A Multi-Agent Decision-Making Approach for Automated Inter-Airline Slot Swapping in Ground Delay Program with Airline Privacy Preservation

1st Hoang-Minh Le, 2nd Duc-Thinh Pham, 3rd Vu N. Duong
Air Traffic Management Research Institute
School of Mechanical and Aerospace Engineering
Nanyang Technological University
Singapore
lehoangm001@e.ntu.edu.sg, dtpham@ntu.edu.sg, vu.duong@ntu.edu.sg

***Abstract*—The inter-airline slot-swapping phase in the Ground Delay Program (GDP) allows airlines to collaboratively revise schedules and manage delayed flights. However, due to the airlines' reluctance to disclose sensitive information, most approaches to identify optimal slot swaps between airlines face practical challenges. In addition, current practices for slot-swapping are largely manual, requiring submitted offers from airlines to operate. To achieve privacy-preserving and automated properties, we propose a multi-agent inter-airline slot swapping framework, where each airline has an intelligent agent assisting it to do the swaps. Two types of agents are implemented: the heuristic agent and the reinforcement learning agent. The framework operates in a multi-round setting, allowing an agent to gradually disclose information. In each round, agents submit lists of offers to the pool, and the coordinator subsequently matches the offers. An agent's objective is to minimize the airline's disruption costs: delay costs and cancellation costs. The results show that the total cost of the system is reduced by 9.4 percent after using the framework, and the reinforcement learning agent achieves a 12.8 percent cost reduction on average for an individual airline.**

***Index Terms*—inter-airline slot-swapping, multi-agent system, ground delay program, air traffic flow management**

I. Introduction

The Air Traffic Flow Management (ATFM) system has improved the efficiency of air traffic around the world and has been implemented by major organizations such as the Federal Aviation Administration (FAA), and Eurocontrol. The ATFM operates in three phases: strategic, pre-tactical, and tactical. The strategic phase focuses on long-term planning, with strategic slot allocation being one of its key activities. Typically 6 months before the operational day, airlines submit their flight plans, and the system assigns slots accordingly. The pre-tactical phase typically takes place one day before operations, when more data are available such as weather conditions and airline disruptions information. The pre-tactical ATFM is implemented to ensure that traffic goes as planned. The tactical ATFM phase is executed on the day of operations to manage the real-time traffic. Since the goal of ATFM is to ensure the safe and efficient flow of traffic, and with the high demand predicted in 2050, innovative technologies must be introduced to tackle the evolving, complex traffic.

Ground Delay Program (GDP) is an effective pre-tactical and tactical decision support tool for managing demand-capacity imbalances. Since 1998, GDP has been implemented under the Collaborative Decision Making (CDM) paradigm [1]. This paradigm includes three main steps: ration-by-schedule, substitutions and cancellations, and compression. The core idea of ration-by-schedule is to distribute delays based on the original schedule arrival time, which is widely accepted as a fairness standard by the industry [2]. In terms of system efficiency, multiple optimization-based approaches are developed to minimize the total delays [3]–[5]. The second step focuses more on the airline side, allowing airlines to exchange flights in their owned slots or to cancel low-value, excessively delayed flights to reduce the delay costs. Some studies have been conducted on developing mathematical models to minimize airlines' costs under deterministic setting [6], stochastic setting [7], [8], or a reinforcement learning model for aircraft swapping [9]. After airlines take recovery actions and report their vacant slots, the system runs the compression algorithm to move flights upward in the arrival sequence to avoid under-capacity. Indeed, the compression algorithm is a simple mechanism for airlines to exchange slots [10]. Although the report of vacant slots is needed for the effective run of the compression algorithm, in fact, airlines are usually hesitant to share this information. Besides, an airline can submit a conditional request to cancel a flight in exchange for moving another flight to the earlier slot through the Slot Credit Substitution (SCS) process. This procedure in practice is manually done, where airlines submit requests, which are later handled by the Air Traffic Control System Command Center (ATCSCC).

The inter-airline slot swapping procedure has been shown to effectively enhance the system performance both in theoretical research and practical applications, for example, the significant delay reduction gained by the use of compression algorithm [11]. In addition, the optimization model proposed to match submitted offers has resulted in significant improvements in passenger delay costs and on-time performance [10]. Further research works extended the model, capable of handling at-most, at-least (AMAL) trades, applying it to the network of traffic [12] and the Airspace Flow Program [13]. While these methods show improvement in the system, these studies only focus on the offer-matching model rather than how airlines generate the trades.

