33RD CONGRESS
OF THE INTERNATIONAL COUNCIL
OF THE AERONAUTICAL SCIENCES
STOCKHOLM, SWEDEN, 4-9 SEPTEMBER, 2022

# QUANTUM ARRIVAL AND DEPATURE MANAGEMENT

Nobuyuki Yoshikawa[1]

[1]Information Technology R&D center, Mitsubishi Electric Corporation, Kanagawa, Japan

## Abstract

The arrival management (AMAN) and departure management (DMAN) system solved by quantum annealer is investigated to improve computation efficiency. The ising model for AMAN/DMAN is built by converting the aircraft sequencing problem formulated by mixed integer programming. To reduce number of decision variables, discretized time slots and slot blocking function to represent separation manner is introduced. Furthermore some implementation technique is shown in this paper. These technique greatly reduce the computation time to 0.01 times, and it helps iterative computation required in actual airport operations. As a result, more than 100 times faster solving and same quality optimality is confirmed by numerical experiment.

**Keywords:** AMAN/DMAN, Quantum Annealing

## 1. Introduction

Aviation industry is in an unprecedented predicament under the influence of COVID-19 from 2020. However the air traffic demand is expected to grow by 4.0(Boeing) [**?**]% per year until 2040. According to the air traffic increasing, the number of the flight is expected to increase. Capacities of major airports are close to limit. Improving operational efficiency is required. Especially the runway operation is known as the bottleneck of airport. All aircraft pair using same runway must keep minimum separation depending on them wake turbulence categories. So the runway throughput is quite limited.

Arrival management (AMAN) and depature management (DMAN) optimize aircraft order of runeay assignment and improve the runway capacity. AMAN/DMAN can be expected not only improving the capacity but also reducing delays. Reducing relay times leads improving passenger satisfactions and reducing $CO_2$ consumption. Many formula of ANAMN/DMAN system is proposed [2, 3]. Runway capacity management is one of the aircraft sequencing problem (ASP) formulation described with mixed integer programming (MIP), and it can find an ideal optimal solution. On the other hand solving MIP requires large computation resource and long time. Estimated Landing Time (ELDT) or Expected Take-Off Time (ETOT) can be canged depending on many causes, e.g. wind conditions, passenger behavior, and ground handling. Therefore iterative computation is necessary. In this paper, the ising model to solve the aircraft sequencing problem around an airport with multiple runways in the arrival/depature mixed mode is investigated, and find the optimal runway assignment and the optimal assignment time. The ising model is solved with a quantum annealer, and this method is mentiond as quantum airctaft sequencing problem (QASP). Discrete time slot is introduced to reduce number of decision variables in QASP. Numerous decision variables are required if the inequalities for separation constraint of the ASP are simply converted into ising model. To avoid this, slot blocking function for separation is introduced. These techniques reduce the number of variables and improve an efficiency of computation, and solvability.

## 2. Preliminary

### 2.1 Aircraft Sequencing Problem

In this paper an aircraft sequencing problem (ASP) around runways is considered to improve the runway usage efficiency. The ASP is formulated in many stdies, and one is formulated as a mixed integer programming called runway capacity management (RCM) [4] and described as followings:

$$\text{Minimize} \sum_{f \in F} t_f - \underline{t}_f \tag{1a}$$

$$\text{subject to}$$

$$\sum_{f \in F} z_{rf} = 1, \quad \forall r \in R, \tag{1b}$$

$$\underline{t}_f \leq t_f \leq \bar{t}_f, \quad \forall f \in F, \tag{1c}$$

$$t_{f_2} \geq t_{f_1} + p_{f_1 f_2} - M\left(1 - y_{f_1 f_2}\right), \quad \forall f_1 \neq f_2 \in F, \tag{1d}$$

$$y_{f_1 f_2} + y_{f_2 f_1} \geq z_{rf_1} + z_{rf_2} - 1, \quad \forall r \in R, \forall f_1 < f_2 \in F, \tag{1e}$$

