# Theory of Ising Machines and a Common Software Platform for Ising Machines

Shu Tanaka

Green Computing Systems Research Organization
Waseda University
27, Wasedamachi, Shinjuku-ku, Tokyo 162-0042, Japan
e-mail: shu.tanaka@aoni.waseda.jp

Yoshiki Matsuda

Fixstars

1–11–1 Osaki, Shinagawa-ku, Tokyo 141-0032, Japan

Nozomu Togawa
Dept. of Computer Science and Communications Engineering
Waseda University
3-4-1 Ookubo, Shinjuku-ku, Tokyo 169-8555, Japan
e-mail: ntogawa@waseda.jp

**Abstract— Ising machines are a new type of non-Neumann computer that specializes in solving combinatorial optimization problems efficiently. The input form of Ising machines is the energy function of the Ising model or quadratic unconstrained binary optimization form, and Ising machines operate to search for a condition to minimize the energy function. We describe the theory of Ising machines and the present status of the Ising machines, software for Ising machines, and applications using Ising machines.**

## I. Introduction

Toward a high-performance computation for combinatorial optimization problems, a new type of computation technology called Ising machines has been developed. Combinatorial optimization problem consists of the set of solution candidates $\mathcal{S}$, constraints, and the objective function $f(z)$, where $z \in \mathcal{S}$. Here, the set of solution candidates $\mathcal{S}$ has combinatorial structure, and a solution candidate $z$ can be represented by multiple discrete variables. The purpose of combinatorial optimization problems is to find the best solution under the given constraints among a large number of solution candidates. More precisely, the combinatorial optimization problem is to find $z^*$ defined as follows:

$$z^* = \mathrm{argmin}_z f(z), \quad z \in \mathcal{S}_\mathrm{f} \subseteq \mathcal{S}, \tag{1}$$

where $\mathcal{S}_\mathrm{f}$ is the set of feasible solutions satisfying the given constraints. We can represent the objective function and given constraints of combinatorial optimization problems by the Ising model. The Ising model is a theoretical model in statistical physics. Ising machines can treat combinatorial optimization problems if we represent combinatorial optimization problems by the Ising model. The detail of the Ising model will be explained in Sec. II. Ising machines operate to minimize the energy of the Ising model according to the principle of physics (see Sec. III).

Currently, various types of Ising machines have been developed. In 2011, the first commercial quantum annealing machine, D-Wave, came out [1], which triggered the attention of Ising machines. Existing quantum annealing machines are quantum hardware inspired by the theory of quantum annealing [2] and that of adiabatic quantum computation [3], [4]. After the launch of the first commercial quantum annealing machine, Ising machines inspired by not only quantum annealing but also other operation principle have been developed [5]–[10]. Thus, the hardware development of Ising machines has been energetically advanced in recent years.

Now, not only hardware but also software development is progressing. Software for Ising machines plays an important role in various parts of computation by Ising machines, especially for the preprocessing and postprocessing of computational results of Ising machines. Besides, we have to tune appropriate hyperparameters to obtain a high precision solution of the combinatorial optimization problem, which would be helped by software for Ising machines. As with software development for Ising machines, applications by using an Ising machine have been studied.

In this paper, we describe the theory of Ising machines and the recent progress of Ising machines. The organization of this paper is as follows. In Sec. II, how to use Ising machines is also shown. In Sec. III, the theory related to Ising machines is explained. In Sec. IV, the development of software for Ising machines is described. Sec. V shows some studies to apply Ising machines to some combinatorial optimization problems. Sec. VI is devoted to the conclusion and future perspective.

## II. How to use Ising machines

As explained in the previous section, the primary purpose of Ising machines is to solve combinatorial optimization problems given by Eq. (1). In this section, we describe how to use Ising machines. We have five steps to utilize Ising machines as follows (see Fig. 1).

**Step 1** Extract a combinatorial optimization problem from the overall issue.

**Step 2** Represent the extracted combinatorial optimization problem by an Ising model.

**Step 3** Decompose and embed the Ising model according to the specifications of Ising machines. Choose appropriate hyperparameters.

**Step 4** Search for lower energy states automatically according to the principle of physics.

**Step 5** Interpret the final state generated by Ising machines so that there is no contradiction with the combinatorial optimization problem.



978-1-7281-4123-7/20/$31.00 

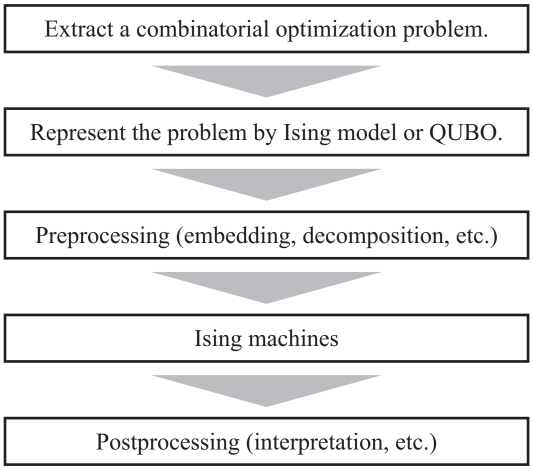


Fig. 1. The flow of the operation of Ising machines.

In the above flow of computation by Ising machines, from step 2 to step 5 are essential as calculation techniques. Thus, we explain the detail of each step in the following subsections. Before the explanation of each step, the review of the Ising model is shown.

### A. Ising model and quadratic unconstrained binary optimization (QUBO)

As stated in Sec. I, the Ising model is a theoretical model in statistical physics and an input form of Ising machines. Statistical physics is to investigate emerging macroscopic phenomena in systems with interacted a large number of microscopic elements.

