our Ising model embedding example of a fully-connected logical Ising model ($n = 5$) onto a physical Ising model on the CMOS annealing machine. In Fig. 4(b), the physical spins colored by the same color correspond to one logical spin which has the same color in Fig. 4(a). For example, the logical spin σ_1 in Fig. 4(a) (red-colored) corresponds to the physical spins $\{\sigma'_{1,1,2}, \sigma'_{1,2,2}, \sigma'_{1,3,2}, \sigma'_{1,4,2}, \sigma'_{1,5,2}, \sigma'_{1,5,1}\}$ in Fig. 4(b). The physical spins colored by gray in Fig. 4(b) are unused.

We can firstly prove four lemmas according to (Condition 1)–(Condition 4) in Definition 1.

Lemma 1. *Our proposed embedding method satisfies (Condition 1) in Definition 1.*

Proof. In Eq. (2) in Step 1, the spins in the first term are connected along x -axis since it is a set of spins whose x -coordinate is increased by one from 1 to i . Similarly, the spins in the second term are connected along y -axis since it is a set of spins whose y -coordinate is increased by one from 1 to $(n-i+1)$.

Furthermore, $\sigma'_{i,n-i+1,1}$ which is the physical spin where $x = i$ in the first term and $\sigma'_{i,n-i+1,2}$ which is the physical spin where $y = n-i+1$ in the second term in Eq. (2) are adjacent along z -axis.

Thus, by performing Step 2.1, for any logical spin σ_i ($i \in V_L$), the physical spins $\sigma'_{x,y,z}$ existed on $s_{x,y,z} \in \varphi(i)$ are connected. \square

Lemma 2. *Our proposed embedding method satisfies (Condition 2) in Definition 1.*

Proof. In Eq. (2) in Step 1, the y -coordinate $n-i+1$ of the physical spins in the first term is a linear function depending

on i . Similarly, the x -coordinate i of the physical spins in the second term is also a linear function depending on i . This means that these coordinates must not become the same value when i is different. z -coordinate is always different between the first and the second terms of Eq. (2). Hence, physical spins do not overlap with each other.

Thus, for any pair of logical vertexes $i, j \in V_L$ ($i \neq j$), $\varphi(i) \cap \varphi(j) = \emptyset$ holds. \square

Lemma 3. *Our proposed embedding method satisfies (Condition 3) in Definition 1.*

Proof. For any pairs of logical vertexes (i, j) where $i < j$, these spins are connected in Step 2.2 one time.

Thus, one physical vertex in $\varphi(i)$ and one physical vertex in $\varphi(j)$ are connected on M_p . \square

Lemma 4. *Our proposed embedding method satisfies (Condition 4) in Definition 1.*

Proof. In Eq. (2) in Step 1, the number of physical spins in the first term becomes i and the number of physical spins in the second term becomes $(n-i+1)$. The number of total physical spins which correspond to one logical spin is then calculated by $i + n - i + 1 = n + 1$.

For every logical spin, the number of physical spins which correspond to the logical spin becomes the same value, which is $(n+1)$.

Thus, for any pair of logical vertexes $i, j \in V_L$ ($i \neq j$), $|\varphi(i)| = |\varphi(j)|$ holds. \square

Then we can prove the following theorem:

Theorem 1. *Our proposed embedding methods satisfies (Condition 1)–(Condition 4) in Definition 1.*

Proof. As Lemma 1 to Lemma 4 describes, our proposed embedding method satisfies (Condition 1)–(Condition 4) in Definition 1. \square

Note that, our proposed embedding method requires several redundant physical spins to embed logical Ising models onto physical Ising models. The state (or value) of

spins in the logical Ising model can be equivalent to the state (or value) of spins in the physical Ising model since physical spins in the same spin-chains are connected with strong ferromagnetic interactions (J_F) as we described in Sect. 3.3.

However, all the physical spins in the same spin-chain do not always have the same value in a practical annealing process, since the behavior of annealing machines is a finite-time stochastic process. In this case, we determine the logical spin value by the majority vote of the corresponding physical spins to take this problem into account. If the number of $(+1)$'s and that of (-1) 's are the same, we can choose two types of strategy. One is that we randomly determine the corresponding logical spin value as $(+1)$ or (-1) . The other is that we determine either $(+1)$ or (-1) according to the local energy which comes from the nearest neighbor interactions and the local magnetic field. In our experiments, we adopt the first strategy for simplicity.

Let us consider the first strategy. For example, assume that $\varphi(i) = \{s_a, s_b, s_c, s_d\} \rightarrow \{\sigma'_a, \sigma'_b, \sigma'_c, \sigma'_d\}$. If we obtain the annealing result of $\sigma'_a = \sigma'_b = \sigma'_c = \sigma'_d = +1$, then $\sigma_i = +1$ is obtained obviously. If we obtain the annealing result of $\sigma'_a = \sigma'_b = +1$ and $\sigma'_c = \sigma'_d = -1$, then we randomly determine $\sigma_i = +1$ or -1 .

