https://doi.org/10.1016/j.tre.2018.12.002

Data analytics in managing aircraft routing and maintenance staffing with price competition by a Stackelberg-Nash game model

Abdelrahman E.E. Eltoukhy[a], Z.X. Wang[b,*], Felix T.S. Chan[a] and X. Fu[c]

[a]*Department of Industrial and Systems Engineering, The Hong Kong Polytechnic University, Hung Hum, Hong Kong*
[b]*School of Business Administration, Institute of Supply Chain Analytics, Dongbei University of Finance and Economics, Dalian, China*
[c]*Institute of Transport and Logistics Studies, University of Sydney, Australia*

*Corresponding Author

elsayed.abdelrahman@connect.polyu.hk
wangzhengxu@dufe.edu.cn
f.chan@polyu.edu.hk
xiaowen.fu@sydney.edu.au

Abstract

This study develops a Stackelberg-Nash game model (SNGM) to capture the interdependence between aircraft routing of airlines and maintenance staffing of maintenance providers, and to consider the price competition among maintenance providers. The SNGM's overall Nash equilibrium is obtained using an iterative game algorithm. The SNGM effectiveness is demonstrated with a case study, in which a neural network-based algorithm is developed to forecast accurate non-propagated delays, and a multiple linear regression algorithm is adopted to predict demand-price relationship for each maintenance provider. The results reveal cost savings of about 26% and 22% for the airline and the maintenance providers, respectively.

Keywords: Data analytics, Aircraft routing problem, Maintenance staffing problem, Game theory

1. Introduction

In the early twentieth century, the Wright brothers successfully achieved the first powered flight with their newly designed aircraft. In addition to being a technological triumph, this event marked the birth of the aviation industry that shaped the world during the twentieth century. Indeed, the aviation industry is one of the pillars of the global economy, as evidenced by the sustained growth in passenger volumes. Statistics released by the International Air Transport Association (IATA) indicated that the number of air passengers reached 3.5 billion in 2015, and this figure is expected to maintain an annual growth rate of 5%. The significant number of passengers reflects the economic significance of the aviation industry, which contributes around US$2.4 trillion annually to the worldwide economy, representing about 3.5% of the worldwide GDP[1]. To cope with the expected growth in traffic, the worldwide fleet is expected to increase from 24,597 aircraft in 2014 to 29,955 in 2022. Consequently, in 2014, airlines paid around US$62.1 billion to maintenance providers as maintenance cost, which is expected to increase to US$90 billion by 2024[2]. Despite the promising industry prospects, airlines and maintenance providers face great challenges in managing the increasing numbers of aircraft. In this regard, the aircraft maintenance routing problem (AMRP) is of critical importance to airlines because it builds the routes for aircraft and schedules the aircraft

[1] http://www.iata.org/about/Documents/iata-annual-review-2015.pdf
[2] https://www.iata.org/whatwedo/workgroups/Documents/ACC-2015-GVA/1630-1650-mtc-cost-trends.pdf

maintenance visits. Similarly, the maintenance staffing problem (MSP) is recognized as a key issue in managing the workforce capacity of the maintenance providers that service the aircraft.

The AMRP has been extensively discussed in the literature. Aircraft routing problems are classified into three main types: tactical (TARP), operational (OARP), and flight delay-based operational (FDARP). First, the TARP specifies the generic arrangement of flight legs or the cyclic schedule to be repeated by each aircraft, while ignoring some of the operational maintenance restrictions [1, 2]. Thus, the application of TARP rotations may not be viable due to ignorance of some of the operational maintenance restrictions. Therefore, the OARP has been developed to explicitly consider various operational maintenance restrictions [3, 4]. Nonetheless, it is questionable whether the OARP can effectively generate routes, because the AMRP models often overlook flight delays that frequently occur in aviation markets. As a result, the generated routes can be sensitive to disruptions. Thus, the FDARP combines flight delays with operational considerations to generate routes that better withstand disruptions [5]. The flight delays considered by the FDARP can be categorized as a propagated delay (PD) or a non-propagated delay (NPD). NPDs are often ascribed to bad weather, maintenance station congestion, technical problems, peak seasons, and passenger issues, which are generalized as non-routing reasons. A PD is described as a flight delay that is due to a delay of a previous flight performed by the same aircraft. Generally, the role of the FDARP is to minimize the PD after forecasting the NPD. In the literature, the NPD is forecasted using the expected value approach, which only focuses on analyzing the historical flight delay data.

In practice, airlines use the FDARP to build a routing plan. This plan involves the determination of the routes to be flown by each aircraft, while taking into account the flight delays and operational maintenance restrictions. It should be noted that the operational maintenance restrictions are respected by including some scheduled maintenance visits for the aircraft. To implement the determined routes as planned, the airlines need to send the times of the scheduled maintenance visits outlined in the routing plan to the maintenance providers. Referring to the scheduled maintenance times, the maintenance providers use the MSP to determine the team sizes required to maintain the aircraft, which is known as the staffing plan.

Because the FDARP of airlines and the MSP of maintenance providers are interrelated, the airlines and maintenance providers need to work together to implement the routing and staffing plans. First, to implement the routing plan as determined, the airlines are responsible for performing the flight legs, whereas the maintenance providers are responsible for maintaining the aircraft and releasing them from the maintenance station on time. If the maintenance providers fail to release the aircraft on time, the aircraft will not be able to cover the next flight, resulting in a delay for this flight. Second, to fulfil the staffing plan as determined the maintenance providers need to supply the appropriate teams and the airlines need to deliver the aircraft on time. If an aircraft arrives late at the maintenance station, the staffing plan will be interrupted because extra workers may be needed to finish the service on time. Thus, the FDARP of the airlines and the MSP of the maintenance providers are clearly interdependent.

Maintenance providers operate in a highly competitive environment and are continually striving to improve their profitability by attracting more business from airlines. To do so, maintenance providers may slightly reduce their service prices to encourage airlines to increase their maintenance visits. As airlines increase their maintenance visits to providers that offer lower prices, the routing plan will be affected, because the maintenance is performed in different locations. Thus, the price competition among maintenance providers is another factor that can interrupt the routing plans of airlines.

To our best knowledge, no studies have examined the interdependence between the FDARP of airlines and the MSP of maintenance providers while considering the price competition among the maintenance providers. This study aims to investigate this setup. For this purpose, we develop a Stackelberg-Nash game model (SNGM), which consists of two sub-games: a leader-follower Stackelberg game (LFSG), which captures the interdependence between the FDARP and the MSP; and a Nash game (NG), which reflects the price competition among the maintenance providers. We combine a bi-level ant colony optimization (ACO)-based algorithm and an analytical method to develop an iterative game algorithm to find the overall Nash equilibrium of the proposed SNGM. The effectiveness of the proposed model is demonstrated with a case study in which the SNGM is applied to a major airline and four maintenance providers located in the Middle East. In undertaking this case study, it is necessary to forecast the NPD for the airline and the demand-price function for each maintenance provider. To achieve this, we exploit data analytics by developing a neural network-based algorithm to forecast an accurate NPD. The algorithm analyzes one-year data collected on flight delays and other external factors such as bad weather and maintenance station congestion. In addition, we use a data analytics tool, called a multiple linear regression algorithm, to predict the relationship between the demand and price for each maintenance provider.

The remainder of this study is organized as follows. In Section 2, we review the relevant literature. Section 3 defines the research gap and the contribution of this study. Section 4 provides the scope, the description and formulation of the SNGM. The solution algorithm for the SNGM is presented in Section 5, and the neural network-based algorithm for predicting the NPD is described in Section 6. In Section 7, we report the case study of the SNGM for a major airline and four maintenance providers located in the Middle East, along with the performance analysis and managerial implications. Finally, Section 8 concludes the study.

2. Literature review

In this section, we briefly describe and discuss some of the major studies in the following areas; aircraft maintenance routing problem, maintenance staffing problem, game theory, and data analytics.

2.1. Aircraft maintenance routing problem

The AMRP is an effective tool for airlines to generate feasible aircraft routes. The AMRP model is generally classified into three problem types: tactical (TARP), operational (OARP), and flight delay-based operational (FDARP). First, the TARP specifies the generic arrangement of the flight legs, called rotations, to be repeated by each aircraft, while ignoring some of the operational maintenance restrictions. Second, the OARP determines the aircraft routes taking into account the operational maintenance restrictions (e.g. those mandated the Federal Aviation Administration (FAA) in the U.S.), such as the maximum number of flying hours, the maximum number of take-offs, and the maximum number of days since the last maintenance operation. Lastly, the FDARP is similar to the OARP except that flight delays are explicitly considered.

In one of the earliest studies on the TARP, Kabbani and Patty [6] proposed a set portioning model for a 3-day AMRP. To handle k-days AMRP, Gopalan and Talluri [7] developed an innovative polynomial time algorithm to solve the static and dynamic formulations of the problem. To solve the 4-day AMRP, Talluri [1] developed an effective heuristic based on the polynomial time algorithm developed by Gopalan and Talluri [7]. Clarke et al. [8] formulated the AMRP as an asymmetric travelling salesman problem to find feasible routes with maximized profit. Liang et al. [2] presented a new time space network for daily AMRP, which formed the basis for their proposed integer linear programming model. Despite the success of the TARP in generating feasible aircraft rotations, application of these rotations may not be viable due to the ignorance of some of the operational maintenance restrictions and the difficulty of repeating the rotations due to the airlines' fluctuating passenger demand. Consequently, the OARP was developed as an alternative classification of the AMRP.

Sriram and Haghani [9] developed an OARP model that considered the maximum number of days since the last maintenance operation as an operational restriction. They used an effective heuristic that solved the proposed model in reasonable computational time compared with CPLEX. In another OARP study, Sarac et al. [3] modeled the AMRP as a set-partitioning model and used branch-and-price as a solution method. The proposed model considered the maximum number of flying hours as an operational maintenance restriction. Haouari et al. [10] proposed a non-linear OARP that obeyed all three operational maintenance restrictions. To solve the proposed model, they linearized the problem using a reformulation-linearization technique. Başdere and Bilge [4] presented an OARP model in which the maximum number of flying hours was considered as an operational maintenance restriction. The proposed model was solved using branch and bound (B&B) for small sized problems, whereas large sized problems were solved using compressed annealing. Although these approaches manage to generate routes that respect the operational maintenance restrictions, as in the OARP, the route plans are questionable because the flight delays, which frequently occur in the aviation industry, are not explicitly considered. Therefore, the generated routes are sensitive to disruptions. To generate routes that better withstand disruptions, a number of studies have jointly modeled the flight delays with the operational constraints. This leads to the third classification of the AMRP.

Lan et al. [11] developed a FDARP model with the objective of minimizing the expected propagated delay. They proposed an approach in which the departure times of the flight legs were retimed to significantly reduce the PD. Dunbar et al. [12] incorporated the stochastic delay information in the model proposed by Lan et al. [11] to achieve accurate calculation of the expected PD. Liang et al. [5] proposed a model that aimed at mitigating the expected PD by inserting time buffers between the flight legs. Because the FDARP generates routes that respect both the operational maintenance restrictions and the flight delays, these routes can better withstand disruptions, resulting in good applications in the airline industry. Therefore, we use the FDARP as the basis of the model proposed in this study.

The FDARP categorizes flight delays as either a PD or an NPD. Generally, the role of the FDARP is to minimize the PD after forecasting the NPDs. In the literature, the NPD is forecasted using the expected value approach, which only focuses on analyzing the historical flight delay data. For example, Liang et al. [5] collected the NPD data for the top three fleets with the longest average PDs, and constructed the probability mass functions (PMFs) of the NPDs for each fleet. Next, they constructed a single NPD PMF by taking the average of the constructed three PMFs, and then used the single PMF to calculate the expected NPD.

In summary, numerous studies have examined the different classifications of the AMRP. In addition, researchers have examined the AMRP while considering the disruptions that frequently occur in practice. For a survey of the research on airline schedule planning disruptions in relation to the AMRP and other planning problems, we refer interested readers to the study by S.H. et al. [13].

2.2. Maintenance staffing problem

The MSP plays a critical role in planning the manpower required to service aircraft. Dietz and Rosenshine [14] proposed a MSP model with the objective of finding the optimal team sizes required to maintain military aircraft. Beaumont [15] developed a MSP model that aimed to minimize the workforce supply. Brusco and Jacobs [16] developed an algorithm for solving set partitioning formulations by eliminating the redundant columns. This algorithm showed a good performance as it reduced the number of columns by about 56%, while solving a case from United Airline. In a follow-up paper, Brusco [17] presented a way to evaluate the performance of the dual all-integer cutting plane that used for solving the set covering formulations. The results reveled an outperformance of the dual all-integer cutting plane over the commercial B&B. Alfares [18] presented a model that aimed at satisfying the maintenance demand with the objective of minimizing the number of workers. The proposed model was solved by developing a two-phase algorithm. Yang et al. [19] also proposed a model to determine the team sizes, the number of shifts required each day, and the starting and closing times of each shift. Yan et al. [20] extended the model of Yang et al. [19] by incorporating some constraints regarding the level of training and the abilities of the workers. Beliën et al. [21] developed a mixed integer programming model for the MSP, and showed a successful application for while building team sizes for Sabena Technics, a major aircraft maintenance company in Belgium. The work by Beliën et al. [21] was extended by Van den Bergh et al. [22], in which the uncertainty of the scheduled arrival times of aircraft was considered. Beliën et al. [23] presented a model with the aim of building team sizes required to service the line maintenance. The authors developed an enumerative algorithm with bounding techniques to solve the proposed model. Similar to the work by Yang et al. [19], De Bruecker et al. [24] developed a model that considered the skills and training of the workers.