$$y_{f_1 f_2}, z_{rf} \in \mathbb{B}. \tag{1f}$$

(1a) the objective is to minimize a total delay for all aircraft set $F$ requesting to use the runway, (1b) each flight $f \in F$ should be assigned to exact one runway from available runway set $r \in R$, (1c) the assign time $t_f \in \mathbb{N}$ should be set between its time window, (1d) any flight pair assigned in the same runway must keep minimum separation $p_{f_1 f_2}$ depends on the wake turbulence category defined by the separation manner such like RECAT-1, and (1e) has a precedence. $y_{f_1 f_2} = 1$ indicates that the aircraft $f_1$ precedes aircraft $f_2$, and $z_{rf} = 1$ indicates that the aircraft $f$ assigned into the runway $r$.

### 2.2 Quantum Annealing

The quantum annealing (QA) has much attention as the nouvel computation tool to solve large combinatorial problem quickly [5]. In the quantum annealing the optimization problem is required to be described as a the quadratic unconstrained binary optimization (QUBO) form as the following:

$$f_q(x) = x^T Q x, \tag{2}$$

where $f_q(\cdot) : \mathbb{B}^n \to \mathbb{R}$ is the objective function to be minimzed, $x \in \mathbb{B}^n$ is a binary decision variables, and $Q \in \mathbb{R}^{n \times n}$ is a coefficient matrix to represent the problem. This coefficient matrix is mentioned as QUBO matrix.

To solve optimization problems with quantum computing, original problems is required to convert to QUBO form. Therefore linear programming is not suit to treat with QA method. Because difficulty of represent continuous decision variables with binary variables. In other hand optimization problem which can be described as integer programming or mixed integer programming form is suitable for converting into QUBO form.

## 3. Quantum ASP

In this paper, an ising model is formulated to solve ASP with quantum annealing machine.

### 3.1 Time slot

Time slots are introduced to reduce the size of solution space to be considered in ASP. The time slot is a time section divided by the discretized time width $\Delta t$. The set of time slots is described as bellow:

$$S = \{s_0, \cdots, s_k, \cdots, s_{\max}\} \tag{3}$$

where index $k$ belonging to the set of slot indices is defined as:

$$K = \left\{ k \mid \forall k \in \mathbb{N}_0, \, k\Delta t \leq \max_{f \in F} \bar{t}_f \right\}. \tag{4}$$

The ready time slot and the due time slot are also represented as $s_{\underline{k}_f}$ and $s_{\bar{k}_f}$ with following slot indices, respectively.

$$\underline{k}_f = \left\lfloor \frac{\underline{t}_f}{\Delta t} \right\rfloor, \tag{5a}$$

$$\bar{k}_f = \left\lfloor \frac{\bar{t}_f}{\Delta t} \right\rfloor. \tag{5b}$$

## 3.2 QASP formulation

### 3.2.1 Decision variables

Decision variable $x_{frk} \in \mathbb{B}$ represents that the aircraft $f$ is assigned into the runway $r$ at the time slot $k \in K_f \subset K$. $K_f$ is an available index subset for aircraft $f$, which containing indices between the ready slot and due slot of the aircraft as the following:

$$K_f = \left\{k - \underline{k}_f\right\} \mid \forall k \in K,\, \underline{k}_f \leq k \leq \bar{k}_f. \tag{6}$$

Here these indices indicate numbers of delayed slot from the ready slot for the aircraft $f$.

### 3.2.2 Total delay

The ASP aims to minimize a sum of delays or total fuel consumption. Here the total delay is adopted as the criteria in the QASP. In the QASP, the decision variable directory means the number of delayed slot from fastest assignment of each aircraft. The delay time of aircraft $f$ can describe with the delayed slot and the time width as $k_f \Delta t$. Then the total delay is written as the following:

$$\sum_{f \in F} k_f \Delta t = \sum_{f \in F} \sum_{r \in R} \sum_{k \in K_f} (k\Delta t) x_{frk}. \tag{7}$$

Here the ising model shall be composed of a quadratic form. Therefore the cost function to be minimized is described as below:

$$H_d(x) = \sum_{f \in F} \sum_{r \in R} \sum_{k \in K_f} (k\Delta t) x_{frk} x_{frk} \tag{8}$$