The required physical spins when embedding fully-connected logical Ising model (K_n) by our proposed method is clearly calculated as $(n+1) \cdot n = n^2 + n$. The number of required spins is in proportion to the square of n , i.e., $O(n^2)$.

5. Experimental Results

In this section, we evaluate the efficiency of our proposed embedding method through several experiments.

5.1 Evaluation of Our Embedding Method

To evaluate the quality of our proposed embedding method, we compare the results of the de facto standard conventional embedding method [15] and our proposed method targeting the CMOS annealing machine.

We have implemented the conventional method [15] and our proposed method in Python language on CentOS 6.8 and Intel Xeon CPU E5-2680 v3 2.50GHz \times 40 machine with 270GB memory.

For every fully-connected logical Ising model which has 4 to 12 logical spins ($n = 4, 5, \dots, 12$), we compare the embedding results obtained from the conventional method [15] and our proposed method targeting the CMOS annealing machine. Since the method [15] is a heuristic one, the results may be different every time. Then we ran the method [15] ten times and picked up the best result. The criterion to find the best result is minimizing the number of total physical spins. Table 1 and Fig. 5 summarize the comparison results. In Table 1, “Min” and “Max” show the minimum and maximum number of physical spins corresponding to a single logical spin, respectively, when using the method [15]. “Total spin” shows the number of total physical spins used. Note that, our embedding method al-

Table 1 Comparison of spin embeddings of fully-connected logical Ising models (K_n) for the CMOS annealing machine.

n	Method [15]		Our proposed method	
	Min	Max	Total spins	Total spins
4	1	3	7	20
5	1	6	13	30
6	1	7	24	42
7	2	11	39	56
8	3	13	67	72
9	2	19	93	90
10	4	27	146	110
11	5	36	182	132
12	—	—	— [†]	156

[†] Computation of [15] does not finish within one hour.

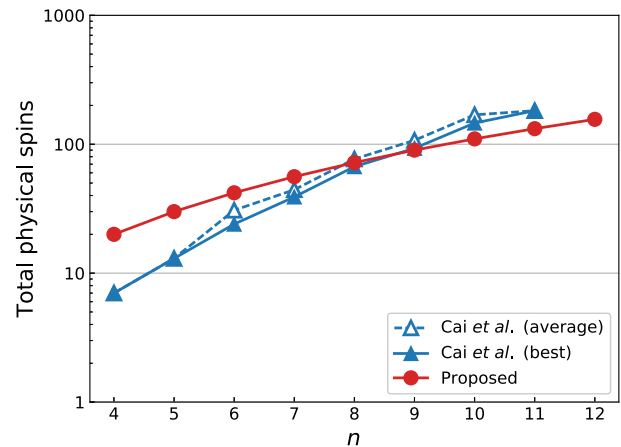


Fig. 5 Comparison of required physical spins.

ways embeds one logical spin onto $(n+1)$ physical spin.

As seen in the results, we can conclude that our proposed method is superior to the conventional method [15] in terms of the following three points:

1. The method [15] requires a smaller number of physical spins when n is small ($n \leq 8$). On the other hand, our proposed embedding method requires a smaller number of physical spins when n is large ($n \geq 9$), while the method [15] requires more physical spins in these cases.
2. In practical problems, an Ising model with eight or less spins is considered to be too small. We can conclude that our proposed embedding method requires smaller physical spins in the practical problem size compared to the conventional method [15].
3. The method [15] assigns logical spins to different lengths of physical spins (spin-chains) which may cause a worse annealing result, while our proposed method assigns every logical spins to the same lengths of physical spins (spin-chains), which may affect to a better annealing result (see Sect. 5.2 in the detailed discussion).
4. The method [15] did not generate a solution when $n \geq 12$, while our proposed method can generate a spin embedding when n becomes large. We can obtain a stable