It should be noted that the work by Beliën et al. [21], Van den Bergh et al. [22], Beliën et al. [23] and De Bruecker et al. [24] was formulated not only to consider the MSP, but also to consider the maintenance rostering problem. The main difference between the two problems is as follows. The MSP aims to build the team sizes required to maintain the aircraft, while considering the workforce capacity and the scheduled arrival and departure times for aircraft. The rostering problem, on the other side, aims to determine each individual working load, while considering the shift succession constraints and the limits on the number of worked hours. In this paper, the rostering problem is beyond the scope of this study, as there is no direct connection with the FDARP of the airline.

2.3. Game theory

Game theory (GT) is concerned with the study of the strategic interactions among rational decision makers who must make decisions that potentially influence the interests of other decision makers. In this study, we focus on two types of game: a leader-follower Stackelberg game (LFSG), and a Nash game (NG).

The LFSG focuses on the interaction between two self-interested players, known as the leader and the follower [25]. The leader holds a powerful position that allows him/her to make decisions that are observable by the follower. The follower then reacts rationally to the decisions of the leader. This game has been successfully applied in many areas, such as seller-buyer supply chains [26, 27], product families and supply chains [25, 28], inventory policies in the vendor managed inventory [29], and pricing [30].

The NG is usually used to capture the competition among different players when setting their product/ service price [31]. This price competition proceeds among the players as follows. First, each player sets a price that is judged based on the customer demand. Next, by observing the prices chosen by the other players, each player reacts by adjusting the price to maximize his/her own profit. So, the price set by one player is not only dependent on his/her decision, but is also dependent on the prices offered by the other players. Similar to the LFSG, the NG has been successfully applied in many fields, such as manufacturer-retailer supply chains [32, 33], pricing in supply chains [34], advertising in vendor managed inventory [31], and rail transportation systems [35].

2.4. Data analytics

Data analytics involves using various tools, including data mining and statistical tools, to discover the correlations, trends, and other valuable information in a body of data [36, 37]. Among the various data analytics tools, neural networks and regression algorithms are very efficient in capturing the relationships between a response variable and one or multiple predictors. The efficiency of the neural network has been reported in various fields, such as demand forecasting [38], liquidity risk assessment in banking [39], prediction of organ status in the healthcare industry [40], time series forecasting [41], and prediction of traveler behavior[42]. The regression algorithm has also been successfully applied in areas such as delay and demand forecasting in the railroad industry [43, 44], price prediction in the warm-water fish supply chain [45], acceleration prediction for railway wagons [46], forecasting air concentrations [47], and forecasting heat demand for district heating system [48]. All the aforementioned studies reveal the fact that, data analytics has been successfully applied in different fields. Therefore, it is worth enough to apply data analytics in this study.

3. Research gap and contribution

3.1. Research gap

Although the FDARP of airlines and the MSP of maintenance providers are clearly interdependent, the literature has focused on independent solutions of the two problems [5, 11, 19, 20]. This interdependence stems from the interrelated requirements of the airlines and maintenance providers. Airlines use the FDARP as a tool to determine their routing plans, which involves determining the aircraft routes including visits to the maintenance providers [5]. For instance, if an aircraft undergoes a maintenance check by a maintenance provider who suffers a workforce shortage, this may result in prolonged maintenance, which in turn will delay the next flight performed by the aircraft, leading to an interruption to the routing plan. On the other hand, maintenance providers use the MSP to build staffing plan, which includes the team sizes required to maintain the received aircraft [20]. This staffing plan may be interrupted in the case of late arrival of the aircraft, which may necessitate extra workers being added to the planned teams to release the aircraft from the maintenance station without a long delay. Clearly, the FDARP of airlines and the MSP of maintenance providers are closely related. Therefore, if both problems are solved independently, the desired routing and staffing solutions may not be achieved, leading to severe flight delays for airlines and disruption of the staffing plans of the maintenance providers.

Maintenance providers have faced tight competition and need to improve their profitability by attracting more demand from the airlines. The maintenance providers can cut their service prices to encourage airlines to increase maintenance visits. However, the increased maintenance visits to lower priced providers will affect the airlines' routing plans, including changing the locations for maintenance. Therefore, the price competition among the maintenance providers is another factor that can interrupt the routing plans of airlines.

In addition, the NPD in the FDARP has been forecasted using the expected value approach. However, in the literature, only historical flight delay data are considered and factors that affect the NPD, such as bad weather and maintenance station congestion, are not dealt with [5, 11, 12]. This approach results in forecasting inaccurate NPDs. Lastly, although game theory and data analytics have been applied in many fields, few studies have used game models and data analytics techniques jointly to handle issues relating to the aviation industry.

To the best of our knowledge, no studies have examined the interdependence between the FDARP of airlines and the MSP of maintenance providers while considering the price competition among the maintenance providers. Accordingly, in this study, we try to fill these gaps by developing a SNGM that can capture the relationship between the airlines and the maintenance providers. We also use data analytics tools to forecast accurate NPDs for the airlines and the demand-price relationship for each maintenance provider. In this sense, this study is a step forward addressing the real needs and practical problems of the aviation industry.

3.2. Contribution

This study makes a number of contributions to the literature. First, as mentioned, the majority of the reviewed studies solve the FDARP and MSP using independent approaches that ignore the interdependence between these problems [5, 11, 19, 20]. In contrast to the existing approaches, we propose a new approach that captures this interdependence. Achieving this goal is quite challenging because the FDARP and MSP address the needs of two different sectors with conflicting goals. For the airlines, the FDARP is solved with the goal of minimizing the expected cost of the PDs. To achieve this, it is necessary for the aircraft to complete the maintenance operations punctually, which requires maximizing the team sizes planned by the maintenance providers and thus increases the labor costs incurred by the maintenance providers. In contrast, the maintenance providers use the MSP to minimize their labor costs by reducing the team sizes required to maintain each aircraft. This may prolong the maintenance and delay the next flight of the aircraft, which in turn would increase the expected PD costs incurred by the airline. Consequently, due to the conflicting goals, it is not reasonable to model the FDARP and the MSP using an "all-in-one" method that combines the two objective functions. Accordingly, a coordinated system needs to be used to model the FDARP and the MSP while considering their interdependence and conflicting goals. Thus, the features of the FDARP and the MSP need to be examined in detail to determine the formulation of the coordinated system. We find that the FDARP of airlines determines the number of maintenance visits for each aircraft. These visits, in turn, form the demand for the maintenance providers who use the demand as an input for the MSP to determine the staffing plan. Accordingly, the FDARP clearly has the dominant position due to the determination of the demand, whereas the MSP holds the subordinate position because it uses the demand determined by the FDARP. To capture these features, in which the FDARP behaves as a leader and the MSP acts as a follower, we formulate the coordinated system as a LFSG model. To represent this model, we use a bi-level model in which the FDARP forms the upper-level and the lower-level is represented by the MSP.

Second, as mentioned, ignoring the price competition among the maintenance providers may lead to disruptions to the routing plans of the airlines. Therefore, the competition is explicitly modelled in our analysis. Price cuts by a maintenance provider lead to increased maintenance demands (i.e., the number of aircraft to be maintained). Observing the prices of the other providers motivates the maintenance providers to adjust their prices to attract more demand from the airlines. This price setting process naturally leads to the formulation of the price competition among the maintenance providers as a NG model.

Lastly, as opposed to the expected value approach, which only focuses on the historical flight delay data, we propose considering some external factors that affect the NPD in practice, including bad weather, maintenance station congestion, and peak season effects. To consider this enormous amount of data, we develop a neural network-based data analytic algorithm that combines historical data and the external factors, which in turn results in accurate prediction of the NPD.

The two sub-games, namely the LFSG and NG, form the SNGM. This model is of practical use for airlines and maintenance providers. The LFSG model helps in constructing the routing and staffing plans while meeting the conditions of each party, and ultimately minimizes the expected PD costs and the labor costs for the airline and maintenance providers, respectively. The NG model enables airlines to select cheaper providers, resulting in reduced maintenance service costs. The NG model also allows maintenance providers to attract more demand, resulting in increased net profits. Finally, the neural network-based algorithm is of use in the aviation industry because it provides airlines with a means of accurately predicting the NPD and developing efficient routing plans.

4. The Stackelberg-Nash game model

In this section, we mainly propose the two sub-games that form the SNGM. Before presenting these games, we first define the model scope and the notations.

4.1. Model scope and notations

The scope of the proposed model can be summarized as follows:

- An airline and multiple maintenance providers are considered in the games.
- The airline focuses on solving the FDARP with a 4-day planning horizon.
- The FDARP of the airline considers only the Type A maintenance check as it is the most frequently used check [4].
- The FDARP of the airline considers the expected value of a non-propagated delay, which is defined as any delay resulting from bad weather, congested maintenance stations, and technical problems. Note that this value is considered for each flight leg in the schedule. In other words, when constructing the routing plan, we consider not only the flight duration of each flight leg, but also the expected NPD that might occur after each flight leg.
- The maintenance providers focus on solving the MSP with deterministic workforce capacity .
- The maintenance providers offer the maintenance service in a competitive market. This competition stems from the Type A maintenance check being the simplest as it includes visual inspection of major parts such as the aircraft engines, and thus the vast majority of providers can provide this check. Therefore, the airlines seek the cheapest provider because any provider can perform this service. Thus, due to the competition, each maintenance provider’s revenue is not only dependent on its own price, but is also affected by the price decisions of the other providers.
- The maintenance service demand for a maintenance provider is not only a function of the price offered by the provider, but is also a function of the prices of all other providers. This is also due to the competition.
- The information exchange among the maintenance providers is limited in that each maintenance provider does not have complete information about the other providers because each provider

thinks that it is risky to disclose much information. The only action that can be taken by each maintenance provider is to react to the price decisions of the other providers.

After presenting the scope of the model, we summarize the notations used throughout this model as follows:

<u>Airline (Leader of the LFSG)</u>

Sets and indices:

I	Set of flight legs, indexed by i or j.
K	Set of aircraft, indexed by k.
MT	Set of maintenance providers, indexed by m or g.
A	Set of airports, indexed by a.
$v \in \{1,2,\dots,\Psi\}$:	The average number of maintenance operations such that each aircraft should receive during the planning horizon.
$\{o,t\}$:	Starting and ending nodes of the connection network.

Parameters

DT_i:	Departure time from the origin airport of flight leg i.
O_{ia}:	Binary integer. It takes value of 1 if the origin airport of flight leg i is airport a and 0 otherwise.
AT_i:	Arrival time at the destination airport of flight leg i.
D_{ia}:	Binary integer. It takes value of 1 if the destination airport of flight leg i is airport a and 0 otherwise.
FT_i:	Duration of flight leg i.
TRT:	Turn-around time.
T_{max}:	Maximum number of allowable cumulative flying hours since last maintenance operation.
C_{max}:	Maximum number of allowable take-offs since last maintenance operation.
$E(NPD_i)$:	Expected value of the non-propagated delay of flight leg i.
Mb_{ma}:	Binary indicator. It takes value of 1 if maintenance provider m is located at airport a and 0 otherwise.
MAT:	Duration required to complete Type A maintenance check. This time should be determined by maintenance provider. However, it is assumed by the airline in the first round of the LFSG between the FDARP of the airline and the MSP of the maintenance providers.
FS:	Fleet size
Ψ:	Maximum average number of maintenance operations should be received by each aircraft. It is calculated by following this rule; $\Psi = \sum_{i\in I} FT_i/(T_{max}\,FS)$.
M:	A big number.
PD_{ijkv}:	Value of propagated delay appeared when aircraft k covers flight leg i and j consecutively, before receiving the maintenance operation number v.

PD_{ikv} :	Value of accumulated propagated delay that appears before covering flight leg i by aircraft k, before receiving the maintenance operation number v.
C_{pD}:	Expected cost incurred by the airline for each minute of propagated delay.

Decision variables

$x_{ijkv} \in \{0,1\}$:	It takes value of 1 if aircraft k covers two consecutive flight legs i and j, before receiving the maintenance operation number v, and 0 otherwise.
$y_{imkv} \in \{0,1\}$:	It takes value of 1 if flight leg i is covered by aircraft k, then the aircraft proceeds to maintenance provider m to receive the maintenance operation number v, and 0 otherwise.
$z_{mjkv} \in \{0,1\}$:	It takes value of 1 if aircraft k leaves maintenance provider m to cover flight leg j, after receiving the maintenance operation number v, and 0 otherwise.
$ATBM_{kvm} > 0$	Scheduled arrival time of aircraft k at maintenance provider m to receive the maintenance operation number v.
$RTAM^{*}_{kvm} > 0$:	Scheduled departure time of the aircraft k from maintenance provider m, after receiving the maintenance operation number v.