### 3.2.3 Constraints and Penalties

Any aircraft pairs are required to keep separation minima. So some slots after assigned are blocked to avoid to assign aircraft according to the separation manner. The number of blocked slot is determined depending on the wake turbulence categories of aircraft pair as the following:

$$P_{D f_1 f_2} = \left\lceil \frac{p_{f_1 f_2}}{\Delta t} \right\rceil. \tag{9}$$

Then the separation table is represented as the function as the following:

$$P(f_1, f_2, k_1, k_2) = \begin{cases} 1 & \text{if } k_1 + P_{D f_1 f_2} \geq k_2 \\ 0 & \text{otherwise} \end{cases}. \tag{10}$$

Therefore the separation constraints is given as the following penalty function:

$$H_s(x) = \sum_{r \in R} \sum_{f_1 \in F} \sum_{f_2 \in F/\{f_1\}} \sum_{k_1 \in K_{f_1}} \sum_{k_2 \in K_{f_2}} P(f_1, f_2, k_1, k_2) x_{f_1 r k_1} x_{f_2 r k_2}. \tag{11}$$

And aircraft shall be assigned only once. It leads the constraint and penalty function as followings:

$$\sum_{r \in R} \sum_{k \in K_f} x_{frk} = 1, \quad \forall f \in F \tag{12}$$

$$\longrightarrow H_u(x) = \sum_{f \in F} \left( \sum_{r \in R} \sum_{k \in K_f} x_{frk} - 1 \right)^2. \tag{13}$$

### 3.2.4 Objective function

A Hamiltonian as the objective function consist of the total delay Equation 8, and 2 penalties Equation 11 and Equation 13:

$$\begin{aligned} H(x) &= H_d(x) + \lambda_s H_s(x) + \lambda_u H_u(x) \qquad (14) \\ &= \sum_{f \in F} \sum_{r \in R} \sum_{k \in K_f} (k\Delta t) {x_{frk}}^2 + \lambda_u \sum_{f \in F} \left( \sum_{r \in R} \sum_{k \in K_f} x_{frk} - 1 \right)^2 \\ &\quad + \lambda_s \sum_{r \in R} \sum_{f_1 \in F} \sum_{f_2 \in F / \{f_1\}} \sum_{k_1 \in K_{f_1}} \sum_{k_2 \in K_{f_2}} P(f_1, f_2, k_1, k_2) x_{f_1 r k_1} x_{f_2 r k_2}. \qquad (15) \end{aligned}$$

where $H$ is a Hamiltonian of the ising model for QASP, $\lambda_s$ and $\lambda_u$ are coefficients for penalty terms, respectively.

## 3.3 QUBO matrix

To apply the ising model to the quantum annealer, building the QUBO matrix representing the Hamiltonian Equation 14 is required. Thanks to the quantum computer, the computation time of ASP is significantly reduced. However, composing the QUBO matrix takes enormous time if it is simply implemented with for-loop statement for all sum function in Equation 14. This long time composing impairs the benefit of using the quantum annealer which can solve combinatorial optimizations rapidly. Some implementation techniques are shown in this section for high-level programming language, especially for Python which is commonly utilized in machine learning and quantum annealing studies.

The Hamiltonian is transformed with the QUBO matrix and decision variable vector as the following:

$$H(x) = x^T Q x, \qquad (16)$$

the $Q \in \mathbb{R}^{N \times N}$ is the QUBO matirx, and $N \in \mathbb{Z}$ is the number of decision variables. As a first step, let us consider a QASP with single runway. The the decision variable vector is formulated as:

$$x = \left[ x_{10},\, x_{11},\, \cdots,\, x_{1\bar{k}_1},\, x_{20},\, \cdots,\, x_{fk},\, \cdots,\, x_{N_f \bar{k}_{N_f}} \right] \in \mathbb{B}^N, \qquad (17)$$

where $N_f$ is a number of flights to be assigned. And the indices $r$ to represent assigned runway is omitted. And the vertical axis of QUBO matrix indicates preceding aircraft and horizontal axis indicates following aircraft. Then the $(i,\, j)$ component of matrix $Q(i,\, j) = Q(\sum_{f=1}^{f_1 - 1} N_f + k_1,\, \sum_{f=0}^{f_2 - 1} N_f + k_2)$ means the cost value when the leader aircraft $f_1$ is assigned at the time slot $k_1$ and the following aircraft $f_2$ is assigned at the time slot $k_2$.