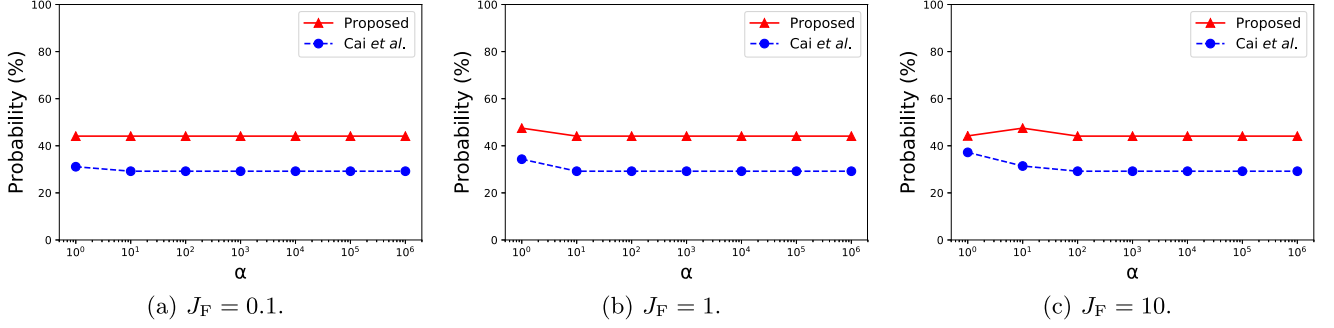


Fig. 6 Comparison of the probabilities of obtaining feasible solutions between [15] and our proposed method in the MAX-CUT problem (Graph: SE3). For each α , the simulations were performed 1000 times.

spin embedding easily.

5.2 Evaluation of Our Method Applied to Combinatorial Optimization

To evaluate our proposed embedding method in more practical uses, we solve a MAX-CUT problem [23], which is one of the NP-hard combinatorial optimization problems, using our proposed embedding method targeting the CMOS annealing machine.

The MAX-CUT problem is described as follows: Let $G = (V, E)$ be an undirected graph where V and E are sets of vertices and edges of G , respectively. We partition the graph G into disjoint two sub-graphs $S = (V', E')$ and $T = (V'', E'')$ under the constraints of $|V'| = |V''|$ (when $|V|$ is even) or $|V'| = |V''| \pm 1$ (when $|V|$ is odd). The objective is to maximize the number of cut-edges between S and T .

This problem can be mapped onto (formulated as) a logical Ising model. As in [12], we prepare $|V|$ logical spins which corresponds to each vertex in V . Now, σ_i shows a logical spin for $i \in V$. The Ising model of this problem is mapped onto (formulated as) follows:

$$H = - \sum_{(i,j) \in E} \frac{1 - \sigma_i \sigma_j}{2} + \frac{\alpha}{2} \left(\sum_{i \in V} \sigma_i \right)^2 \quad (3)$$

where α is a weight parameter and should be positive[†]. When the solution is optimal, Eq. (3) becomes minimum (ground-state). The first term of Eq. (3) represents the objective function. For every edge, the first term becomes -1 (if the edge is a cut-edge, i.e., $\sigma_i \neq \sigma_j$) or 0 (if the edge is not a cut-edge, i.e., $\sigma_i = \sigma_j = \pm 1$) from H . This means that when the number of cut-edges is the largest, the energy H is minimized. In the second term of Eq. (3), when $|S| = |T|$, it becomes zero. As the difference between $|S|$ and $|T|$ increases, the second term becomes large and it will be a penalty increasing the energy H . Here, we introduce a positive value α which is a weight parameter of the constraint term. Equation (3) can be written as:

[†]In order to simplify the J_{ij} value in Eq. (5), we give $\alpha/2$ to the second term in Eq. (3).

Table 2 Simulation parameters.

Parameter	Value
Initial spin value	Random
Spin flipping probability (at the beginning)	0.75
Spin flipping probability (at the end)	0.001
Annealing steps	100,000 ^{††}

^{††} It takes 10×8 msec on a real machine (100MHz) which inverts a $1/8$ part of total spins in one step [1]. Thus the execution time would be 80msec when total annealing steps become 100,000.

$$H = \frac{1}{2} \sum_{(i,j) \in E} \sigma_i \sigma_j + \frac{\alpha}{2} \sum_{i \in V, j \in V, i \neq j} \sigma_i \sigma_j + \text{const} \quad (4)$$

where const shows a constant value which does not depend on σ_i . Comparing Eq. (4) to the energy of Ising model shown in Eq. (1), the interactions and the external magnetic fields of the Ising model are:

$$J_{ij} = \begin{cases} -\alpha - \frac{1}{2} & \text{if } (i, j) \in E \\ -\alpha & \text{otherwise} \end{cases} \quad (5)$$

$$h_i = 0 \quad \forall i \in V.$$

5.2.1 Comparison between [15] and Our Method

We solved several MAX-CUT problems using spin embeddings obtained from [15] and our proposed method targeting the CMOS annealing machine. In order to compare and evaluate these spin embeddings, we used CMOS annealing machine simulator [1].

The benchmark graph SE3 ($|V| = 8$) is picked up from Graph Collection [24]. The parameters are set to be $\alpha \in \{1, 10, 100, 1000, 10000, 100000\}$ and $J_F \in \{0.1, 1, 10\}$. For each parameter, the simulation was performed 1000 times. According to [1], the simulation setting parameters are shown in Table 2.