<u>Maintenance providers (Follower of the LFSG + Forming the NG)</u>

Sets and indices

$kv \in \{11, \ldots., 1\Omega, \ldots bv, \ldots, K\Psi\}$:	Set of aircraft that are planned to visit the maintenance provider to receive a specific number of maintenance operation. Each element in this set consists of two parts such that the first part, k, represents the aircraft, whereas the second part, v, represents the number of the maintenance operation that will be received by the aircraft.
S	Set of shifts operated by maintenance provider, indexed by s.
$\{o', t'\}$	Starting and ending node of the layered graph.

Parameters

SAT_{kvm}:	Scheduled arrival time of an aircraft k to receive the maintenance operation number v by maintenance provider m.
SDT_{kvm}:	Scheduled departure time of an aircraft k after receiving the maintenance operation number v by maintenance provider m.
w^{l}_{sm}:	Minimal team size (number of worker) that can be formed by maintenance provider m during shift s.
w^{u}_{sm}:	Maximal team size (number of worker) that can be formed by maintenance provider m during shift s.
Q^{max}_{s}:	Capacity of workforce during shift s.
l_{kv}:	Workload (man-hours) required to maintain an aircraft k that receives maintenance operation number v.

C_{wkvsm}:	Cost incurred by maintenance provider m when assigns w workers to maintain an aircraft k that receives maintenance operation number v during shift s.
De_m:	Demand volume of maintenance service for maintenance provider m.
θ_m:	A positive constant that reflects the potential size of maintenance service that can be offered by maintenance provider m.
ϑ_m:	Sensitivity of maintenance service demand for maintenance provider m to its maintenance service price.
δ_{mg}:	Sensitivity of maintenance service demand for maintenance provider m to the maintenance service price set by competitor g.
TA_m:	Total number of aircraft served by maintenance provider m.
NP_m:	Profit of maintenance provider m.

Decision variables for maintenance providers

$wf_{kvsm} \in \{w_{sm}^{l}, \dots, w_{sm}^{u}\}$:	Number of workers (team size) assigned by maintenance provider m to maintain an aircraft k that receives maintenance operation number v during shift s.
$RTAM_{kvm} > 0$:	Real departure time of an aircraft k after receiving maintenance operation number v by maintenance provider m.
$P_m > 0$:	Price of the maintenance service offered by maintenance provider m.

4.2. Framework of the SNGM

Figure 1 illustrates the framework of the SNGM, which consists of two sub-games: a vertical LFSG and a horizontal NG.

The vertical LFSG captures the interdependence between two main players;

1. An airline that acts as a leader with the following characteristics:
    a. Objective function (payoff): Minimization of the expected PD cost and the maintenance service cost.
    b. Decision variables (strategies); x_{ijkv}, y_{imkv}, z_{mjkv}, $ATBM_{kvm}$, and $RTAM_{kvm}^{*}$.
    c. Constraints: Coverage constraints, the balance constraints, and the operational maintenance constraints, as explained in section 4.3.

2. Multiple maintenance providers that acts as followers with the following characteristics:
    a. Objective function (payoff): Minimization of the labor cost.
    b. Decision variables (strategies): wf_{kvsm} and $RTAM_{kvm}$
    c. Constraints: Workforce capacity constraints, the scheduled arrival and departure times for the aircraft.

LFSG proceeds as follows. Acting as a leader, the airline starts the game by solving the FDARP to determine the routing plan decisions (referred to as x_{ijkv}, y_{imkv}, z_{mjkv}, $ATBM_{kvm}$, and $RTAM_{kvm}^{*}$). The last two decisions determine the scheduled arrival and departure times for the aircraft. The last two decisions are sent to the maintenance providers, who in turn act as followers and solve the MSP to determine their staffing plan decisions (referred to as wf_{kvsm} and $RTAM_{kvm}$). The last decisions indicate the real departure times for the aircraft, which is sent back to the airline. The decision process is iterated until the Stackelberg equilibrium is reached, at which point both players no longer have the intention to change their decisions, because any change might have a negative impact on their objective functions.

The horizontal NG captures the competition among multiple players that have the following characteristics:

a. Objective function (payoff): Maximization of the profit.
b. Decision variable (strategies): P_m.
c. Constraints: Prices set by competitors, as explained in section 4.4.

The horizontal NG interacts with the LFSG as follows. The game starts with two decisions from the LFSG. First, the routing decision, y_{imkv}, which reflects the demand for the maintenance providers. Second, the staffing plan decision, wf_{kvsm}, which is used to calculate the labor cost with the help of C_{wkvsm}. The previous two decisions are shown in Figure 1 as an arrow from the LFGG to the NG. In fact, these decisions help the NG to determine the maintenance service price for each provider (referred to as P_m). Note that these prices are determined not only by considering the self-pricing decisions, but also by considering the competitors' price decisions. Therefore, no provider has the intention to change the price, resulting in the Nash Equilibrium.

The equilibrium pricing decisions are then sent back to the LFSG. Note that the pricing decisions are shown in Figure 1 as an arrow from the NG to the LFSG. If these prices change the Stackelberg decisions, the LFSG will generate new routing and staffing decisions. Next, the new decisions are sent to the NG and the process continues. This process is iterated until a stable situation is reached in which none of the players are willing to change their decisions because any deviation will not improve their own benefits. This stable situation, in which the LFSG and NG are both in equilibrium, is called the overall Nash equilibrium.

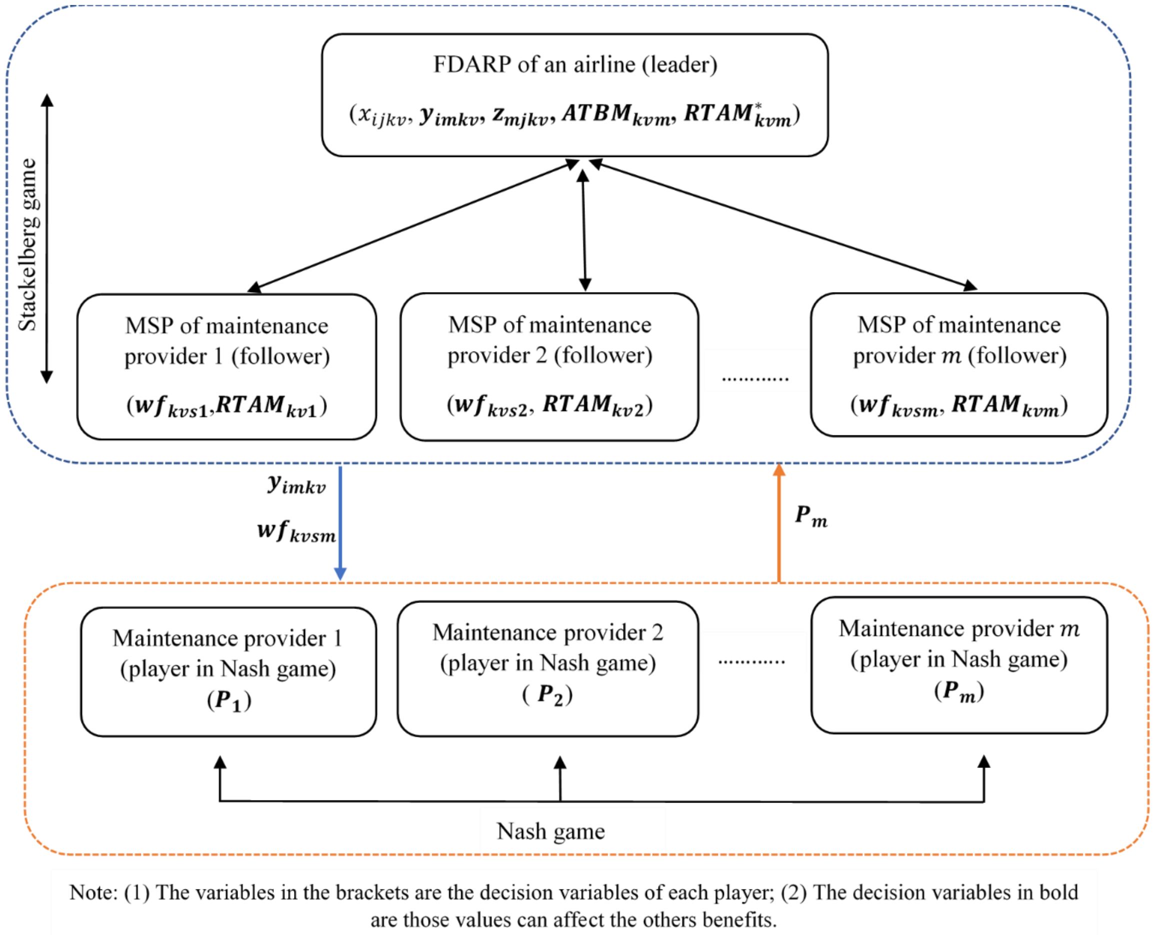


Figure 1: Framework of the Stackelberg-Nash game

4.3. Formulation of the LFSG

In this section, we present the LFSG, which is modeled as a bi-level model, in which the FDARP of the airline forms the upper-level and the lower-level is represented by the MSP. It is important to mention here that, usually, the LFSG aims to find the equilibrium solution, whereas the bi-level programming model aims to find the optimal solution. Since the bi-level programming is used to model the LFSG, the main aim is to find the equilibrium solution and is not to find out the optimal solution.

The upper-level includes the FDARP, which aims to construct the routing plan with the objective of minimizing the expected PD cost and the maintenance cost. Note that the FDARP is formulated by following the connection network because it is one of the most efficient networks used for representing the AMRP [3, 4, 10]. This network includes two node sets and three arc sets. For the node sets, the first set represents the flight legs (I) and the second set denotes the maintenance providers (MT), as shown in Figure 2. The arc sets are compressed into three different types: the coverage arc set (COV), the visiting maintenance arc set (VMA), and the leaving maintenance arc set (LMA). The coverage arc $cov(i,j) \in COV$ can be used to cover the flight legs because it helps the aircraft to cover two consecutive flight legs. The other arc sets, VMA and LMA, can be used to visit the maintenance providers. The visiting maintenance arc $vma(i,m) \in VMA$ helps the aircraft to visit the maintenance providers after the flight coverage, whereas the leaving maintenance arc $lma(m,j) \in LMA$ plays the role of covering the flight legs after completing the maintenance operations. The nodes and arcs included in the connection network ease the formulation of the FDARP as a multi-commodity network flow model, in which each aircraft denotes a single commodity moving throughout the network. To build the routing plan, the FDARP uses x_{ijkv}, y_{imkv}, and z_{mjkv} as decision variables to represent the coverage arcs, the visiting maintenance arcs, and the leaving maintenance arcs, respectively. The FDARP also uses other decision variables, called $ATBM_{kvm}$ and $RTAM^{*}_{kvm}$, to determine the scheduled arrival and departure times for the aircraft.

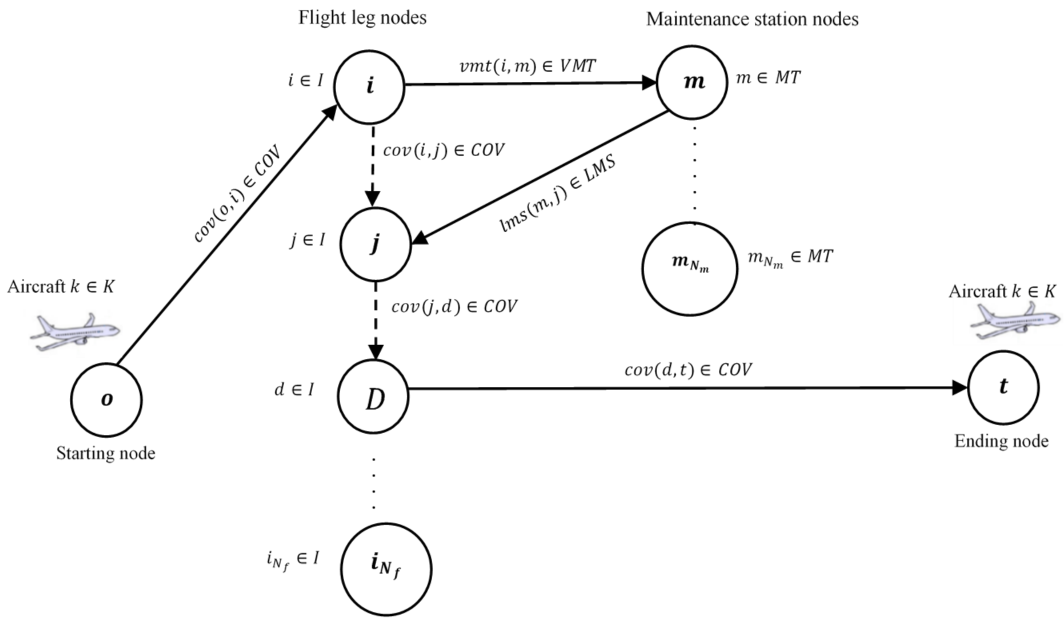


Figure 2: Construction of the connection network

The lower-level represents the MSP for constructing the staffing plan, with the objective of minimizing the labor cost. We formulate the MSP as a layered graph, because this graph has shown efficient performance in capturing the staffing and worker allocation problem [49]. The graph is constructed using three components: the layers, nodes, and arcs, as shown in Figure 3. The layers represent the aircraft that will receive the maintenance operations, in addition to representing the starting and ending points of the layered graph. The nodes are incorporated in each layer to represent the potential number of workers that can maintain the aircraft. Lastly, the arcs are inserted in the graph to connect the layers, so that the model can be easily solved using metaheuristics such as ACO.

