IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received 28 July 2023, accepted 8 September 2023, date of publication 12 September 2023, date of current version 19 September 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3314498

RESEARCH ARTICLE

Trip Planning Based on subQUBO Annealing

TATSUYA NOGUCHI, KEISUKE FUKADA, (Member, IEEE), SIYA BAO, (Member, IEEE), AND NOZOMU TOGAWA, (Member, IEEE)

Department of Computer Science and Communications Engineering, Waseda University, Shinjuku, Tokyo 169-8555, Japan

Corresponding authors: Tatsuya Noguchi (tatsuya.noguchi@togawa.cs.waseda.ac.jp) and Nozomu Togawa (ntogawa@waseda.jp)

This work was supported in part by the Core Research for Evolutional Science and Technology (CREST) Program of the Japan Science and Technology Agency (JST), Japan, under Grant JPMJCR19K4.

ABSTRACT The trip planning problem (TPP) can be formulated as a combinatorial optimization problem that searches for the best route to visit a series of landmarks and hotels. Meanwhile, Ising machines have attracted attention due to their efficiency in solving combinatorial optimization problems. The Ising machines solve the combinatorial optimization problems by transforming the problems into quadratic unconstrained binary optimization (QUBO) models. However, the possible input QUBO size of current Ising machines is quite limited. Thus, it is hard to directly embed a large-scale TPP onto the current Ising machines. In this paper, we propose a novel subQUBO annealing method based on the combined variable selection method to solve the TPP. The proposed method finds a quasi-optimal solution to a large problem by repeatedly partitioning the original QUBO model into small subQUBOs that can be embedded onto the Ising machine. Specifically, to construct a subQUBO, we select variables from the original QUBO model, which have small deviation values. Further, we select variables randomly from the original QUBO model, so as not to fall into the local optimum. We have conducted an evaluation experiment using Ising machines on TPP and confirmed that the proposed method outperforms the state-of-the-art methods in terms of POI satisfaction and POI cost.

INDEX TERMS Trip planning problem, Ising machine, quantum computer, Ising model, QUBO model, subQUBO.

I. INTRODUCTION

A. TRIP PLANNING PROBLEM

The trip planning problem (TPP) in this paper is the problem that finds the optimal route to visit a series of point-of-interests (POIs) and hotels over multiple days [1], [2], [3], [4]. For example, Sylejmani et al. [1] presented a method that solves the trip planning problem using a heuristic algorithm based on tabu search; Saeki et al. [2] presented a method for planning a multi-objective trip using antcolony optimization; Fournier et al. [3] showed a method that solves the bus passenger trip planning problem using an A*-guided and Pareto dominance-based heuristic; Garcia et al. [4] presented two different methods to solving the time-dependent team orienteering problem with time windows; Shuai et al. [5] presented a method that solves multiple traveling salesman problems by applying an NSGA-II framework; He et al. [6] presented a hybrid method based on tabu search and intra-tour optimization to solve the multiple traveling salesman problems.

B. ISING MACHINE

A combinatorial optimization problem [7] finds the optimal combination among a large number of options under various constraints. Famous combinational optimization problems include the traveling salesman problem, knapsack problem and integer programming problem. In most combinatorial optimization problem, as the number of combinations increases, the variable combination increases exponentially, which makes it difficult for a von Neumann architecture to find an optimal or a quasi-optimal solution in a realistic time. On the other hand, Ising machines have recently been developed which are specialized for solving combinatorial optimization problems [8], [9]. The Ising machines transform the original problem onto the Ising models or quadratic unconstrained binary optimization (QUBO) models [10], and the Ising machines search for the ground state of the Ising or QUBO models, that leads to the optimal solution

The associate editor coordinating the review of this manuscript and approving it for publication was Wei Huang.

of the original problem. Many studies have confirmed the superiority of Ising machines in solving combinational optimization problems [11], [12], [13], [14], [15], [16], [17], [18], [19].

On the other hand, the input QUBO size of current Ising machines and quantum computers is quite limited [20]. For example, D-Wave Advantage has an input QUBO size from 100 to 150 variables [21], and IBM Quantum Eagle has a 127-bit input size [22]. In other words, it is difficulty to directly solve a large-scale TPP by current Ising machines as the size of the POIs and hotels becomes large.

C. SUBQUBO ANNEALING

To deal with the problem above, three have been proposed several methods to partition a QUBO model and perform a subQUBO annealing. *qbsolv* [23] partitions a QUBO into subQUBOs based on the *impact* value, which is defined by how much the objective function increases when the binary variable is flipped. The subQUBOs extracted based on the impact value are solved by the subsolver, and the entire QUBO is optimized by tabu search, which updates the current value and enables efficient search. However, since the impact value is the degree of influence on the current solution, it is not globally optimized and is likely to lead to a local optima. Rosenberg et al. [24] presented a meta-heuristic method for solving larger problems by repeatedly solving sub-problems while keeping the remaining variables fixed. However, the selection of variables, such as a gain-based selection and a fusion-guides selection, uses information from the current solution, and then, if the solution is biased, it may still fall into a local solution. Jo et al. [25] presented a problem-focused embedded partitioning technique that improves the obtained partial solution by intentionally worsening typical embedding quality measures. Bass et al. [26] presented a heterogeneous computation stack combining quantum annealing and classical machine learning. With this method of variable selection, the variables selected are fixed and may not reach the globally optimal solution due to the local solution. Atobe et al. [27] presented a hybrid annealing method using multiple solution instances with subQUBO extraction based on a theoretical background. However, because it uses the variation in the binary variables of the quasi-optimal solutions generated by the classical computer, if the classical computer is stuck on a local solution, the subQUBO to be extracted may be biased and global search cannot be performed.

D. OUR PROPOSAL

Based on the discussions above, we propose a method for solving large-scale TPPs by the subQUBO annealing method based on the combined variable selection. The proposed subQUBO annealing method repeatedly partitions an original large QUBO model into several small subQUBOs. Then, these subQUBOs are solved by an Ising machine and their results are collected. Finally, we can find a quasi-optimal solution to the original TPP. The proposed method extracts subQUBOs by the following two steps: (1) local-solution-based variable selection which aims at extracting potential quasi-optimal solution based on statistical analysis and (2) random variable selection which aims at avoiding a local optimum. By introducing these two steps, the proposed method can find a globally optimal solution without falling into a local minima.

For evaluation, the proposed method was evaluated on two types of real Ising machines for the TPP. As a result, the proposed method outperforms the existing methods in terms of POI satisfaction and POI cost.

E. CONTRIBUTION

The contributions of this paper are shown as follows:

1) We propose a subQUBO annealing method that solves the TPP by partitioning the QUBO model into sub-QUBOs. The proposed subQUBO annealing method is composed of (1) local-solution-based variable selection and (2) random variable selection which guarantee to generate a better quasi-optimal solution.
2) We evaluated the proposed method by real Ising machines in several real datasets. Evaluation results demonstrate that the proposed method outperforms the existing methods in terms of POI satisfaction and POI cost.

The rest of the paper is organized as follows: In Section II, we present the definitions of the Ising model and the QUBO model. In Section III, we define the TPP and formulate it as the QUBO model. In Section IV, we propose a new subQUBO annealing method for TPP. In Section V, the proposed method is evaluated using real Ising machines. Section VI gives the the conclusion of this paper.

II. ISING MODEL AND QUBO MODEL

This section briefly describes the Ising model and the QUBO model.

A. ISING MODEL

The Ising model is a model with magnetic properties in statistical mechanics. Fig. 1 shows an example of the Ising model. The Ising model consists of spins in the positive status of which value is $+1$ or negative status of which value is -1. Also, there are interactions $a_{i,j}$ between two spins, and an external magnetic field b_i acting on each single spin.