According to the first term of Equation 14 doesn't contain any cross term, then this term affects only diagonal component of the matrix.

$$Q(i,\, i) = k\Delta t, \quad \forall f \in F,\, \forall k \in K_f, \qquad (18a)$$

$$i = I(f,\, k) = \sum_{g=1}^{f-1} N_g + k. \qquad (18b)$$

The second term of Equation 14 can be transformed as:

$$\sum_{f \in F} \left( \sum_{r \in R} \sum_{k \in K_f} x_{frk} - 1 \right)^2 = \sum_{f \in F} \left( \sum_{k_1 \in K_f} \sum_{k_2 \in K_f} x_{fk_1} x_{fk_2} \right) - 2 \sum_{f \in F} \sum_{k \in K_f} {x_{jk}}^2 + N \qquad (19)$$

The first term of Equation 19 gives penalties for diagonal blocks excluding diagonal components (the orange area of Figure 1). The second term of Equation 19 is for diagonal component. From Equation 18 and 19, the diagonal components are written as:

$$Q(I(f,\, k),\, I(f,\, k)) = k\Delta t - 2\lambda_u, \quad \forall f \in F,\, \forall k \in K_f. \qquad (20)$$

The first term of Equation 19 represents that non-diagonal components of diagonal block corresponding to the intra-flight variables indicated with red area in Figure 1 are filled by penalty $\lambda_u$.

$$Q(I(f,\,k_1),I(f,\,k_2)) = \lambda_u, \quad \forall f \in F,\, \forall k_1 \in K_f,\, \forall k_2 \in K_f/\{k_1\}\,. \tag{21}$$

The third term of Equation 14 defines the penalty of inter-aircraft separation. Then the penalty is applied to the components of non-diagonal block. The blocked components of the aircraft are decided by the wake turbulence categories of leading and following aircraft, the assigned time of pair aircraft and the available time window of aircraft itself. This block pattern table can be prepared before composing the matrix Figure 1 right. The table should be defined for all assumed time slot from the earliest $s_0$ to the latest $s_{\max}$. Therefore the non-diagonal block is written as:

$$Q(I(f_1,k_1),I(f_2,k_2)) = P(f_1,f_2,k_1,k_2), \quad \forall f_1 \in F,\, \forall f_2 \in F/\{f_1\},\, k_1 \in K_{f_1},\, \forall k_2 \in K_{f_2}. \tag{22}$$

However with the modern programming language, this can be implemented as a matrix range specified copy. Then computation time is expected greatly shorten. Figure 1 shows an example of the matrix composing for 3 aircraft scenario, each aircraft has 4 assignable slots, and any pair of aircraft has 3 slots as the minimum separation. The area where the index indicating the slot matches can be copied from the pattern table on the right to the QUBO matrix, in Figure 1 the area shown with green and light blue rectangle for separation between aircraft 1 and 2. The light blue rectangle indicates blocked slot for following aircraft 2 blocked by the leading aircraft 1, and vise versa for the light green rectangle.

