

---

# ENHANCING MULTIPLE OBJECT TRACKING ACCURACY VIA QUANTUM ANNEALING

---

**Yasuyuki Ihara**

Innovation Promotion Division  
NEC Solution Innovators, Ltd.  
1-18-7 Shinkiba, Koto-ku, Tokyo, 136-8627 Japan.  
ihara\_mxk[ ]nec.com\*

March 29, 2024

## ABSTRACT

Multiple object tracking (MOT), a key task in image recognition, presents a persistent challenge in balancing processing speed and tracking accuracy. This study introduces a novel approach that leverages quantum annealing (QA) to expedite computation speed, while enhancing tracking accuracy through the ensembling of object tracking processes. A method to improve the matching integration process is also proposed. By utilizing the sequential nature of MOT, this study further augments the tracking method via reverse annealing (RA). Experimental validation confirms the maintenance of high accuracy with an annealing time of a mere  $3 \mu\text{s}$  per tracking process. The proposed method holds significant potential for real-time MOT applications, including traffic flow measurement for urban traffic light control, collision prediction for autonomous robots and vehicles, and management of products mass-produced in factories.

**Keywords** Multiple object tracking · Quantum annealing · Reverse annealing · Combinatorial optimization problem · Image recognition

## 1 Introduction

Multiple object tracking (MOT) is a key task in image recognition, holding an advantage over single-object tracking Fan et al. [2019], Zheng et al. [2020] in scenarios with crowded detected objects and frequent occlusions. Because of its superiority, it has been the subject of active research in recent years Zhang et al. [2008], Li et al. [2009], Bewley et al. [2016], Wojke et al. [2017], Zhang et al. [2022], Dai et al. [2019], Gebregziabher [2023], Gudauskas and Matusevičius [2021], Ward et al. [2021]. MOT is also being used in a variety of real-world situations, such as counting the number of traveling vehicles to obtain traffic flow information Dai et al. [2019], predicting approaches of nearby objects for autonomous robots Gebregziabher [2023], and analyzing the movements of players in sports videos Gudauskas and Matusevičius [2021]. Not only accuracy in object tracking but also latency-free real-time processing are needed in situations where processing of control is required immediately after tracking. Such cases include traffic flow control in big cities, where traffic flow is measured as input to control traffic light; collision avoidance in autonomous robotics and vehicles Gebregziabher [2023]; and quality control in manufacturing Ward et al. [2021]., where abnormal objects must be detected from large quantities of manufactured items in factories.

In research to date, MOT has been frequently formulated as a multi-dimensional assignment problem, a type of combinatorial optimization problem (COP). The optimal solution is often obtained by the Hungarian method Kuhn [1955], Munkres [1957]. However, given that the order is  $O(n^3)$  (where  $n$  represents the number of tracks), real-time processing becomes difficult as the scale of computation increases. While research has been conducted to enhance tracking accuracy by integrating multiple types of MOT, the suppression of computational costs has not yet been achieved.

---

\*Replace [ ] with @.

In recent years, quantum annealing (QA) has emerged as a new method for solving COPs, and has become an active area of research. Unlike simulated annealing, which searches for optimal solutions using thermal fluctuations, QA is an algorithm that uses quantum fluctuations to search for optimal solutions to COPs. It is a heuristic algorithm that utilizes quantum physics phenomena and has drawn attention for its ability to evaluate all solution combinations simultaneously using the quantum superposition effect, and to pass through potential barriers using the tunneling effect. As such, it can find the global optimal solution quickly and accurately without being trapped in local solutions. A technique analogous to QA, called adiabatic quantum computation (AQC), has also been proposed Farhi et al. [2001], Childs et al. [2001]. In AQC, a system starts from the ground state, and adiabatic time evolution is used to find the combinatorial optimization solution. Both QA and AQC use adiabatic time evolution, but AQC is a logical computational model that can also be implemented using quantum gates.

Research on MOT using QA or adiabatic quantum computation has been carried out in recent years Zaeck et al. [2022], McCormick et al. [2022a,b]. However, these studies have been limited to the examination of the reduction of computational costs and have not reported improvements in accuracy.

This paper proposes a method that not only accelerates object tracking calculations using QA, but also enhances accuracy by ensemble processing of object tracking by leveraging the characteristics of QA. Another method that deploys reverse annealing (RA) to further boost the efficiency of object tracking processing is also proposed.

The organization of this paper is as follows. Section 2 provides an overview of conventional MOT. Section 3 discusses the general theory of combinatorial optimization using QA. Section 4 formulates the MOT problem in a manner that can be addressed by QA. Section 5 introduces the first proposed method, Quantum-Ensemble MOT, which enhances accuracy through ensemble processing and accelerates object tracking. The second method, aimed at further optimizing MOT efficiency using RA, is presented in Section 6. Both Sections 5 and 6 also describe evaluation experiments using the UA-DETRAC dataset, a benchmark for object detection and tracking. The work is summarized in Section 7.

## 2 Multiple Object Tracking (MOT)

This section provides an overview of traditional MOT. The tracking problem is approached through the paradigm of "Tracking by Detection". In this method, objects are detected in each individual video frame. Detected objects are then temporally associated through a process that operates independently of the detection process, thereby creating distinct object tracks.

### 2.1 Formulation of MOT

The MOT problem is often tackled using bipartite graph matching to establish the correspondence of the same object across consecutive video frames. The general pseudocode for this approach to MOT is presented in Algorithm 1 Gebregziabher [2023].

#### 2.1.1 DeepSORT

This subsection discusses DeepSORT Wojke et al. [2017], a representative and popular method of MOT in recent years. The SORT (Simple Online and Realtime Tracking) algorithm, a precursor to DeepSORT Bewley et al. [2016], associates the same object using the Hungarian method. This is based on the positional relationship between the object

---

**Algorithm 1** Pseudocode for MOT

---

```

1: Initialize tracks
2: for each frame do
3:   Obtain new detections from observation data
4:   Construct a bipartite graph denoted as  $G = (U \cup V, E)$ . ( $U$ : set of current tracks,  $V$ : set of new detections,  $E$ :
      correspondence between tracks and detections).
5:   Compute the cost of edges (e.g., Euclidean distance, similarity of appearance features, etc.).
6:   Determine the optimal matching using a bipartite graph matching algorithm (e.g., Hungarian method).
7:   Assign new detections to current tracks.
8:   Terminate tracks that could not be assigned to new detections.
9:   Start new tracks for detections that could not be assigned.
10: end for
11: return All tracks

```

---

Table 1: Experimental Environment

CPU	Intel Core i7-7700K, 4 cores 8 threads 4.20 Ghz
GPU	NVIDIA GeForce RTX 2080 Ti 11GB
Memory	32.0GB
OS	Windows 10 Enterprise (64bit)

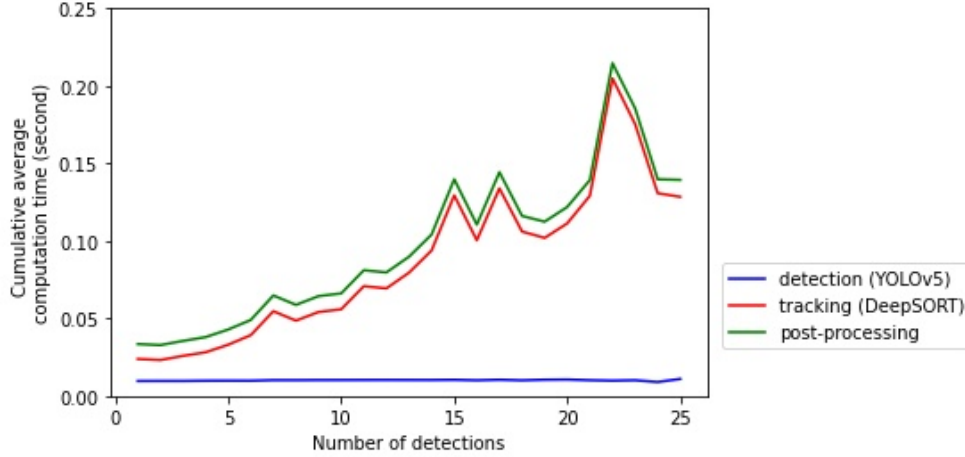


Figure 1: The cumulative processing time for detection by YOLOv5 and tracking by DeepSORT. The right axis represents the number of detected vehicles, and the vertical axis represents the total processing time for detection and tracking (unit: seconds).

detection result in the latest frame and the position prediction from the object detection result in past frames by the Kalman filter. Although it is a simple and fast method, there are challenges such as difficulty in tracking when an object is missed mid-process due to phenomena such as occlusions, or when other phenomena interrupt tracking.