Ising machines search for an optimal solution by finding the combination of σ_i, called the ground state, that minimizes the energy function by Eq. (1). The energy function of the Ising model $E(\boldsymbol{\sigma})$ is represented as follows [28]:

$$E(\boldsymbol{\sigma}) = \sum_{i=1}^{n-1} \sum_{j>i}^{n} a_{i,j}\sigma_i\sigma_j + \sum_{i=1}^{n} b_i\sigma_i \tag{1}$$

where $\boldsymbol{\sigma} = \{\sigma_1, \sigma_2, \ldots, \sigma_n\}$ is the set of n spins and $\sigma_i \in \{-1, 1\}$. $a_{i,j}$ is the interaction between two spins σ_i and σ_j, and b_i is the external magnetic field of spin σ_i.

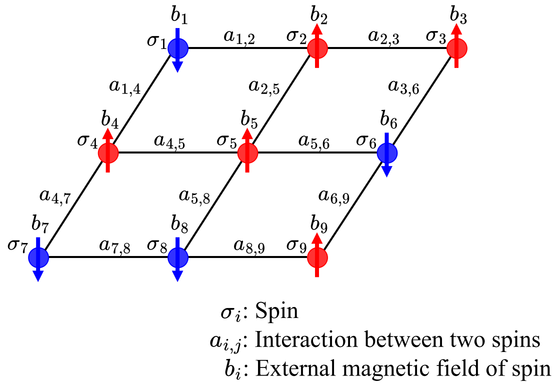


FIGURE 1. Example of the Ising model.

B. QUBO MODEL

The QUBO model is equivalent to the Ising model and the energy function $f(\boldsymbol{x})$ is represented as follows [29]:

$$f(\boldsymbol{x}) = \boldsymbol{x}^t Q \boldsymbol{x} = \sum_{i=1}^{n-1} \sum_{j \geq i}^{n} q_{i,j} x_i x_j \tag{2}$$

where $\boldsymbol{x} = \{x_1, x_2, \ldots, x_n\}$ is the set of n binary variables and $x_i \in \{0, 1\}$. Q is an $n \times n$ matrix of $q_{i,j}$, and $q_{i,j}$ is the interaction coefficient between binary variables x_i and x_j.

The binary variable x_i can be linearly transformed to the corresponding σ_i by Eq. (3).

$$\sigma_i = 2x_i - 1 \tag{3}$$

Since the QUBO model has an equivalent structure to the Ising model, it is also possible to map the combinational optimization problem onto the QUBO models.

Hereafter, the QUBO model is used in this paper, and the size of the QUBO model refers to the number of binary variables in Eq. (2).

III. TRIP PLANNING PROBLEM AND ITS QUBO MODEL FORMULATION

A. TRIP PLANNING PROBLEM (TPP)

The TPP finds the best route to visit multiple POIs and hotels over days.

Let $C = \{c_1, c_2, \ldots, c_p\}$ be the set of p POIs and $L = \{h_1, h_2, \ldots, h_q\}$ be the set of q hotels. Each POI c_k $(1 \leq k \leq p)$ has a satisfaction rate r_k, a staying cost e_{c_k}, and duration time of stay t_k, and each hotel h_l $(1 \leq l \leq q)$ is given an accommodation fee e_{h_l}. We attempt to find the routes from day 0 to day $(m+1)$. Day 0 is the arrival day and the route for day 0 is the shortest route from the arrival ariport to the hotel to stay at day 0. Days 1 to m are sightseeing days, and for each sightseeing day i $(1 \leq i \leq m)$, we find the route which starts from the hotel stayed on day $(i-1)$, visits a series of POIs and arrives at the hotel to stay on day i. Day $(m+1)$ is the departure day to the departure airport, and we find the shortest route between the hotel stayed on day m and the airport. The arrival and departure airports are the same.

We hence explain three important and practical factors in the TPP:

1) Total trip cost is the sum of POI visit costs and hotel stay costs over all the days.
2) Total POI satisfaction is the total ratings of POIs to be visited over all the sightseeing days.
3) Total trip time is the total time spent at POIs, travel time between POI and hotel, and travel time between POIs for each day.

Based on the descriptions on the three factors above, the objective and constrains for the TPP are shown as follows:

1) Objective 1: Maximize the total POI satisfaction.
2) Objective 2: Minimize the total trip cost.
3) Constraint 1: A user cannot visit each POI no more than once over trip days.
4) Constraint 2: A user must not visit more than one POI at the same time nor stay more than one hotel at the same time.
5) Constraint 3: Travel time for each day must not exceed the time limit.

Then the TPP is defined as follows:

Definition 1: For given m trip days and trip area including a set C of POIs and a set L of hotels, the TPP is to find the route over m days, satisfying Constraint 1, Constraint 2, and Constraint 3 above, so as to maximize Objective 1 and minimize Objective 2.

B. QUBO MODEL MAPPING OF TPP

In this section, we describe how to map the TPP to the QUBO model based on [30]. We define two type of binary variables $x_{i,j,k}$ and $y_{i,k}$ as Eq. (4) and Eq. (5), respectively.

$$x_{i,j,k} = \begin{cases} 1 & \text{if POI } c_k \text{ is visited at time } j \text{ on day } i \\ 0 & \text{otherwise} \end{cases} \tag{4}$$

$$y_{i,k} = \begin{cases} 1 & \text{if hotel } h_k \text{ is used on day } i \\ 0 & \text{otherwise} \end{cases} \tag{5}$$

Fig. 2 shows an example of mapping the TPP onto the QUBO model. The energy function H of the TPP is defined by Eq. (6).

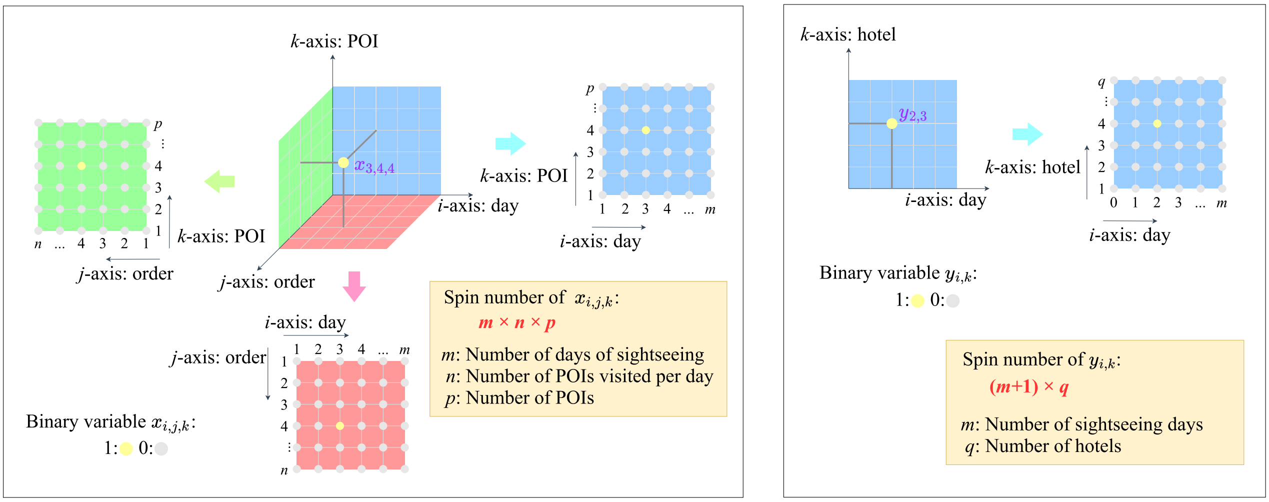
$$H(\boldsymbol{x}, \boldsymbol{y}) = \alpha H_A + \beta H_B + \gamma H_C + \delta H_D + \epsilon H_E \tag{6}$$

where α, β, γ, δ, and ϵ are positive hyperparameters, $\boldsymbol{x} = \{x_{i,j,k}\}$, and $\boldsymbol{y} = \{y_{i,k}\}$. When we input the QUBO model in Eq. (6) into an Ising machine, it searches for the solution of $\boldsymbol{x}$ and $\boldsymbol{y}$ minimizing Eq. (6).