Figure 6 and Fig. 7 show the results. Figure 6 shows the probability of feasible solutions (solutions which satisfy the constraint of equally partitioning a given graph) when varying the J_F value. As seen in Fig. 6, we can see that our proposed method has higher probabilities of obtaining

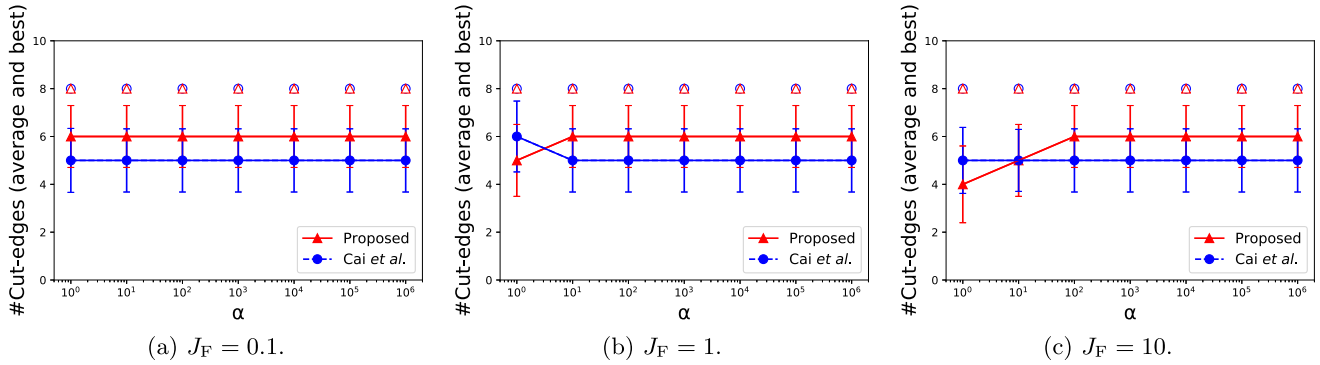


Fig. 7 Comparison of the qualities of solutions (i.e. the numbers of cut-edges) between [15] and our proposed method in the MAX-CUT problem (Graph: SE3). The white markers represent the best solutions, the colors markers and lines represent the average solutions, and the error bars represent the standard deviations. For each α , the simulations were performed 1000 times.

Table 3 Results of our method applied to MAX-CUT problem on the CMOS annealing machine.

Graph	α	#cut-edges ^{†††}		
		$J_F = 0.1$	$J_F = 1$	$J_F = 10$
SE3 ($ V = 8, E = 10$)	1	6.20 / 8 / 2 (44.1%)	5.41 / 8 / 2 (47.5%)	4.26 / 6 / 2 (44.2%)
	10	6.20 / 8 / 2 (44.1%)	6.20 / 8 / 2 (44.1%)	5.41 / 8 / 2 (47.5%)
	100	6.20 / 8 / 2 (44.1%)	6.20 / 8 / 2 (44.1%)	6.20 / 8 / 2 (44.1%)
	1000	6.20 / 8 / 2 (44.1%)	6.20 / 8 / 2 (44.1%)	6.20 / 8 / 2 (44.1%)
	10000	6.20 / 8 / 2 (44.1%)	6.20 / 8 / 2 (44.1%)	6.20 / 8 / 2 (44.1%)
	100000	6.20 / 8 / 2 (44.1%)	6.20 / 8 / 2 (44.1%)	6.20 / 8 / 2 (44.1%)
BFLY3 ($ V = 24, E = 48$)	1	25.34 / 34 / 18 (23.7%)	15.01 / 28 / 8 (31.0%)	10.44 / 16 / 8 (53.8%)
	10	25.34 / 34 / 18 (23.7%)	25.34 / 34 / 18 (23.7%)	15.01 / 28 / 8 (31.0%)
	100	25.34 / 34 / 18 (23.7%)	25.34 / 34 / 18 (23.7%)	25.34 / 34 / 18 (23.7%)
	1000	25.34 / 34 / 18 (23.7%)	25.34 / 34 / 18 (23.7%)	25.34 / 34 / 18 (23.7%)
	10000	25.34 / 34 / 18 (23.7%)	25.34 / 34 / 18 (23.7%)	25.34 / 34 / 18 (23.7%)
	100000	25.34 / 34 / 18 (23.7%)	25.34 / 34 / 18 (23.7%)	25.34 / 34 / 18 (23.7%)
bcsstk01 ($ V = 48, E = 176$)	1	90.05 / 108 / 74 (18.7%)	69.98 / 94 / 44 (20.6%)	60.63 / 70 / 58 (55.8%)
	10	90.05 / 108 / 74 (18.7%)	90.05 / 108 / 74 (18.7%)	69.98 / 94 / 44 (20.6%)
	100	90.05 / 108 / 74 (18.7%)	90.05 / 108 / 74 (18.7%)	90.05 / 108 / 74 (18.7%)
	1000	90.05 / 108 / 74 (18.7%)	90.05 / 108 / 74 (18.7%)	90.05 / 108 / 74 (18.7%)
	10000	90.05 / 108 / 74 (18.7%)	90.05 / 108 / 74 (18.7%)	90.05 / 108 / 74 (18.7%)
	100000	90.05 / 108 / 74 (18.7%)	90.05 / 108 / 74 (18.7%)	90.05 / 108 / 74 (18.7%)
Grid8x8 ($ V = 64, E = 112$)	1	58.13 / 79 / 43 (16.2%)	13.04 / 28 / 8 (21.5%)	9.19 / 12 / 8 (50.3%)
	10	58.13 / 79 / 43 (16.2%)	58.13 / 79 / 43 (16.2%)	13.04 / 28 / 8 (21.5%)
	100	58.13 / 79 / 43 (16.2%)	58.13 / 79 / 43 (16.2%)	58.13 / 79 / 43 (16.2%)
	1000	58.13 / 79 / 43 (16.2%)	58.13 / 79 / 43 (16.2%)	58.13 / 79 / 43 (16.2%)
	10000	58.13 / 79 / 43 (16.2%)	58.13 / 79 / 43 (16.2%)	58.13 / 79 / 43 (16.2%)
	100000	58.13 / 79 / 43 (16.2%)	58.13 / 79 / 43 (16.2%)	58.13 / 79 / 43 (16.2%)
FFT4 ($ V = 80, E = 128$)	1	65.26 / 78 / 56 (13.6%)	19.23 / 44 / 16 (30.8%)	17.52 / 24 / 16 (48.9%)
	10	65.26 / 78 / 56 (13.6%)	65.26 / 78 / 56 (13.6%)	19.23 / 44 / 16 (30.8%)
	100	65.26 / 78 / 56 (13.6%)	65.26 / 78 / 56 (13.6%)	65.26 / 78 / 56 (13.6%)
	1000	65.26 / 78 / 56 (13.6%)	65.26 / 78 / 56 (13.6%)	65.26 / 78 / 56 (13.6%)
	10000	65.26 / 78 / 56 (13.6%)	65.26 / 78 / 56 (13.6%)	65.26 / 78 / 56 (13.6%)
	100000	65.26 / 78 / 56 (13.6%)	65.26 / 78 / 56 (13.6%)	65.26 / 78 / 56 (13.6%)