The DeepSORT algorithm, an extension of the SORT algorithm, was developed to address this problem. It first performs matching using appearance features, and for detected objects that did not match, it carries out matching similar to SORT. However, as shown in Figure 1, experiments conducted with DeepSORT in the experimental environment depicted in Table 1 revealed a challenge: the computation time increased as the number of detected objects grew. For instance, when the number of detected objects exceeds 14, achieving detection and tracking at 10fps becomes difficult.

### 3 Combinatorial Optimization using Quantum Annealing

#### 3.1 Combinatorial Optimization Problems (COPs)

COPs involve the efficient search for the optimal combination from various possible patterns, using an evaluation function that assesses the quality of a combination. As the size of the problem increases, the total number of combinations also grows exponentially, leading to a phenomenon known as combinatorial explosion. This makes the problem increasingly difficult to solve. Examples of COPs that can be solved in polynomial computation time include the minimum spanning tree problem and the shortest path problem for graphs. Conversely, there are also COPs that require exponential computation time to solve, such as the traveling salesman problem.

Many COPs can be formulated as minimization problems using a cost function  $E$  in the form of an Ising model, which is defined with spin variables ( $\sigma = \pm 1$ ) Lucas [2014].

$$\begin{aligned} \sigma^* &= \underset{\sigma}{\operatorname{argmin}} E(\sigma), \\ E(\sigma) &:= - \sum_{i \neq j} J_{ij} \sigma_i \sigma_j - \sum_i h_i \sigma_i. \end{aligned} \quad (1)$$

Here,  $\sigma = (\sigma_1, \dots, \sigma_N)^\top$  represents a vector of spins,  $J_{i,j}$  denotes the interaction variables between spins, and  $h_i$  signifies the strength of an external magnetic field. The formulation of combinatorial optimization problems using an

Ising model is equivalent to a quadratic binary integer programming problem. This equivalence is achieved through the use of a Quadratic Unconstrained Binary Optimization (QUBO) formulation  $E'$  with binary variables  $q_i$ , which can take on values of 1 or 0. The transformation of variables is given by  $q_i = \frac{\sigma_i + 1}{2}$ , where  $\sigma_i$  are the spin variables in the Ising model.

$$\begin{aligned} \mathbf{q}^* &= \underset{\mathbf{q}}{\operatorname{argmin}} E'(\mathbf{q}), \\ E'(\mathbf{q}) &:= - \sum_{i \neq j} a_{ij} q_i q_j - \sum_i b_i q_i. \end{aligned} \quad (2)$$

### 3.2 Basic Principles of QA

Simulated annealing (SA), which utilizes thermal fluctuations, is recognized as a traditional method for solving COPs. In contrast, QA is an algorithm that leverages quantum fluctuations to identify the optimal solution of a COP as the minimum energy solution Kadowaki and Nishimori [1998]. It is capable of traversing potential barriers during the search for the minimum energy solution, thereby enabling a search for the globally optimal solution with the lowest energy without being trapped in local optimum.

In this context, the  $x$ ,  $y$ , and  $z$  components of the Pauli spin matrix  $\sigma$ , and the  $2 \times 2$  identity matrix  $I$  are defined as follows:

$$\begin{aligned} \sigma^x &:= \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \quad \sigma^y := \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}, \\ \sigma^z &:= \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}, \quad I := \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}. \end{aligned} \quad (3)$$

The  $n$ -fold tensor product of the identity matrix  $I$  is denoted as  $I^{\otimes n}$ , and the  $N$ -fold tensor product of the Pauli spin matrix  $\sigma$  and the identity matrix  $I$  is expressed as:

$$\sigma_i^* = I^{\otimes(i-1)} \otimes \sigma^* \otimes I^{\otimes(N-i)}. \quad (* = x, y, z) \quad (4)$$

Under the above setting, the Hamiltonian of the Ising model, denoted as

$$H_1 = - \sum_{i \neq j} J_{ij} \sigma_i^z \sigma_j^z - \sum_i h_i \sigma_i^z \quad (5)$$

is introduced in relation to the cost function (equation (1)) of the COP to be solved. The tensor product  $\sigma_i^z$  in equation (5) corresponds to the spin variable  $\sigma_i$  in equation (1).

To initiate the search for the solution to the COP from a quantum superposition state, the effect of quantum fluctuations due to a transverse magnetic field is utilized to realize a quantum superposition state. The Hamiltonian of the transverse magnetic field is expressed as follows:

$$H_0 = \alpha \sum_i \sigma_i^x. \quad (6)$$

Subsequently, the Hamiltonian, which integrates  $H_0$  and  $H_1$ , is introduced as:

$$H(t) := \left(1 - \frac{t}{T}\right) H_0 + \frac{t}{T} H_1, \quad (7)$$

This Hamiltonian evolves over time from  $H_0$  at time  $t = 0$  to  $H_1$  at time  $t = T$ . At the time  $t = 0$  when the search for the optimal solution begins, a quantum superposition state is realized. If the time evolution is sufficiently slow, according to the adiabatic theorem of quantum mechanics, in an ideal system free from external noise, the state is always maintained in the ground state. Therefore, it is believed that the ground state of  $H_1$ , which is desired to be solved, can be obtained with a high probability at time  $t = T$  Kadowaki and Nishimori [1998].

## 4 Formulation of MOT via QUBO Representation

In this section, the MOT problem described in Section 2 is formulated as the problem of minimum weight maximal matching in a bipartite graph. In existing research on MOT, methods using maximum matching, maximal matching, and stable matching have been proposed. However, this paper presents a methodology that employs maximal matching. This approach was chosen for two primary reasons:

1. Maximal matching can be expressed in the QUBO (Quadratic Unconstrained Binary Optimization) formula mentioned later.
2. In the situation of MOT, the degree of each node in the bipartite graph is often uneven, and compared to the maximum matching prioritizing maximizing the number of matchings, maximal matching tends to yield more stable results.

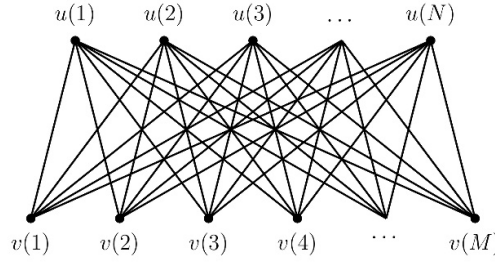


Figure 2: Illustration of matching

Let  $U = \{u(i)\}_{i=1}^N$  denote the set of tracks at the detection frame at time  $t - 1$ , and  $V = \{v(j)\}_{j=1}^M$  denote the set of detected objects at the detection frame at time  $t$ . The time  $t$  is counted for each detection frame and is assumed to take an integer value. At this time, a bipartite graph  $G = (U \cup V, E)$  is introduced, where edges are only placed between  $u \in U$  and  $v \in V$ . Here,  $U \cap V = \emptyset$ , and  $E$  is the set of edges  $E = \{(u(i), v(j)) \mid u(i) \in U, v(j) \in V\}$ . An illustration of the bipartite graph  $G$  is shown in Figure 2.

In MOT problems, when the correspondence of the same object between  $U, V$  is represented by  $M(\subseteq E)$ , different  $e_1, e_2 \in M (e_1 \neq e_2)$  do not share the same node. Therefore, MOT problems can be considered as a type of matching problem in bipartite graph  $G$ .

To quantitatively evaluate the quality of matching, a function  $w$  on  $E$  is introduced. The similarity between objects  $u(i), v(j)$  in adjacent detection frames is measured by some indicator, such as the distance between objects or the similarity of image features. This indicator is denoted by  $w(u(i), v(j))$ , which takes values in  $[0, 1]$ , where a larger value implies a higher similarity.