To provide a flexible model, that can be used both by airlines and the central system, the generic mathematical model has been developed [2]. Airlines can use this model to find the desired schedules, and then manually generate the slot-swapping offers to submit. From the system perspective, given all required inputs, the method could find the optimal solution, for example, as an objective to minimize public delay and private delay costs. However, such a system faces difficulty in application since airlines are hesitant to reveal their operational costs due to market competitiveness. For an inter-airline slot-swapping system, the ability to swap slots without revealing airline private information is required, a concept known as privacy-preserving [14]. Delay Ledger was proposed to allow that feature, allowing airlines to submit a coarse version of their flight values and let airlines take turns to assign delays to all airlines. The framework also ensures airlines' delay costs are always reduced, typically mentioned as the *individual rationality* property, which encourages airlines to use the system. Privacy-preserving systems are also proposed in the market setting [15], and the auction mechanism [16]. Nevertheless, these systems do not address how airlines should generate their bids.

Our motivation is to create an inter-airline slot-swapping system with these properties: fully automated, privacy-preserving, individual-rational, and efficient. To achieve this, we propose a multi-agent inter-airline slot-swapping framework, where each airline has an intelligent agent helping it to swap slots. The intelligent agent observes the airline's private operational costs and offers of other agents. Then the agent automatically submits offers to the pool, which are eventually matched by the coordinator. To minimize the disclosure of private information, the mechanism is designed in a multi-round setting, where agents gradually release offers over time. Our main contributions are:

- Propose a multi-agent inter-airline slot-swapping mechanism.
- Train an agent to make good offers using the reinforcement learning algorithm.

II. INTER-AIRLINE SLOT SWAPPING IN COLLABORATIVE DECISION MAKING

Ground Delay Program (GDP) currently operates under the Collaborative Decision Making (CDM) paradigm. After the prediction of airport capacity shortage is carried out, two following steps are performed: ration-by-schedule, flight substitutions and cancellations. The first step ensures that capacity limits are not exceeded, while the second step allows airlines to take recovery actions to mitigate **disruption costs**, including delay costs and cancellation costs. Once two steps are completed, the inter-airline slot-swapping procedure is implemented to further reduce costs.

A. Ration-by-schedule

Based on the predicted airport arrival rate (PAAR), the arrival slots are stretched out. The algorithm assigns flights to new arrival slots based on the order of their original schedules. To illustrate the concept, Fig. 1 shows that the number of flights that arrive in one 15-minute time slot reduces from two to one flight. In the next step, airlines can take action to mitigate costs caused by the disruption.

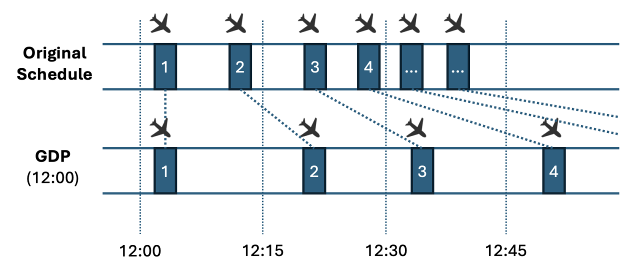

Fig. 1. An Example of Ration-By-Schedule Step in GDP

B. Flight Substitutions and Cancellations

In this step, an airline can select two flights to swap slots. Typically, high-value flights are prioritized to be in the best slots, even by delaying or canceling low-value flights. The airline can use an automated tool to optimize multiple objectives: delay cost, on-time performance, or total passenger delays. In this paper, we assume airlines are willing to minimize the *disruption costs*. The objective function is the combination of objective functions proposed by Vossen (2006) and Balakrishnan (2022) [10], [14].

$$Min \sum_{i}^{f} \sum_{j}^{s} x_{i,j} \cdot (T_{slot_j} - T_{flight_i}) \cdot C_{delay_i} + \sum y_i \cdot C_{cancel_i} \quad (1)$$

subject to

$$\sum_{i}^{f} x_{i,j} + y_i = 1 \quad (2)$$

$$\sum_{j}^{s} x_{i,j} <= 1 \quad (3)$$

$$x_{i,j}, y_i >= 0 \quad (4)$$

In formulation (1), $x_{i,j}$ and y_i are binary decision variables. $x_{i,j}$ takes value 1 if flight f_i is assigned to slot s_j; and y_i takes value 1 if flight f_i is canceled. The arrival time for slot s_j and the scheduled arrival time of flight f_i are denoted as T_{slot_j} and T_{flight_i} respectively. Both T_{slot_j} and T_{flight_i} are measured in 15-minute time units, meaning that delay costs are incurred per time slot rather than per minute. The constraint (2) ensures each flight is assigned to at most one slot, while constraint (3) guarantees that one slot is allocated to at most one flight.