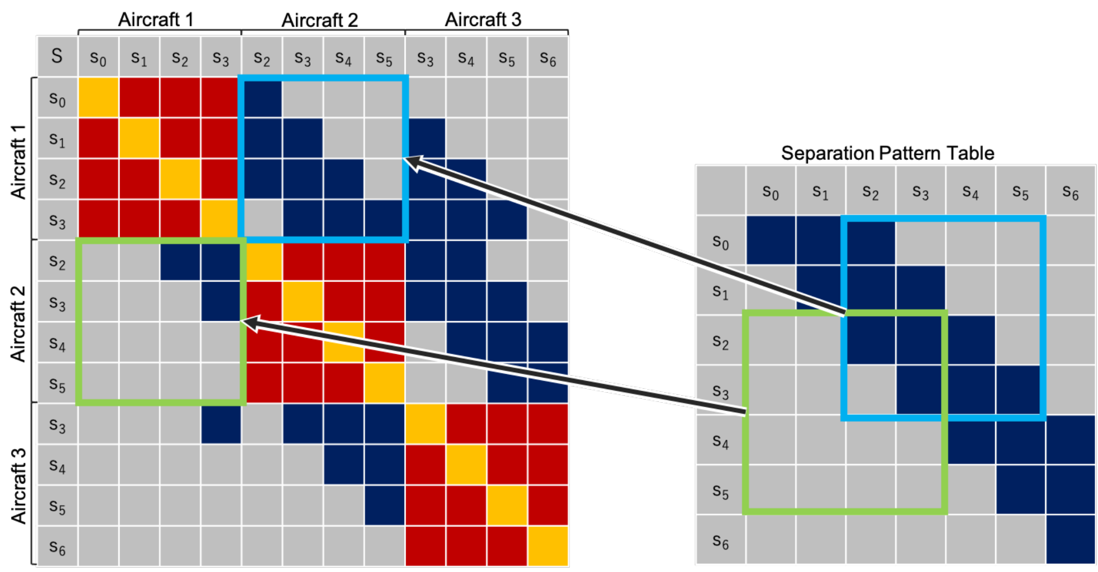


Figure 1 – QUBO matrix composing diagram.

The QUBO matrix described above is for the single runway condition, here it is mentioned as SR-matrix. To adopt the multiple runway airport, a simple extension is needed. The SR-matrix is placed as the diagonal block. This diagonal block indicates that restriction in the same runway. On the other hand, non-diagonal block restricts inter-runway relation-ship. In this paper, only independent runways are considered. Therefore the slot for same aircraft is blocked. This area is a diagonal block of sub block, which size is same to SR-matrix, see Figure 2.

## 4. Numerical Experiment

The aircraft sequencing problem with multiple runway is assumed in this experiments. The matrix building and MIP solving is performed on the Intel core i9 2.3GHz 8 cores cpu, 16BG memory.

### 4.1 QUBO matrix calculation time

The computation time reduction of the QUBO matrix by composing techniques are measured. Figure 3 shows computation times of the QUBO matrix for scenario including various number of aircraft,

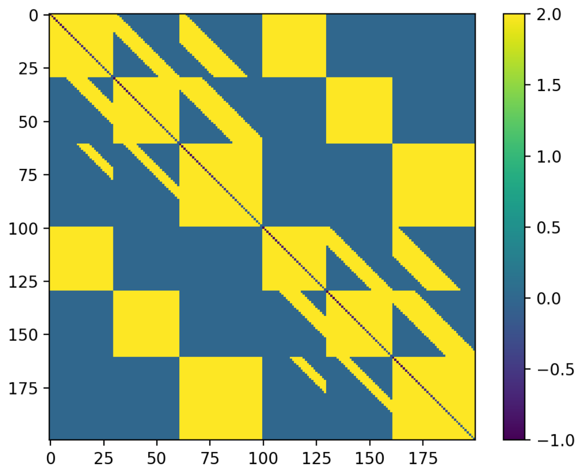


Figure 2 – Example of QUBO matrix for multiple runway with 3 flight.

from 3 to 25 aircraft. Blue line shows the calculation time of ordinal implementations with for-loop statements corresponding to sums in Equation 14, and red line indicates the calculation time for implementations with proposed techniques. For each plots, 5 different random scenario are generated and the QUBO matrices building time is measured.