1) OBJECTIVE FUNCTION FOR TOTAL POI SATISFACTION H_A

The objective function H_A is a function to maximize the total visited POI satisfaction from day 1 to day m by the sum of all ratings r_k of POIs visited. H_A is expressed by Eq. (7).

$$H_A = -\sum_{i=1}^{m} \sum_{j=1}^{n} \sum_{k=1}^{p} r_k x_{i,j,k} \tag{7}$$



(a) QUBO model of binary variable $x_{i,j,k}$.

(b) QUBO model of binary variable $y_{i,k}$.

FIGURE 2. Two QUBO models in the TPP [30].

2) OBJECTIVE FUNCTION FOR TOTAL TRIP COST H_B

The objective function H_B is a function to minimize the total trip cost, by the sum of the staying cost e_{c_k} at each POI visited from day 1 to day m and the accommodation fee e_{h_k} at each hotel from day 0 to day m. H_B is expressed by Eq. (8).

$$H_B = \sum_{i=1}^{m}\sum_{j=1}^{n}\sum_{k=1}^{p} e_{c_k} x_{i,j,k} + \sum_{i=0}^{m}\sum_{k=1}^{q} e_{h_k} y_{i,k} \tag{8}$$

3) POI MULTIPLE-VISIT PROHIBITION CONSTRAINT FUNCTION H_C

The constraint function H_C is a function that a user cannot visit each POI no more than once. H_C is expressed by Eq. (9).

$$H_C = \sum_{k=1}^{p}\left[\left(1 - 2 \times \sum_{i=1}^{m}\sum_{j=1}^{n} x_{i,j,k}\right)^2 - 1\right] \tag{9}$$

H_C takes the minimum value 0 when each POI is visited no more than once over trip days.

4) SIMULTANEOUS VISIT PROHIBITION CONSTRAINT FUNCTION H_D

The constraint function H_D is a function that prohibits visiting more than one POI at the same time or staying at more than one hotel at the same time. H_D is expressed by Eq. (10).

$$H_D = \sum_{i=1}^{m}\sum_{j=1}^{n}\left(1 - \sum_{k=1}^{p} x_{i,j,k}\right)^2 + \sum_{i=0}^{m}\left(1 - \sum_{k=1}^{q} y_{i,k}\right)^2 \tag{10}$$

The first term of H_D is a constraint that assigns only one POI to each time and the second term is a constraint that assigns only one hotel to each day.

5) TIME CONSTRAINT FUNCTION H_E

The constraint function H_E restricts the total travel time on day i to be less than or equal to the time limit $T^i_{\max}$ on day i. H_E is expressed by Eq. (11).

$$H_E = \sum_{i=1}^{m}[H_{E1}(i) + H_{E2}(i) + H_{E3}(i) - T^i_{\max}] \tag{11}$$

where

$$H_{E1}(i) = \sum_{j=1}^{n}\sum_{k=1}^{p} t_k x_{i,j,k} \tag{12}$$

$$H_{E2}(i) = \sum_{j=1}^{n-1}\sum_{u=1}^{p}\sum_{v=1}^{p} t(u,v) x_{i,j,u} x_{i,j+1,v} \tag{13}$$

$$H_{E3}(i) = \sum_{u=1}^{p}\sum_{z=1}^{q} t(u,z)\left(x_{i,1,u} y_{i-1,z} + x_{i,n,u} y_{i,z}\right) \tag{14}$$

H_{E1}, H_{E2}, and H_{E3} represent the total staying time at POIs, the total transfer time between POIs c_u and c_v, and the total transfer time between the POI c_u and the hotel h_z, respectively, at day i.

Here, we define the transfer time between two points p_1 and p_2 as follows:

$$t(p_1, p_2) = \frac{d(p_1, p_2)}{v(p_1, p_2)} \tag{15}$$

where $d(p_1, p_2)$ is the Euclidean distance between p_1 and p_2 and $v(p_1, p_2)$ is the transfer speed between p_1 and p_2.

IV. PROPOSED SUBQUBO ANNEALING

In this section, we propose a subQUBO annealing method based on (1) local-solution-based variable selection and (2) random variable selection.

As mention in Section I, the current Ising machines are limited by their sizes of the input QUBO models due to

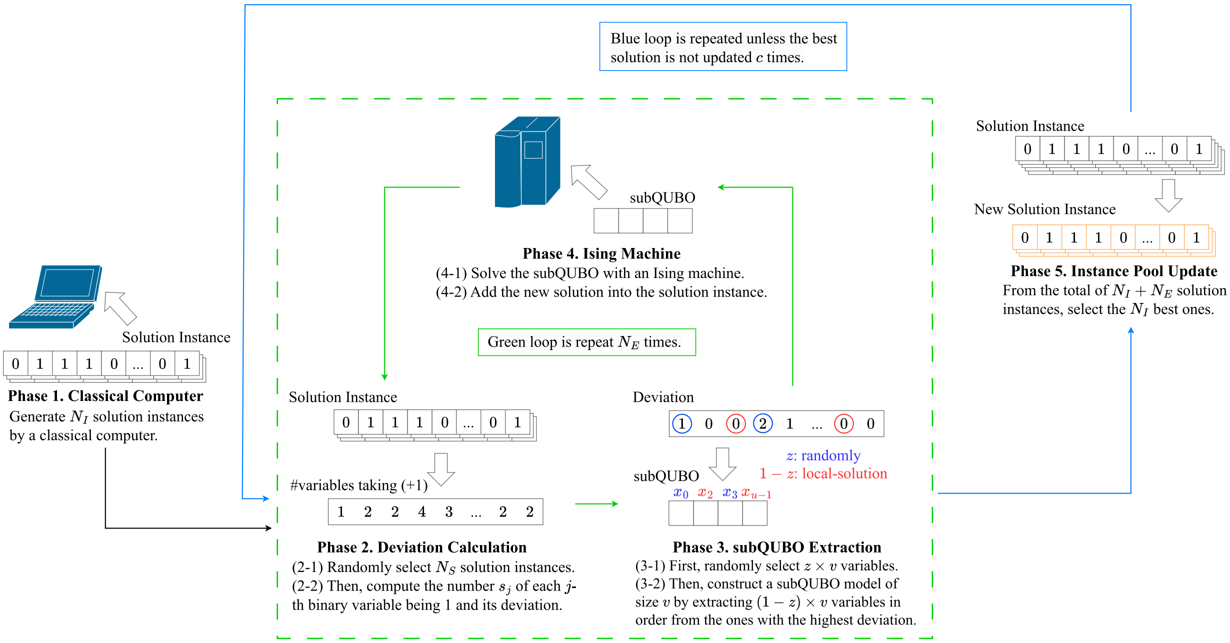


FIGURE 3. The flowchart of the proposed subQUBO annealing method.

	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	x_{10}
Solution 1	0	1	1	1	0	1	1	0	0	1
Solution 2	0	0	0	0	1	1	0	0	1	1
Solution 3	0	1	0	1	0	1	1	0	0	1
Solution 4	0	1	0	1	0	1	1	1	0	1
Solution 5	0	0	0	1	0	1	0	1	1	0

Least deviated   Most deviated

FIGURE 4. Local-solution-based variable selection.

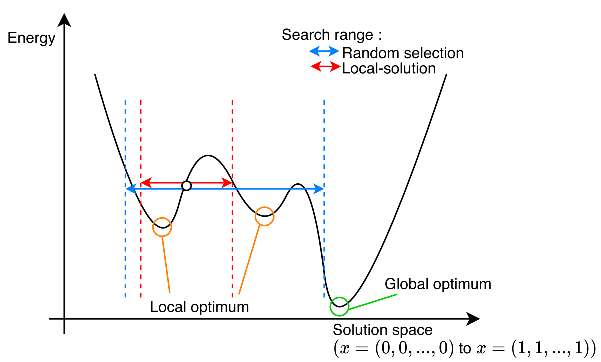


FIGURE 5. Example of energy search range.

hardware constraints. Therefore, we partially extract some variables from the original QUBO model, and then input them to an Ising machine, and solve the subQUBO model repeatedly. Finally, we can obtain a quasi-optimal state for problems that exceed the computable size of the Ising machine.