Let us introduce the Ising model. The Ising model is defined on an undirected graph $G = (V, E)$, where $V$ and $E$ are the set of vertices and the set of edges, respectively. Here the number of vertices is $|V|$. Microscopic elements in the Ising model are called spins which are placed on each vertex and take either $+1$ or $-1$. Let $s_i$ be the spin on the vertex $i \in V$. The energy function of the Ising model is given by

$$E(\boldsymbol{s}) = \sum_{i \in V} h_i s_i + \sum_{(i,j) \in E} J_{ij} s_i s_j, \tag{2}$$

$$s_i \in \{+1, -1\},\ \forall i \in V, \tag{3}$$

where $\boldsymbol{s}$ represents $|V|$ spin variables in vector form, i.e., $\boldsymbol{s} = (s_1, s_2, \cdots, s_{|V|})$. The coefficients in RHS of Eq. (2), $h_i$ and $J_{ij}$, are called local magnetic field and interaction, respectively. Here, both of them are real values. According to the principle in physics, lower energy states are more stable. Here, the lowest energy state is called the ground state.

Depending on the combinatorial optimization problem to be considered, it is more convenient to use a binary variable that takes either 0 or 1 instead of a spin variable which takes either $+1$ or $-1$. If this is the case, we use a form of quadratic unconstrained binary optimization (QUBO). Let $x_i$ be the binary variable defined on the vertex $i \in V$. QUBO is given by

$$E(\{\boldsymbol{x}\}) = \sum_{i \in V} c_i x_i + \sum_{(i,j) \in E} w_{ij} x_i x_j, \tag{4}$$

$$x_i \in \{0, 1\},\ \forall i \in V, \tag{5}$$

where $\boldsymbol{x}$ represents $|V|$ binary variables in vector form, i.e., $\boldsymbol{x} = (x_1, x_2, \cdots, x_{|V|})$. The coefficients in RHS of Eq. (4) are real values. Using the transformation between the spin variables and the binary variables,

$$x_i = \frac{s_i + 1}{2}, \quad \forall i \in V, \tag{6}$$

the energy function of Ising model and the form of QUBO are equivalent except for the constant value which does not depend on states.

### B. How to represent combinatorial optimization problem by Ising model or QUBO

In this subsection, methods to represent two typical combinatorial optimization problems – number partitioning problem and graph partitioning problem – by the Ising model are shown.

*1) Combinatorial optimization problem without constraints:* We consider the number partitioning problem as an example of combinatorial optimization problems without constraints. The definition of number partitioning problem is as follows. Let $\mathcal{N}_{\rm W}$ be a set of $N$ positive integers, i.e., $\mathcal{N}_{\rm W} = \{n_1, n_2, \cdots, n_N\}$. The purpose of number partitioning problem is to find two disjoint subsets so that the difference between the sum of elements in the subset $\mathcal{N}_{\rm A}$ and the sum of elements in the subset $\mathcal{N}_{\rm B}$ is minimized. The difference is given by

$$D = \left| \sum_{n_i \in \mathcal{N}_{\rm A}} n_i - \sum_{n_i \in \mathcal{N}_{\rm B}} n_i \right|. \tag{7}$$

In order to consider the case that the difference $D$ is minimized, we introduce spin variables $\boldsymbol{s} = (s_1, s_2, \cdots, s_N)$. When $s_i = +1$, the positive integer $n_i$ belongs to the subset $\mathcal{N}_{\rm A}$, whereas when $s_i = -1$, the positive integer $n_i$ belongs to the subset $\mathcal{N}_{\rm B}$. Using the spin variables, we consider the following energy function:

$$E_{\rm NP} = \left( \sum_{i=1}^{N} n_i s_i \right)^2. \tag{8}$$

The above equation is the energy function of Ising model. When the energy function given by Eq. (8) is minimized, the difference given by Eq. (7) is also minimized. Thus, to solve the number partitioning problem, it is sufficient to find a state that minimizes the energy function given by Eq. (8).

*2) Combinatorial optimization problem with an equality constraint:* We consider the graph partitioning problem as an example of a combinatorial optimization problem with an equality constraint. The definition of graph partitioning problem as follows. Let $G_{\rm W} = (V_{\rm W}, E_{\rm W})$ be an undirected graph, where $V_{\rm W}$ and $E_{\rm W}$ are the set of vertices and the set of edges, respectively. Here we assume that the number of vertices $|V_{\rm W}| (= N)$ is even. Let us consider to divide the undirected graph into two disjoint subgraphs $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$, where $V_i$ and $E_i$ are the set of vertices and the set of edges of $G_i$ $(i = 1, 2)$, respectively. The graph partitioning problem is to find two disjoint subgraphs $G_1$ and $G_2$ so that the number of edges between two subgraphs is minimized under the constraint $|V_1| = |V_2|$. To represent the number of edges between two subgraphs, we introduce spin variables $\boldsymbol{s} = (s_1, s_2, \cdots, s_N)$. When $s_i = +1$, the vertex $i$ belongs to the subgraph $G_1$, whereas when $s_i = -1$, the vertex $i$ belongs to the subgraph $G_2$. The number of edges between the two subgraphs is represented by

$$E_{\text{edges}} = \sum_{(i,j)\in E_{\text{W}}} \frac{1 - s_i s_j}{2}. \tag{9}$$

The above equation is the energy function of Ising model.