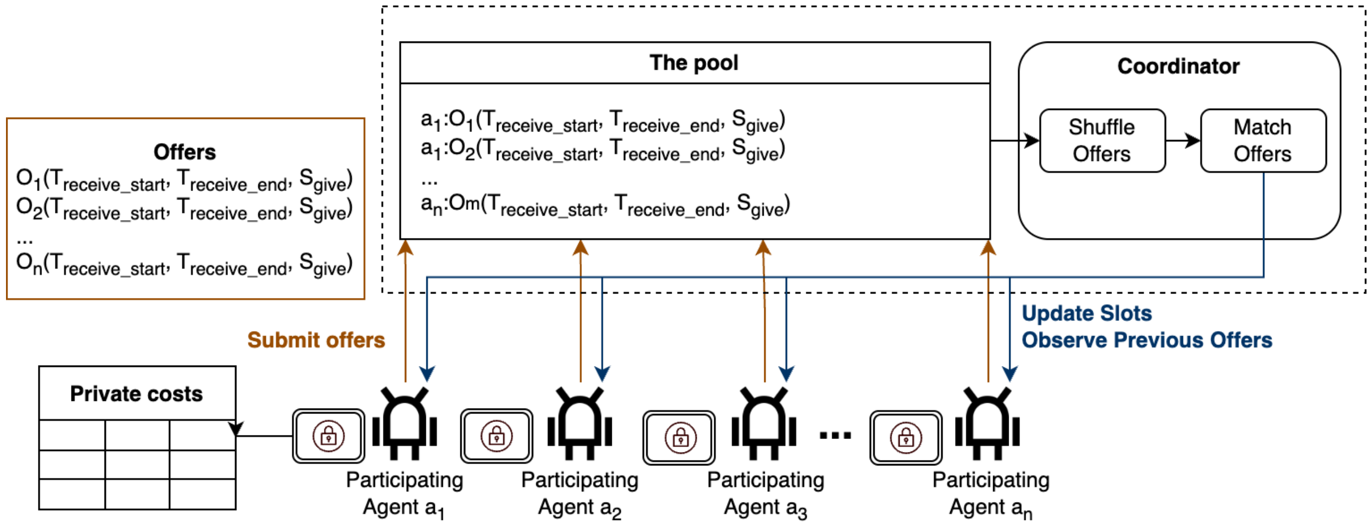

Fig. 2. The Multi-Agent Inter-Airline Slot-Swapping Framework

C. Multi-Agent Inter-Airline Slot Swapping

Following the local optimized schedules provided in the flight substitutions and cancellations step, the inter-airline slot-swapping process is initiated. The practical approach involves the Slot Credit Substitution (SCS) process, where an airline submits an offer in the form of "Willing to cancel flight f_i to reduce the delay of another flight f_j." We formulate this request as an offer O:

$$O = (T_{receive\_start}, T_{receive\_end}, S_{give})$$

where an airline is willing to relinquish slot S_{give} to receive a slot within a time window [$T_{receive\_start}$,$T_{receive\_end}$]. The received slot is assigned to the high-value flight, while the relinquished slot is from either a vacant slot or a low-value flight slot.

Fig. 2 illustrates the framework in a multi-round setting. Each round, multiple agents submit their lists of offers to the centralized system, referred to as *the pool*. Each agent represents an airline and has access to the airline's private delay and cancellation costs. Following the submission of all offers, *the coordinator* performs the match. Two offers O_i and O_j can be matched if:

$$T^{O_i}_{\text{give}} \in \left[T^{O_j}_{\text{receive\_start}}, T^{O_j}_{\text{receive\_end}}\right]$$

and

$$T^{O_j}_{\text{give}} \in \left[T^{O_i}_{\text{receive\_start}}, T^{O_i}_{\text{receive\_end}}\right]$$

where $T^{O_i}_{give}$ and $T^{O_j}_{give}$ are the time of slot $S^{O_i}_{give}$ and $S^{O_j}_{give}$ accordingly. After one round, the agents observe offers made by other agents in the previous round and proceed to make the next offers.