In this case, the selection and construction of the subQUBO model is the key to finding the optimal solution of the original QUBO model. Therefore, this section proposes a subQUBO annealing method that combines local-solution-based variable selection and random variable selection.

A. LOCAL-SOLUTION-BASED VARIABLE SELECTION

Local-solution-based variable selection is based on the method of Atobe et al. [27]. In [27], multiple runs of the QUBO solver on a classical computer such as [31], are assumed to yield multiple quasi-optimal solutions to the original problem. These quasi-optimal solutions are called *solution instances*.

When focusing on each variable x_i of the QUBO model, the variation of x_i across the multiple solution instances obtained is called the *deviation*. If the deviation of x_i is small, x_i is considered to be taking the optimal value. On the other hand, when the deviation is large, the value of x_i is not fixed and non-optimal. Therefore, to construct a subQUBO, variables with large deviations must be selected as many as possible.

For example, assume that we have five solution instances as depicted in Fig. 4 and the subQUBO size is four. Every solution instance is composed of 10 variables, which is larger than the subQUBO size. When we look at the 1st variable x_1 in these solutions instances, all of them take the value of 0. In this case, x_1 is the least deviated. In the same way, when we look at the 7th variable x_7, three of them take the value of $+1$ and the remaining two of them take the value of 0. In this case, around the half of the solution instances takes the value of $+1$ and the half takes the value of 0, i.e., x_7 is the most deviated. Therefore, the four variables, x_2, x_7, x_8, and x_9, from the most deviated one to the least are selected, which can fit into the subQUBO size.

As shown in Fig. 5, the subQUBO extracted by the local-solution-based variable selection can search for a small range in the solution space efficiently (the red range in Fig. 5). However, if there is a globally optimal solution outside the search range, it may fall back to a local solution.

B. RANDOM VARIABLE SELECTION

Local-solution-based variable selection produces a quasi-optimal solution with a high probability but converges to a biased quasi-ground-state solution instead of yielding a ground state. Therefore, in addition to the local-solution-based variable selection, we additionally include the random variable selection.

We randomly select variables from the original QUBO model. The random variable selection is supposed to select variables that are not selected during the local-solution-based variable selection. Also, as shown in Fig. 5, the random variable selection expands the solution search range, facilitating the process of approaching the globally optimal solution and reducing the possibility of falling into a local solution.

C. METHODOLOGY

Based on the discussion above, we propose a method that combines (1) local-solution-based variable selection and (2) random variable selection to extract the subQUBO model.

Let u be the size of the original QUBO model, v be the size of the subQUBO model to be extracted, and random selection ratio z ($0 \leq z \leq 1$) be the proportion of variables randomly selected for the subQUBO model. Fig. 3 shows the flowchart of the proposed method.

Phase 1: First, N_I solution instances are randomly generated and added to the empty instance pool. In the TPP, two types of binary variables, $x_{i,j,k}$ and $y_{i,k}$, are used. Hence, we arrange all of these variables in a one-dimensional array as depicted in Fig. 6, which gives a solution instance. Then, we retrieve every solution instance X_i from the instance pool, search for the quasi-ground-state solution by a classical computer as its initial solution, and return it to the instance pool. This operation is performed for each solution instance in the instance pool. Among the solution instances in the instance pool, the one with the smallest energy value is the tentative optimal solution X_{best}.

Phase 2: Then, we randomly select N_S solution instances $(X_1, X_2, \ldots, X_{N_S})$ from the instance pool. We focus on the j-th binary variable x_j of the selected N_S solution instances and find the number s_j of solution instances for which $x_j = 1$. Then, the absolute value, showing the deviation of x_j, of the difference between s_j and $N_S/2$ is computed.

Phase 3: After Phase 2, we select $z \times v$ variables from the original QUBO model randomly where z is random selection ratio and v is the size of the subQUBO model. After that, $(1 - z) \times v$ variables are extracted, from the most deviated ones to the least. The subQUBO model of size v is constructed with these variables.

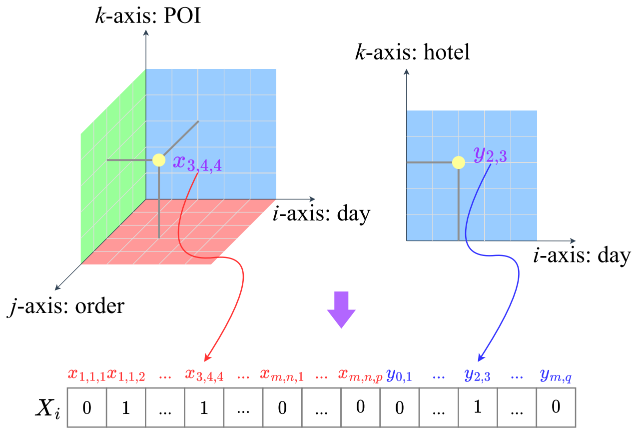


FIGURE 6. Mapping the binary variables $x_{i,j,k}$ and $y_{i,k}$ to a one-dimensional array.

TABLE 1. Dataset details.

Dataset	Size	Time limit [h]	Travel speed [km/h]
Tokyo	p104-q10-m3-n5[a]	8	35
Kyoto	p86-q8-m3-n5	8	35
Sapporo	p66-q6-m3-n5	8	35

[a] pX-qY-mZ-nW shows that the total number of POIs is X, the total number of hotels is Y, the number of tourist days (excluding an arrival day and a departure days) is Z, and the number of POIs visited per day is W.

Phase 4: For the extracted subQUBO model, a (quasi-) ground-state solution is obtained by an Ising machine, and the solution is combined with a randomly selected solution instance X_t from the instance pool to generate a new solution instance X'.

Phase 5: The generated solution instance X' is added to the instance pool as a new solution instance. These operations are performed N_E times to generate N_E new solution instances. Among $(N_I + N_E)$ solution instances where N_I is the number of original solutions and N_E is the number of newly generated solutions, we select the best N_I solution instances.

The above operations in Phase 2 to Phase 5 are repeated until the tentative optimal solution converges, and the final X_{best} is output. In the proposed method, if the optimal solution is not updated $c = 3$ times, we consider that the solution is well converged.

V. EXPERIMENT

In this section, we apply the proposed subQUBO annealing to the TPP. We compare our method with the existing methods [27], [30], proposed recently:

1) Bao et al. [30] is a one-step annealing method for the TPP. If the annealing machine has enough spins and can accept the entire QUBO of the TPP at once, this method can be applied to the TPP.
2) Atobe et al. [27] is a subQUBO annealing method which only uses local-solution-based variable

TABLE 2. Hyperparameter setting.

Dataset	α	β	γ	δ	ϵ
Tokyo	2	1	100	100	0.5
Kyoto	2	1	100	100	0.5
Sapporo	2	1	100	100	0.5

selection. This method can be applied to various kinds of QUBOs including the TPP.

As an Ising machine, we use Fixstars Amplify Annealing Engine (AE) [32] and D-Wave Advantage quantum annealer [33].

A. ENVIRONMENT

To run our proposed method, we use dwave-neal SimulatedAnnealingSampler [31] as the QUBO solver for Phase 1 and the Intel Xenon processor (CPU: 104 cores, 2.10 GHz/RAM: 1.5TB) as the classical computer for all the Phases except for Phase 4.

For Phase 4, we use two types of physical Ising machines, Fixstars Amplify AE which is a GPU-based annealing machine and can accept more than 100K QUBO variables and D-Wave Advantage quantum annealer which can accept around 100 to 150 QUBO variables. Annealing time is 1000 milliseconds for Fixstars Amplify AE and 2000 microseconds for D-Wave Advantage quantum annealer.