Next, we construct the energy function of the Ising model corresponding to the given constraint, $|V_1| = |V_2|$. The energy function should be constructed to be minimized when the given constraint is satisfied. That is,

$$E_{\text{constraint}} = \left( \sum_{i \in V_{\text{W}}} s_i \right)^2. \tag{10}$$

The above energy function is minimized and takes zero when the constraint is satisfied. Thus, the total energy function of the Ising model corresponding to the graph partitioning problem is given by

$$E_{\text{total}} = E_{\text{edges}} + \alpha E_{\text{constraint}}, \tag{11}$$

where $\alpha$ is a hyperparameter. When $\alpha$ is sufficiently large, the ground state of the Ising model given by Eq. (11) corresponds to the optimal solution of the original graph partitioning problem. In practice, since the performance of Ising machines depends on the choice of hyperparameter, we should develop an efficient method to tune the hyperparameter.

*3) Other types of combinatorial optimization problems:* In the above, methods to represent two typical combinatorial optimization problems by the Ising model were explained. Some combinatorial optimization problems are more convenient to represent the form of QUBO than the Ising model, e.g., the traveling salesman problem and the quadratic assignment problem. Ising model or QUBO mapping for other combinatorial optimization problems is explained in [11]–[13].

Depending on the combinatorial optimization problem, the products of three or more spin/binary variables appear in the objective function. If this is the case, the reduction of products must be done. Let us consider the product of three binary variables, $x_1 x_2 x_3$, where $x_i \in \{0, 1\}$ for $i = 1, 2, 3$. We introduce an auxiliary binary variable $\tilde{x}_{12}$. If $\tilde{x}_{12} = x_1 x_2$, $x_1 x_2 x_3 = \tilde{x}_{12} x_3$. The following function is minimized and has the value zero when the condition $\tilde{x}_{12} = x_1 x_2$, i.e., $\tilde{x}_{12} = x_1 \wedge x_2$ is satisfied:

$$\text{AND}(\tilde{x}_{12}; x_1, x_2) = x_1 x_2 - 2\tilde{x}_{12}(x_1 + x_2) + 3\tilde{x}_{12}. \tag{12}$$

Using the auxiliary binary variable $\tilde{x}_{12}$, the cost function can be represented by the form of QUBO:

$$x_1 x_2 x_3 = \tilde{x}_{12} x_3 + \lambda \text{AND}(\tilde{x}_{12}; x_1, x_2), \tag{13}$$

where $\lambda$ is a hyperparameter as well as in Eq. (11).

### C. Preprocessing of computation by Ising machines

In the previous subsection, how to represent the combinatorial optimization problem by the Ising model or QUBO is described. Even if the Ising model or QUBO corresponding to the combinatorial optimization problem is prepared, in many cases, it is difficult to directly perform the search of the ground state of the Ising model or QUBO with Ising machines. This difficulty comes from restrictions in Ising machines, e.g., the limited graph structure, the limited number of spins, the limited dynamic range of the coefficients in the Ising model. Among them, some studies on preprocessing to overcome limited graph structures and the limited number of spins have been performed.

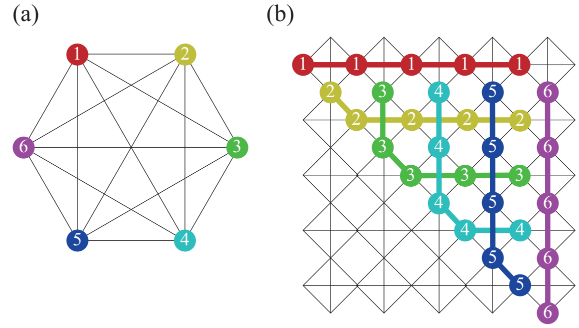


Fig. 2. An example of graph embedding. (a) A logical Ising model with six spins on a complete graph. (b) A physical Ising model corresponding to the logical Ising model depicted in (a) on the king's graph.

Herein, we refer to the Ising model corresponding to the combinatorial optimization problem as *logical* Ising model. The logical Ising model is defined on an undirected complete graph $G_{\text{L}} = (V_{\text{L}}, E_{\text{L}})$, where $V_{\text{L}}$ and $E_{\text{L}}$ are respectively the set of vertices and the set of edges on $G_{\text{L}}$. Since $G_{\text{L}}$ is an undirected complete graph, $(i, j) \in E_{\text{L}}$ for arbitrary $i$ and $j$ except for $i \neq j$. In contrast to the logical Ising model, we introduce *physical* Ising model corresponding to an Ising machine. The physical Ising model is defined on an undirected graph $G_{\text{P}} = (V_{\text{P}}, E_{\text{P}})$, where $V_{\text{P}}$ and $E_{\text{P}}$ are respectively the set of vertices and the set of edges on $G_{\text{P}}$. It should be noted that $G_{\text{P}}$ is not necessarily a complete graph.

First, let us explain preprocessing to overcome limited graph structures. When we solve a combinatorial optimization problem using an Ising machine, we need to perform the graph embedding $G_{\text{L}} \to G_{\text{P}}$ [14]. Figure 2 shows an example of graph embedding. Figure 2 (a) represents a logical Ising model with six spins on a complete graph. Also, Fig. 2 (b) represents a physical Ising model corresponding to the logical Ising model depicted in Fig. 2 (a) on a king's graph. The numbers in the circles in Fig. 2 (b) corresponds to those in Fig. 2 (a). Thick lines connecting the same number of spins in Fig. 2 (b) indicates a strong interaction. When the spins connected by thick lines take the same value, the energy is minimized. Thus, a single spin of the logical Ising model is expressed by multiple spins of the physical Ising model. There are two main types of graph embedding methods. The first one is to generate a graph that interacts with all the spins using the geometric properties of the graph $G_{\text{P}}$ [15]–[18]. The second one is to generate a graph $G_{\text{P}}$ by using heuristics [19], [20].