The MOT problem can now be considered as a maximum-weight maximal matching problem that searches for a matching  $M$ :

$$M^* = \operatorname{argmax}_M \sum_{m \in M} w(m). \quad (8)$$

Whether or not  $(u, v) \in E$  is included in matching  $M$  can be represented by

$$x_{u,v} = \begin{cases} 1 & (u, v) \in M \\ 0 & (u, v) \notin M \end{cases} \quad (9)$$

in terms of the binary variable  $x_{u,v}$  on  $E$ . Using the cost function  $F_w$  on  $E$ , the optimization problem (8) can then be rewritten as an optimization problem

$$\begin{aligned} \mathbf{x}^* &= \operatorname{argmin}_{\mathbf{x} = \{x_{u,v} \mid (u,v) \in E\}} F_w(\mathbf{x}) \\ F_w(\mathbf{x}) &:= - \sum_{(u,v) \in E} w(u,v) x_{u,v} \end{aligned} \quad (10)$$

under constraints

$$\sum_{v \in V} x_{u,v} \leq 1 \quad (\forall u \in U), \quad \sum_{u \in U} x_{u,v} \leq 1 \quad (\forall v \in V), \quad (11)$$

which represent matching constraints (matching edges never share the same node).

Here, the variable pair  $\mathbf{x} = \{x_{u,v} \mid (u,v) \in E\}$  that satisfies the constraint conditions (11) can be considered as one of the sets of variables that minimize the following formulas:

$$F_U(\mathbf{x}) := \frac{1}{2} \sum_{u \in U} \left( \sum_{v \in V} x_{u,v} - 0.5 \right)^2, \quad F_V(\mathbf{x}) := \frac{1}{2} \sum_{v \in V} \left( \sum_{u \in U} x_{u,v} - 0.5 \right)^2. \quad (12)$$

Since the binary variable  $x$  always satisfies  $x^2 = x$ , ignoring constant terms allows equations (12) to be simplified as the following quadratic homogeneous polynomial:

$$F_U(\mathbf{x}) := \sum_{u \in U} \left( \sum_{i < j} x_{u,v(i)} x_{u,v(j)} \right), \quad F_V(\mathbf{x}) := \sum_{v \in V} \left( \sum_{i < j} x_{u(i),v} x_{u(j),v} \right). \quad (13)$$

Using Lagrange's method of undetermined multipliers, the optimization problem (10) under the constraints (11) can be summarized as

$$\begin{aligned} \mathbf{x}^* &= \underset{\mathbf{x} \in \{x_{u,v} \mid (u,v) \in E\}}{\operatorname{argmin}} F(\mathbf{x}) \\ F(\mathbf{x}) &:= F_w(\mathbf{x}) + \lambda F_U(\mathbf{x}) + \lambda F_V(\mathbf{x}) \end{aligned} \quad (14)$$

using constraint expressions (13). Here,  $\lambda (> 0)$  is a hyperparameter that adjusts the strength of the constraint equation. In the subsequent sections, the parameter  $\lambda$  is consistently set to 0.7. Since  $F(\mathbf{x})$  is a QUBO formulation, it can be implemented on a D-Wave machine to solve combinatorial optimization problems.

## 5 Proposed Method 1: Quantum-Ensemble MOT to Improve Accuracy

### 5.1 Integration of Multiple Types of Detectors

To improve the accuracy of MOT, several methods have been developed to integrate the outputs of multiple types of trackers. The Layer-wise Aggregation Discriminative Model (LADM) Peng et al. [2020] uses a weighted average of predictors in the detection process, while the GIAOTracker Du et al. [2021] fuses multiple trackers using the score of TrackNMS, which is inspired by SoftNMS. Moreover, EnsembleMOT Du et al. [2022], which is inspired by ensemble learning in machine learning but does not require a model or training, has also been proposed. However, these developments fail to account for the escalation in computational cost that accompanies the increase in both the number of detected objects and the diversity of trackers.

### 5.2 Multiplication of Matching

This paper proposes a method that can reduce the computational cost of bipartite graph matching for object tracking, even when the number of detected objects and the types of trackers increase. The core idea is to use quantum annealing to speed up the search for the optimal matching, as shown in Figure 3. Quantum annealing is utilized to suppress the computation time even when the number of nodes in the bipartite graph increases.

Section 4 treats the MOT problem as a maximal matching problem in the bipartite graph  $G = (U \cup V, E)$  using a single tracker. By employing  $F_w, F_U, F_V$  in Equations (10) and (12), it formulates an optimization problem with the QUBO formula  $F$  in Equation (14), which is solvable by quantum annealing.

Considering the scenario where there are  $P$  types of trackers  $\{\mathcal{T}_p\}_{p=1}^P$ , each of which can be formulated as maximal matchings in a bipartite graph, an extended bipartite graph  $\hat{G} = \left( \bigcup_{p=1}^P U_p \cup \bigcup_{p=1}^P V_p, \bigcup_{p=1}^P E_p \right)$  is constructed. This graph duplicates the set of tracks  $U = \{u(i)\}_{i=1}^N$  in the detection frame at time  $t-1$  and the set of detected objects  $V = \{v(j)\}_{j=1}^M$  in the detection frame at time  $t$  into  $P$  copies. Here,  $E_p$  is a set of edges drawn only between  $U_p, V_p$ , and the weight  $w_p$  of  $E_p$  is determined according to the type of tracker  $\mathcal{T}_p$ . Multiplication of Matching is illustrated in Figure 4.

The problem of  $P$  types of tracking is considered, which is combined into a single QUBO equation and is to be solved using quantum annealing:

$$F_p(\mathbf{x}^p) := F_{w_p}(\mathbf{x}^p) + \lambda F_{U_p}(\mathbf{x}^p) + \lambda F_{V_p}(\mathbf{x}^p),$$

$$(\mathbf{x}^1, \mathbf{x}^2, \dots, \mathbf{x}^P)^* = \underset{\mathbf{x}_p \in \{\mathbf{x}_{u,v}^p \mid (u,v) \in E_p\}}{\operatorname{argmin}} \sum_{p=1}^P F_p(\mathbf{x}^p). \quad (15)$$

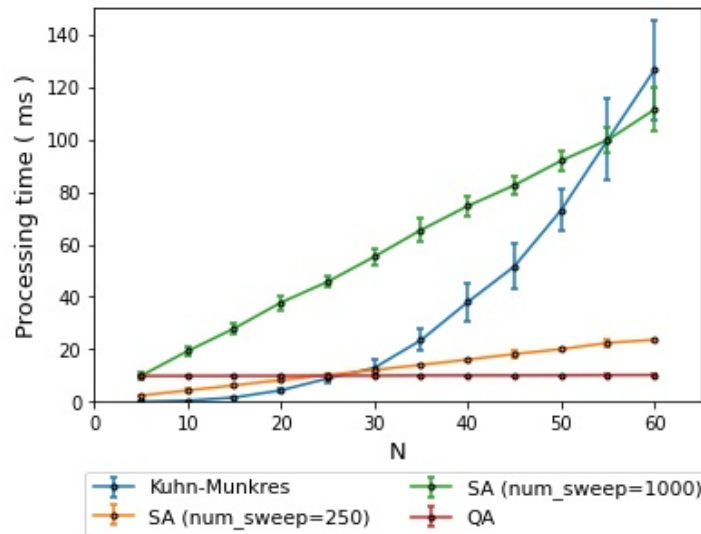


Figure 3: Number of detected objects and processing time for matching

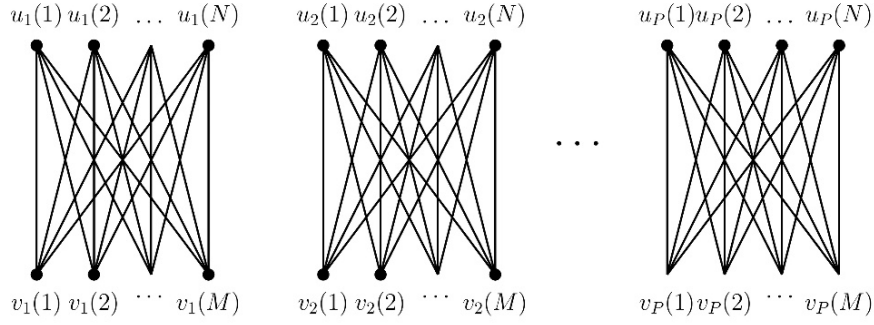


Figure 4: Illustration of multiplexing matching

For the detected object  $v(j) \in V = \{v(j)\}_{j=1}^M$  at time  $t$ , the optimal solution  $(x^1, x^2, \dots, x^P)^*$  of optimization problem (15) yields a set of track candidates

$$\{u_p(i) \mid x_{u_p(i), v_p(j)}^p = 1, 1 \leq i \leq N, 1 \leq p \leq P\} \quad (16)$$

related to  $v(j) = v_p(j)$ . Using the integration method of multiplexed matching described in the next subsection 5.3, the track candidates are considered for aggregation into a single  $i^*$ . Then, the track  $u(i^*)$  at time  $t - 1$  is associated with the detected object  $v(j)$  at time  $t$ .