Table 1 shows the dataset used in the experiment. The dataset includes 104 POIs and 10 hotels in Tokyo area, 86 POIs and 8 hotels in Kyoto area, and 66 POIs and 6 hotels in Sapporo area. In this experimental evaluation, we set the travel speed at 35 km/h[1] between POIs and/or hotels.

The spin size of Bao et al. [30] is 1600 for the Tokyo dataset, 1322 for Kyoto dataset, and 1014 for Sapporo dataset. When we use Fixstars Amplify AE, the corresponding QUBO models are input to it directly and hence the method of Bao et al. [30] can be applied to them. The sizes of the subQUBOs for our proposed method and Atobe et al. [27] are set to 100, 200, and 400 for any datasets, when we use Fixstars Amplify AE. The size of the subQUBOs is set to 100, when we use D-Wave Advantage quantum annealer. The random selection ratio z of the proposed method is set to 0.3, 0.5, 0.7, and 1.0.

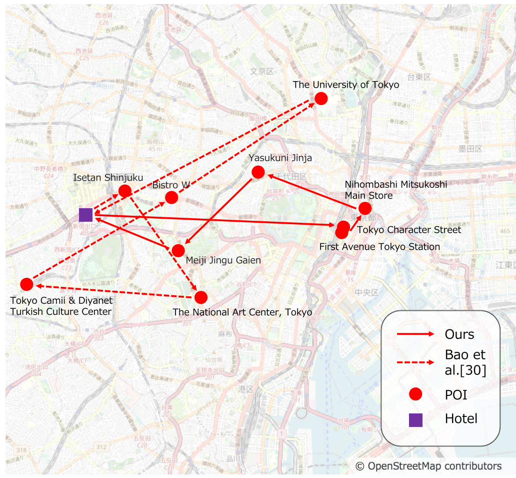
According to [30], we set the hyperparameters of $\alpha, \beta, \gamma, \delta$, and ϵ as shown in Table 2. We set $N_I = 20$, $N_E = 5$, and $N_S = 5$ in the subQUBO annealing step, since these parameters give the best solution in [27].

B. RESULTS USING FIXSTARS AMPLIFY AE

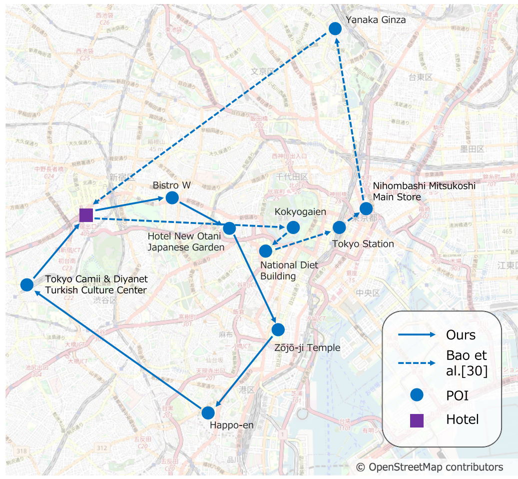
1) SOLUTION QUALITY

Tables 3–5 list the solution quality results for (1) POI satisfaction, (2) trip cost, and (3) energy calculated by Eq. (6).

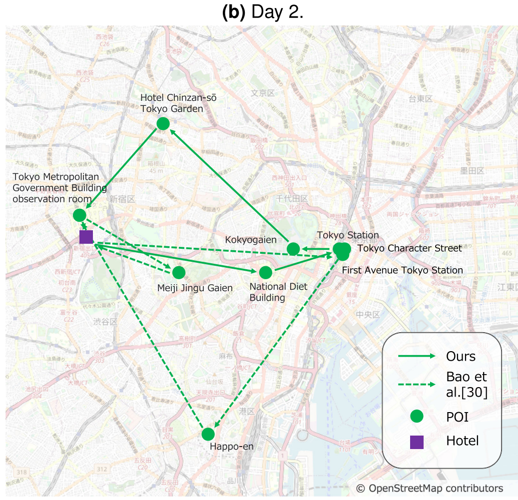
[1] In this paper, we assume that the user travels by train. Since the average speed of the JR Yamanote Line is 34.5 km/h, the travel speed is set to be 35 km/h.



(a) Day 1.



(b) Day 2.



(c) Day 3.

FIGURE 7. 5-day trip plans of Bao et al. [30] and the proposed method. The subQUBO size and the random selection ratio of the proposed method are (200, 0.3).

We run the proposed method, the method by Bao et. al [30], and the method by Atobe et al. [27] 5 times and obtain the averaged results.

TABLE 3. Experimental results of TPP on POI satisfaction, trip cost and energy: subQUBO size = 100.

Dataset	Method	Random selection ratio z	POI satisfaction	Trip cost	Energy
Tokyo	Bao et al. [30]	-	62.44	60.1	-1955.2433
	Atobe et al. [27]	0.0	62.78	60	-1956.4600
	Ours	**0.3**[b]	62.66	60	-1956.6620
		0.5	62.6	60.68	-1955.8525
		0.7	62.18	61.92	-1952.3804
		1.0	62.28	71.54	-1940.6014
Kyoto	Bao et al. [30]	-	63.3	37.58	-1978.8993
	Atobe et al. [27]	0.0	63.26	37.14	-1979.8967
	Ours	**0.3**[b]	63.56	37.4	-1980.2512
		0.5	63.56	37.6	-1980.1319
		0.7	63.34	39.4	-1976.4320
		1.0	63.1	46	-1964.4956
Sapporo	Bao et al. [30]	-	61.66	40.08	-1973.7079
	Atobe et al. [27]	0.0	62	40	-1974.4543
	Ours	0.3	62	40	-1974.5501
		0.5[b]	61.98	40	-1974.5573
		0.7	61.82	41.7	-1972.0142
		1.0	61.64	45.22	-1967.1774

[b] The best solution in each area is shown in bold.
* Results that are better than or equal to the existing methods are underlined.

TABLE 4. Experimental results of TPP on POI satisfaction, trip cost and energy: subQUBO size = 200.

Dataset	Method	Random selection ratio z	POI satisfaction	Trip cost	Energy
Tokyo	Bao et al. [30]	-	62.44	60.1	-1955.2433
	Atobe et al. [27]	0.0	62.7	60	-1956.7570
	Ours	**0.3**[b]	62.72	60	-1956.7570
		0.5	62.68	60	-1956.6698
		0.7	62.74	60.2	-1956.7076
		1.0	62.46	69.62	-1948.3399
Kyoto	Bao et al. [30]	-	63.3	37.58	-1978.8993
	Atobe et al. [27]	0.0	63.3	37.12	-1979.8816
	Ours	**0.3**[b]	63.46	37.24	-1980.3113
		0.5	63.48	37.28	-1980.2582
		0.7	63.44	37.18	-1980.2133
		1.0	63.14	44.48	-1967.1311
Sapporo	Bao et al. [30]	-	61.66	40.08	-1973.7079
	Atobe et al. [27]	0.0	61.96	40	-1974.4524
	Ours	**0.3**[b]	62	40	-1974.5028
		0.5	62	40	-1974.5017
		0.7	61.96	40	-1974.4328
		1.0	61.84	43.8	-1969.2709

[b] The best solution in each area is shown in bold.
* Results that are better than or equal to the existing methods are underlined.

Ours vs Bao et al. [30]: Out of a total of 36 possible outcomes with three different datasets, four different random selection ratios, and three different subQUBO sizes, *Ours* has higher POI satisfaction than Bao et al. [30] in 31 of them. Also, *Ours* has lower trip costs than Bao et al. [30] in 23 outcomes and smaller energy than Bao et al. [30] in 26 outcomes. Therefore, rather than solving the TPP directly, *Ours* is possible to generate a higher quality solution by partitioning the QUBO using the subQUBO annealing. Since quantum annealing is based on a stochastic search, the method by Bao et al. [30] has the disadvantage of generating poor solutions. In contrast, the proposed method is better in that it generates multiple solutions in a single iteration, and by repeating the operation of picking up the best solution many times, it converges to a more stable optimal solution.