^{†††} In the “#cut-edges” column, each cell shows the number of cut-edges of feasible solutions in a format of “average / best / worst (probability of feasible solutions).”

feasible solutions in the evaluation cases. The main reason is that we embed every logical spin to the same length of physical spins by our embedding method (i.e., we satisfy

the *Condition 4* in Definition 1).

Figure 7 shows the quality of solution (the number of cut-edges) in feasible solutions when varying the J_F values.

5.2.2 Evaluation of Our Method Using Other Graphs

We solved several MAX-CUT problems using our proposed embedding method also using the annealing machine simulator [1] for other benchmarks. The four benchmark graphs are picked up from Graph Collection [24]. The parameters are set to be $\alpha \in \{1, 10, 100, 1000, 10000, 100000\}$ and $J_F \in \{0.1, 1, 10\}$. For each parameter, the simulation was performed 1000 times. The simulation setting parameters are the same as in Table 2.

Table 3 shows the results. For most benchmark graphs in $\alpha = 1$ or $\alpha = 10$, when J_F becomes larger, the probability of obtaining feasible solutions tends to be increased while the quality of solutions tends to be decreased. For $J_F = 1, 10$, when α becomes larger, the probability of obtaining feasible solutions tends to be decreased while the quality of solutions tends to be increased. Consideration on these parameter setting is one of the important future works.

6. Discussion

6.1 Updating a Spin and Its Effect to α and J_F in CMOS Annealing Machine

The performance of CMOS annealing may be independent of J_F and α expect for the α value being close to J_F in Fig. 6, both in [15] and our proposed method. As we mentioned in Sect. 5.1, we use the CMOS annealing simulator [1], where physical spins are updated by Algorithm 1[†]. In Algorithm 1, N refers to a spin to be updated, and Nx ($x = U, L, R, D, F$) refers to each spin adjacent to N , as depicted in Fig. 8(a). Each spin has a value of (+1) or (−1). Ix ($x = U, L, R, D, F$) refers to an interaction between the spins N and Nx , and S refers to external magnetic field of the spin N .

Algorithm 1 The algorithm of updating a spin N

```

SUM = 0;
X = {U, L, R, D, F};
for x in X do
    SUM = SUM + Nx × Ix;
end for
SUM = SUM + S;
if SUM > 0 then
    N = 1;
else if SUM < 0 then
    N = −1;
else
    Randomly
    N = +1 or −1;
end if

```

In Algorithm 1, the value of spin N is determined to be (+1) or (−1) depending on the values of Nx , interactions Ix , and external magnetic field S of spin N . Note that, after Algorithm 1 is over, the value of the spin N may be inverted randomly to avoid falling into local minimum values [1].