### 5.3 Integration of Multiplexed Matching

In conventional research, a method has been adopted to improve the accuracy of solutions by integrating multiple patterns of solutions to combinatorial optimization problems through a majority voting method Apaydin et al. [2008], Yoshida et al. [2022], Shim et al. [2017]. In the case of maximal matching problems, the integration by majority voting can be showed as in Figure 5.

However, in the case of integration by majority voting, there is an effect of integration if it is a double matching, but no effect of accuracy improvement is observed in the case of triple or more matchings. Ten patterns per  $N$  were artificially generated for  $N \times N$  bipartite graph where all nodes have a degree of 4, and the results of the effect of integration of multiple matchings by majority voting were demonstrated by the error rate of the solution in Figure 6. Trial computations were conducted 250 times per graph.

Note that the weights of the edges were randomly assigned in the range of -1.0 to 0.0. The reason for fixing the degree of nodes to 4 is because it is assumed that the average number of neighboring vehicles in the vehicle tracking problem is around 4.

This paper proposes a method for cyclically integrating solutions as follows. The integration of tracking results by multiple types of tracking methods, as mentioned in the previous subsection, allows for the types of matching problems to be heterogeneous (the same nodes and edges are used for each, but the weights can be different).

For the same bipartite graph  $G = (V, E)$  ( $V$ : set of nodes,  $E$ : set of edges), weights  $W_i$  are assigned to  $E$  and denoted as  $G_i = (V, E, W_i)$ . The maximal matching solution of  $G_i$  is denoted as  $M_i = (V_i, E_i)$  ( $V_i, E_i$  are subsets of  $V, E$  respectively).

---

**Algorithm 2** Algorithm for proposed integration method
 

---

- 1: Initialize  $K$  with  $M_1$ .
  - 2: **for** each iteration  $i$  from 1 to  $P$  **do**
  - 3:     Integrate  $K = (V_K, E_K)$  and  $M_j$  ( $j := i + 1 \bmod P$ ) to obtain a subgraph  $H_i = (V_K \cup V_j, E_K \cup E_j)$  of  $G$ .
  - 4:     **if** a path containing a node of degree 2 is found within  $H_i$  **then**
  - 5:         Locate an alternating path that minimizes the sum of weights, analogous to finding the solution for maximum matching through augmenting path search.
  - 6:         Establish this as the new integrated solution  $K$ .
  - 7:     **end if**
  - 8: **end for**
-



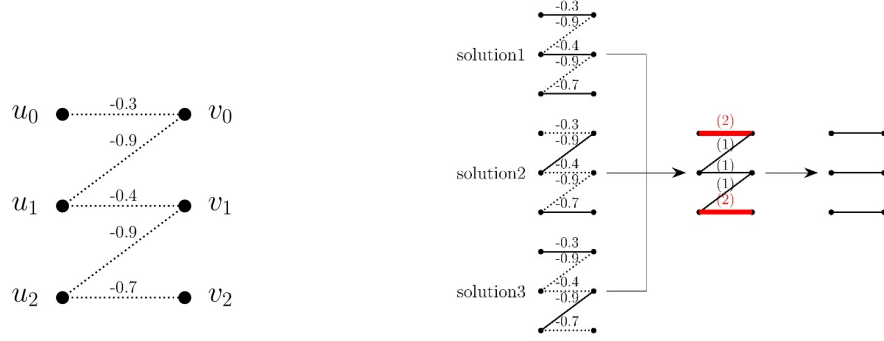


Figure 5: This is an illustration of the integration of multiple matchings by majority voting. For a bipartite graph on the left, three types of matching solutions (solution1-3) are obtained. The voting results are shown in the central graph. The numbers in parentheses on the edges of the central graph represent the number of votes, and the red lines indicate the edges with the most votes. The final solution depicted on the right is a solution of maximum matching, but not of

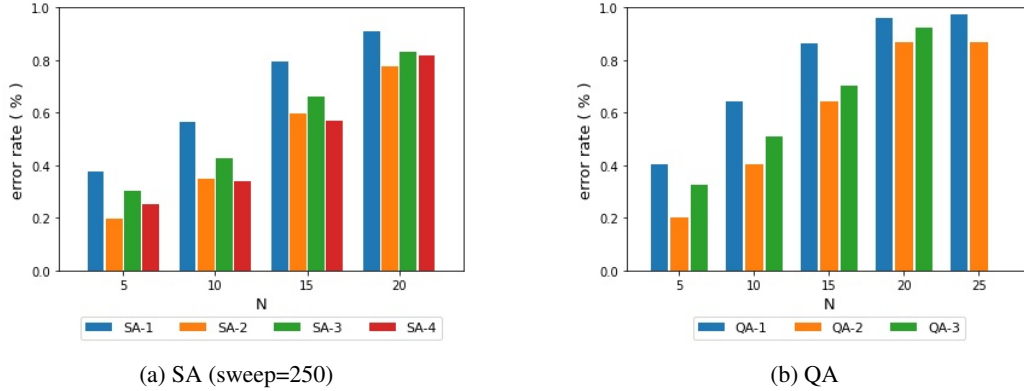


Figure 6: Comparison of error rates by majority voting method. The abbreviations SA and QA denote Simulated Annealing and Quantum Annealing, respectively. Furthermore, SA- $k$  and QA- $k$  each represent the integration of solutions for  $k$ -multiplexed matching. In cases of triple or higher order matching, satisfactory results have not been achieved in comparison to double matching.

Then, as described in Algorithm 2 and Figure 7, the set of maximal matching solutions  $\{M_i\}_{i=1}^P$  is integrated cyclically.

The results of the proposed method's effect on the integration of multiple matchings, verified by the error rate of the solution using the same sample as the experimental results in Figure 6, are shown in Figure 8. In the case of double matching, the accuracy is the same and settings as that of the previously mentioned majority vote integration. However, as the multiplicity of matching increases, the effect of integration becomes apparent. Furthermore, the effect of matching integration, measured by TTS, is shown in Figure 9.

**TTS** TTS (Time-to-solution) Albash and Lidar [2018] is utilized as a metric to measure the efficiency of the simulated or quantum annealing algorithm in finding the optimal solution. TTS signifies the average computation time required to obtain the true optimal solution, given that the annealing time  $T$  per trial and the target probability  $P_R$  of obtaining the true optimal solution at least once are predetermined. When the probability  $P_S(t)$  of a successful annealing at time  $T$  is considered, TTS is defined as follows:

$$\text{TTS} := \begin{cases} T \cdot \frac{\log(1-P_R)}{\log(1-P_S(t))} & \text{as } P_S(t) < P_R \\ T & \text{as } P_S(t) \geq P_R \end{cases} \quad (17)$$

Even when verified with TTS, as shown in Figure 9, it was found that the effect of multiple matching is significantly demonstrated by the method using QA, which has a less increase in QPU access time.

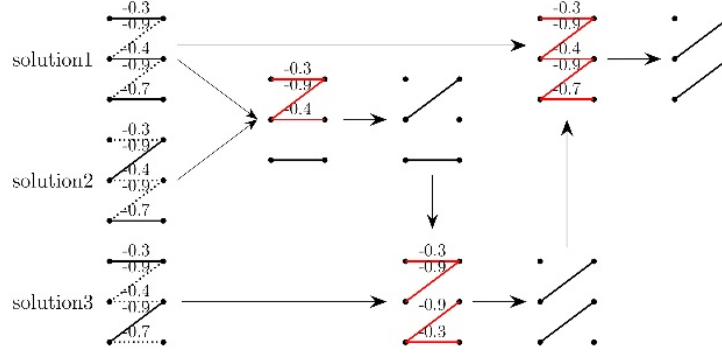


Figure 7: This is an illustration of the integration of multiple matchings by the proposed method. Similar to Figure 5, when three types of matching solutions (solution1-3) are obtained, first, solution1-2 are integrated. As a result, paths containing nodes of degree 2 that appear are indicated by red lines. Among the red lines, the alternating path whose sum of weights is minimal is retained as the integrated solution. Ultimately, the correct solution of maximal matching is obtained.