The decision variables used by the MSP mainly include wf_{kvsm} and $RTAM_{kvm}$. wf_{kvsm} helps to determine the number of workers required to maintain the aircraft, whereas $RTAM_{kvm}$ specifies the real departure times for the aircraft from the maintenance station. These MSP decisions are finally leading to generate an efficient staffing plan.

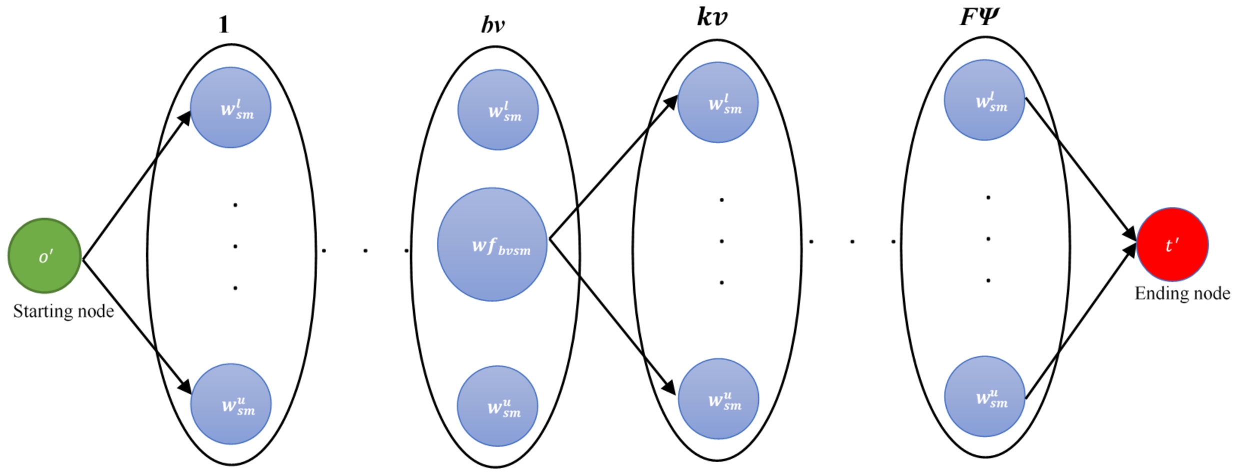


Figure 3: Construction of the layered graph

Based on the predefined notations, the LFSG can be formulated as a bi-level model as follows:

$$\min \sum_{v=1,\dots,\Psi} C_{pD}\left(\sum_{k\in K}\sum_{i\in I}\sum_{j\in I} PD_{ijkv}\, x_{ijkv}\right) + \sum_{m\in MT} P_m \left(\sum_{i\in I}\sum_{k\in k}\sum_{v=1,\dots,\Psi} y_{imkv}\right) \qquad (1.0)$$

s.t.

$$PD_{ijkv} = PD_{ikv} + \left(E(NPD_{ik}) - (DT_j - AT_i - TRT)\right)^+ \qquad \forall\, i,j \in I, \forall\, k \in K, \forall\, v = 1,\dots,\Psi \qquad (1.1)$$

$$\sum_{k\in K}\left(\sum_{j\in I\cup\{t\}}\sum_{v\in\Psi} x_{ijkv} + \sum_{m\in MT}\sum_{v\in\Psi} y_{imkv}\right) = 1 \qquad \forall\, i \in I \qquad (1.2)$$

$$\sum_{j\in I} x_{ojkv} + \sum_{m\in MT} y_{omkv} = 1 \qquad \forall\, k \in k, \forall\, v = 1,\dots,\Psi \qquad (1.3)$$

$$\sum_{i\in I} x_{itkv} + \sum_{m\in MT} z_{mtkv} = 1 \qquad \forall\, k \in K, \forall\, v = 1,\dots,\Psi \qquad (1.4)$$

$$\sum_{j\in I\cup\{o\}} x_{jikv} + \sum_{m\in MT} z_{mikv} = \sum_{j\in I\cup\{t\}} x_{ijkv} + \sum_{m\in MT} y_{imkv} \qquad \forall\, i \in I, \forall\, k \in K, \forall\, v = 1,\dots,\Psi \qquad (1.5)$$

$$\sum_{j\in I}\sum_{v=1,\dots,\Psi} y_{jmkv} = \sum_{j\in I\cup\{t\}}\sum_{v=1,\dots,\Psi} z_{mjkv} \qquad \forall\, m \in MT, \forall\, k \in K \qquad (1.6)$$

$$AT_i + TRT - DT_j \le M(1 - x_{ijkv}) \qquad \forall\, i,j \in I, \forall\, k \in K, \forall\, v = 1,\dots,\Psi \qquad (1.7)$$

$$\sum_{k\in K} x_{ijkv} \le \sum_{a\in A} D_{ia} O_{ja} \qquad \forall\, i,j \in I, \forall\, v = 1,\dots,\Psi \qquad (1.8)$$

$$\sum_{k\in K} y_{imkv} \le \sum_{a\in A} D_{ia} Mb_{ma} \qquad \forall\, i \in I, \forall\, m \in MT, \forall\, v = 1,\dots,\Psi \qquad (1.9)$$

$$ATBM_{kvm} \ge \sum_{i\in I\cup\{o\}}\sum_{m\in MT} (AT_i + TRT + PD_{ikv}) y_{imkv} \qquad \forall\, k \in K, \forall\, v = 1,\dots,\Psi \qquad (1.10)$$

$$\sum_{k\in K} z_{mjkv} \le \sum_{a\in A} Mb_{ma}\, O_{ja} \qquad \forall\, m \in MT, \forall\, j \in I, \forall\, v = 1,\dots,\Psi \qquad (1.11)$$

$$RTAM^*_{kvm} - DT_j \le M\left(1 - z_{mjkv}\right) \qquad \forall\, m \in MT, \forall\, j \in I, \forall\, k \in K, \forall v = 1,\dots,\Psi \qquad (1.12)$$

$$RTAM^*_{kvm} \geq \sum_{i\in I\cup\{o\}} \sum_{m\in MT} (AT_i + TRT + PD_{ikv} + MAT) y_{imkv} \qquad \forall\, k \in K, \forall\, v = 1, \dots, \Psi \qquad (1.13)$$

$$RTAM^*_{kvm} \geq RTAM_{kvm} \qquad \forall k \in K, \forall\, v = 1, \dots, \Psi, \forall\, m \in MT \qquad (1.14)$$

$$\sum_{i\in I\cup\{o\}} \sum_{j\in I} x_{ijkv} \leq C_{max} \qquad \forall\, k \in K, \forall\, v = 1, \dots, \Psi \qquad (1.15)$$

$$\sum_{i\in I\cup\{o\}} \sum_{j\in I} FT_j x_{ijkv} \leq T_{max} \qquad \forall\, k \in K, \forall\, v = 1 \qquad (1.16)$$

$$\sum_{i\in I} \sum_{j\in I} FT_j x_{ijkv} + \sum_{m\in MT} \sum_{j\in I} FT_j z_{mjkv} \leq T_{max} \qquad \forall\, k \in K, \forall\, v = 2, \dots, \Psi \qquad (1.17)$$

$$x_{ijkv} \in \{0,1\} \qquad \forall\, i, j \in I, \forall\, k \in K, \forall\, v = 1, \dots, \Psi \qquad (1.18)$$

$$y_{imkv} \in \{0,1\} \qquad \forall\, i \in I, \forall\, m \in MT, \forall\, k \in K, \forall\, v = 1, \dots, \Psi \qquad (1.19)$$

$$z_{mjkv} \in \{0,1\} \qquad \forall\, m \in MT, \forall\, j \in I, \forall\, k \in K, \forall\, v = 1, \dots, \Psi \qquad (1.20)$$

$$ATBM_{kvm} > 0 \qquad \forall k \in K, \forall\, v = 1, \dots, \Psi, \forall\, m \in MT \qquad (1.21)$$

$$RTAM^*_{kvm} > 0 \qquad \forall k \in K, \forall\, v = 1, \dots, \Psi, \forall\, m \in MT \qquad (1.22)$$

where given decision variables ($ATBM_{kvm}$ and $RTAM^*_{kvm}$) are used for solving:

$$\min \sum_{m\in MT} \sum_{s\in S} \sum_{kv\in k\Psi} C_{wkvsm}\, wf_{kvsm} \qquad (2.0)$$

s.t.

$$RTAM_{kvm} \geq SDT_{kvm} + \left(SAT^{\xi}_{kvm} + TRT + \frac{l_{kv}}{wf^{\xi}_{kvsm}} - SDT^{\xi}_{kvm} \right)^{+} \qquad \forall kv \in k\Psi, \forall m \in MT \qquad (2.1)$$

$$\sum_{kv\in k\Psi} wf_{kvsm} \leq Q^{max}_{s} \qquad \forall s \in S, \forall m \in MT \qquad (2.2)$$

$$SAT_{kvm} = ATBM_{kvm} \qquad \forall kv \in k\Psi, \forall s \in S, \forall m \in MT \qquad (2.3)$$

$$SDT_{kvm} = RTAM^*_{kvm} \qquad \forall kv \in k\Psi, \forall s \in S, \forall m \in MT \qquad (2.4)$$

$$wf_{kvsm} \in \{w^{l}_{sm}, \dots, w^{u}_{sm}\} \qquad \forall kv \in k\Psi, \forall s \in S, \forall m \in MT \qquad (2.5)$$

$$RTAM_{kvm} > 0 \qquad \forall kv \in k\Psi, \forall m \in MT \qquad (2.6)$$

The FDARP, as the upper-level, is represented by Eqs. (1.0) – (1.20), whereas the lower-level is represented by the MSP in Eqs. (2.0) – (2.8).

The objective function of the upper-level is to minimize the expected PD cost and the maintenance service cost, as described by Eq. (1.0). Note that the maintenance cost is determined based on the maintenance service prices specified by the NG. Constraints (1.1) describe the calculation of the PD.

To build a feasible routing plan, the aircraft need to cover all the flight legs. For this reason, coverage constraints (1.2), (1.3), and (1.4) are formulated. Constraints (1.2) indicate that each flight leg is covered exactly once, whereas constraints (1.3) and (1.4) ensure the route initiation and the route completion for each aircraft, respectively.

When constructing the routing plan, the circulation of the aircraft throughout the network should be maintained. This can be achieved by using the balance constraints (1.5) and (1.6). Constraints (1.5) maintain the balance when the aircraft covers the flight legs, whereas constraints (1.6) ensure the balance when the aircraft visits the maintenance providers.

The time and place issues are important factors to be considered when constructing the routing plan. Therefore, constraints (1.7) and (1.8) are formulated. Constraints (1.7) indicate the time constraints because they manage the timing when connecting two consecutive flight legs using the same aircraft. Constraints (1.8) indicate the place constraints because they handle the origin and destination considerations when connecting two consecutive flight legs using the same aircraft.

As mentioned, an applicable routing plan should include some maintenance visits for the aircraft, as mandated by the FAA. To achieve this, we use constraints (1.9). These constraints guarantee that each aircraft can visit the maintenance provider if the destination airport of the last covered flight leg and the location of the maintenance provider are the same. In real practice, to prepare a suitable maintenance operation for the aircraft, the airline should inform the maintenance provider with the scheduled arrival time of the aircraft. For this purpose, constraints (1.10) are cast. These constraints ensure that the scheduled arrival time of the aircraft at the maintenance provider is later than or equal the arrival time of the last covered flight leg plus the turn-around time and the value of accumulated propagated delay.