Next, let us explain preprocessing to overcome the limited number of spins. Since the number of spins in Ising machines is limited in general, we need to decompose the original combinatorial optimization problem to the smaller combinatorial optimization problems which can be inputted at once. Software tools for the decomposition of the Ising model will be introduced in Sec. IV.

### D. Ising machines

By performing the above scheme, a physical Ising model to be input to an Ising machine is generated. Ising machines behave to search the ground state of the Ising model given by Eq. (2) or the QUBO given by Eq. (4). Ising machines can be categorized according to the operating principle from the viewpoint of physics. Ising machines currently being developed are based on simulated annealing, quantum annealing, or a kind of equation of motion. Ising machines operate according to the dynamics in which the vector expressing the probability or the state changes with time. In Sec. III, we explain the theory of simulated annealing and quantum annealing which are mainly used to the operation principle of Ising machines.

### E. Postprocessing of computation by Ising machines

Ising machines do not necessarily obtain the ground state of the Ising model or QUBO. The above fact comes from the following reasons: the behavior of Ising machines based on simulated annealing or quantum annealing is stochastic, and the initial state of Ising machines based on the equations of motion is randomly generated. In addition, although the convergence theorem of simulated annealing and that of quantum annealing exist, the actual computation time is shorter than the time to satisfy the convergence theorem. Thus, we obtain not necessarily the ground state but a lower energy state by using Ising machines in practice. Because of this stochastic behavior, we should develop a method to interpret the obtained state using Ising machines. In the following, we explain two methods which are often used in various cases. The first one is related to the inverse transformation from the obtained state of the physical Ising model to the state of the logical Ising model. The second one is used when we return from the state of the logical Ising model to the solution of the original combinatorial optimization problem.

*1) A method related to the inverse transformation from the physical Ising model to the logical Ising model:* The logical Ising model on $G_{\mathrm{L}}$ corresponding to the combinatorial optimization problem is embedded to the physical Ising model on $G_{\mathrm{P}}$, as shown in Fig. 2. The state of logical Ising model on $G_{\mathrm{L}}$ is obtained by performing an inverse transformation to the obtained state of physical Ising model on $G_{\mathrm{P}}$.

Figure 3 (a) shows an example of a physical Ising model with six spins interacted with each other. As explained in Sec. II-C, spins on the physical Ising model coupled with strong interaction must take the same value. However, some spins may take different values due to the stochastic behavior of Ising machines, as shown in Fig. 3 (a). In this case, the spin of the logical Ising model is determined by the majority vote applying to the obtained state of Ising machines [see Fig. 3 (b)].

*2) A method related to the given constraints:* Next, we explain the postprocessing that is necessary when a state which does not satisfy the given constraints is obtained. Let us consider the case that a combinatorial optimization problem is represented by the Ising model on vertical and horizontal grids. The situation corresponds to some kinds of combinatorial optimization problems such as the traveling salesman problem

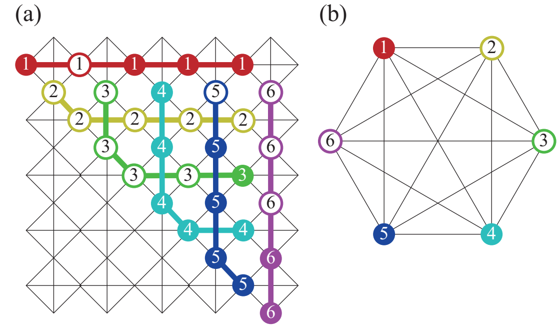

Fig. 3. An example of an inverse transformation from the state in the physical Ising model to the state in logical Ising model. The solid and open circles represent $+1$ state and $-1$ state, respectively. The bold line indicates a strong interaction. (a) An example of a physical Ising model with six spins interacted with each other. (b) The logical Ising model with six spins. The spins are decided by the majority vote.

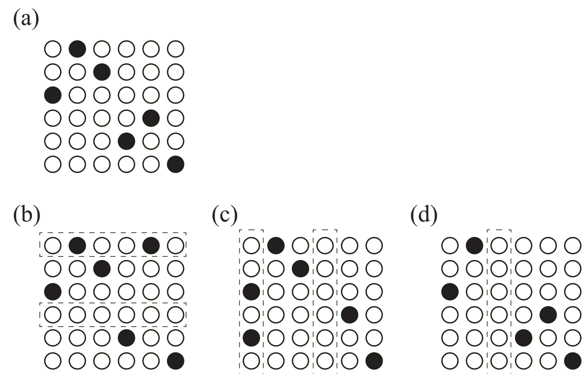

Fig. 4. The solid and open circles represent $+1$ state and $-1$ state, respectively. (a) Only one spin in each row and column takes +1 state. (b), (c), (d) Examples of the violation of constraints. The dotted rectangles indicate the position of the violation of constraints.

and quadratic assignment problem [21]. Suppose the constraint is satisfied when there is a $+1$ spin in each column and row [see Fig. 4 (a)]. When the states depicted in Figs. 4 (b), (c) and (d) are obtained by using Ising machines, a state satisfying the given constraints can be generated by inverting the spin that causes the violation of constraints. The method is referred to as the interpretation method [21].

## III. Theory of Ising machines

As mentioned in Sec. II-D, Ising machines can be categorized according to the operation principle. In this section, we explain two mainly operation principles of Ising machines – simulated annealing and quantum annealing.