## 5.4 Evaluation Experiments

This subsection performs a quantitative evaluation of MOT using both conventional baseline methods and the proposed method, thereby demonstrating the superiority of the proposed method.

### 5.4.1 Experimental Data

The following data is utilized for the experiment:

- MVI 39271 (1min 18sec, 20fps) and MVI 39401 (1min 9sec, 20fps) from the UA-DETRAC dataset uad, Wen et al. [2020], which is a benchmark for traffic MOT with pre-annotated vehicle locations and IDs (Figure 10 ).

### 5.4.2 Experimental Settings

Experiments were conducted using the D-Wave Advantage2 prototype1.1 for quantum annealing in the environment outlined in Table 1. For vehicle detection, YOLOv5 was employed across all experimental patterns. As depicted in Figure 1, the detection process was carried out at approximately 10 frames per second. For the tracking methods, a comparative evaluation was conducted on the following methods.

- Baseline Method: DeepSORT, which matches by similarity of appearance features of vehicles detected in the previous frame and matches in bounding box with location prediction.
- Proposed Method: As presented in subsection 5.2, a method is proposed to solve multiplex maximal matching using quantum annealing. It should be noted that each maximal matching uses two types:

**tracking1.** Similar to SORT, the Intersection over Union (IoU) between the trajectory forecasting by the Kalman filter and the detection location on the current frame is used as the weight.

**tracking2.** The similarity of appearance features between the most recent detected object (object1) in track and the detected object (object2) on the current frame, calculated by perceptual hash Zauner et al. [2011], is used as the weight. Specifically, let PH1 and PH2 denote the 64-bit Perceptual Hashes that can be obtained from object1 and object2, respectively. When the Hamming distance between PH1 and PH2 is denoted as  $d_H(\text{PH1}, \text{PH2})$ , the similarity  $w_{1,2}$  between object1 and object2 is defined as

$$w_{1,2} := \exp \left( -\frac{d_H(\text{PH1}, \text{PH2})}{32} \right). \quad (18)$$

In the proposed method, quantum annealing calculations are configured to perform 100 trials per execution, each with an annealing time of 10  $\mu s$ .

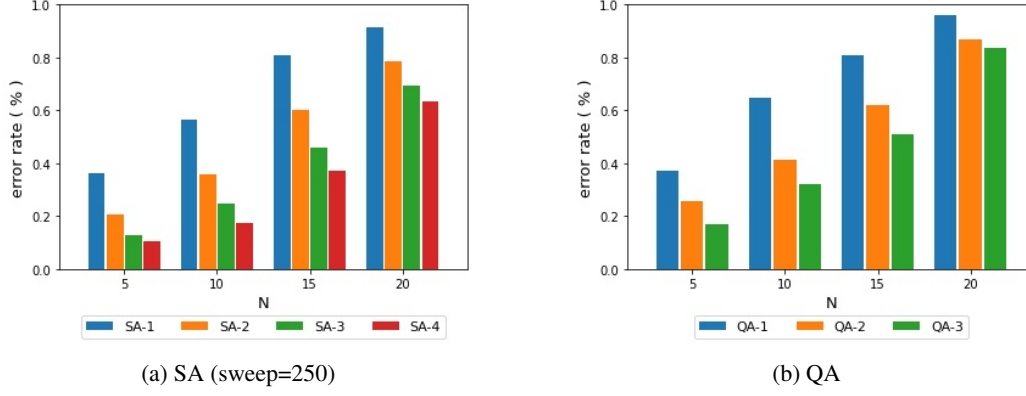


Figure 8: Comparison of error rates by proposed method. Contrary to the case presented in Figure 6, the effect of integration becomes more pronounced as the multiplicity of the matching increases.

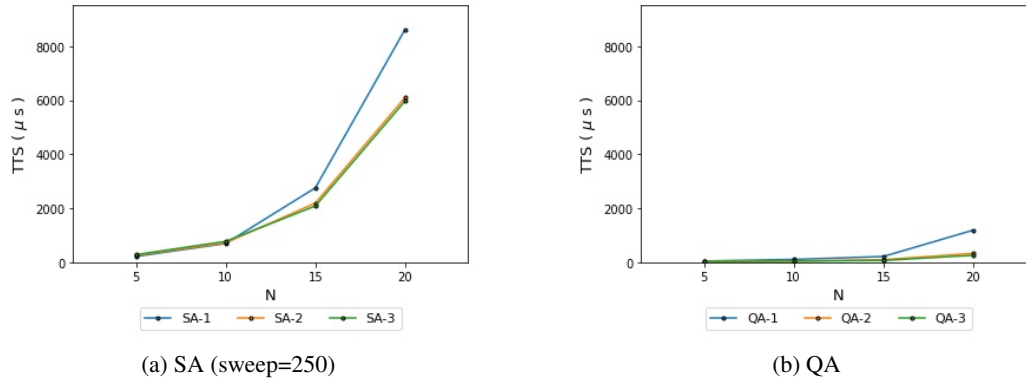


Figure 9: Comparison of TTS by proposed method



Figure 10: Overview of UA-DETRAC dataset

### 5.4.3 Evaluation Criteria

Here, each of the following quantitative evaluation metrics for the accuracy of multi-object tracking is discussed: IDSW, MOTA Bernardin and Stiefelwagen [2008], IDF1 Ristani et al. [2016], and the absolute percentage error of vehicle count.

**IDSW** IDSW (ID Switches) is the total number of times that the identifier (ID) of each object is erroneously switched during tracking.

**MOTA** MOTA (Multiple Object Tracking Accuracy) Bernardin and Stiefelwagen [2008] is a metric that evaluates the frequency of three typical error sources in MOT: misses, false positives, and mismatch errors, also known as non-detection, mis-detection, and swapping of tracking IDs. The value range is from 0 to 1; the closer to 1, the greater

the tracking performance.

$$\text{MOTA} := 1 - \frac{\text{FP} + \text{FN} + \text{IDSW}}{\text{GT}} \quad (19)$$

where

- FP (False Positives) is the total number of objects that are erroneously detected.
- FN (False Negatives) is the total number of objects that are undetected.
- GT (Ground Truths) is the total number of objects that are annotated.

**IDF1** IDF1 Ristani et al. [2016] is an evaluation metric for object tracking that measures not only the consistency of tracking, but also the consistency of identification. It is defined by the following formula:

$$\text{IDF1} = \frac{2 \text{IDTP}}{2 \text{IDTP} + \text{IDFP} + \text{IDFN}} \quad (20)$$

where

- IDTP (ID True Positives) is the number of objects that are correctly tracked. A tracking result is considered correct if it matches well with the ground truth data of the object's trajectory.
- IDFP (ID False Positives) is the number of objects that are wrongly tracked.
- IDFN (ID False Negatives) is the number of objects that are missed by the tracker.

Similar to the F1-score in object detection, IDF1 is defined as the harmonic mean of ID-Precision=IDTP/(IDTP+IDFP) and ID-Recall=IDTP/(IDTP+IDFN). IDF1 ranges from 0 to 1, and a higher value indicates a higher accuracy of tracking.

**Absolute Percentage Error (APE) of Vehicle Count** From the perspective of traffic flow measurement, the absolute percentage error of vehicle count is introduced as an intuitively understandable metric:

$$\text{Absolute Percentage Error of Vehicle Count} = \frac{|\text{NT} - \text{CN}|}{\text{CN}} \quad (21)$$

where

- NT is the number of tracked vehicles
- CN is the correct number of vehicles

This metric quantifies the error magnitude as a percentage relative to the correct number of vehicles, indicating that a value closer to 0.0 signifies superior MOT performance.

## 5.5 Evaluation Results

This subsection presents the quantitative evaluation results of the baseline and proposed methods for MOT in Table 2. It also shows the quantitative evaluation results of the object detection in the pre-processing stage. The ground truth data are obtained from the annotation results of vehicle detection and tracking (vehicle bounding boxes and IDs).

The proposed methods (QA $k$  ( $k = 1, 2, 12$ )) generally outperform the conventional method (DeepSORT) in all metrics. For the movie MVI 39271, the tracking method QA2 based on image features has worse IDSW and APE values than QA1, due to the presence of many taxis with similar appearance. However, the integrated method QA12 can mitigate the errors of QA2 by combining QA1 and QA2.