Also, Fig 7 shows the 5-day trip plans of Bao et al. [30] and the proposed method. The proposed method has a more compact route than the Bao et al. [30], and it tours more satisfactory POIs in a shorter travel time.

Ours vs Atobe et al. [27]: In the proposed method, when subQUBO size is 100, the best random selection ratio is $z = 0.3$ for the Tokyo and Kyoto datasets, and $z = 0.5$ for the Sapporo dataset. As well, when subQUBO size is 200, the best random selection ratio is $z = 0.3$ for all the datasets. When subQUBO size is 400, the best random selection ratio is $z = 0.3$ for the Tokyo and dataset, $z = 0.0$ for the Kyoto dataset, and $z = 0.7$ for the Sapporo dataset. Except for the Kyoto dataset with subQUBO size of 400, *Ours* realizes smaller energy than Atobe et al. [27]. Therefore, we can have a conclusion that better solutions can be obtained by adding randomness during the subQUBO extraction.

TABLE 5. Experimental results of TPP on POI satisfaction, trip cost and energy: subQUBO size = 400.

Dataset	Method	Random selection ratio z	POI satisfaction	Trip cost	Energy
Tokyo	Bao et al. [30]	-	62.44	60.1	-1955.2433
	Atobe et al. [27]	0.0	62.64	60	-1956.6203
	Ours	**0.3**[b]	62.74	60	-1956.6701
		0.5	62.72	60	-1956.6427
		0.7	62.72	60	-1956.6460
		1.0	62.68	61.48	-1956.3334
Kyoto	Bao et al. [30]	-	63.3	37.58	-1978.8993
	Atobe et al. [27]	**0.0**[b]	63.44	37.3	-1980.1934
	Ours	0.3	63.52	37.4	-1980.0473
		0.5	63.54	37.5	-1980.1260
		0.7	63.48	37.32	-1980.1203
		1.0	63.68	38.9	-1977.8507
Sapporo	Bao et al. [30]	-	61.66	40.08	-1973.7079
	Atobe et al. [27]	0.0	62	40	-1974.4367
	Ours	0.3	61.98	40	-1974.4440
		0.5	61.84	40	-1974.3611
		0.7[b]	62	40	-1974.4558
		1.0	61.88	40	-1974.3427

[b] The best solution in each area is shown in bold.
* Results that are better than or equal to the existing methods are underlined.

2) CONSTRAINT SATISFACTION PROBABILITY

Tables 6–8 list the constraint satisfaction probability over (1) #POIs/day; (2) travel time and (3) overall probability. The "Probability" in these tables shows the percentage of days in which the time constraint is satisfied in all three days of sightseeing. "Probability" becomes 100% when the travel times on days 1 to 3 are all less than eight hours. For the travel time, the closer the average travel time each day is to the time limit, the better the results become.

TABLE 6. Experimental results of TPP on constraints: subQUBO size = 100.

Dataset	Method	Random selection ratio z	#POIs	Travel time [h] Day1	Day2	Day3	Probability [%]
Tokyo	Bao et al. [30]	-	5	6.59	6.48	6.00	100
	Atobe et al. [27]	0.0	5	5.90	6.19	6.12	100
	Ours	**0.3**[b]	5	5.97	5.55	5.80	100
		0.5	5	5.66	5.95	5.70	100
		0.7	5	6.42	7.47	6.22	60
		1.0	5	7.63	8.80	8.43	0
Kyoto	Bao et al. [30]	-	5	6.60	6.68	6.97	100
	Atobe et al [27]	0.0	5	6.39	6.00	6.58	100
	Ours	**0.3**[b]	5	6.59	6.21	6.14	100
		0.5	5	5.97	6.23	6.58	100
		0.7	5	6.15	6.78	8.74	80
		1.0	5	8.27	7.95	15.12	0
Sapporo	Bao et al. [30]	-	5	6.49	6.21	6.36	100
	Atobe et al. [27]	0.0	5	6.33	6.72	6.04	100
	Ours	0.3	5	5.83	6.78	6.30	100
		0.5[b]	5	6.18	6.82	5.81	100
		0.7	5	6.58	7.27	5.99	80
		1.0	5	7.23	7.94	6.60	40

[b] The best solution in each area is shown in bold.
[*] Results that are better than or equal to the existing methods are underlined.

TABLE 7. Experimental results of TPP on constraint functions: subQUBO size = 200.

Dataset	Method	Random selection ratio z	#POIs	Travel time [h] Day1	Day2	Day3	Probability [%]
Tokyo	Bao et al. [30]	-	5	6.59	6.48	6.00	100
	Atobe et al. [27]	0.0	5	6.20	5.93	5.64	100
	Ours	**0.3**[b]	5	5.72	5.95	5.86	100
		0.5	5	5.88	6.05	5.50	100
		0.7	5	5.96	6.11	5.80	100
		1.0	5	7.19	7.89	7.63	40
Kyoto	Bao et al. [30]	-	5	6.60	6.68	6.97	100
	Atobe et al [27]	0.0	5	6.16	6.56	6.48	80
	Ours	**0.3**[b]	5	6.24	6.10	6.40	100
		0.5	5	6.18	6.12	6.53	100
		0.7	5	6.08	6.19	6.69	100
		1.0	5	7.78	8.51	13.02	0
Sapporo	Bao et al. [30]	-	5	6.49	6.21	6.36	100
	Atobe et al. [27]	0.0	5	6.24	6.90	5.79	100
	Ours	**0.3**[b]	5	6.64	6.44	5.92	100
		0.5	5	5.94	6.44	6.62	100
		0.7	5	6.32	6.37	6.28	100
		1.0	5	7.47	7.40	6.34	60

[b] The best solution in each area is shown in bold.
[*] Results that are better than or equal to the existing methods are underlined.

Ours vs Bao et al. [30] Both Bao et al. [30] and *Ours* satisfy all the constraints of #POIs/day and the travel time. In 24 of the total results, "Probability" is 100%. Also, the results show that a moderate degree of randomness is

TABLE 8. Experimental results of TPP on constraint functions: subQUBO size = 400.

Dataset	Method	Random selection ratio z	#POIs	Travel time [h] Day1	Day2	Day3	Probability [%]
Tokyo	Bao et al. [30]	-	5	6.59	6.48	6.00	100
	Atobe et al. [27]	0.0	5	6.20	5.93	5.64	100
	Ours	**0.3**[b]	5	5.72	5.95	5.86	100
		0.5	5	5.88	6.05	5.50	100
		0.7	5	5.96	6.11	5.80	100
		1.0	5	7.19	7.89	7.63	40
Kyoto	Bao et al. [30]	-	5	6.60	6.68	6.97	100
	Atobe et al [27]	**0.0**[b]	5	6.16	6.56	6.48	80
	Ours	0.3	5	6.24	6.10	6.40	100
		0.5	5	6.18	6.12	6.53	100
		0.7	5	6.08	6.19	6.69	100
		1.0	5	7.78	8.51	13.02	0
Sapporo	Bao et al. [30]	-	5	6.49	6.21	6.36	100
	Atobe et al. [27]	0.0	5	6.24	6.90	5.79	100
	Ours	0.3	5	6.64	6.44	5.92	100
		0.5	5	5.94	6.44	6.62	100
		0.7[b]	5	6.32	6.37	6.28	100
		1.0	5	7.47	7.40	6.34	60

[b] The best solution in each area is shown in bold.
[*] Results that are better than or equal to the existing methods are underlined.

TABLE 9. Experimental results on the number of iterations and processing time: subQUBO size = 100.