After completing the maintenance operation, the aircraft should leave the maintenance provider and resume covering the next scheduled flight leg. To achieve this, constraints (1.11) – (1.14) are imposed. Constraints (1.11) constitute the place constraints, which handle the origin considerations when selecting the next flight to be covered after leaving the maintenance provider. The time constraints in (1.12) guarantee that the aircraft can cover the next flight leg after leaving the maintenance provider if the departure time of the next flight leg is larger than or equal to the scheduled departure time of aircraft $RTAM^{*}_{kvm}$, which is determined by constraints (1.13) and (1.14). Initially, in the first round of the LFSG between the FADRP of the airline and the MSP of the maintenance providers, $RTAM^{*}_{kvm}$ is determined using constraints (1.13), which include an assumption by the airline regarding the maintenance duration. In reality, this assumption is not applicable, because this duration should be determined by the maintenance provider. Therefore, in the subsequent rounds of the LFSG, constraints (1.13) become redundant and $RTAM^{*}_{kvm}$ is determined using constraints (1.14). These constraints ensure that the airline builds its calculation based on the real departure time of the aircraft received from the MSP of the maintenance provider. In this model, constraints (1.14) constitute the links between the upper and lower-levels of the bi-level model.

Forcing an aircraft to undergo maintenance cannot be achieved using the abovementioned constraints. Therefore, the operational maintenance constraints (1.15) – (1.17) are imposed. Constraints (1.15) ensure that the number of take-offs since the last maintenance operation does not exceed the allowable limit. Similarly, constraints (1.16) and (1.17) are restrictive constraints regarding the accumulated flying hours. Finally, constraints (1.18) – (1.22) represent the domain restrictions imposed on the decision variables.

The lower level has the objective function of minimizing the total labor cost incurred by the maintenance providers, as represented by Eq. (2.0). Constraints (2.1) describe the calculation of the real departure time for the aircraft to leave the maintenance provider.

To build a feasible staffing plan, the worker capacity in each shift needs to be considered. For this purpose, constraints (2.2) are imposed to ensure that the total number of workers allocated to service the aircraft in each shift does not exceed the worker capacity.

Because the MSP acts as a follower of the LFSG, it should receive some information from the leader. For this purpose, constraints (2.3) and (2.4) are incorporated in the model. Constraints (2.3) and (2.4) help to calculate the scheduled arrival and departure times for each aircraft, respectively. These two constraints are formulated based on the decision variables $ATBM_{kvm}$and $RTAM^{*}_{kvm}$ received from the leader. Finally, constraints (2.5) and (2.6) indicate the domain definitions of the decision variables.

4.4. Formulation of the NG

In this section, we propose the NG, which describes how the maintenance providers compete to set the price of the maintenance service. The NG can be modeled as follows:

Maximize (for $\forall m \in MT$):

$$NP_m = De_m P_m - \left(\frac{\sum_{s\in S}\sum_{kv\in k\Psi} C_{wkvsm} wf_{kvsm}}{TA_m}\right) De_m \tag{3.0}$$

where
$$De_m = \theta_m - \vartheta_m P_m + \sum_{g\in MT} \delta_{mg} P_g \tag{3.1}$$

$$TA_m = \sum_{i\in I}\sum_{k\in k}\sum_{v=1,\dots,\Psi} y_{imkv} \tag{3.2}$$

Eq. (3.0) represents maintenance provider m's profit, which is determined by the revenue represented in the first term minus the total labor cost as described in the second term. Note that the total labor cost is determined by multiplying the demand (De_m) by the average labor cost incurred for each aircraft (($\sum_{s\in S}\sum_{kv\in k\Psi} C_{wkvsm} wf_{kvsm}$)$/TA_m$). The average labor cost incurred for each aircraft is calculated using the decision variable wf_{kvsm} that is received from the follower of the LFSG. The leader of the LFSG also helps the NG to determine the average labor cost incurred for each aircraft using the decision variable y_{imkv}, which specifies TA_m, as shown in Eq. (3.2).

Because the maintenance providers compete in setting the maintenance service price, each maintenance provider's demand should be formulated based on its own price and the other observed prices. Therefore, the demand is formulated consistent with this observation, as shown in Eq. (3.1), such that the demand for maintenance provider m is not only a function of its own price P_m, but also of the competitors' prices P_g. Note that ϑ_m and δ_{mg} are given while considering the demand properties $\frac{\partial De_m}{\partial P_m} < 0, \frac{\partial De_m}{\partial P_g} > 0, m, g \in MT$ following Samuelson [50].

5. Solution algorithm for the overall Nash equilibrium

To obtain the overall Nash equilibrium, we have to obtain: (1) the Stackelberg equilibrium for the LFSG and (2) the Nash equilibrium for the NG. Therefore, in Sections 5.1 and 5.2, we discuss how to find the two types of equilibrium. We then provide the algorithm to find the overall Nash equilibrium in Section 5.3.

5.1. Obtaining the Stackelberg equilibrium

To obtain the Stackelberg equilibrium, it is necessary to solve the bi-level model. Before presenting the proposed solution method, we briefly discuss the existing solution methods. There are two main approaches for solving the bi-level model: indirect and direct. The indirect approach changes the model from a bi-level to a single level structure, and solves it using methods such as B&B based on the K times best method [51], the Karushe-Kuhne-Tucker (KTT) conditions method [52], and the penalty function method [53]. However, using the indirect approach to solve our bi-level model is not promising for two reasons. First, it overlooks the fact that each level belongs to a different company with a specific self-interested goal. Second, the leader's decision power might be dominated by the follower's decision, resulting in a different representation for the model. Alternatively, the direct approach solves the bi-level model directly using methods such as the satisfactory solution method [54]. Although the direct approach respects the structure of the bi-level model, it becomes quite challenging for the direct solution methods to handle large scale network problems. This is apparent when the efficiency of the direct solution methods is significantly reduced due to the large number of nodes included in the network model [28]. Because each level of our bi-level model (known as the FDARP and the MSP) belongs to large scale network problem, the application of the direct approach is challenging.

Note that each level of the proposed bi-level model belongs to the class of NP-hard problems [3, 49]. Therefore, it is reasonable to use meta-heuristics to solve the proposed model because they have been successfully applied in solving different problems, such as the crew scheduling problem [55], the vehicle routing problem [56], the aircrew rostering problem [57], and the control attitude behavior problem [58]. As aforementioned, both the FDARP and the MSP are modeled as network-based problems, for which ACO has been successfully applied in solving large and complex network-based problems [59-62]. These observations motivate us to propose a bi-level ACO based-algorithm to solve the bi-level model. This algorithm consists of two levels: the upper-level ACO-based algorithm for the FDARP and the lower-level ACO-based algorithm for the MSP. These two levels are designed because each level of the bi-level model has its unique features and goals.

5.1.1. Upper-level ACO-based algorithm

The upper-level ACO-based algorithm consists of three main steps:

- Covering flight legs; to conduct this step, the ants, which simulate the aircraft, scout throughout the network to cover flight legs using the so-called state transition rule. In other words, suppose that an ant covers a flight leg represented by node i and looks for covering next flight leg represented by node j. To select the next flight leg, we adopt the following state transition rule:

$$j = \begin{cases} \arg\_max_{j \in NFL_i^k} \left\{ \left[\tau_{ij}\right]^{\alpha} \left[\eta_{ij}\right]^{\beta} \right\} & if\ q \leq q_0 \\ J & if\ q > q_0 \end{cases} \quad (4.0)$$

  where NFL_i^k denotes the potential flight legs to be covered by ant k after covering flight leg i. The terms τ_{ij} and η_{ij} represent the pheromone trail and the heuristic function of the coverage arc $cov\ (i,j)$. Note that the η_{ij} can be determined as $1/(C_{pD} * PD_{ijkv})$. The terms α and β are used to express the relative importance of τ_{ij} and η_{ij}, respectively. q is a random number based on uniform distribution $[0{\sim}1]$, whereas q_0 is the exploration parameter $(0 \leq q_0 \leq 1)$. Actually, q guides the ant to select the next flight leg. In other words, if $q \leq q_0$, the flight leg j in which its arc $cov(i,j)$ carries the greatest τ_{ij} and η_{ij} will be selected. On the contrary, if $q > q_0$, the flight leg j is selected using the following probability rule:

$$P_{ij}^k = \frac{\left[\tau_{ij}\right]^{\alpha}\left[\eta_{ij}\right]^{\beta}}{\sum_{j \in NFL_i^k}\left[\tau_{ij}\right]^{\alpha}\left[\eta_{ij}\right]^{\beta}} \qquad if\ j \in NFL_i^k \qquad (4.1)$$

where P_{ij}^{k} is the probability of selecting flight leg j to be covered after flight leg i using the same aircraft k.

- Visiting maintenance providers; this step is similar to the previous step, but here we select flight legs in which their destination airports have maintenance providers. This can be done using the following transition and probability rules:

$$j = \begin{cases} \arg\_max_{j \in NVM_i^k} \left\{ [\tau_{jm}]^{\alpha} [\eta_{jm}]^{\beta} \right\} & if\ q \leq q_0 \\ J & if\ q > q_0 \end{cases} \qquad (4.2)$$

$$P_{jm}^{k} = \frac{[\tau_{jm}]^{\alpha}[\eta_{jm}]^{\beta}}{\sum_{j \in NVM_i^k} [\tau_{jm}]^{\alpha}[\eta_{jm}]^{\beta}} \qquad if\ j \in NVM_i^k \qquad (4.3)$$

where NVM_i^k is similar to NFL_i^k, but the potential flight legs in this step offer maintenance providers in their destination airports. The terms τ_{jm} and η_{jm} are the pheromone trail and the heuristic function of the visiting maintenance arc $vma\ (j, m)$. The η_{jm} can be calculated as $1/P_m$. The term P_{jm}^{k} is the probability of selecting flight leg j that its destination airport offers maintenance provider m to perform the maintenance for aircraft k.

- Updating the pheromone trail; this step is conducted with the objective of reflecting the quality of the obtained solution. To do so, we use the following equations:

$$\tau_{ij,new} = (1-\rho)\tau_{ij,old} + \Delta\,\tau_{ij} \qquad (4.4)$$

$$\tau_{jm,new} = (1-\rho)\tau_{jm,old} + \Delta\,\tau_{jm} \qquad (4.5)$$

$$\Delta\,\tau_{jm} = \Delta\,\tau_{ij} = \frac{Q}{cost(A_{best})} \qquad if\{i,j\}\ and\ \{j,m\} \subseteq A_{best} \qquad (4.6)$$

Eq. (4.4) is used to update the pheromone of the coverage arcs, whereas the pheromone of the visiting maintenance arcs is updated using Eq. (4.5). Note that ρ is the evaporation rate parameter $(0 < \rho < 1)$. The concept of this step is that, at each iteration of the algorithm, the pheromone existing on the coverage and the maintenance arcs are eroded uniformly, so that the ants can ignore the bad paths and look for better paths in the next iterations. This erosion can be achieved by using the terms $(1-\rho)\tau_{ij,old}$ and $(1-\rho)\tau_{jm,old}$ to update the pheromone of the coverage arcs and the visiting maintenance arcs, respectively. To recognize the best solution found so far, the pheromone of the arcs that forms the best solution should be updated by depositing a quantity on these arcs. This can be achieved by the second terms of Eqs. (4.4) and (4.5), $\Delta\,\tau_{ij}$ and $\Delta\,\tau_{jm}$, which reflects the pheromone amount that will be deposited in the arcs. This amount is calculated using Eq. (4.6). It should be noted that Q is a pheromone depositing control factor, in which its value guides the algorithm whether to converge to the local optimal or to search randomly. The term $cost(A_{best})$ is the objective function of the best solution found so far.

5.1.2.Lower-level ACO-based algorithm

The lower-level ACO-based algorithm includes two main steps:

- Forming maintenance teams; similar to the first step of the previous ACO, this step is conducted through the ants that move throughout the layered graph. Indeed, each ant forms a path that starts from the starting node and ends at the ending nodes. Between the starting and ending nodes, each ant visits the layers sequentially and selects the appropriate team sizes. In other word, suppose an ant selects the team size while visiting a layer bv and intends to visit the next layer kv to select its team size. This can be achieved as in the previous ACO, by adoption of the following rules:

$$w = \begin{cases} \arg\_max_{w \in N_{kv}} \left\{ [\tau_{bvkvw}]^{\alpha'} [\eta_{bvkvw,worker}]^{\beta'} \right\} & if\ q \leq q_0' \\ W & if\ q > q_0' \end{cases} \quad (4.7)$$

$$P_{bvkvw} = \frac{[\tau_{bvkvw}]^{\alpha'} [\eta_{bvkvw,worker}]^{\beta'}}{\sum_{w \in N_f} [\tau_{bvkvw}]^{\alpha'} [\eta_{bvkvw,worker}]^{\beta'}} \qquad if\ w \in N_{kv} \quad (4.8)$$

where N_{kv} is the possible team sizes that could be formed for layer kv. The terms τ_{bvkvw} and $\eta_{bvkvw,worker}$ indicate the pheromone trail and heuristic function of the edge between layers bv and kv, whereas α' and β' reflect the relative importance of pheromone trail and heuristic function respectively. Here, $\eta_{bvkvw,worker}$ can be calculated as $1/(wf_{kvsm}^{\xi} * C_{wkvsm})$, and q_0' is the exploration parameter ($0 \leq q_0' \leq 1$).