For the movie MVI 39401, the tracking accuracy of QA1 and QA2 is slightly degraded by the occlusion caused by poles and large vehicles, and the interference of surrounding vehicles with similar appearance. However, the integrated method QA12 can complement the errors of QA1 and QA2 by fusing them. These results demonstrate the effectiveness of the proposed integration method.

Table 2: Evaluation results of object detection (YOLOv5) and MOT (DeepSORT, Quantum Annealing ) using the UA-DETRAC dataset. It should be noted that the same object detection results were used for all multi-object tracking. Furthermore, QA is an abbreviation for Quantum Annealing, and  $QA_k$  ( $k = 1, 2$ ) represents the results obtained by tracking $k$  as explained in 5.4.2 Experimental Settings. QA-12 denotes the integrated results of tracking1 and tracking2. Values in **bold** represent the best numerical results among the compared tracking methods.

movie		MVI 39271			
detection	Precision	0.9997			
	Recall	0.9143			
	F1-score	0.9551			
tracking	method	DeepSORT	QA1	QA2	QA12
	MOTA	0.9031	<b>0.9050</b>	0.9044	<b>0.9050</b>
	IDF1	0.9446	0.9513	<b>0.9520</b>	0.9513
	IDSW	9	<b>3</b>	5	<b>3</b>
	APE	0.1702	<b>0.0000</b>	0.0426	<b>0.0000</b>
	NT	55	<b>47</b>	49	<b>47</b>
	(CN)	(47)			
movie		MVI 39401			
detection	Precision	0.9967			
	Recall	0.9528			
	F1-score	0.9742			
tracking	method	DeepSORT	QA1	QA2	QA12
	MOTA	0.9424	0.9433	<b>0.9436</b>	0.9405
	IDF1	0.9285	0.9563	<b>0.9710</b>	0.9646
	IDSW	20	16	15	<b>14</b>
	APE	0.0976	0.0244	0.0244	<b>0.0000</b>
	NT	90	84	84	<b>82</b>
	(CN)	(82)			

## 6 Proposed Method 2: Efficiency Improvement of MOT by Reverse Annealing

In MOT, tracking processing is typically performed for each detection frame. However, given that the position of an object changes gradually, the tracking result for the current frame is often similar to that of the previous frame, depending on the situation. In this section, by utilizing such properties, a method is proposed to further enhance the efficiency of MOT processing through reverse annealing.

### 6.1 Reverse Annealing (RA)

RA is a technique that employs the inverse process of quantum annealing to search for a more optimal solution when a candidate for the optimal solution is known through other methods Passarelli et al. [2020]. It is utilized for local solution search. In this context, quantum annealing described in subsection 3.2 is called forward annealing (FA) to distinguish it from RA.

In the time evolution of the Hamiltonian (Equation (7)), FA starts from a strong quantum superposition state with a transverse field to select and evaluate all combinations of solutions with equal probability, and gradually reduces the influence of the transverse field. In contrast, RA begins from a state unaffected by the transverse field, with the trial optimal solution as the initial value. It then increases the influence of the transverse field to include the effect of quantum fluctuations. After a temporary pause, it reduces the influence of the transverse field to efficiently search for a more refined solution Passarelli et al. [2020]. This pause is used in open systems like D-Wave machines that are subject to thermal effects, to efficiently search for solutions using thermal relaxation. Examples of annealing schedules for forward annealing and reverse annealing are shown in Figure 11.

### 6.2 Enhanced MOT through RA

RA is utilized to seek a more optimal solution. There have been instances where RA has expedited the search for solutions with high accuracy using shorter annealing schedules than those in QA (FA) Haba et al. [2022], rev. This paper extends this approach and presents a method that employs reverse annealing to enhance the efficiency of MOT.

To rapidly determine the initial values for MOT between the  $(t - 1)$ -th and  $t$ -th detection frames, each node  $u$  of the tracks up to the  $(t - 1)$ -th detection frame and each node  $v$  of the detection in the  $t$ -th frame are encoded as follows. Initially, the frame image is partitioned into regions both vertically and horizontally, with a position code assigned to each region (Figure 12 (a)). For each node  $u$  of the tracks, a position code  $c(u)$ , corresponding to the center of the bounding box that represents the predicted position in the  $t$ -th frame, is assigned. This prediction is based on the detection and tracking results up to the  $(t - 1)$ -th detection frame and is made by the Kalman filter in **tracking1**. method described in Subsection 5.4. For each node  $v$  of the detection, a position code  $c(v)$ , corresponding to the center of the bounding box that represents the detection result, is assigned. Only when  $c(u) = c(v)$ , a subgraph that matches  $u$  and  $v$  is constructed and used as the initial value (Figure 12 (b)). When formulating the MOT problem between the  $(t - 1)$ -th

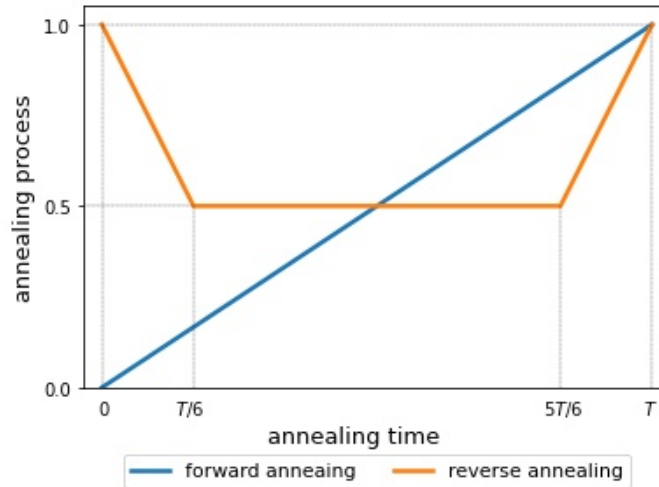
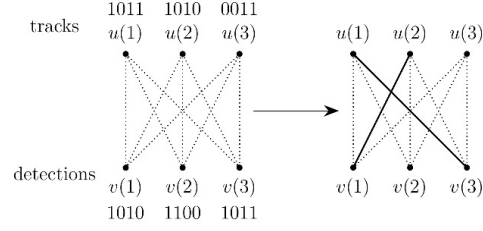


Figure 11: An example of an annealing schedule. Here, both forward annealing and reverse annealing are set to the same annealing time  $T$ .

0000	0001	0010	0011
0100	0101	0110	0111
1000	1001	1010	1011
1100	1101	1110	1111

(a) location code table



(b) construction of initial value

Figure 12: (a) Example of location code table (when the image is divided into four parts both vertically and horizontally), and (b) construction of initial value from local code (The four-digit number, noted adjacent to the nodes of the graph on the left, represents the position code.).

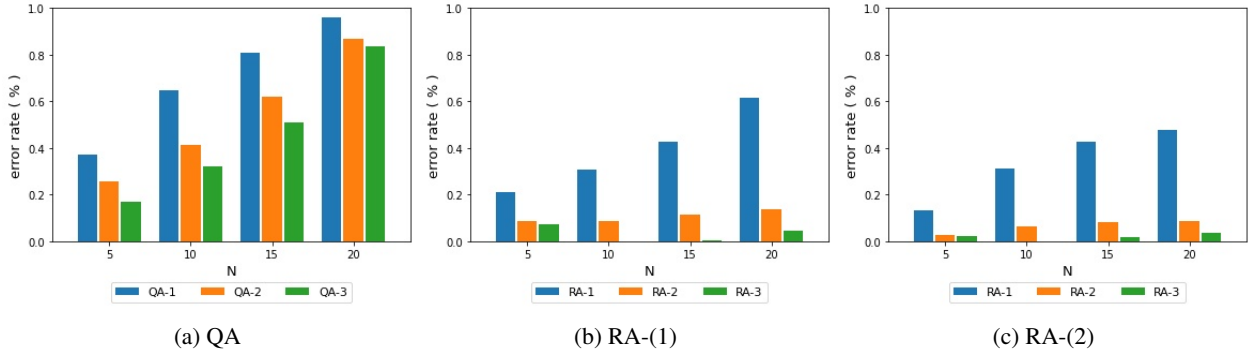


Figure 13: Comparison of error rates by QA (Quantum Annealing) and RA (Reverse Annealing). The multiplicity  $k$  of the integration of matching solutions is denoted as QA- $k$  and RA- $k$ . The annealing time per trial in (b) RA-(1) and (c) RA-(2) was 10  $\mu$ s and 3  $\mu$ s, respectively.

and  $t$ -th detection frames in the same manner as Equations (14) and (15), the initial value  $\hat{x} = \{x_{u,v} \mid (u, v) \in E\}$  can be expressed as follows:

$$x_{u,v} = \begin{cases} 1 & \text{as } c(u) = c(v) \\ 0 & \text{otherwise} \end{cases} \quad (22)$$

Starting from this initial value, a more refined solution is sought by RA.