### A. Simulated annealing

Simulated annealing is a metaheuristic for optimization problem and is applied to the Markov chain Monte Carlo (MCMC) method [22]–[24]. In the simulated annealing based on MCMC, the state changes stochastically according to a designed transition probability. Herein we focus on the MCMC-based simulated annealing for searching the ground state of the Ising model given by Eq. (2). The procedure of the simulated annealing is as follows:

**Step 1** A random state is prepared as an initial state, and the parameter called temperature $T$ is set to a large value compared to the coefficients in the energy function.
**Step 2** A spin is chosen randomly or sequentially.
**Step 3** Update the state of the chosen spin according to the transition probability which satisfies the balance condition. Well-known methods to construct the transition probability are called the heat-bath method and the Metropolis method.
**Step 4** Step 2 and step 3 are repeatedly performed $t_{\text{inner}}$ times.
**Step 5** The parameter $T$ decreases, e.g., $T \leftarrow rT$, where $r < 1$.
**Step 6** The above procedure from step 2 to step 5 are repeatedly performed $t_{\text{outer}}$ times.

Here we explain the detail of step 3. In MCMC, the master equation, which is an equation of stochastic dynamics, is used. Let $\boldsymbol{s}_k$ be a macroscopic state, where $1 \leq k \leq 2^N$ when the number of spins is $N$. Using the existence probability of state $\boldsymbol{s}_k$ at time $t$, $P(\boldsymbol{s}_k, t)$, the master equation at time $t$ is given by

$$P(\boldsymbol{s}_k, t+\Delta t) = -\sum_{\ell \neq k} w_{k\to\ell} \Delta t P(\boldsymbol{s}_k, t) + \sum_{\ell \neq k} w_{\ell\to k} \Delta t P(\boldsymbol{s}_\ell, t) + P(\boldsymbol{s}_k, t), \quad (14)$$

where $w_{k\to\ell}$ is a transition probability from the state $\boldsymbol{s}_k$ to the state $\boldsymbol{s}_\ell$. To converge to the equilibrium probability distribution at the long-time limit of the above stochastic process, the following equation called *balance condition* must be satisfied:

$$\sum_{\ell \neq k} w_{\ell\to k} P_{\text{eq}}(\boldsymbol{s}_\ell; T) = \sum_{k \neq \ell} w_{k\to\ell} P_{\text{eq}}(\boldsymbol{s}_k; T), \quad (15)$$

$$P_{\text{eq}}(\boldsymbol{s}_k; T) = \frac{\exp[-E(\boldsymbol{s}_k)/T]}{\sum_k \exp[-E(\boldsymbol{s}_k/T)]} \quad (16)$$

where $P_{\text{eq}}(\boldsymbol{s}_k; T)$ is the equilibrium probability of the state $\boldsymbol{s}_k$ at temperature $T$.

Two well-known and well-used methods to determine the transition probability $w_{k\to\ell}$; the heat-bath method and the Metropolis method. The transition probability according to the heat-bath method is given by

$$w_{k\to\ell} = \frac{\exp[-E(\boldsymbol{s}_\ell)/T]}{\exp[-E(\boldsymbol{s}_k)/T] + \exp[-E(\boldsymbol{s}_\ell)/T]}. \quad (17)$$

On the other hand, the transition probability according to the Metropolis method is given by

$$w_{k\to\ell} = \begin{cases} \exp\{-[E(\boldsymbol{s}_\ell) - (\boldsymbol{s}_k)]/T\} & [E(\boldsymbol{s}_\ell) > E(\boldsymbol{s}_k)] \\ 1 & [E(\boldsymbol{s}_\ell) \leq E(\boldsymbol{s}_k)] \end{cases}. \quad (18)$$

Here both the transition probability based on the heat-bath method and that based on the Metropolis method are satisfied by the balance condition.

### B. Quantum annealing

Quantum annealing is a metaheuristic similar to the simulated annealing [2]. In the simulated annealing, we introduce the temperature as a source of fluctuation, whereas a quantum field is introduced as a source of fluctuation in the quantum annealing [25]–[31]. In the original paper on quantum annealing [2], the authors compared the performance of simulated annealing and that of quantum annealing for searching the ground state of some Ising models. A similar proposal was made in Ref. [3], [4], which is called adiabatic quantum computation.

Here we define the following matrices called the Pauli matrices:

$$\sigma^x = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \quad \sigma^z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}, \quad (19)$$

Here we introduce the *classical* Hamiltonian corresponding to the Ising model given by Eq. (2) as follows:

$$\mathcal{H}_{\text{c}} = \sum_{i \in V} h_i \sigma_i^z + \sum_{(i,j)\in E} J_{ij} \sigma_i^z \sigma_j^z, \quad (20)$$

$$\sigma_i^z = \bigotimes_{k=1}^{i-1} I_2 \otimes \sigma^z \otimes \left( \bigotimes_{k=i+1}^{N} I_2 \right), \quad (21)$$

$$\sigma_i^z \sigma_j^z = \bigotimes_{k=1}^{i-1} I_2 \otimes \sigma^z \otimes \left( \bigotimes_{k=i+1}^{j-1} I_2 \right) \otimes \sigma^z \otimes \left( \bigotimes_{k=j+1}^{N} I_2 \right), \quad (22)$$

where $I_2$ is the identity matrix of $2 \times 2$. Here $\otimes$ represents the Kronecker product. Thus, the classical Hamiltonian is a matrix of $2^N \times 2^N$. Next, we introduce a quantum field which causes fluctuation. A *quantum* Hamiltonian is given by

$$\mathcal{H}_{\text{q}} = \sum_{i \in V} \sigma_i^x, \quad (23)$$

$$\sigma_i^x = \bigotimes_{k=1}^{i-1} I_2 \otimes \sigma^x \otimes \left( \bigotimes_{k=i+1}^{N} I_2 \right). \quad (24)$$

Thus the quantum Hamiltonian is also a matrix of $2^N \times 2^N$ as well as the classical Hamiltonian. By using $\mathcal{H}_{\text{c}}$ and $\mathcal{H}_{\text{q}}$, the total Hamiltonian is generated as follows:

$$\mathcal{H}(t) = A(t)\mathcal{H}_{\text{c}} + B(t)\mathcal{H}_{\text{q}}, \quad (25)$$

where $A(t)$ and $B(t)$ are a monotonically increasing function and a monotonically decreasing function, respectively. Quantum annealing is realized by the Schrödinger equation:

$$\frac{d}{dt} |\Psi(t)\rangle = \mathcal{H}(t) |\Psi(t)\rangle, \quad (26)$$

where $|\Psi(t)\rangle$ is a wavefunction which is a $2^N$-dimensional complex vector. According to the principle of quantum physics, the norm of the wavefunction corresponds to the probability amplitude.