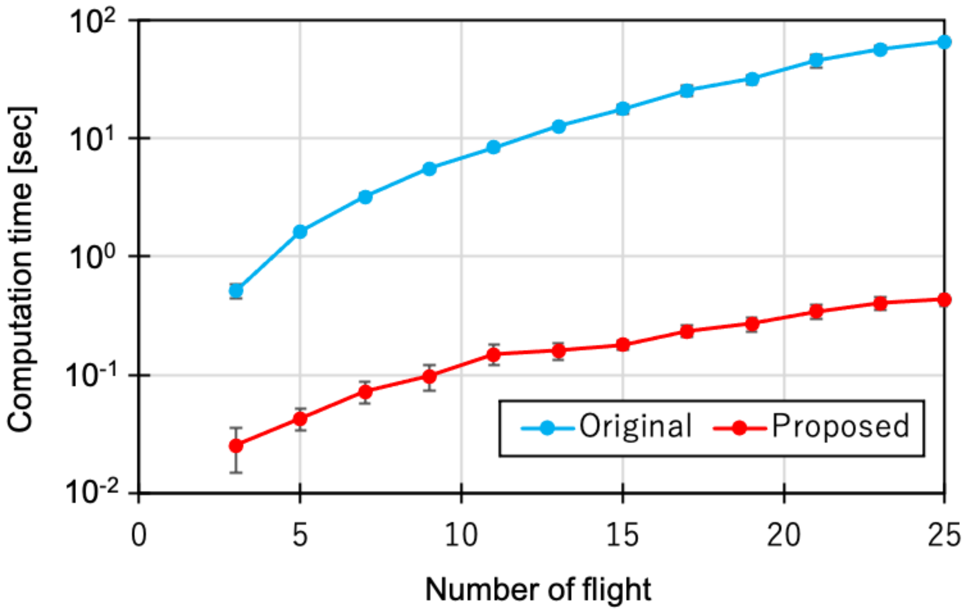


Figure 3 – Affect of the QUBO matrix composing techniques.

## 4.2 Calculation time with quantum annealer

Aircraft sequencing performance is shown in this section. Figure 4 shows the result of calculation time of ASP with MIP and QA. The blue line indicated the result of MIP, and red line is QA. MIP calculation time exponentially increased depend on the number of flight. Each condition is an average result of 5 random scenario. And more than 15 flight, MIP can not finish calculations untill $10^3$ second, therefore optimization is terminated before reach optimal solution. In other hand QA can finish calculation by less than 10 seconds for all conditions in this paper. The calculation time of QA also increased depend on increasing of number of flight. The increase is almost linear. Therefore QASP can be said that it has possibility to treat larger number of flight such like 100.

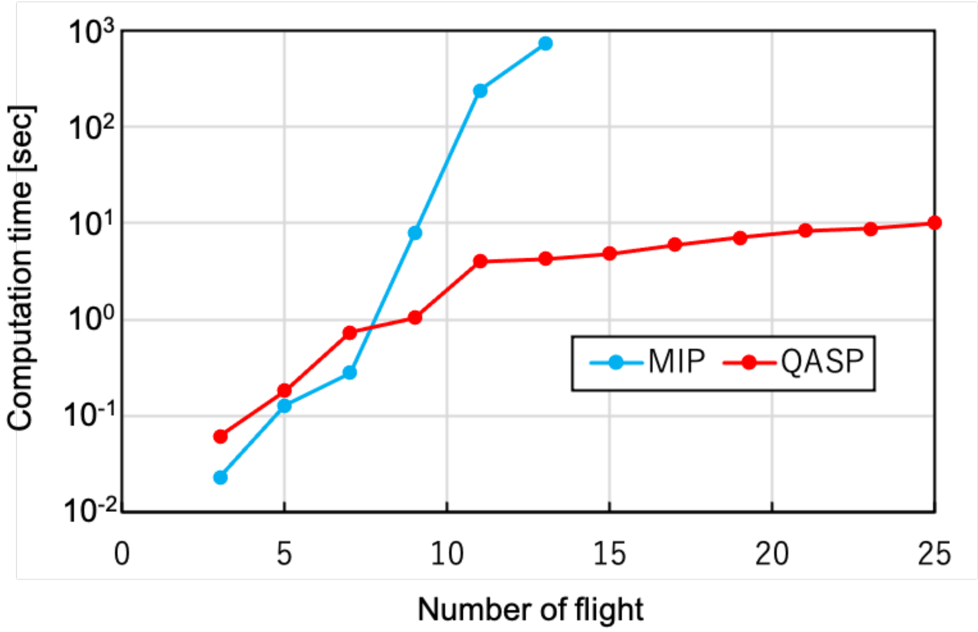


Figure 4 – Affect of the QUBO matrix composing techniques.