- Updating the pheromone; by using the same concept as in the previous ACO, this step can be done in accordance to the following equations:

$$\tau_{bvkvw,new} = (1 - \rho')\tau_{bvkvw,old} + \Delta\,\tau_{bvkvw} \quad (4.9)$$

$$\Delta\,\tau_{bvkvw} = \frac{Q'}{cost(B_{wm,best})} \qquad if\ edge \subseteq B_{wm,best} \quad (4.10)$$

where ρ' is the evaporation rate factor of the lower-level ACO-based algorithm, whereas $B_{wm,best}$ reflects the objective function of the best solution found so far. The factor Q' is a pheromone depositing control factor of the lower-level ACO-based algorithm.

So far, we present the main steps for the bi-level ACO-based algorithm to solve both the FDARP and the MSP. However, it is not clear how the Stackelberg equilibrium can be calculated and achieved. Therefore, in the next sub-section, we explain how the Stackelberg equilibrium can be calculated during the implementation of the bi-level ACO-based algorithm.

5.1.3. Calculating the Stackelberg equilibrium

As mentioned, the LFSG includes two main players: the FDARP of the airline and the MSP of the maintenance providers. To simplify the explanation of the Stackelberg equilibrium, we let X_0 represent the decision variables (x_{ijkv}, y_{imkv}, z_{mjkv}, $ATBM_{kvm}$, and $RTAM_{kvm}^{*}$) taken by the FDARP of the airline, whereas $X_{m,Stack}$ denotes the decision variables (wf_{kvsm} and $RTAM_{kvm}$) used by the MSP of the maintenance providers. Based on the previous definitions, the response functions of the FDARP and the MSP can be defined as Eqs. (4.11), and (4.12), respectively. This means that the decision, X_0, of the FDARP is a function, $r_0(.)$, of the variable $X_{m,Stack}$ used by the MSP. Similarly, Eq. (4.12) indicates that the decision taken by the MSP is a function of the decision taken by the FDARP.

$$X_0 = r_0(X_{m,Stack}) \quad (4.11)$$

$$X_{m,stack} = r_m(X_0) \quad (4.12)$$

To achieve the Stackelberg equilibrium, the upper-level ACO based algorithm and the lower-level ACO-based algorithm are used in a dynamic reaction process with multiple iterative stages. Suppose that at a given iterative stage t, with the decision $X_{m,Stack}^{t}$ taken by the MSP and the decision X_0^t taken by the FDARP, the FDARP and the MSP make the responses as shown in Eqs. (4.13) and (4.14) to obtain their decisions at the iterative stage t+1.

$$X_0^{t+1} = r_0(X_{m,Stack}^{t}) \qquad (4.13)$$

$$X_{m,stack}^{t+1} = r_m(\, X_0^t) \qquad (4.14)$$

The Stackelberg equilibrium can be achieved if the following conditions exist and are satisfied [31, 63]:

$$\left\| X_0^{t+1} - r_0(X_{m,Stack}^{t}) \right\| == 0 \qquad (4.15)$$

$$\left\| X_{m,stack}^{t+1} - r_m(\, X_0^t) \right\| == 0 \qquad (4.16)$$

This means that both the FDARP and the MSP are unwilling to change their decisions because any change may have a negative impact on their objective functions. Therefore, the bi-level ACO-based algorithm is terminated.

5.2. Obtaining the Nash equilibrium

Achieving the Nash equilibrium necessitates solving the NG model expressed in Eqs. (3.0) – (3.2). In the NG model, the net profit is calculated as a continuous and differentiable function. Therefore, the NG model can be solved using the standard optimization approaches, such as partial differentiation with respect to prices [31, 35]. Note here that the discrete decision variables C_{wkvsm} and y_{imkv} are determined by the LFSG, and their values are used in the NG. Therefore, the profit function is still continuous and differentiable. Based on the previous observation, we can obtain the optimal decision on P_m for all the maintenance providers by using the following equation:

$$\frac{\partial NP_m}{\partial P_m} = 0 \qquad \forall\, m \in MT \qquad (5.0)$$

To simplify the calculation of the Nash equilibrium, we design $X_{m,Nash}$ to represent the decision variable P_m taken by maintenance provider m. For any maintenance provider m, the decision variables of all other maintenance providers can be expressed as $X_{-m,Nash}$. Based on the previous definitions, the response functions of maintenance provider m can be defined as:

$$X_{m,Nash} = r_m(X_{-m,Nash}) \qquad (5.1)$$

This means that the decision, X_m, taken by maintenance provider m is a function of the variable $X_{-m,Nash}$ taken by all other providers. To achieve the Nash equilibrium, the maintenance providers normally behave in a dynamic manner with multiple iterative stages. Suppose that at a given iterative stage t, with the decision $X_{-m,Nah}^{t}$ taken by all other maintenance providers, the maintenance provider m makes the response shown in Eq. (5.2) to obtain its decisions at the iterative stage t+1.

$$X_{m,Nash}^{t+1} = r_m(X_{-m,Stack}^{t}) \qquad (5.2)$$

The Nash equilibrium can be achieved if the following conditions exist and are satisfied [31, 63]:

$$\sum_{m\in MT}\left\|X_0^{t+1} - r_0(X_{m,Stack}^{t})\right\| == 0 \qquad (5.3)$$

This means that none of the maintenance providers are willing to change their pricing decisions because any change may result in a loss of profit.

5.3. Obtaining the overall Nash equilibrium

To obtain the overall Nash equilibrium, we need to use an algorithm that can simultaneously achieve both the Stackelberg equilibrium and the Nash equilibrium. For this purpose, we propose an iterative game algorithm that couples the bi-level ACO-based algorithm and the analytical method described in the previous sections. The detailed procedures of the iterative game algorithm are as follows:

<u>The iterative game algorithm</u>

Step 0: Initialize the parameter values of the bi-level ACO-based algorithm (i.e., $\alpha, \beta, q_0, \rho, Q, \alpha', \beta', q_0', \rho', Q'$, and the number of ants). Then, set a value for the maximum number of iterative stages, $t \in T$.

Step 1: Initialize the number of iterative stages $t = 1$.

Step 2: Determine the routing plan decisions by applying Steps 2.1 – 2.14.

- Step 2.1: Construct two lists such that the first one stores the aircraft (K) and the second one contains the flight legs (I). Note that each aircraft is represented by an ant.
- Step 2.2: Examine the status of the K list. In the case of a nonempty K list, select a single aircraft or ant from the K list to start its route construction, otherwise proceed to Step 2.13.
- Step 2.3: Using the I list, examine its status. In the case of a nonempty I list, go to Step 2.4, otherwise go to Step 2.13.
- Step 2.4: Initiate the route construction for the ant k by picking a random flight leg i from the I list.
- Step 2.5: By considering the constraints described by Eqs. (1.7) and (1.8), scan through the I list and identify the possible flight legs to be covered. If there are no more possible flight legs for coverage, go to Step 2.12, otherwise go to Step 2.6.
- Step 2.6: Using the state transition and probability rules described in Eqs. (4.0) and (4.1), pick out the next flight leg j.
- Step 2.7: Check whether the operational constraints expressed in Eqs. (1.15) – (1.17) are violated after selecting the flight leg j. In the case of violation, proceed to Step 2.8, otherwise go to Step 2.11.
- Step 2.8: Scan through the I list to identify the possible flight legs in which the destination airports offer maintenance providers. This can be done by using constraints in Eqs. (1.9) and (1.10). If there are no more possible choices for coverage, go to Step 2.12, otherwise go to Step 2.9.
- Step 2.9: Using the state transition and probability rules described in Eqs. (4.2) and (4.3), pick out the next flight leg j such that the ant k covers that flight and then receives the maintenance operation. Note that when $t = 1$, the P_m included in this step can be assumed by the FDARP of the airline, but when $t > 1$, the P_m is determined based on the NG prices, as calculated in Step 4.
- Step 2.10: After completing the maintenance operation, the ant should resume covering the flight legs by following the constraints described in Eqs. (1.5), (1.6), and (1.11) – (1.14). Note that when $t = 1$, the departure time for the aircraft from the maintenance station is assumed

by the FDARP of the airline through constraints (1.13), but when $t > 1$, this time is determined by the MSP of the maintenance providers, as stored in Step 3.14.

Step 2.11: Add the chosen flight leg to the route of ant k, then exclude that flight leg from the I list and proceed to Step 2.5.

Step 2.12: Terminate the route for ant k by following the constraints stated in Eq. (1.4). Next, proceed to Step 2.2, after excluding the ant or aircraft k from the K list.

Step 2.13: Conduct the pheromone trail updating process for the coverage and the visiting maintenance arcs, by following Eqs. (4.4) – (4.6).

Step 2.14: For the existing iterative stage t, store the routing plan decisions in X_0^t. Next, calculate the solution of this stage and update the best solution found so far.

Step 3: Determine the staffing plan decisions by applying Steps 3.1 – 3.15.

Step 3.1: Construct a list called (MT), which includes all the maintenance providers.

Step 3.2: Examine the status of the MT list. In the case of an empty MT list, go to Step 3.15, otherwise pick a random maintenance provider m and go to Step 3.3.

Step 3.3: Based on the decisions stored in X_0, extract the aircraft that will be maintained by the selected maintenance provider and store them in a list named ($K\Psi$).

Step 3.4: Prepare a list called ANT to store the ants designed to construct the staffing plan.

Step 3.5: Using the ANT list, examine its status. In the case of an empty ANT list, proceed to Step 3.14, otherwise randomly pick an ant from the ANT list and put it in the starting node of the graph as its current position.

Step 3.6: Check the condition of the $K\Psi$ list. If all the aircraft stored in the $K\Psi$ list are visited by the selected ant, go to Step 3.11, otherwise proceed to Step 3.7.

Step 3.7: By using the $K\Psi$ list, select the first unvisited aircraft kv from the list and make it the next position to be covered by the ant.

Step 3.8: For the selected aircraft, determine its scheduled arrival and departure times by following the constraints stated in Eqs. (2.3) and (2.4). Next, determine the possible team sizes while considering the constraints described by Eqs. (2.2) and (2.5).

Step 3.9: Using the state transition and probability rules described in Eqs. (4.7) and (4.8), select the team size required to maintain the selected aircraft.

Step 3.10: Change the status of aircraft kv to a visited aircraft in the $K\Psi$ list. Then, let the ant move to layer kv as its current position and go to Step 3.6.

Step 3.11: Terminate the role of the selected ant by putting it in the ending node of the layered graph.

Step 3.12: Conduct the pheromone trail updating process by following Eqs. (4.9) and (4.10).

Step 3.13: For all the aircraft stored in the $K\Psi$ list, change their status to unvisited and go to Step 3.5.

Step 3.14: For the existing iterative stage t, store the staffing plan decisions of maintenance provider m in $X_{m,Stack}^t$ and go to Step 3.2.

Step 3.15: For the existing iterative stage t, calculate the solution and update the best solution found so far.

Step 4: Determine the maintenance service price P_m for each maintenance provider by applying Eq. (5.0)

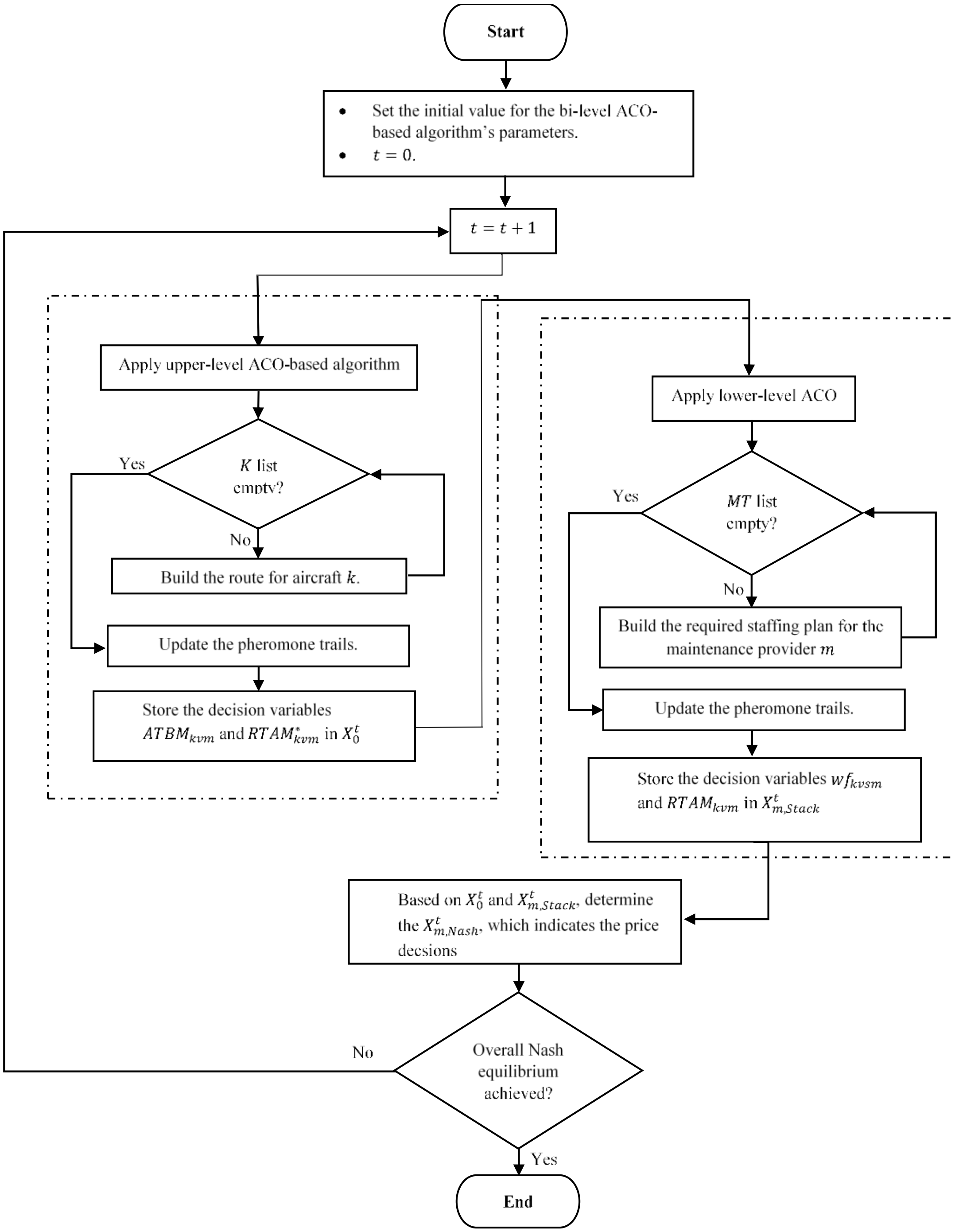
Step 5: Using Eqs. (4.15), (4.16), and (5.2), check whether the Stackelberg equilibrium and the Nash equilibrium are achieved. If both equilibriums are achieved, go to Step 6, otherwise increment the iterative stage and go to Step 2.