[†]In [1], there is no explicit algorithm description but we summarize the CMOS annealing behavior as in Algorithm 1.

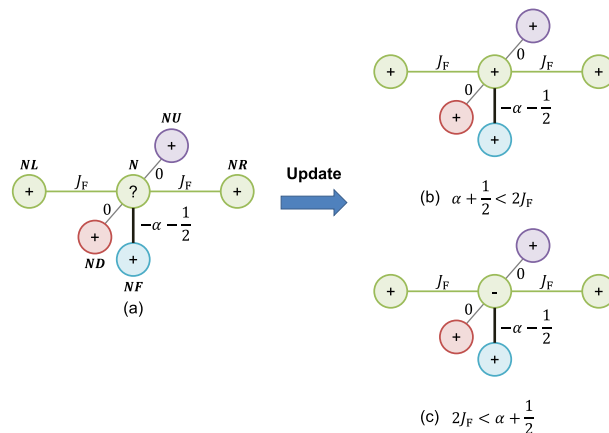


Fig. 8 An example of updating spin N .

The example of updating the spin N is shown in Fig. 8. Figure 8 represents a part of Fig. 4(b), where one logical spin is embedded onto the three physical spins NL, N, NR . In Fig. 8(a), the interactions between N and ND and between N and NU are set to be zero according to Step 2 of the our proposed method (See Sect. 4 in detail). In Fig. 8, according to Algorithm 1, SUM is calculated as follows:

$$\begin{aligned}
 SUM &= J_F + J_F - \alpha - \frac{1}{2} + 0 + 0 \\
 &= 2J_F - \alpha - \frac{1}{2}
 \end{aligned}$$

In the case of Fig. 8(b), i.e., $\alpha + 1/2 < 2J_F$ holds, the spin N becomes (+1) since SUM becomes a positive value. On the other hand, in the case of Fig. 8(c), i.e., $2J_F < \alpha + 1/2$ holds, the spin N becomes (−1) since SUM becomes a negative value.

Here, for example, assume that $J_F = 1$. When $\alpha > 3/2$, $2J_F < \alpha + 1/2$ always holds and hence the spin N becomes (−1). Of course, if α ($> 3/2$) increases, the result does not change. Therefore, it is considered that there is almost no difference in the result, when the value of α is $10 \times J_F$ and when the value of α is $10^6 \times J_F$. This result is almost the same even when $J_F = 0.1$ and $J_F = 10$.

For this reason, the parameters J_F and α seem to be independent of the performance of the CMOS annealing results of Fig. 6.

6.2 Expected Quality of Solutions

The goal of combinatorial optimization is to find the optimum solution. The performances of the proposed method and the method [15] seem to be equivalent since the best solutions in the MAX-CUT problem (Graph: SE3) using the CMOS annealing simulator between the both methods are the same.

Figure 7 represents the results of 1000 times simulation. However, it is unknown whether an optimal solution is always obtained when the number of simulation times or the annealing time is limited. If the average quality of solutions

is better, it can be expected that the solution obtained by just one trial or several times trial will be better. As a result, it can be expected that the results obtained by the limited number of trials or limited annealing times become better. Hence, we believe that evaluating the average quality must be reasonable enough. Actually, [25] also uses the average quality of solutions to evaluate the annealing methods. Our proposed method is superior to the method [15] in terms of the average quality of solutions.

In addition to the above, the method [15] does not guarantee that graph embedding always succeeds and has the worst case complexity $O(n^9)$ (the average case complexity $O(n^3)$) when a logical Ising model with the number n of nodes is embedded onto the physical Ising model [26]. This means that the method [15] itself may become a bottleneck against total annealing time. On the other hand, our proposed method can always embed the logical Ising model with number n of nodes into the CMOS annealing machine using $O(n^2)$ spins, and can embed the graphs in Table 3 onto the CMOS annealing machine with around one second, which must be short enough.

7. Conclusion

In this paper, we proposed a fully-connected Ising model embedding method for the CMOS annealing machine. Experimental results effectively show that our proposed method embeds Ising models using less physical spins compared to the conventional de facto standard method in the practical problem size, and that our method is efficient when solving practical combinatorial optimization problems.

Considerations and optimizations of the annealing parameter setting, and applying our method to other combinatorial optimization problems are important future works.

Acknowledgements

This paper is based on the results obtained from a project commissioned by the New Energy and Industrial Technology Development Organization (NEDO). One of the authors (S. T.) was supported by JST, PRESTO Grant Number JPMJPR1665, Japan and JSPS KAKENHI Grant Numbers 15K17720, 15H03699.