Dataset	Method	Random selection ratio z	#iterations	Processing time [s] Ising machine	Classical computer	Total
Tokyo	Bao et al. [30]	-	-	0.932	-	0.932
	Atobe et al. [27]	0.0	12.8	60.667	93.600	154.267
	Ours	0.3	18.4	87.167	121.771	208.937
		0.5	20.4	97.873	131.836	229.709
		0.7	10.2	47.253	78.803	126.056
		1.0	6.4	29.658	58.615	88.273
Kyoto	Bao et al. [30]	-	-	0.900	-	0.900
	Atobe et al. [27]	0.0	12.8	59.236	71.750	130.987
	Ours	0.3	15.6	74.021	84.347	158.368
		0.5	19.6	92.937	100.974	193.911
		0.7	13.8	64.842	77.524	142.366
		1.0	4.8	22.290	38.561	60.851
Sapporo	Bao et al. [30]	-	-	0.889	-	0.889
	Atobe et al. [27]	0.0	12.2	57.039	48.339	105.378
	Ours	0.3	17.4	82.426	64.550	146.975
		0.5	15.8	74.732	59.311	134.043
		0.7	14	65.403	53.894	119.297
		1.0	10.2	46.665	41.853	88.517

necessary, since the stronger the randomness is, the more likely it is that the constraints will not be satisfied.

Ours vs Atobe et al. [27]: The travel time constraint is not satisfied for Atobe et al. [27] of Kyoto dataset with the subQUBO size of 200. Meanwhile, *Ours* with a random selection ratio other than 1.0 satisfies the travel time constraint. As *Ours* includes the randomness in the subQUBO extraction, it is more likely to satisfy the constraints.

TABLE 10. Experimental results on the number of iterations and processing time: subQUBO size = 200.

Dataset	Method	Random selection ratio z	#iterations	Processing time [s] Ising machine	Classical computer	Total
Tokyo	Bao et al. [30]	-	-	0.932	-	0.932
	Atobe et al. [27]	0.0	12.6	59.763	146.994	206.757
	Ours	0.3	17	81.214	185.623	266.838
		0.5	16.2	77.174	178.291	255.465
		0.7	14.4	68.343	161.993	230.337
		1.0	10.6	47.810	125.209	173.019
Kyoto	Bao et al. [30]	-	-	0.900	-	0.900
	Atobe et al. [27]	0.0	11.6	54.857	108.977	163.834
	Ours	0.3	16.8	79.843	145.059	224.903
		0.5	15.4	73.062	136.904	209.966
		0.7	15.6	73.835	136.817	210.653
		1.0	6	27.164	63.679	90.843
Sapporo	Bao et al. [30]	-	-	0.889	-	0.889
	Atobe et al. [27]	0.0	12.4	58.750	79.794	138.544
	Ours	0.3	12.6	59.619	80.906	140.525
		0.5	14	65.919	89.671	155.590
		0.7	13.2	62.061	84.443	146.504
		1.0	7.4	33.748	52.684	86.432

TABLE 11. Experimental results on the number of iterations and processing time: subQUBO size = 400.

Dataset	Method	Random selection ratio z	#iterations	Processing time [s] Ising machine	Classical computer	Total
Tokyo	Bao et al. [30]	-	-	0.932	-	0.932
	Atobe et al. [27]	0.0	12.2	57.127	243.852	300.979
	Ours	0.3	11.6	54.531	237.706	292.237
		0.5	14	65.832	279.524	345.356
		0.7	14	65.741	278.921	344.662
		1.0	16.2	76.270	324.471	400.741
Kyoto	Bao et al. [30]	-	-	0.900	-	0.900
	Atobe et al. [27]	0.0	16	74.930	259.403	334.333
	Ours	0.3	11.6	54.490	191.667	246.157
		0.5	11.8	55.255	190.703	245.958
		0.7	14.2	66.466	231.759	298.225
		1.0	10.6	49.775	174.685	224.461
Sapporo	Bao et al. [30]	-	-	0.889	-	0.889
	Atobe et al. [27]	0.0	10.2	47.525	116.940	164.465
	Ours	0.3	12	56.302	130.570	186.873
		0.5	8.2	38.413	88.585	126.998
		0.7	10.6	49.887	110.073	159.960
		1.0	10.2	47.958	106.508	154.466

3) ANNEALING ITERATION

Tables 9–11 list the annealing iteration and processing time. The number of iterations with random selection ratios of $z = 0.3$, 0.5, and 0.7 tends to converge to a higher number of iterations and lower energy, while the number of iterations with random selection ratio $z = 1.0$ tends to converge to a suboptimal solution at a lower number and earlier stage of solution. The total execution time of the proposed method is several hundred times longer than that of the Bao et al. [30], since the proposed method repeatedly anneals the subQUBOs using an Ising machine. Compared to the Atobe et al. [27], the runtime of

TABLE 12. Experimental results using D-Wave Advantage on POI satisfaction, trip cost, energy, and constraints (subQUBO size = 100).

Dataset	Random selection ratio z	POI satisfaction	Trip cost	Energy	#POI	Travel time [h]		
						Day1	Day2	Day3
Tokyo	0.3	61.7	66.7	-1945.7163	5	7.71	7.35	6.91
Kyoto	0.3	62.6	50.6	-1963.9938	5	8.00	6.24	7.08
Sapporo	0.3	60.6	45.1	-1965.686	5	6.28	7.26	7.22

TABLE 13. Experimental results using D-Wave Advantage on the number of iterations and processing time (subQUBO size = 100).

Dataset	Random selection ratio z	#Iterations	Processing time [s]		
			Ising machine	Classical computer	Total
Tokyo	0.3	6	0.06	53.620	53.680
Kyoto	0.3	6	0.06	41.881	41.941
Sapporo	0.3	6	0.05	22.900	22.950

the proposed method tends to decrease as the subQUBO size increases.

C. RESULTS USING D-WAVE ADVANTAGE

Tables 12–13 show the experimental results on D-Wave Advantage. In the experiment using D-Wave Advantage, to satisfy the POI multiple visit prohibition constraint and the simultaneous visit prohibition constraint, the solution solved by the Ising machine was corrected for each iteration using the method propesed in [34]. We set the random selection ratio $z = 0.3$ as this value gives almost the best in the previous subsection.

As a result, we obtained a solution comparable to Fixstars Amplify AE. This result clearly indicates that, the proposed method can effectively solves the TPP utilizing the spin-size-limited physical quantum annealer.

VI. CONCLUSION

In this paper, we proposed a method to solve the TPP based on partitioning a large QUBO into several subQUBOs effectively. Evaluation experiments were conducted using real Ising machines and compared with the existing methods. The proposed method is effective in solving the TPP and obtained better results in terms of POI satisfaction, trip cost, energy, and constraint satisfaction probability.

Future work includes applying the proposed method to different areas other than Japan. Also, how to setup the optimal hyperparameters in the proposed method is another important future work.