Step 6: Because the Stackelberg equilibrium and the Nash equilibrium are achieved, the overall Nash equilibrium is obtained. Then, terminate the algorithm.

Figure 4 shows the flowchart of the iterative game algorithm. The upper-part of the figure shows the bi-level ACO-based algorithm, which is used to handle the LFSG, whereas the lower-part of the figure illustrates the analytical method used to solve the NG. For the sake of computational convenience, we set the maximum number of iterative stages to be 500. The description of the iterative game algorithm shows that the existence of the overall Nash equilibrium is mainly dependent on two main equilibriums; the Stackelberg equilibrium and the Nash equilibrium. To get the Stackelberg equilibrium, it is necessary to find out feasible solutions for the FDARP and the MSP, then reach the convergence points, in which no player intends to change the taken decisions. To get the Nash equilibrium, it can be achieved by reaching the points, in which no player wants to change the pricing decision.

As mentioned, finding out the feasible solutions for the FDARP and the MSP is first step to get the Stackelberg equilibrium. Imagine, if the maximum number of iterations is completed, and the feasible solutions cannot be achieved. This means that there is no feasible routing plan to meet the requirement of the staffing plan or vice versa. In this situation, the iterative game algorithm should be re-run for maximum 100 runs as recommended by Yu and Huang [31], while setting the most sensitive factor, MAT, to be around 6 and 8 hours. If the maximum number of runs is completed before finding out the feasible solutions, which was very rare during our preliminary experiments, the Stackelberg equilibrium cannot be got. Consequently, the overall Nash equilibrium cannot be derived. As mentioned earlier, reaching the convergence point is the second step after finding out the feasible solutions. Imagine, if the maximum number of iterations is completed while the feasible solutions are found out before reaching the convergence points. This necessitates re-running the iterative game algorithm while using the same conditions the first run with the same maximum number of runs that is mentioned earlier. In case of completing the maximum number of runs before reaching the convergence points, which was also rare in our preliminary experiments, the Stackelberg equilibrium cannot be got. Consequently, the overall Nash equilibrium cannot be derived.

Lastly, to find the Nash equilibrium, we need to reach the point in which all players do not change their decisions. To do so, we need feasible solutions for the FDARP and the MSP, as they contribute in the calculation of the pricing decisions. If the maximum number of iterations is completed and the feasible solutions are achieved before getting the Nash equilibrium, the iterative game algorithm should be re-run while using the same conditions the first run with the same maximum number of runs that is mentioned earlier. When the maximum number of runs is completed before getting the Nash equilibrium, which was again rare in our preliminary experiments, the Nash equilibrium cannot be got. Consequently, the overall Nash equilibrium cannot be derived.



6. Data analytics for non-propagated delay forecasting

Figure 4: Flowchart of the iterative game algorithm.

Using the iterative game algorithm to solve the proposed the SNGM is insufficient, because it misses how to calculate the NPD. Instead of the expected value approach, which focuses on the historical data, we use a data analytics technique that is able to consider massive amounts of information to forecast the NDP. This approach includes developing a neural network-based algorithm to capture the nonlinear relationship among the various factors that affect the NPD. The main steps of this algorithm are as follows.

The neural network-based algorithm

Step a: Data collection. The data are collected from a major airline in the Middle East. The data include certain features such as the flight number, departure airport, arrival airport, arrival time, departure time, flight duration, the NPD for each flight, and other factors.

Step b: Data preprocessing. For the collected data, a flight is considered delayed even if its related NPD time is less than 15 minutes, because any NPD time can easily cause a propagated delay in practice. Moreover, NPDs longer than 170 minutes are discarded, because this indicates a severe disruption, which is beyond the scope of our study [64].

Step c: Define the input sets. These sets include historical information and other factors that affect the NPD, and can be summarized as follows.

i. Set 1: flight number, departure airport, departure time, arrival airport, arrival time, visited maintenance station, day of operation, and flight duration.

ii. Set 2: bad weather indicator. It is known that NPDs frequently occur during bad weather. Because it is difficult to predict the time of bad weather, a 3-point scale indicator is proposed, in which the values of 1, 2, and 3 indicate low chance, medium chance, and high chance of bad weather occurrence, respectively.

iii. Set 3: maintenance station congestion indicator. An NPD can be caused by a delay in the maintenance station in cases of congestion. To capture this situation, we use a 4-point scale indicator, with values of 1, 2, 3, and 4, which indicate below 30% station utilization, 30%-60% station utilization, 60%-80% station utilization, and over 80% station utilization, respectively.

iv. Set 4: season indicator. It is known that NPDs frequently occur during holiday seasons, such as Christmas and summer vacations. Accordingly, a 3-point scale indicator is used, in which the values of 1, 2, and 3 indicate a normal day, one week before or after the holiday season, and the holiday season, respectively.

Step d: Design the structure of the neural network. We use a multilayer feed-forward neural network because this structure is commonly adopted. The network consists of an input layer, hidden layer, and output layer. For the activation function, we use the sigmoid function, due to its efficiency in capturing the non-linear relationships between different factors.

Step e: Train the neural network. To achieve this, we use the supervised learning method, in which 70% of the data is used for training and the rest is used for validation.

7. Case study

7.1. Problem context

After proposing the SNGM to capture the coordination between the FDARP of the airline and the MSP of the maintenance providers, and the competition among the maintenance providers, it is necessary to demonstrate the effectiveness of the proposed model as a decision tool for airlines and maintenance providers. For this purpose, a case study based on real data acquired from a major airline and four maintenance providers located in the Middle East is presented. Note that the case selected from the airline represents the fleet with the longest average PD. This is to assess the performance of the proposed model to minimize the PD. The details of the collected data are presented in Table 1. The proposed algorithm and model were coded in MATLAB R2014a, and tested on an Intel i7 CPU processor with 2.50 GHz CPU clock speed and an 8 GB RAM laptop running Windows 10.

Table 1: Characteristics of the collected data

Airline	
I	320 flight legs
FS	36 aircraft
A	8 airports
C_{max}	10 take-offs
T_{max}	40 hours
MT	4 maintenance providers
TRT	45 minutes
MAT	8 hours
C_{pD}	$C_{pD} = \begin{cases} 750 & PD_{ijkv}^{\xi} \leq 15\ minutes \\ 1250 & PD_{ijkv}^{\xi} > 16\ minutes \end{cases}$

	Maintenance provider 1	Maintenance provider 2	Maintenance provider 3	Maintenance provider 4
	• l_{kv}=50 hrs	• l_{kv}=50 hrs	• l_{kv}=50 hrs	• l_{kv}=50 hrs
	Morning shift (8 hrs)	Morning shift (8 hrs)	Morning shift (12 hrs)	Morning shift (12 hrs)
w_{Sm}^{l}	8 workers	6 workers	8 workers	7 workers
w_{sm}^{u}	15 workers	12 workers	15 workers	12 workers
Q_{s}^{max}	150 workers	120 workers	100 workers	80 workers
	Afternoon shift (8 hrs)	Afternoon shift (8 hrs)	Night shift (12 hrs)	Night shift (12 hrs)
w_{Sm}^{l}	5 workers	4 workers	5 workers	5 workers
w_{sm}^{u}	10 workers	8 workers	10 workers	10 workers
Q_{s}^{max}	100 workers	80 workers	100 workers	80 workers
	Night shift (8 hrs)	Night shift (8 hrs)		
w_{Sm}^{l}	2 workers	2 workers		
w_{sm}^{u}	5 workers	4 workers		
Q_{s}^{max}	50 workers	40 workers		

7.2. Non-propagated delay forecasting

The data presented in the previous section are insufficient to conduct the experiments because they miss the NPD. To obtain the NDP, the proposed neural network-based algorithm is applied. For this purpose, we collected the information for all of the flights recorded by the airline from January 2017 to December 2017, including the flight number, departure airport, departure time, arrival airport, arrival time, visited maintenance station, day of operation, and flight duration. The data comprise 292,000 flights flown by 12 fleets. After analyzing the data, the top fleet with the longest average PD was selected to test the capability of the proposed model in minimizing the PD and test the potential of the proposed neural network-based algorithm for forecasting accurate NPDs. The features of the selected fleet are summarized in Table 2. Note that the selected fleet in this section and the previous section are same.

Table 2: Features of the selected fleet.

Fleet	**Total flights**	**Delayed flights**		**NPD (minutes)**		**PD (minutes)**	
		No.	%	Total	Average	Total	Average
A300	29,440	6,597	22.41	333,849	11.34	287,628	9.77

7.3. Data analytics for predicting the demand-price function for the maintenance providers

The experiments necessitate predicting the demand-price function for the maintenance providers. For this purpose, data analytics in the form of a regression is adopted, because this is one of the most efficient ways to capture the relationship between a response variable and one or multiple predictors. Because the demand for a maintenance provider is a function of its own price and the prices offered by all other providers, the demand is a function of multiple predictors. Therefore, the multiple linear regression algorithm, as a part of the Minitab standard tools, is reasonable to be used in this study. Note that this regression algorithm is based on real data collected from the maintenance providers for the January 2017 to December 2017 period.

The multiple linear regression algorithm is used to obtain the demand-price function for each maintenance provider, as shown in Table 3. Two indicators are used to assess the quality of the obtained relationship. The first indicator is the R-squared, which indicates how well the obtained model fits the collected data. The results in Table 3 show that the R-squared indicators for all of the functions are larger than 90%, indicating that the regression model fits the collected data very well. The second indicator is the p-value, which indicates the relationship between the response variable and the predictors. If the obtained p-value is larger than the selected significance level, there is no significant relationship between the response variable and the predictors. The p-values presented in Table 3 indicate that there is a significant relationship between the demand and the prices because the p-values are smaller than the significance level, which is 5% in this study.

Table 3: Regression analysis between the demand and the related prices.

Maintenance provider	**Regression fitted line** $\boldsymbol{De_m = \theta_m - \vartheta_m P_m + \sum_{g \in MT} \delta_{mg} P_g}$	**R-squared**	**Predictor variable**	$\boldsymbol{p}$**-value**
1	De_1= 35 - 0.00428 P_1+ 0.00117 P_2+ 0.00150 P_3 + 0.000842 P_4	98.7%	θ_1	0.001
			P_1	0.000
			P_2	0.000
			P_3	0.000
			P_4	0.000
2	De_2= 30 - 0.00296 P_2+ 0.00105 P_1+ 0.00097 P_3 +0.000604 P_4	94.5%	θ_2	0.000
			P_2	0.000
			P_1	0.000
			P_3	0.000
			P_4	0.000
3	De_3= 32 - 0.00348 P_3+ 0.00121 P_1+ 0.00120 P_2 +0.000996 P_4	95.4%	θ_3	0.000
			P_3	0.000
			P_1	0.000
			P_2	0.000
			P_4	0.000
4	De_4= 27 - 0.00235 P_4+ 0.000900 P_1+ 0.00049 P_2 +0.00059 P_3	91.1%	θ_4	0.000
			P_4	0.000
			P_1	0.000
			P_2	0.000
			P_3	0.000

7.4. Results of the Stackelberg-Nash model

In this section, the results obtained by solving the Stackelberg-Nash model are reported using the iterative game algorithm. The values of the parameters of the iterative game algorithm first need to be set. For the sake of computational convenience, the iterative game algorithm uses the following values: α=1, β=2, q_0=0.95, ρ=0.05, Q=0.01, α'=2, β'=2, q'_0=0.85, ρ'=0.05, Q'=0.01, ant size for upper-level ACO=fleet size, and ant size for lower-level ACO=number of flights in which their aircraft are maintained.-The results of the iterative game algorithm are provided in Figures 5 and 6. Figure 5 shows the results of the LFSG that acts as a coordinated system between the FDARP of the airline and the MSP of the maintenance providers. Figure 5 indicates that after 450 iterative stages, the algorithm reaches the convergence point, meaning that none of the players are willing to change their decisions, resulting in overall Nash equilibrium values of 477,295 for the airline and 207,520 for the maintenance providers. Figure 6 shows the results of the NG among the maintenance providers, including the prices and net profits achieved at the overall Nash equilibrium.