III. METHODOLOGY

Our proposed method includes two main components: the coordinator and the participating agents. The coordinator shares similarities with other proposed works to match offers such as optimization models from Vossen 2006, Bertsimas 2016, and Erkan 2019. The participating agents generate offers automatically based on the observation of private disruption costs and public offers.

A. The coordinator

An objective of the coordinator is to match airlines' offers efficiently and within a timely manner, given the process is run in the pre-tactical phase. In addition, as the system runs in multi-round, the participating agents require sufficient time to proceed with each iteration. In our proposed method, we employ a simple top-to-bottom slot-matching approach. The algorithm iterates all available offers using two nested loops, with the order of offers randomized beforehand. As the focus of this paper is not on the coordinator algorithm, we plan to keep this simple. Other matching algorithms can be substituted to provide better matching.

B. The participating agent

In each round, an agent needs to generate a list of offers to minimize disruption costs. The objective of generating an offer is to move a flight, especially a high-value one to the earliest possible slot, even by releasing a low-value slot. As depicted in Fig. 3, for a specific already rescheduled flight, the best possible slot is the slot at its scheduled time of arrival (STA), denoted as t_{ori}. The maximum arrival slot, denoted t_{max}, is assumed to be 20 time slots after the STA. This assumption is based on the fact that flights are typically canceled due to long delay hours. For the delayed flights, the current time slot is denoted as t_{res}. If the flight is canceled, the t_{res} is not applicable. If this flight can move to a new earlier slot t_{new} (Fig. 4, Fig. 5), the cost reduction by obtaining this slot is:

$$C_{\text{reduced}} = \begin{cases} (t_{\text{res}} - t_{\text{new}}) \cdot C_d & \text{if a flight is delayed} \\ (t_{\text{new}} - t_{\text{ori}}) \cdot C_d - C_{\text{c}} & \text{if a flight is canceled} \end{cases}$$

where C_d is the delay costs per time slot, and C_c is the cancellation costs of that flight. Then, the agent needs to choose a slot to relinquish. The incurred cost of the relinquished slot will be:

$$C_{\text{incurred}} = \begin{cases} 0 & \text{if a slot is vacant} \\ C_c^r - (t_{\text{res}} - t_{\text{ori}}) \cdot C_d^r & \text{if a slot is occupied} \end{cases}$$

where C_c^r is the cancellation costs and C_d^r is the delay costs per time slot of the flight in that relinquished slot. If the slot is vacant, the $C_{incurred}$ cost is zero.

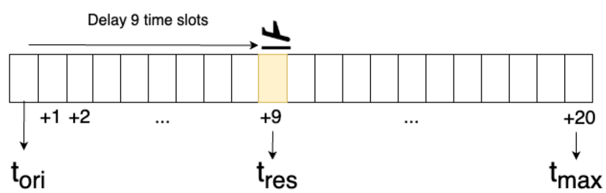

Fig. 3. Illustration of a Rescheduled Flight Arrival Slot Relative to Its Scheduled Arrival and Maximum Arrival Slot

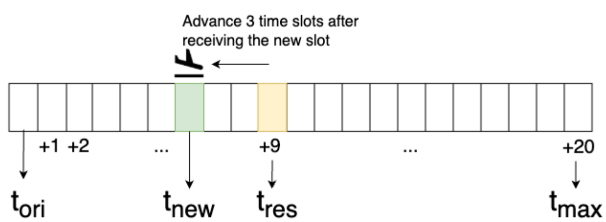

Fig. 4. Illustration of How a Rescheduled Flight Shifts to an Earlier Arrival Slot

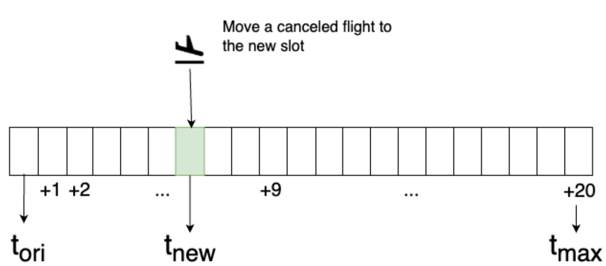

Fig. 5. Illustration of How a Canceled Flight Moves to an Arrival Slot

To generate an offer as a tuple O, $T_{receive\_start} = t_{ori}$ because the earlier slot, the better the cost reduction. The task of an agent is to pick $T_{receive\_end}$ so that it receives the highest reduction costs while ensuring the matching chance. Lower $T_{receive\_end}$ leads to lower $C_{reduced}$, but the low range creates difficulty in matching. After picking the range of the received slot, the agent needs to pick a slot to relinquish. In this case, the agent also needs to balance between relinquishing the low-value slots and high-demand slots. An offer has a potential total reduction cost, C_{total} as follow:

$$C_{total} = C_{reduced} - C_{incurred}$$

if C_{total} is positive, this offer is called *valid offer*. If C_{total} is negative, this offer is called *invalid offer*. Since the system is individual-rational, an agent only submits valid offers to the *pool*. Note that although it is attracted to submit an offer with high C_{total}, the cost reduction only happens if the offer is accepted, referred to as a *smart offer*. The objective is to develop an agent that is capable of making smart offers while avoiding invalid offers.

C. The participating heuristic agent

For the participating heuristic agent, for each offer, the agent will choose the middle option, defined as:

$$T_{received\_end} = T_{received\_start} + 10$$

The middle point is selected to balance the cost reduction and the likelihood of an offer being successfully matched. For the relinquished slot, the agent chooses the slot with the highest demand. This approach shows better performance compared to the lowest-value slot relinquished approach.

D. The participating reinforcement learning agent

For the reinforcement learning algorithm, we train an agent to learn to pick an effective $T_{receive\_end}$ slot. The agent observation is structured as a 3 x 40 grid, denoted as s, which captures available slots in the pool and the agent's owned slot can be relinquished (Fig. 6). The first 3 x 20 grid represents the current slots in the pool, with the first row representing the *cost reduced* if the agent obtains that slot. The second row represents the number of offers relinquishing these slots, and the third row represents the number of offers requesting these slots. The latter 3 x 20 grid represents the slots owned by the agent. The first row is the *cost incurred* if the agent gives up that slot. The second and the third rows are similar to the previous grid. The action space consists of 21 actions, with the first 20 actions representing the $T_{receive\_end}$ from $t_{ori} + 1$ to t_{max}, the final action represents the decision not to submit an offer. An action is denoted as a. The reward function is designed to encourage agents to make *valid offers* and *smart offers* while penalizing *invalid offers*. The reward is shown as follows:

$$R(s, a) = \begin{cases} -5, & \text{if an invalid offer is made} \\ 0, & \text{if no offer is submitted,} \\ 1, & \text{if a valid offer is made,} \\ 30, & \text{if the offer is successfully matched.} \end{cases}$$

To easily analyze the agent behavior across airlines with different sizes, the reward is normalized by dividing by the total number of flights operated by the airline. As a result, the average reward per round is defined as follows: -5 if an agent makes all invalid offers, 0 if no offers are made, 1 if

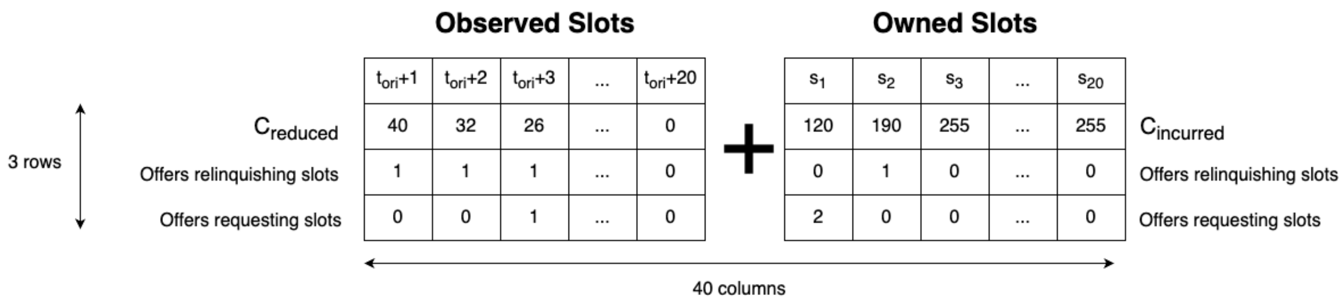

Fig. 6. The Observation Grid of a RL-Agent by combining Observed Slots and Owned Slots information

all submitted offers are valid, and 30 if all submitted offers are matched. We use the Deep-Q Learning algorithm to train the model due to its ability to handle high-dimensional state spaces.