To verify the effect of maximal matching by reverse annealing, the same artificially generated  $N \times N$  bipartite graph as used in Subsection 5.3 is utilized. The results of the quantitative evaluation by error rate are shown in Figure 13. Trial computations were conducted 250 times per graph. The initial values for reverse annealing were obtained by randomly flipping the bits of the optimal solution for maximal matching with a probability of 5%. This was premised on the assumption that the initial value predictions, based on the position prediction by the tracker and the object detection results, would be incorrect at a rate of less than 5% per quantum bit.

From these results, it can be anticipated that if the initial value can be predicted almost accurately, the accuracy of matching by reverse annealing can be expected to be higher than that of forward annealing. By integrating the results of multiple matchings, a further improved accuracy can be expected.

Also, similar to Figure 14, from the results evaluated by TTS, it can be inferred that the solution accuracy per unit time of reverse annealing is better than that of forward annealing, and the correct answer can be obtained with fewer trials of annealing computation.

### 6.3 Evaluation Experiments

This Subsection describes the results of evaluation experiments carried out to demonstrate the effectiveness of MOT by RA.

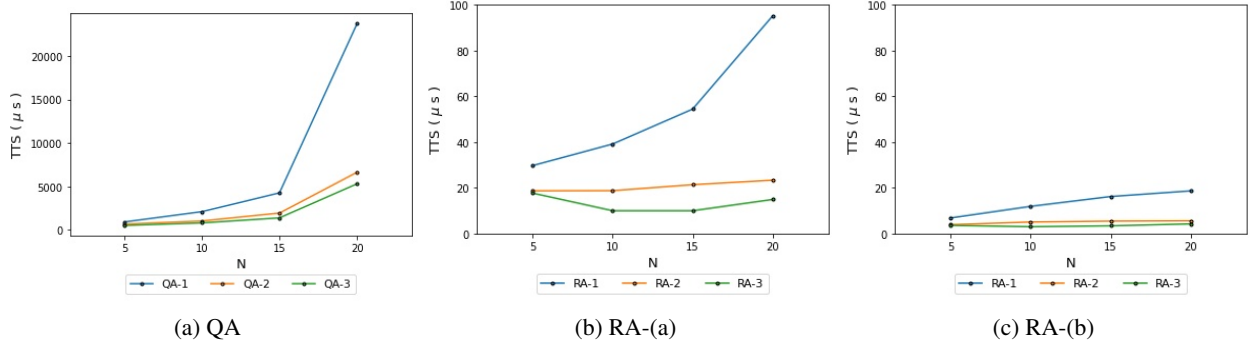


Figure 14: Comparison of TTS by QA and RA

Table 3: The accuracy when performing MOT by RA, using the same tracking results as in Table 2.  $RA_k$  ( $k = 1, 2$ ) represents the results obtained by tracking  $k$  as explained in 5.4.2 Experimental Settings. RA12 denotes the integrated results of **tracking1** and **tracking2**.

movie		MVI 39271					
tracking	method	QA1	QA2	QA12	RA1	RA2	RA12
	MOTA	<b>0.9050</b>	0.9044	<b>0.9050</b>	0.9047	0.9040	0.9047
	IDF1	0.9513	<b>0.9520</b>	0.9513	0.9513	<b>0.9520</b>	0.9513
	IDSW	<b>3</b>	5	<b>3</b>	<b>3</b>	5	<b>3</b>
	APE	<b>0.0000</b>	0.0426	<b>0.0000</b>	<b>0.0000</b>	0.0426	<b>0.0000</b>
	NT	<b>47</b>	49	<b>47</b>	<b>47</b>	49	<b>47</b>
	(CN)	(47)					
	method	QA1	QA2	QA12	RA1	RA2	RA12
tracking	MOTA	0.9433	<b>0.9436</b>	0.9405	0.9402	0.9402	0.9409
	IDF1	0.9563	0.9710	0.9646	0.9563	<b>0.9712</b>	0.9366
	IDSW	16	15	14	16	14	<b>12</b>
	APE	0.0244	0.0244	<b>0.0000</b>	0.0244	0.0244	<b>0.0000</b>
	NT	84	84	<b>82</b>	84	84	<b>82</b>
	(CN)	(82)					

### 6.3.1 Experimental Data and Settings

The same experimental data and settings as described in subsection 5.4 was used. The annealing time for RA was set to  $3 \mu s$ , and the number of repeated trials per tracking process was limited to one.

### 6.3.2 Evaluation Results

The same object tracking results and evaluation criteria as in Subsection 5.4 are used and presented in Table 3. Although the results are comparable to those of QA, it can be observed from the results using the video MVI 39401 that the method RA12 has the least IDSW, among others. This indicates that good results can be obtained with a short annealing time ( $3 \mu s$  per tracking process).

In the method of RA12, the error rate per quantum bit of the initial value predicted using the location code described in Subsection 6.2 was 2.76%. In the method of RA111 (triple matching), the QPU access time per tracking process had an average value (AV) of 6.66 ms and a standard deviation (SD) of 1.33 ms. The pre-processing time for tracking, which includes initial value prediction, had an AV of 2.60 ms and a SD of 2.29 ms. The post-processing time for



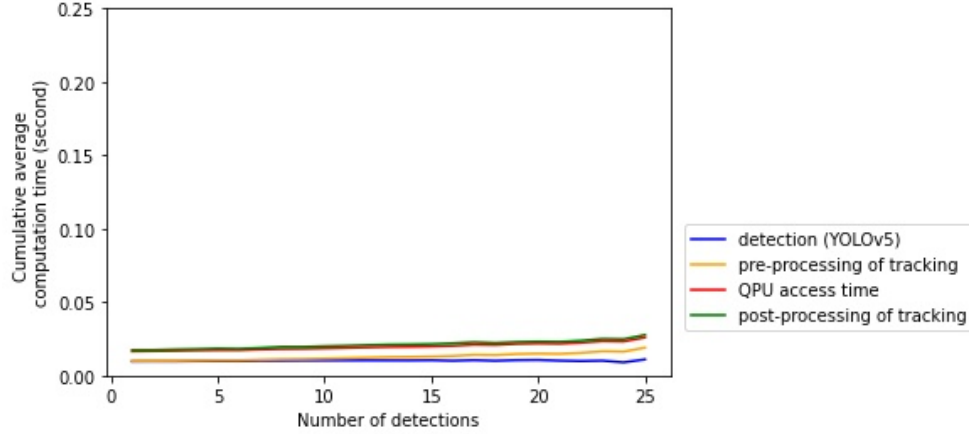


Figure 15: The cumulative processing time for detection by YOLOv5 and tracking by RA111 (triple matching). The right axis represents the number of detected vehicles, and the vertical axis represents the total processing time for detection and tracking (unit: seconds).

tracking, which includes integration processing, had an AV of 1.27 ms and a SD of 0.38 ms. The cumulative sum of these processing times and the tracking processing times is depicted in Figure 15.

## 7 Conclusions

In this paper, a method that leverages quantum annealing to enhance both the speed and accuracy of MOT by ensembling the MOT process was proposed and demonstrated. A method that further improves the efficiency of MOT by using RA was also proposed and demonstrated. Although there are still challenges in implementing QA on edge computing platforms and ensuring its cost-effectiveness, it is expected that proposed methods will be useful for real-time MOT applications, such as traffic flow measurement and control, collision prediction for autonomous driving, and quality control of mass-produced products. It is anticipated that proposed methods will contribute to the advancement of next-generation transportation and automation fields.

## Acknowledgments

In the execution of this research, substantial assistance was received from colleagues at NEC Solution Innovators, Ltd. The author express the profound gratitude.