Equation (26) corresponds to an ideal Ising machine based on quantum annealing. In an actual Ising machine based on quantum annealing includes temperature effects and external noise effects. Unfortunately, no theoretical equation has been established that perfectly reproduces the behavior of an actual Ising machine based on quantum annealing. Thus, theoretical analysis toward the understanding of an actual Ising machine based on quantum annealing has been performed from the viewpoint of quantum statistical physics [32].

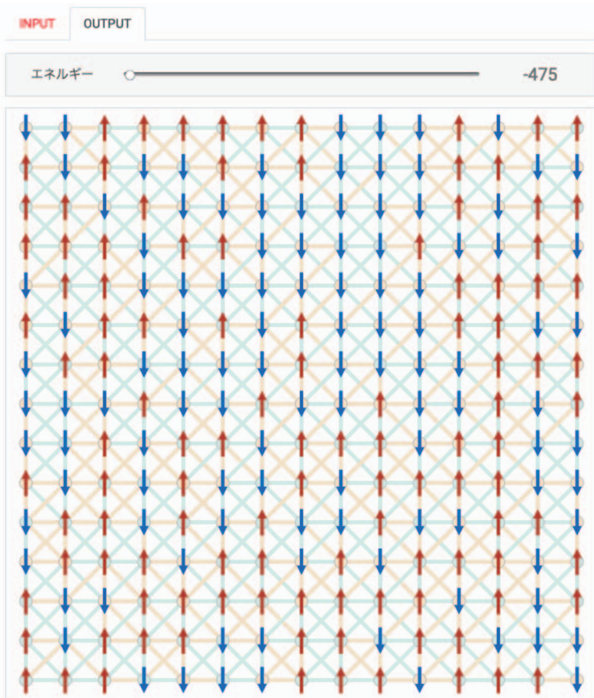


Fig. 5. “Ising editor” on the website called “Annealing Cloud Web” [33].

## IV. Software for Ising machines

In the previous sections, we explain how to solve combinatorial optimization problems by using Ising machines. In this section, we show the current state of software development for using Ising machines.

A website called “Annealing Cloud Web” is provided for beginners [33]. On the website, we can use the user interface for intuitive input of the Ising model and the demonstration of CMOS annealing machines. Figure 5 shows an example of the obtained result by using the CMOS annealing machine in “Ising editor” on the website. On the website, some demonstrations, such as image restoration and network robustness construction, can be done. The website also provides an operation of the CMOS annealing machine via WebAPI.

Some software for developers is also available. D-Wave Systems, a developer of Ising machines based on quantum annealing, has released a software called Ocean Software [34]. The software supports an environment that can easily implement a combinatorial optimization problem using the Ising model and QUBO and the simulated annealing for the implemented Ising model or QUBO. Software for graph embedding and decomposition of large-scaled problem explained in Sec. II-C is included in Ocean Software [35]. In addition, software for Ising machines using Julia [36], and PyQUBO have been developed as a domain-specific language that enables automatic generation of the Ising model and QUBO [13], [37].

The commissioned project of the New Energy and Industrial Technology Development Organization (NEDO), “research and development of Ising machine common software platform”, started in 2018. In the platform, software and middleware connecting the actual combinatorial optimization problem and Ising machines will be prepared.

## V. Application search using Ising machines

The development of Ising machines and that of software for Ising machines are progressing. From such a background, studies on application searches have been done to examine in which scene Ising machines should be used. Applications for Ising machines are expected to be combinatorial optimization problems, machine learning, and model simulation in physics and chemistry. In the following, we explain the above three topics.

### A. Combinatorial optimization problems

Experts in various fields have done studies on combinatorial optimization problems using Ising machines, e.g., the route selection problem aimed at avoiding traffic jam by an automaker [38], route optimization of automatic guided vehicles in factory by an auto parts manufacturer [39], advertisement delivery optimization by an advertising company [13], rectangular packing problem [40] and slot placement problem [21] by a group of researchers in the field of integrated circuits. It should be noted that a method for a combinatorial optimization problem in an area is diverted to a method for a combinatorial optimization problem in another area. For example, the Ising model representation of route optimization of automatic guided vehicles in factory [39] is inspired by the study on the Ising model representation of the route selection problem proposed in Ref. [38]. Since the Ising model is simple, the formulation in one field can be transferred to another field.

### B. Machine learning

Two types of studies in the application of Ising machines toward speed up machine learning exist. One is to process learning in machine learning by using Ising machines, whereas the other is to process inference in machine learning by using Ising machines. In the former case, the motivation is to minimize the loss function defined by the difference between real data and predicted data. Typical examples of studies on the former case are the application to some machine learning methods for feature selection, e.g., the proposal of QBoost which is a modified method of boosting to utilize Ising machines [41], [42], image analysis using QBoost [43], and nonnegative binary matrix factorization [44]. In the latter case, the motivation is to minimize the prediction function. An example of studies on the latter case is the black-box optimization by using Ising machines [45].