Figure 5 shows the total delay as objective function. There is no significant deference between MIP and QA solution. This fact imply that QA solution mostly reach or close to optimal solution.

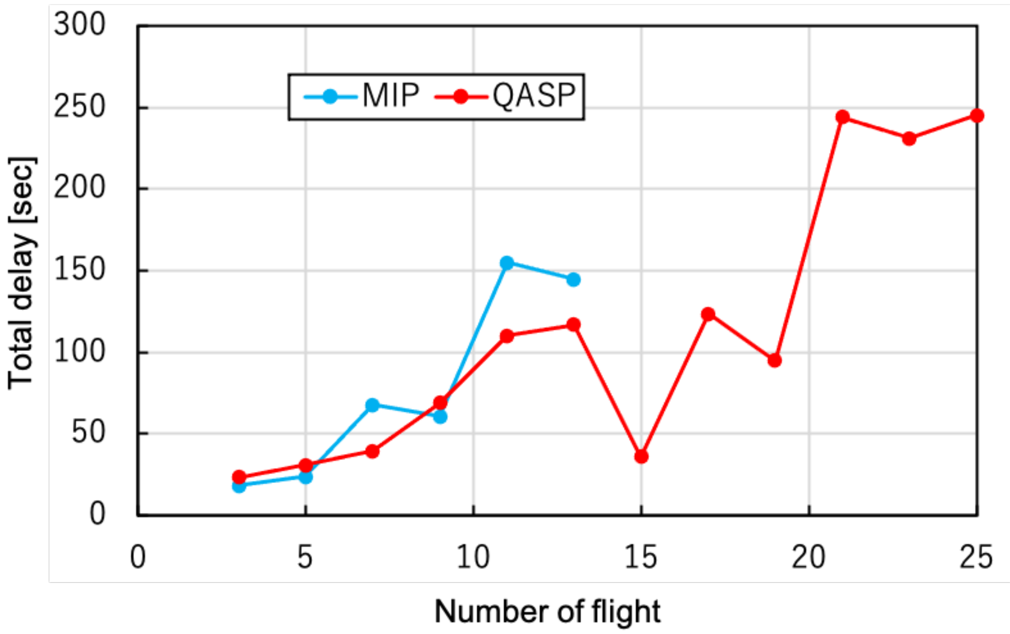


Figure 5 – Affect of the QUBO matrix composing techniques.

## 5. Conclusion

The quantum aircraft sequencing problem (QASP) to determine the assigned time to assign the runway. In this paper, the Hamiltonian as objective function for QA is defined based on MIP ASP problem. Furthermore some implementation techniques to reduce computation time of QUBO matrix is shown. These technique reduce calculation time less than 1/100 time than ordinal implementation. And QASP solve the APS greatly short time that is enough to utilize in the actual airport operation. Solution optimal is also confirmed.

## Copyright Statement



## References

[1] Boeing. *Commercial Market Outlook 2021-2040*. 2021.

[2] Di Mascio P., Cervelli D., Correra A. C., Frasacco, L., Luciano, E., Moretti, L. Effects of departure manager and arrival manager systems on airport capacity. *Journal of Airport Management*, vol. 15 No. 2, pp 204-218, 2021.

[3] Rodríguez-Sanz, Á., Valdes, R. M. M. A., Pérez-Castán, J. A., Cózar, P. L., Comendador, V. F. G. Tactical runway scheduling for demand and delay management. *Aircraft Engineering and Aerospace Technology*, 2021.

[4] Farhadi F., and Ghoniem A. Runway capacity management–an empirical study with application to Doha International Airport. *Transportation Research Part E: Logistics and Transportation Review*, Vol. 68, pp 53-63, 2014.

[5] Masayuki O, Chako T., Shuntaro O., Masayoshi T., Shinichiro T., amd Kazuyuki T. Quantum annealing: next-generation computation and how to implement it when information is missing. *Nonlinear Theory and Its Applications, IEICE*, Vol. 9, No. 4, pp 392-405, 2018.