References

- [1] M. Yamaoka, C. Yoshimura, M. Hayashi, T. Okuyama, H. Aoki, and H. Mizuno, "A 20k-spin Ising chip to solve combinatorial optimization problems with CMOS annealing," *IEEE Journal of Solid-State Circuits*, vol.51, no.1, pp.303–309, Jan. 2016.
- [2] C. Yoshimura, M. Hayashi, T. Okuyama, and M. Yamaoka, "FPGA-based Annealing Processor for Ising Model," *Proc. Fourth International Symposium on Computing and Networking (CANDAR 2016)*, pp.436–442, Nov. 2016.
- [3] S. Matsubara, H. Tamura, M. Takatsu, D. Yoo, B. Vatankhahghadim, H. Yamasaki, T. Miyazawa, S. Tsukamoto, Y. Watanabe, K. Takemoto, and A. Sheikholeslami, "Ising-Model Optimizer with Parallel-Trial Bit-Sieve Engine," *Advances in Intelligent Systems and Computing*, vol.611, pp.432–438, Springer International Publishing, Cham, 2018.
- [4] M. Aramon, G. Rosenberg, E. Valiante, T. Miyazawa, H. Tamura, and H.G. Katzgraber, "Physics-Inspired Optimization for Quadratic Unconstrained Problems Using a Digital Annealer," *Frontiers in Physics*, vol.7, p.48, 2019.
- [5] M.W. Johnson, M.H.S. Amin, S. Gildert, T. Lanting, F. Hamze, N. Dickson, R. Harris, A.J. Berkley, J. Johansson, P. Bunyk, E.M. Chapple, C. Enderud, J.P. Hilton, K. Karimi, E. Ladizinsky, N. Ladizinsky, T. Oh, I. Perminov, C. Rich, M.C. Thom, E. Tolkacheva, C.J.S. Truncik, S. Uchaikin, J. Wang, B. Wilson, and G. Rose, "Quantum annealing with manufactured spins," *Nature*, vol.473, no.7346, pp.194–198, May 2011.
- [6] P.I. Bunyk, E.M. Hoskinson, M.W. Johnson, E. Tolkacheva, F. Altomare, A.J. Berkley, R. Harris, J.P. Hilton, T. Lanting, A.J. Przybysz, and J. Whittaker, "Architectural Considerations in the Design of a Superconducting Quantum Annealing Processor," *IEEE Transactions on Applied Superconductivity*, vol.24, no.4, pp.1–10, Aug. 2014.
- [7] M. Maezawa, K. Imafuku, M. Hidaka, H. Koike, and S. Kawabata, "Design of quantum annealing machine for prime factoring," *Proc. 2017 16th International Superconductive Electronics Conference (ISEC)*, pp.1–3, June 2017.
- [8] M. Maezawa, G. Fujii, M. Hidaka, K. Imafuku, K. Kikuchi, H. Koike, K. Makise, S. Nagasawa, H. Nakagawa, M. Ukibe, and S. Kawabata, "Toward Practical-Scale Quantum Annealing Machine for Prime Factoring," *J. Phys. Soc. Jpn.*, vol.88, no.6, Sept. 2018.
- [9] T. Inagaki, Y. Haribara, K. Igarashi, T. Sonobe, S. Tamate, T. Honjo, A. Marandi, P.L. McMahon, T. Umeki, K. Enbutsu, O. Tadanaga, H. Takenouchi, K. Aihara, K.I. Kawarabayashi, K. Inoue, S. Utsunomiya, and H. Takesue, "A coherent Ising machine for 2000-node optimization problems," *Science*, vol.354, no.6312, pp.603–606, Oct. 2016.
- [10] P.L. McMahon, A. Marandi, Y. Haribara, R. Hamerly, C. Langrock, S. Tamate, T. Inagaki, H. Takesue, S. Utsunomiya, K. Aihara, R.L. Byer, M.M. Fejer, H. Mabuchi, and Y. Yamamoto, "A fully programmable 100-spin coherent Ising machine with all-to-all connections," *Science*, vol.354, no.6312, pp.614–617, Oct. 2016.
- [11] E. Ising, "Beitrag zur Theorie des Ferromagnetismus," *Zeitschrift für Physik*, vol.31, no.1, pp.253–258, Feb. 1925.
- [12] A. Lucas, "Ising formulations of many NP problems," *Frontiers in Physics*, vol.2, pp.1–15, 2014.
- [13] K. Terada, D. Oku, S. Kanamaru, S. Tanaka, M. Hayashi, M. Yamaoka, M. Yanagisawa, and N. Togawa, "An Ising model mapping to solve rectangle packing problem," *Proc. 2018 International Symposium on VLSI Design, Automation and Test (VLSI-DAT)*, pp.1–4, April 2018.
- [14] S. Tanaka, R. Tamura, and B.K. Chakrabarti, *Quantum Spin Glasses, Annealing and Computation*, 1st ed., Cambridge University Press, New York, NY, USA, 2017.
- [15] J. Cai, W.G. Macready, and A. Roy, "A practical heuristic for finding graph minors," *arXiv preprint arXiv:1406.2741*, pp.1–16, 2014.
- [16] D. Venturelli, S. Mandrà, S. Knys, B. O’Gorman, R. Biswas, and V. Smelyanskiy, "Quantum Optimization of Fully Connected Spin Glasses," *Physical Review X*, vol.5, no.3, pp.031040:1–031040:8, Sept. 2015.
- [17] W. Lechner, P. Hauke, and P. Zoller, "A quantum annealing architecture with all-to-all connectivity from local interactions," *Science Advances*, vol.1, no.9, pp.e1500838:1–e1500838:5, Oct. 2015.
- [18] A. Zaribafian, D.J.J. Marchand, and S.S. Changiz Rezaei, "Systematic and deterministic graph minor embedding for Cartesian products of graphs," *Quantum Information Processing*, vol.16, no.5, p.136, May 2017.
- [19] TheoryInPractice, "aqc-virtual-embedding," <https://github.com/TheoryInPractice/aqc-virtual-embedding>
- [20] A.D. King, J. Carrasquilla, J. Raymond, I. Ozfidan, E. Andriyash, A. Berkley, M. Reis, T. Lanting, R. Harris, F. Altomare, K. Boothby,