### C. Model simulation in physics and chemistry

Ising model, which is the input format of Ising machines, is a model representing magnetic materials. Thus, phase transitions in random magnetic systems are simulated by using an Ising machine based on quantum annealing [46]. Moreover, the Ising model is the basis of various theoretical models in statistical physics. In other words, various theoretical models in statistical physics can be constructed by generalizing the Ising model. By using the above concept, an exotic phenomenon called topological phase transition was observed by using an Ising machine based on quantum annealing [47]. In addition, application to quantum chemistry has also been reported. The calculation in quantum chemistry uses the energy function representing the electron-electron interaction. Since the energy function can be approximated as the Ising model, the quantum chemical calculations on molecules were done [48], [49].

## VI. Conclusion and future perspective

In this paper, we introduced the development of Ising machines, the development of software for Ising machines, and applications using Ising machines. The operation principle of Ising machines was also described in detail. Research and development of the Ising machine and its surrounding areas are currently underway. From the hardware development aspect, it is significant to increase the number of spins that can be used at once. Next, from the viewpoint of software development, it is important to develop software that compensates for the shortcomings of hardware and software that incorporates algorithms to extract the potential of Ising machines. Moreover, in terms of applications, it is necessary to study the superiority gained by using Ising machines by comparison with existing methods and to prepare for the coming era of large-scale Ising machines.

## Acknowledgement

This paper is based on the results obtained from a project commissioned by the New Energy and Industrial Technology Development Organization (NEDO).

## References

[1] 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, pp. 194-198, 2011.

[2] T. Kadowaki and H. Nishimori, "Quantum annealing in the transverse Ising model," Physical Review E, Vol. 58, pp. 5355-5363, 1998.

[3] E. Farhi, J. Goldstone, S. Gutmann and M. Sipser, "Quantum computation by adiabatic evolution," arXiv, quant-ph/0001106, 2000.

[4] E. Farhi, J. Goldstone, S. Gutmann, J. Lapan, A. Lundgren, and D. Preda, "A quantum adiabatic evolution algorithm applied to random instances of an NP-complete problem," Science, Vol. 292, pp. 472-476, 2001.

[5] 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-Stae Circuits, Vol. 51, pp. 303-309, 2016.

[6] 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. Kawarabayashi, K. Inoue, S. Utsunomiya, and H. Takesue, "A coherent Ising machine for 2000-node optimization problems," Science, Vol. 354 pp. 603-606, 2016.

[7] 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 pp. 614-617, 2016.

[8] 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," Conference on Complex, Intelligent, and Software Intensive Systems, pp. 432-438, 2017.

[9] 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," Journal of the Physical Society of Japan, Vol. 88, pp. 061012-1-7, 2019.

[10] H. Goto, K. Tatsumura, and A. R. Dixon, "Combinatorial optimization by simulating adiabatic bifurcations in nonlinear Hamiltonian systems," Science advances, Vol. 5, eaav2372-1-8, 2019.

[11] A. Lucas, "Ising formulations of many NP problems," Frontiers in Physics, Vol. 2, pp. 5-1-15, 2014.

[12] S. Tanaka, R. Tamura, and B. K. Chakrabarti, Quantum spin glasses, annealing and computation, Cambridge University Press, 2017.

[13] K. Tanahashi, S. Takayanagi, T. Motohashi, and S. Tanaka, "Application of Ising machines and a software development for Ising machines," Journal of the Physical Society of Japan, Vol. 88, p. 061010-1-10, 2019.

[14] V. Choi, "Minor-embedding in adiabatic quantum computation: I. The parameter setting problem," Quantum Information Processing, Vol. 7, pp. 193-209, 2008.

[15] V. Choi, "Minor-embedding in adiabatic quantum computation: II. Minor-universal graph design," Quantum Information Processing, Vol. 10, pp. 343-353, 2011.

[16] T. Boothby, A. D. King, and A. Roy "Fast clique minor generation in Chimera qubit connectivity graphs," Quantum Information Processing, Vol. 15, pp. 495-508, 2016.

[17] N. Dattani and N. Chancellor, "Embedding quadratization gadgets on Chimera and Pegasus graphs," arXiv arXiv:1901.07676, 2019.

[18] D. Oku, K. Terada, M. Hayashi, M. Yamaoka, S. Tanaka, and N. Togawa, "A fully-connected Ising model embedding method and its evaluation for CMOS annealing machines," IEICE Transactions on Information and Systems, Vol. 102, 1696-1706, 2019.

[19] J. Cai, W. G. Macready, and A. Roy, "A practical heuristic for finding graph minors," arXiv arXiv:1406.2741, 2014.

[20] S. Okada, M. Ohzeki, M. Terabe, S. Taguchi, "Improving solutions by embedding larger subproblems in a D-Wave quantum annealer," Scientific Reports, Vol. 9, pp. 2098-1-10, 2019.

[21] S. Kanamaru, D. Oku, M. Tawada, S. Tanaka, M. Hayashi, M. Yamaoka, M. Yanagisawa, and N. Togawa, "Efficient Ising model mapping to solving slot placement problem," Proceedings of 2019 IEEE International Conference on Consumer Electronics (ICCE), 2019.

[22] S. Kirkpatrick, C. D. Gelatt Jr., M. P. Vecchi, "Optimization by simulated annealing," Science, Vol. 220, pp. 671-680, 1983.

[23] D. S. Johnson, C. R. Aragon, L. A. McGeoch, and C. Schevon, "Optimization by simulated annealing: an experimental evaluation; Part I, graph partitioning," Operations Research, Vol. 37, pp. 865-892, 1989.