## References

- Heng Fan, Liting Lin, Fan Yang, Peng Chu, Ge Deng, Sijia Yu, Hexin Bai, Yong Xu, Chunyuan Liao, and Haibin Ling. Lasot: A high-quality benchmark for large-scale single object tracking. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5374–5383, 2019.
- Linyu Zheng, Ming Tang, Yingying Chen, Jinqiao Wang, and Hanqing Lu. Learning feature embeddings for discriminant model based tracking. In *Computer Vision—ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XV 16*, pages 759–775. Springer, 2020.
- Li Zhang, Yuan Li, and Ramakant Nevatia. Global data association for multi-object tracking using network flows. In *2008 IEEE conference on computer vision and pattern recognition*, pages 1–8. IEEE, 2008.
- Yuan Li, Chang Huang, and Ram Nevatia. Learning to associate: Hybridboosted multi-target tracker for crowded scene. In *2009 IEEE conference on computer vision and pattern recognition*, pages 2953–2960. IEEE, 2009.
- Alex Bewley, Zongyuan Ge, Lionel Ott, Fabio Ramos, and Ben Uprocft. Simple online and realtime tracking. In *2016 IEEE international conference on image processing (ICIP)*, pages 3464–3468. IEEE, 2016.
- Nicolai Wojke, Alex Bewley, and Dietrich Paulus. Simple online and realtime tracking with a deep association metric. In *2017 IEEE international conference on image processing (ICIP)*, pages 3645–3649. IEEE, 2017.
- Yifu Zhang, Peize Sun, Yi Jiang, Dongdong Yu, Fucheng Weng, Zehuan Yuan, Ping Luo, Wenyu Liu, and Xinggang Wang. Bytetrack: Multi-object tracking by associating every detection box. In *European Conference on Computer Vision*, pages 1–21. Springer, 2022.
- Zhe Dai, Huansheng Song, Xuan Wang, Yong Fang, Xu Yun, Zhaoyang Zhang, and Huaiyu Li. Video-based vehicle counting framework. *IEEE Access*, 7:64460–64470, 2019.
- Bruk Gebregziabher. Multi object tracking for predictive collision avoidance. *arXiv preprint arXiv:2307.02161*, 2023.
- Julius Gudauskas and Žygimantas Matusevičius. Multiple object tracking for video-based sports analysis. In *CEUR workshop proceedings: IVUS 2021: Information society and university studies 2021: Proceedings of the 26th international conference on information society and university studies (IVUS 2021)*, Kaunas, Lithuania, April 23, 2021, volume 2915, pages 1–10. CEUR-WS, 2021.
- Robert Ward, Payam Soulatiantork, Shaun Finneran, Ruby Hughes, and Ashutosh Tiwari. Real-time vision-based multiple object tracking of a production process: Industrial digital twin case study. *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture*, 235(11):1861–1872, 2021.
- Harold W Kuhn. The hungarian method for the assignment problem. *Naval research logistics quarterly*, 2(1-2):83–97, 1955.
- James Munkres. Algorithms for the assignment and transportation problems. *Journal of the society for industrial and applied mathematics*, 5(1):32–38, 1957.
- Edward Farhi, Jeffrey Goldstone, Sam Gutmann, Joshua Lapan, Andrew Lundgren, and Daniel Preda. A quantum adiabatic evolution algorithm applied to random instances of an np-complete problem. *Science*, 292(5516):472–475, 2001.
- Andrew M Childs, Edward Farhi, and John Preskill. Robustness of adiabatic quantum computation. *Physical Review A*, 65(1):012322, 2001.
- Jan-Nico Zaeck, Alexander Liniger, Martin Danelljan, Dengxin Dai, and Luc Van Gool. Adiabatic quantum computing for multi object tracking. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8811–8822, 2022.
- Timothy M McCormick, Zipporah Klain, Ian Herbert, Anthony M Charles, R Blair Angle, Bryan R Osborn, and Roy L Streit. Multiple target tracking and filtering using bayesian diabatic quantum annealing. In *2022 Sensor Data Fusion: Trends, Solutions, Applications (SDF)*, pages 1–9. IEEE, 2022a.
- Timothy M McCormick, Bryan R Osborn, R Blair Angle, and Roy L Streit. Implementation of a multiple target tracking filter on an adiabatic quantum computer. In *2022 IEEE Aerospace Conference (AERO)*, pages 1–14. IEEE, 2022b.
- Andrew Lucas. Ising formulations of many np problems. *Frontiers in physics*, 2:5, 2014.
- Tadashi Kadowaki and Hidetoshi Nishimori. Quantum annealing in the transverse ising model. *Physical Review E*, 58(5):5355, 1998.
- Jinlong Peng, Yueyang Gu, Yabiao Wang, Chengjie Wang, Jilin Li, and Feiyue Huang. Dense scene multiple object tracking with box-plane matching. In *Proceedings of the 28th ACM International Conference on Multimedia*, pages 4615–4619, 2020.

- Yunhao Du, Junfeng Wan, Yanyun Zhao, Binyu Zhang, Zhihang Tong, and Junhao Dong. Giaotracker: A comprehensive framework for mcmot with global information and optimizing strategies in visdrone 2021. In *Proceedings of the IEEE/CVF International conference on computer vision*, pages 2809–2819, 2021.
- Yunhao Du, Zihang Liu, and Fei Su. Ensemblomot: A step towards ensemble learning of multiple object tracking. *arXiv preprint arXiv:2210.05278*, 2022.
- Mehmet Serkan Apaydin, Vincent Conitzer, and Bruce Randall Donald. Structure-based protein nmr assignments using native structural ensembles. *Journal of Biomolecular NMR*, 40:263–276, 2008.
- Asaki Yoshida, Tsukasa Miki, Moe Shimada, Yuri Yoneda, and Jun-Ichi Shirakashi. Experimental evaluation of performance improvement by sparse operation in ising spin computing. In *2022 IEEE Nanotechnology Materials and Devices Conference (NMDC)*, pages 29–32. IEEE, 2022.
- Yong Shim, Akhilesh Jaiswal, and Kaushik Roy. Ising computation based combinatorial optimization using spin-hall effect (she) induced stochastic magnetization reversal. *Journal of applied physics*, 121(19), 2017.
- Tameem Albash and Daniel A Lidar. Adiabatic quantum computation. *Reviews of Modern Physics*, 90(1):015002, 2018.
- The UA-DETRAC Benchmark Suite*. [Online]. Available: <https://detrac-db.rit.albany.edu/>. [Accessed: 14 - Jul - 2023].
- Longyin Wen, Dawei Du, Zhaowei Cai, Zhen Lei, Ming-Ching Chang, Honggang Qi, Jongwoo Lim, Ming-Hsuan Yang, and Siwei Lyu. Ua-detrac: A new benchmark and protocol for multi-object detection and tracking. *Computer Vision and Image Understanding*, 193:102907, 2020.
- Christoph Zauner, Martin Steinebach, and Eckehard Hermann. Rihamark: perceptual image hash benchmarking. In *Media watermarking, security, and forensics III*, volume 7880, pages 343–357. SPIE, 2011.
- Keni Bernardin and Rainer Stiefelhagen. Evaluating multiple object tracking performance: the clear mot metrics. *EURASIP Journal on Image and Video Processing*, 2008:1–10, 2008.
- Ergys Ristani, Francesco Solera, Roger Zou, Rita Cucchiara, and Carlo Tomasi. Performance measures and a data set for multi-target, multi-camera tracking. In *European conference on computer vision*, pages 17–35. Springer, 2016.
- Gianluca Passarelli, Ka-Wa Yip, Daniel A Lidar, Hidetoshi Nishimori, and Procolo Lucignano. Reverse quantum annealing of the p-spin model with relaxation. *Physical Review A*, 101(2):022331, 2020.
- Renichiro Haba, Masayuki Ohzeki, and Kazuyuki Tanaka. Travel time optimization on multi-agv routing by reverse annealing. *Scientific reports*, 12(1):17753, 2022.
- Reverse Quantum Annealing for Local Refinement of Solutions (white paper)*. [Online]. Available: [https://www.dwavesys.com/media/5hsjmvom/14-1018a-a\\_reverse\\_quantum\\_annealing\\_for\\_local\\_refinement\\_of\\_solutions.pdf](https://www.dwavesys.com/media/5hsjmvom/14-1018a-a_reverse_quantum_annealing_for_local_refinement_of_solutions.pdf). [Accessed: 14 - Jul - 2023].