IV. EXPERIMENTS

For the data and experiment setup, we first simulate the ration-by-schedule and Intra-Airline slot swapping procedures to generate the input for the inter-airline slot swapping system. We selected the dataset from the Bureau of Transportation Statistics' (BTS) Marketing Carrier On-Time Performance [17] for July 20th, 2024, which includes the origin, destination, airline, scheduled and actual departure and arrival times. We took only flights arrived at Hartsfield-Jackson Atlanta International (ATL) airport to do the experiment. We selected ATL airport because it is one of the biggest airports in the United States. The chosen time period is in the big disruption event with more than 400 flights being canceled, where the need for slot-swapping takes place to deal with disruption. To simplify the experiment, we selected the 9 smallest airlines to train and test the system.

To simulate the post-disruption landing slots using ration-by-schedule, we map them directly with the actual landing slots in the data. We simplify the arrival time into 15-minute bins. We assume no priority flights, and the order of flights is maintained. After this procedure, we get a new Calculated In-Block Time (CIBT) for the flight plans. This is when the airlines need to substitute, cancel, or swap flights to minimize the costs.

To demonstrate the reduction of delay costs, each flight has to be assigned a flight value. However, this is private information so we do not have the value. To evaluate the model, we assign delay costs randomly for a flight with a uniform distribution from 1 to 10. A similar method has been conducted by Balakrishnan (2022). Note that we assign random costs as a real cost for flights only once and then use it for all the algorithms to compare. Another assumption is that the airline will cancel any flight that is delayed more than 20 time slots (300 minutes). We assume the cancellation cost is equal to the delay costs of 21 time slots, which makes sense for the previous assumption.

A. *Intra-airline slot swapping*

Firstly, airlines will run the Intra-Airline swapping algorithm to optimize their disruption costs. Using the objective function from equation (1), we use linear programming to solve the optimization problem from the PuLP library.

B. *Inter-airline slot swapping*

To evaluate the capability of agents to learn and make offers, we design scenarios in which one agent operates using the Reinforcement Learning (RL) method, whereas the remaining agents employ the heuristic method. We run the system in 10 rounds, as Heuristic-Based agents cease no further improvement after round 7. For the RL agent, we use the Stable-Baselines library and train it using MLPPolicy. A discount factor of 0.5 is applied. We evaluate the model every 8000 time-steps, with the best model saved. Note that since offer-matching by the coordinator is inherently randomized in practice, we introduce a control evaluation setting where the analyzing agent is assigned the lowest priority. This ensures that the RL agent is assessed under the worst-case scenario, with real-world performance expected to be equal or superior.

V. RESULTS

In terms of model learning, Fig. 7 shows the convergence after 300000 time steps. Initially, the average total reward is approximately -30, as the agent explores possible actions and randomly generates sup-optimal offers, resulting in negative rewards. Gradually, the agent learns to make valid offers, which will yield a reduction if the matches are made. By 50000 time steps, the average reward is 0, equivalent to an agent that makes no offers. Beyond this point, the agent starts to generate smart offers, convergent to an average reward of 20 (equivalent to 2 per round). This shows that the agent successfully learns not only to avoid invalid offers but also to generate valid and smart offers.

We evaluate the performance of the multi-agent slot-swapping system by computing the disruption costs for the Inter-Airline Only, Heuristic, and Reinforcement Learning (RL) cases. Table I shows that, on average, the Heuristic-Based Multi-agent Slot-swapping system reduces the total costs by 8.7 percent, while the Reinforcement Learning approach achieves a 9.4 percent reduction compared to the Intra-Airline Only case. Overall, the multi-agent slot swapping system

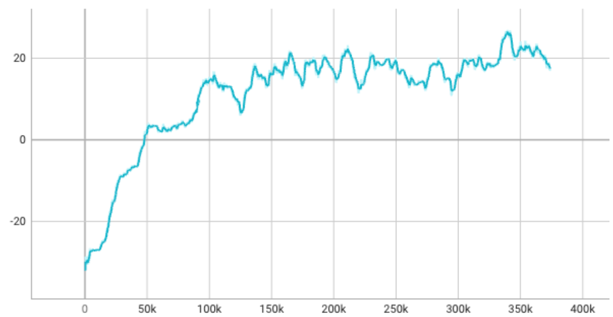

Fig. 7. Convergence Curve Over 300000 Time Steps

significantly reduced the disruption costs compared to the Intra-Airline case, with the Reinforcement Learning approach showing slight improvement over the Heuristic method.