- P.I. Bunyk, C. Enderud, A. Fréchet, E. Hoskinson, N. Ladizinsky, T. Oh, G. Poulin-Lamarre, C. Rich, Y. Sato, A.Y. Smirnov, L.J. Swenson, M.H. Volkman, J. Whittaker, J. Yao, E. Ladizinsky, M.W. Johnson, J. Hilton, and M.H. Amin, "Observation of topological phenomena in a programmable lattice of 1,800 qubits," *Nature*, vol.560, no.7719, pp.456–460, Aug. 2018.
- [21] D. Eppstein, "Finding Large Clique Minors is Hard," *Journal of Graph Algorithms and Applications*, vol.13, no.2, pp.197–204, 2009.
- [22] T. Boothby, A.D. King, and A. Roy, "Fast clique minor generation in Chimera qubit connectivity graphs," *Quantum Information Processing*, vol.15, no.1, pp.495–508, Jan. 2016.
- [23] R.M. Karp, "Reducibility among combinatorial problems," in *Complexity of computer computations*, pp.85–103, Springer, 1972.
- [24] T.S.M. Collection, "Ag-monien graph collection." <https://sparse.tamu.edu/AG-Monien>
- [25] R. Martoňák, G.E. Santoro, and E. Tosatti, "Quantum annealing of the traveling-salesman problem," *Physical Review E*, vol.70, no.5, pp.1–4, 2004.
- [26] K.E. Hamilton and T.S. Humble, "Identifying the minor set cover of dense connected bipartite graphs via random matching edge sets," *Quantum Information Processing*, vol.16, no.4, pp.1–17, 2017.



Daisuke Oku received the B. Eng., and M. Eng. degrees from Waseda University in 2016 and 2017, respectively, all in Computer Science and Communications Engineering. He is presently working towards D. Eng. degree there. His research interests are LSI design, cryptography architecture, and Ising computer. He is a student member of IPSJ.



Kotaro Terada received the B. Eng., M. Eng., and Dr. Eng. degrees from Waseda University in 2014, 2015, and 2018, respectively, all in Computer Engineering. He is presently a software engineer at Yahoo Japan Corporation. His research interests are combinatorial optimization and high-level synthesis.



Masato Hayashi received the B.S., and M.S. degrees in computer science from Waseda University, Tokyo, Japan, in 2008 and 2010, respectively. In 2010, he joined the Central Research Laboratory, Hitachi Ltd., Tokyo, Japan. He has been engaged in the research of accelerator architecture.



Masanao Yamaoka received the B.E., M.E., and Ph.D. degrees in electronics and communication engineering from Kyoto University, Kyoto, Japan, in 1996, 1998, and 2007, respectively. In 1998, he joined the Central Research Laboratory, Hitachi Ltd., Tokyo, Japan, where he was engaged in the research and development of low-power embedded SRAM and CMOS circuits. Since 2012, he has been engaged in the research of new-paradigm computing using CMOS circuits.



Shu Tanaka received the B. Sci. degree from Tokyo Institute of Technology in 2003 and the M. Sci. and Dr. Sci. degrees from The University of Tokyo in 2005 and 2008, respectively. He is presently an associate professor in Green Computing Systems Research Organization, Waseda University and a PRESTO researcher in Japan Science and Technology Agency. His research interests are quantum annealing, Ising machine, statistical mechanics, and materials science. He is a member of JPS.



Nozomu Togawa received the B. Eng., M. Eng., and Dr. Eng. degrees from Waseda University in 1992, 1994, and 1997, respectively, all in electrical engineering. He is presently a Professor in the Department of Computer Science and Communications Engineering, Waseda University. His research interests are VLSI design, graph theory, and computational geometry. He is a member of IEEE and IPSJ.