REFERENCES

[1] K. Sylejmani and A. Dika, "Solving touristic trip planning problem by using taboo search approach," *Int. J. Comput. Sci. Issues*, vol. 8, no. 5, p. 139, 2011.
[2] E. Saeki, S. Bao, T. Takayama, and N. Togawa, "Multi-objective trip planning based on ant colony optimization utilizing trip records," *IEEE Access*, vol. 10, pp. 127825–127844, 2022.
[3] S. M. R. Fournier, E. O. Hülse, and É. V. Pinheiro, "A*-guided heuristic for a multi-objective bus passenger trip planning problem," *Public Transp.*, vol. 13, no. 3, pp. 557–578, Oct. 2021.
[4] A. Garcia, P. Vansteenwegen, O. Arbelaitz, W. Souffriau, and M. T. Linaza, "Integrating public transportation in personalised electronic tourist guides," *Comput. Oper. Res.*, vol. 40, no. 3, pp. 758–774, Mar. 2013.
[5] Y. Shuai, S. Yunfeng, and Z. Kai, "An effective method for solving multiple travelling salesman problem based on NSGA-II," *Syst. Sci. Control Eng.*, vol. 7, no. 2, pp. 108–116, Nov. 2019.
[6] P. He and J.-K. Hao, "Hybrid search with neighborhood reduction for the multiple traveling salesman problem," *Comput. Oper. Res.*, vol. 142, Jun. 2022, Art. no. 105726.
[7] B. H. Korte, J. Vygen, B. Korte, and J. Vygen, *Combinatorial Optimization*. Cham, Switzerland: Springer, 2011, vol. 1, pp. 1–12.
[8] M. Parizy and N. Togawa, "Analysis and acceleration of the quadratic knapsack problem on an Ising machine," *IEICE Trans. Fundamentals Electron., Commun. Comput. Sci.*, vol. E104.A, no. 11, pp. 1526–1535, 2021.
[9] S. Kanamaru, K. Kawamura, S. Tanaka, Y. Tomita, and N. Togawa, "Solving constrained slot placement problems using an Ising machine and its evaluations," *IEICE Trans. Inf. Syst.*, vol. E104.D, no. 2, pp. 226–236, 2021.
[10] M. Lewis and F. Glover, "Quadratic unconstrained binary optimization problem preprocessing: Theory and empirical analysis," *Networks*, vol. 70, no. 2, pp. 79–97, Sep. 2017.
[11] T. Wang and J. Roychowdhury, "Oscillator-based Ising machine," 2017, *arXiv:1709.08102*.
[12] N. Mohseni, P. L. Mcmahon, and T. Byrnes, "Ising machines as hardware solvers of combinatorial optimization problems," *Nature Rev. Phys.*, vol. 4, no. 6, pp. 363–379, May 2022.
[13] T. Wang, L. Wu, P. Nobel, and J. Roychowdhury, "Solving combinatorial optimisation problems using oscillator based Ising machines," *Natural Comput.*, vol. 20, no. 2, pp. 287–306, Jun. 2021.
[14] T. Shirai, S. Tanaka, and N. Togawa, "Guiding principle for minor-embedding in simulated-annealing-based Ising machines," *IEEE Access*, vol. 8, pp. 210490–210502, 2020.
[15] 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, Nov. 2016.
[16] T. Wang and J. Roychowdhury, "OIM: Oscillator-based Ising machines for solving combinatorial optimisation problems," in *Proc. Int. Conf. Unconventional Comput. Natural Comput.* Cham, Switzerland: Springer, 2019, pp. 232–256.
[17] Z. Wang, A. Marandi, K. Wen, R. L. Byer, and Y. Yamamoto, "Coherent Ising machine based on degenerate optical parametric oscillators," *Phys. Rev. A, Gen. Phys.*, vol. 88, no. 6, Dec. 2013, Art. no. 063853.
[18] K. Takata, A. Marandi, R. Hamerly, Y. Haribara, D. Maruo, S. Tamate, H. Sakaguchi, S. Utsunomiya, and Y. Yamamoto, "A 16-bit coherent Ising machine for one-dimensional ring and cubic graph problems," *Sci. Rep.*, vol. 6, no. 1, p. 34089, Sep. 2016.
[19] T. Aonishi, K. Mimura, S. Utsunomiya, M. Okada, and Y. Yamamoto, "Statistical mechanics of coherent Ising machine—The case of ferromagnetic and finite-loading Hopfield models —," *J. Phys. Soc. Jpn.*, vol. 86, no. 10, Oct. 2017, Art. no. 104002.
[20] G. Chapuis, H. Djidjev, G. Hahn, and G. Rizk, "Finding maximum cliques on the D-Wave quantum annealer," *J. Signal Process. Syst.*, vol. 91, nos. 3–4, pp. 363–377, Mar. 2019.

[21] D. Willsch, M. Willsch, C. D. Gonzalez Calaza, F. Jin, H. De Raedt, M. Svensson, and K. Michielsen, ''Benchmarking advantage and D-wave 2000Q quantum annealers with exact cover problems,'' *Quantum Inf. Process.*, vol. 21, no. 4, p. 141, Apr. 2022.
[22] IBM. *IBM Quantum Breaks The 100-Qubit Processor Barrier*. Accessed: Jul. 20, 2023. [Online]. Available: https://research.ibm.com/blog/127-qubit-quantum-processor-eagle?social_post=5922928345&linkId=140353937
[23] M. Boost, S. Reinhardt, and A. Roy, ''Partitioning optimization problems for hybrid classical/quantum execution,'' D-wave, Burnaby, BC, Canada, Tech. Rep. 14–1006A-A, 2017.
[24] G. Rosenberg, M. Vazifeh, B. Woods, and E. Haber, ''Building an iterative heuristic solver for a quantum annealer,'' *Comput. Optim. Appl.*, vol. 65, pp. 845–869, 2016.
[25] M. Jo, M. Hanks, and M. S. Kim, ''Divide-and-conquer embedding for QUBO quantum annealing,'' 2022, *arXiv:2211.02184*.
[26] G. Bass, C. Tomlin, V. Kumar, P. Rihaczek, and J. Dulny, ''Heterogeneous quantum computing for satellite constellation optimization: Solving the weighted *k*-clique problem,'' *Quantum Sci. Technol.*, vol. 3, no. 2, Apr. 2018, Art. no. 024010.
[27] Y. Atobe, M. Tawada, and N. Togawa, ''Hybrid annealing method based on subQUBO model extraction with multiple solution instances,'' *IEEE Trans. Comput.*, vol. 71, no. 10, pp. 2606–2619, Oct. 2022.
[28] A. P. Punnen, *The Quadratic Unconstrained Binary Optimization Problem*. Cham, Switzerland: Springer, 2022.
[29] F. Glover, G. Kochenberger, and Y. Du, ''A tutorial on formulating and using QUBO models,'' 2018, *arXiv:1811.11538*.
[30] S. Bao, M. Tawada, S. Tanaka, and N. Togawa, ''Multi-day travel planning using Ising machines for real-world applications,'' in *Proc. IEEE Int. Intell. Transp. Syst. Conf. (ITSC)*, Sep. 2021, pp. 3704–3709.
[31] *Simulated Annealing Sampler*. D-Wave Systems. Burnaby, BC, Canada. Accessed: Jul. 20, 2023. [Online]. Available: https://docs.ocean.dwavesys.com/en/stable/docs_neal/reference/sampler.html
[32] Fixstars. *Fixstars Amplify*. Accessed: Jul. 20, 2023. [Online]. Available: https://amplify.fixstars.com/en/
[33] D-Wave Systems. *D-Wave Announces D-Wave 2000Q Quantum Computer and First System Order*. Burnaby, BC, Canada. Accessed: Jul. 20, 2023. [Online]. Available: https://dwavefederal.com/d-wave-announces-d-wave-2000q-quantum-computer-first-system-order/
[34] K. Fukada, M. Parizy, Y. Tomita, and N. Togawa, ''A three-stage annealing method solving slot-placement problems using an Ising machine,'' *IEEE Access*, vol. 9, pp. 134413–134426, 2021.

TATSUYA NOGUCHI received the B.Eng. degree in communications and computer engineering from Waseda University, in 2023, where he is currently pursuing the M.Eng. degree. His research interests include trip planning and Ising machine.

KEISUKE FUKADA (Member, IEEE) received the B.Eng. and M.Eng. degrees in computer science from Waseda University, in 2021 and 2022, respectively, where he is currently pursuing the Dr.Eng. degree. His research interests include mathematical optimization and quantum computation.

SIYA BAO (Member, IEEE) received the B.Eng., M.Eng., and Dr.Eng. degrees in computer science from Waseda University, in 2015, 2017, and 2020, respectively. She is currently an Assistant Professor with the Department of Computer Science and Communications Engineering, Waseda University. Her research interests include geographic information systems and Ising machine.

NOZOMU TOGAWA (Member, IEEE) received the B.Eng., M.Eng., and Dr.Eng. degrees in electrical engineering from Waseda University, in 1992, 1994, and 1997, respectively. He is currently a Professor with the Department of Computer Science and Communications Engineering, Waseda University. His research interests include quantum computation and integrated system design. He is a member of ACM, IEICE, and IPSJ.