TABLE I
THE TOTAL SYSTEM COST COMPARISON BETWEEN INTRA-AIRLINE ONLY, 9 HEURISTIC-BASED AGENTS, AND 1 RL-BASED AGENT AND 8 HEURISTIC-BASED AGENTS

Scenarios	Total Cost (Intra-Airline)	Total Cost (9 Heuristic-based Agents)	Total Cost (1 RL-based Agent, 8 Heuristic-based Agents)
1	8873	8049	8091
2	8873	8262	8138
3	8873	8061	7782
4	8873	8262	8147
5	8873	8049	7917
6	8873	8049	7999
7	8873	8049	8092
8	8873	8049	8240
9	8873	8049	7956
Average	8873	8097	8040

However, when analyzing the impact on individual airlines, as shown in Table II, on average, the Heuristic algorithm shows an 8.2 percent reduction in disruption costs compared to the Intra-Airline Only case, while the Reinforcement Learning approach achieves a 12.8 percent reduction, showing additional over 4 percentage improvement. In addition, in all scenarios, the RL-based agent performs at least equal, if not better than the Heuristic-based algorithm. In the case of the equal results of both methods, it is typically because no beneficial offers can be generated to further enhance performance. In summary, for a single airline, the RL agent outperforms the Heuristic approach, showing greater cost reduction without compromising the total system costs.

VI. CONCLUSION AND FUTURE WORK

In this study, we proposed a multi-agent inter-airline slot swapping system with the following properties: fully automated, privacy-preserving, and individual-rational. In addition, we train a single agent to generate smart offers based on airline private costs and offers of other agents. The result suggests that the total disruption costs have reduced significantly after using the system, and the reinforcement learning agents save more costs for an airline. This study provides a proof-of-concept for a multi-agent system in slot swapping, allowing better operational efficiency, economic benefits, and scalability.

Although the result is promising, the experiment setup is limited and the model is still in the primary state. Firstly, the models are simplified with a smaller number of airlines. With more airlines, more swaps can be done and the system can be improved more. Secondly, this model is trained by letting agents interact multiple times with one-day data only. The expansion of data can help the agent learn a better generic policy, avoiding being overfitted. Lastly, this system only focuses on how participating agents learn to make good offers, but does not analyze the effect of the coordinating agents. An effective matching coordinator has the potential to affect the strategies of the participating agents.

For future work, the model can be expanded to the multi-airport system, where the airlines can swap slots with other airlines in different airports. In addition, more reinforcement learning agents could be integrated for better optimization. Lastly, the system can include the at-most, at-least (AMAL) offers, which have been demonstrated with higher performance.

ACKNOWLEDGMENT

This work is carried out at the Air Traffic Management Research Institute (ATMRI), Nanyang Technological University (NTU), Singapore. The author acknowledges the NTU Research Scholarship provided by Nanyang Technological University (NTU).

REFERENCES

[1] Cox, Jonathan, and Mykel J. Kochenderfer. "Ground Delay Program Planning Using Markov Decision Processes." Journal of Aerospace Information Systems 13, no. 3 (March 2016): 134–42. https://doi.org/10.2514/1.I010387.
[2] Erkan, Hale, Nesim K. Erkip, and Özge Şafak. "Collaborative Decision Making for Air Traffic Management: A Generic Mathematical Program for the Rescheduling Problem." Computers & Industrial Engineering 137 (November 2019): 106016. https://doi.org/10.1016/j.cie.2019.106016.
[3] Richetta, Octavio, and Amedeo R. Odoni. "Solving Optimally the Static Ground-Holding Policy Problem in Air Traffic Control." Transportation Science 27, no. 3 (August 1993): 228–38. https://doi.org/10.1287/trsc.27.3.228.
[4] Vranas, Peter B., Dimitris J. Bertsimas, and Amedeo R. Odoni. "The Multi-Airport Ground-Holding Problem in Air Traffic Control." Operations Research 42, no. 2 (April 1994): 249–61. https://doi.org/10.1287/opre.42.2.249.
[5] Bertsimas, Dimitris, Guglielmo Lulli, and Amedeo Odoni. "An Integer Optimization Approach to Large-Scale Air Traffic Flow Management." Operations Research 59, no. 1 (February 2011): 211–27. https://doi.org/10.1287/opre.1100.0899.
[6] Luo, Songjun, and Gang Yu. "On the Airline Schedule Perturbation Problem Caused by the Ground Delay Program." Transportation Science 31, no. 4 (November 1997): 298–311. https://doi.org/10.1287/trsc.31.4.298.
[7] Woo, Young-Bin, and Ilkyeong Moon. "Scenario-Based Stochastic Programming for an Airline-Driven Flight Rescheduling Problem under Ground Delay Programs." Transportation Research Part E: Logistics and Transportation Review 150 (June 2021): 102360. https://doi.org/10.1016/j.tre.2021.102360.