[24] D. S. Johnson, C. R. Aragon, L. A. McGeoch, and C. Schevon, "Optimization by simulated annealing: an experimental evaluation; Part II, graph coloring and number partitioning," Operations Research, Vol. 39, pp. 378-406, 1991.

[25] K. Kurihara, S. Tanaka, and S. Miyashita, "Quantum annealing for clustering," Proceedings of the 25th Conference on Uncertainty in Artificial Intelligence (UAI 2009), 2009.

[26] I. Sato, K. Kurihara, S. Tanaka, H. Nakagawa, and S. Miyashita, "Quantum annealing for variational Bayes inference," Proceedings of the 25th Conference on Uncertainty in Artificial Intelligence (UAI 2009), 2009.

[27] S. Miyashita, S. Tanaka, H. de Raedt, and B. Barbara, "Quantum response to time-dependent external fields," Journal of Physics: Conference Series, Vol. 143, pp. 012005-1-10, 2009.

[28] S. Tanaka and S. Miyashita, "Nonmonotonic dynamics in a frustrated Ising model with time-dependent transverse field," Physical Review E, Vol. 81, pp. 051138-1-8, 2010.

[29] I. Sato, S. Tanaka, K. Kurihara, S. Miyashita, and H. Nakagawa, "Quantum annealing for Dirichlet process mixture models with applications to network clustering," Neurocomputing, Vol. 121, pp. 523-531, 2013

[30] M. Nakahara and S. Tanaka, Lectures on quantum computing, thermodynamics and statistical physics, World Scientific, 2013.

[31] Y. Seki, S. Tanaka, and S. Kawabata, "Quantum phase transition in fully connected quantum Wajnflasz—Pick model," Journal of the Physical Society of Japan, Vol. 88, p. 054006-1-12, 2019.

[32] T. Kadowaki and M. Ohzeki, "Experimental and theoretical study of thermodynamic effects in a quantum annealer," Journal of the Physical Society of Japan, Vol. 88, pp. 061008-1-9, 2019.

[33] https://annealing-cloud.com

[34] https://docs.ocean.dwavesys.com/en/latest/

[35] https://docs.ocean.dwavesys.com/projects/qbsolv/en/latest/

[36] D. O'Malley and V. V. Vesselinov, "ToQ.jl: A high-level programming language for D-Wave machines based on Julia," 2016 IEEE High Performance Extreme Computing Conference (HPEC), 2016.

[37] https://pyqubo.readthedocs.io/en/latest/

[38] F. Neukart, G. Compostella, C. Seidel, D. Von Dollen, S. Yarkoni, and B. Parney, "Traffic flow optimization using a quantum annealer," Frontiers in ICT, Vol. 4, pp. 29-1-6, 2017.

[39] M. Ohzeki, A. Miki, M. J. Miyama, and M. Terabe, "Control of automated guided vehicles without collision by quantum annealer and digital devices," Frontiers in Computer Science, Vol. 1, pp. 9-1-9, 2019.

[40] 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," Proceedings of 2018 International Symposium on VLSI Design, Automation and Test (VLSI-DAT), 2018.

[41] H. Neven, V. S. Denchev, M. Drew-Brook, J. Zhang, W. G. Macready, and G. Rose, "Binary Classification using Hardware Implementation of Quantum Annealing," NIPS 2009 Demonstration, 2009.

[42] H. Neven, V. S. Denchev, G. Rose, and W. G. Macready, "QBoost: Large Scale Classifier Training with Adiabatic Quantum Optimization," JMLR Workshop and Conference Proceedings, Vol. 25, pp. 333-348, 2012.

[43] E. Boyda S. Basu, S. Ganguly, A. Michaelis, S. Mukhopadhyay, and R. R. Nemani, "Deploying a quantum annealing processor to detect tree cover in aerial imagery of California," PLOS ONE, Vol. 12, e0172505-1-22, 2017.

[44] D. O'Malley, V. V. Vesselinov, B. S. Alexandrov, and L. B. Alexandrov, "Nonnegative/Binary matrix factorization with a D-Wave quantum annealer," PLOS ONE, Vol. 13, e0206653-1-12, 2018.

[45] K. Kitai, J. Guo, S. Ju, S. Tanaka, K. Tsuda, J. Shiomi, and R. Tamura, "Expanding the horizon of automated metamaterials discovery via quantum annealing," arXiv arXiv:1902.06573, 2019.

[46] R. Harris, Y. Sato, A. J. Berkley, M. Reis, F. Altomare, M. H. Amin, K. Boothby, P. Bunyk, C. Deng, C. Enderud, S. Huang, E. Hoskinson, M. W. Johnson, E. Ladizinsky, N. Ladizinsky, T. Lanting, R. Li, T. Medina, R. Molavi, R. Neufeld, T. Oh, I. Pavlov, I. Perminov, G. Poulin-Lamarre, C. Rich, A. Smirnov, L. Swenson, N. Tsai, M. Volkmann, J. Whittaker, and J. Yao, "Phase transitions in a programmable quantum spin glass simulator," Science, Vol. 361, pp. 162-165, 2018.

[47] 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. Fréchette, E. Hoskinson, N. Ladizinsky, T. Oh, G. Poulin-Lamarre, C. Rich, Y. Sato, A. Y. Smirnov, L. J. Swenson, M. H. Volkmann, 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, pp. 456-460, 2018.

[48] R. Xia, T. Bian, and S. Kais, "Electronic structure calculations and the Ising Hamiltonian," Journal of Physical Chemistry B, Vol. 122, pp. 3384-3395, 2018.

[49] M. Streif, F. Neukart, and M. Leib, "Solving quantum chemistry problems with a D-Wave quantum annealer," arXiv arXiv:1811.05256, 2018.