TABLE II
THE INDIVIDUAL AIRLINE COST COMPARISON BETWEEN INTRA-AIRLINE ONLY, HEURISTIC-BASED, AND RL-BASED APPROACHES

Scenarios	Chosen Airline	Intra-Airline Only Cost	Heuristic-based Cost	RL-based Cost	Heuristic-based Reduction (Compared to Intra-Airline)	RL-based Reduction (Compared to Intra-Airline)
1	YV	488	456	432	-6.6%	-11.5%
2	XY	318	318	231	0.0%	-27.4%
3	OH	680	504	404	-25.9%	-40.6%
4	B6	623	597	564	-4.2%	-9.5%
5	AS	428	288	279	-32.7%	-34.8%
6	OO	1109	976	890	-12.0%	-19.7%
7	UA	992	811	798	-18.2%	-19.6%
8	AA	1886	1876	1876	-0.5%	-0.5%
9	F9	2349	2314	2263	-1.5%	-3.7%
Average		985	904	859	-8.2%	-12.8%

[8] Taylor, Christine, Shin-Lai Tien, Erik Vargo, and Craig Wanke. "Strategic Flight Cancellation Under Ground Delay Program Uncertainty." ATM2019 29, no. 1 (January 2021): 5–15. https://doi.org/10.2514/1.D0178.

[9] Hondet, Gabriel, Luis Delgado, and Gérald Gurtner. "Airline Disruption Management with Aircraft Swapping and Reinforcement Learning." Sesar Innovation Day, 2018.

[10] Vossen, Thomas W. M., and Michael O. Ball. "Slot Trading Opportunities in Collaborative Ground Delay Programs." Transportation Science 40, no. 1 (February 2006): 29–43. https://doi.org/10.1287/trsc.1050.0121.

[11] Chang, Kan, Ken Howard, Rick Oiesen, Lara Shisler, Midori Tanino, and Michael C. Wambsganss. "Enhancements to the FAA Ground-Delay Program Under Collaborative Decision Making." Interfaces 31, no. 1 (February 2001): 57–76. https://doi.org/10.1287/inte.31.1.57.9689.

[12] Bertsimas, Dimitris, and Shubham Gupta. "Fairness and Collaboration in Network Air Traffic Flow Management: An Optimization Approach." Transportation Science 50, no. 1 (February 2016): 57–76. https://doi.org/10.1287/trsc.2014.0567.

[13] Sherali, Hanif D., Justin M. Hill, Michael V. McCrea, and Antonio A. Trani. "Integrating Slot Exchange, Safety, Capacity, and Equity Mechanisms Within an Airspace Flow Program." Transportation Science 45, no. 2 (May 2011): 271–84. https://doi.org/10.1287/trsc.1100.0356.

[14] Balakrishnan, Hamsa. "Airline Disruption Management with Delay Ledgers." (2022).

[15] Schuetz, Christoph G., Eduard Gringinger, Nadine Pilon, and Thomas Lorunser. "A Privacy-Preserving Marketplace for Air Traffic Flow Management Slot Configuration." In 2021 IEEE/AIAA 40th Digital Avionics Systems Conference (DASC), 1–9. San Antonio, TX, USA: IEEE, 2021. https://doi.org/10.1109/DASC52595.2021.9594401.

[16] Feichtenschlager, Paul, Kevin Schuetz, Samuel Jaburek, Christoph G. Schuetz, and Eduard Gringinger. "Privacy-Preserving Implementation of an Auction Mechanism for ATFM Slot Swapping." In 2023 Integrated Communication, Navigation and Surveillance Conference (ICNS), 1–12. Herndon, VA, USA: IEEE, 2023. https://doi.org/10.1109/ICNS58246.2023.10124262.

[17] Bureau of Transportation Statistics, "Airline On-Time Statistics and Delay Causes," 2024.