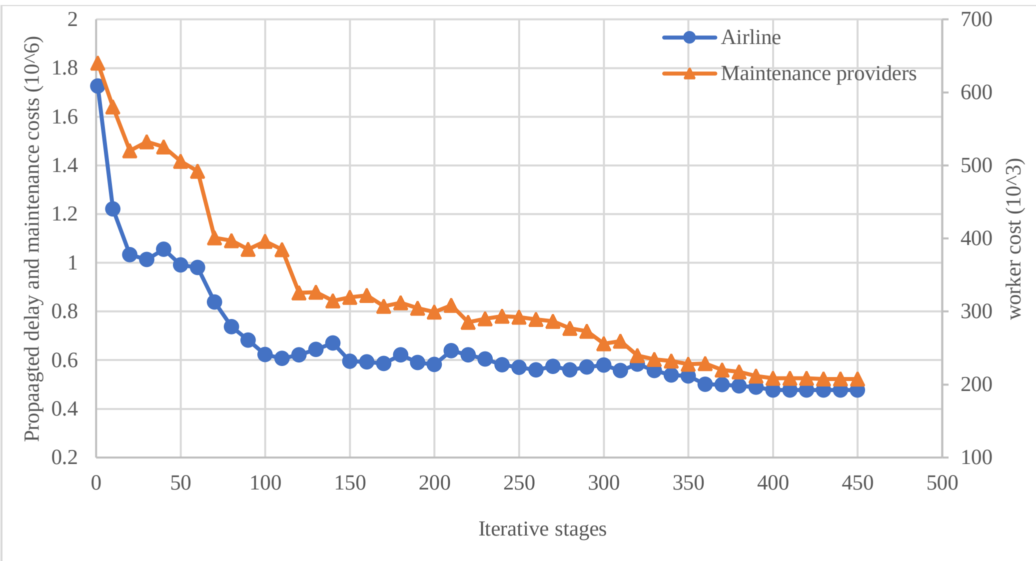


Figure 5: Convergence of the iterative game algorithm

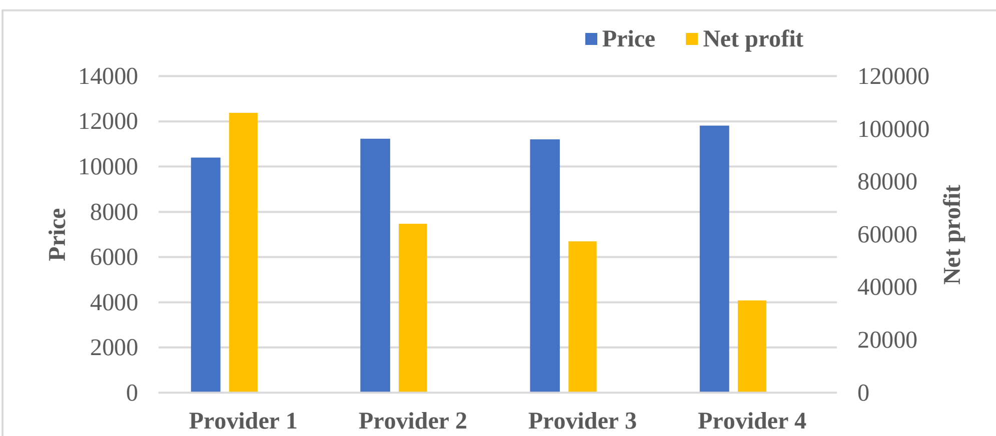


Figure 6: Price and net profit for the maintenance providers.

As mentioned, the proposed model consists of two sub-games: the LFSG and the NG. To determine the overall performance of the proposed model, we need to address the following questions: “what is the role and importance of each game of the model," "what determines the model performance," and "is it due to considering the coordination using the LFSG or to considering the competition through the NG?" To answer these questions, we extend our experiments to compare different settings in the proposed model, as shown in Sections 7.5 and 7.6.

7.5. Importance of the NG

To determine the importance of the NG, which captures the competition among the maintenance providers, two cases are compared: considering the competition and neglecting the competition. The first case can be represented using the model proposed in the previous section, which considers both the coordination through the LFSG and the competition through the NG. The second case can be represented by a model that only considers the coordination through the LFSG, so that the competition is neglected. The second model can be captured by Eqs. (1.0) – (2.6) and solved by the bi-level ACO-based algorithm presented in Section 5.1. This bi-level ACO-based algorithm can be implemented by applying all the steps of the iterative game algorithm, while neglecting step 4, the Nash equilibrium consideration in step 5, and any prices set by the NG throughout the algorithm. The results of the bi-level ACO-based algorithm are summarized as follows. The algorithm converges after 500 iterative stages and returns Stackelberg equilibrium values of 570,381 for the airline and 207,250 for the maintenance providers.

The performance of the two models in handling the airline and maintenance providers' costs is shown in Table 4. The results show that the first model outperforms the second by about 16.32% in handling the airline costs, especially the maintenance costs. The first model outperforms the second because it considers the competition among the maintenance providers, which includes cutting the maintenance service prices, and thus enables the airline to identify the cheaper providers. This results in a reduction in the cost paid by the airline, as in the first model. In contrast to the first model, the second model neglects the competition, and the airline thus loses the opportunity to trace the cheaper prices due to competition, resulting in higher maintenance costs. The labor costs of the maintenance providers are mainly affected by the coordination between the airline and the maintenance providers, which includes adjusting the staffing plan decisions until the Stackelberg equilibrium is reached. Because the two models consider the coordination through the LFSG, there is no expected change in the labor cost.

Table 4: The performance of the first and second models while handling the airline and maintenance providers' costs

Costs	First model (Coordination + Competition)	Second model (Only Coordination)	Outperformance (%)
Propagated delay and maintenance costs by airline	477,295	570,381	16.32
Labor cost by maintenance providers	207,500	207,500	0

Table 5 shows the results of the two models while handling the net profit of the maintenance providers. It can be seen from Table 5 that three out of four providers achieve better profits while using the first model. This improvement is due to the competition, which includes the process of cutting the price of the maintenance service. Indeed, cutting the price leads to more demand from the airline, and thus results in increased net profit. Of course, not all the providers increase their profits due to the competition, because some of them cannot cut their prices due to certain financial obligations. This is why the last provider suffers from the competition effect, with the profit decreased by around 39%.

Table 5: The performance of the first and second models while handling the net profit of maintenance providers.

Maintenance provider	First model (Coordination + Competition)	Second model (Only Coordination)	Improvement (%)
Provider 1	106,100	56,982	86.19
Provider 2	64,082	62,162	3.08
Provider 3	57,431	46,622	23.18
Provider 4	34,931	56,982	-38.69

To summarize, considering the competition through the NG is fruitful for the airline because it leads to reduced maintenance costs. The competition is also useful for the majority of the providers because it helps attract more demand from the airline, resulting in increased net profit.

7.6. Importance of the LFSG

Similar to the previous section, two cases are compared in this section: considering the coordination between the airline and maintenance providers, and neglecting this coordination. The first case can be captured by the first model proposed in the previous section. The second case can be represented by a model that only considers the competition through the NG, while neglecting the coordination, meaning a separate FDARP of the airline and MSP of the maintenance the providers. We call the model for this second case the third model, in which the FDARP of the airline can be represented using Eqs. (1.0) – (1.22), while neglecting the linkage constraints expressed in Eq. (1.14). The MSP of the maintenance providers of the third model can be represented using Eqs. (2.0) – (2.6), while redesigning the constraints in (2.1) as $RTAM_{kvm} = SDT_{kvm}$. Finally, the competition captured by the third model can be represented using Eqs. (3.0) – (3.2). This third model can be solved as follows. First, the FDARP and the MSP can be solved using the upper and lower-level ACO-based algorithms, respectively. These algorithms are expressed in steps 2 and 3 of the iterative game algorithm. Second, the competition part can be handled using the analytical method explained in Section 5.2. The results of the upper and lower-level ACO-based algorithms are as follows. The upper-level algorithm converges after 470 iterations and returns its best value of 526,815 for the airline, whereas the lower-level algorithm converges after 500 iterations and achieves its best value of 265,705 for the maintenance providers.

Table 6 summarizes the performance results of the first and third models while handling the airline and maintenance providers' costs. The results show that the first model outperforms the third model by about 9.40% and 22% while handling the airline and maintenance providers' costs, respectively. This outperformance arises because considering the coordination between the airline and the maintenance providers enables both players to keep adjusting their routing and staffing decisions to improve their results. This finally leads to reductions in the PD cost and the labor costs paid by the airline and the maintenance providers, respectively. In contrast to the first model, the third model neglects the coordination, and thus the airline and maintenance providers lose the opportunity to adjust their routing and staffing decisions, leading finally to higher costs for the airline and the maintenance providers.

Table 6: The performance of the first and third models while handling the airline and maintenance providers costs

Costs	First model (Coordination + Competition)	Third model (Only competition)	Outperformance (%)
Propagated delay and maintenance costs by airline	477,295	526,815	9.40
Labor cost by maintenance providers	207,500	265,705	22

The results of the two models in handling the net profit of the maintenance providers are reported in Table 7. The table shows that all the providers enjoy better profits using the first model instead of the third model, which neglects the coordination. This is mainly due to the coordination, which helps the maintenance providers to minimize the labor costs and results in increased net profits.

Table 7: The performance of the first and third models while handling the net profit of maintenance providers.

Maintenance provider	First model (Coordination + Competition)	Third model (Only competition)	Improvement (%)
Provider 1	106,100	80,271	32.17
Provider 2	64,082	50,488	26.92
Provider 3	57,431	45,196	27.07
Provider 4	34,931	28,134	24.15

In conclusion, considering the coordination between the airline and the maintenance providers through the LFSG is important for the airline because it leads to reduced PD costs. The maintenance providers can also benefit from the coordination because it helps to minimize their labor costs, resulting in increased net profits.

7.7. Performance analysis

The performance of the SNGM, as presented in the previous sections, is not sufficient to demonstrate its importance and superiority over the models in the literature. Therefore, we extend our experiments to compare the performance of the proposed model with that of the traditional models, which do not consider the coordination and competition. We name this traditional approach the fourth model. Our proposed model is the same as the first model presented in the previous two sections, whereas the fourth model is similar to the third model presented in Section 7.6, except it neglects the competition captured by the NG. The results of the upper and lower-level ACO-based algorithms can be summarized as follows. In fact, both algorithms converge after 300 iterations and return their best values of 645,866 for the airline and 265,705 for the maintenance providers.

Table 8 compares the performance of the first and fourth models. The table shows that the first model outperforms the fourth model by about 26.10 % and 22% while handling the airline and maintenance providers' costs, respectively. The first model performs better because it considers the coordination and competition games. The coordination between the airline and maintenance providers helps both players to keep adjusting their routing and staffing decisions, so that the airline can achieve a lower PD and the maintenance providers can reduce their labor costs. The competition game enables the airline to select the cheapest maintenance provider, resulting in reduced maintenance costs. In contrast to the first model, the fourth model neglects both the coordination and the competition, and the airline and maintenance providers lose these opportunities for reducing costs, resulting in higher incurred costs for the airline and maintenance providers.

Table 8: The performance of the first and fourth models while handling the airline and maintenance providers costs

Costs	First model (Coordination + Competition)	Fourth model (No Coordination + No Competition)	Outperformance (%)
Propagated delay and maintenance costs by airline	477,295	645,866	26.10
Labor cost by maintenance providers	207,500	265,705	22

Table 9 reports the results of the two models while handling the net profit of the maintenance providers. The table shows that three out of four providers enjoy better profits while adopting the first model. The providers gain increased profits because the model considers the coordination, which leads to reduced labor costs, whereas taking the competition into account leads to increased demand. These two factors finally lead to the improved net profit of the providers. As aforementioned, some providers cannot reduce their prices due to certain financial restrictions and thus cannot attract more demand, leading to a reduction in their net profit, as shown in the case of Provider 4.

Table 9: The performance of the first and fourth models while handling the net profit of maintenance providers.

Maintenance provider	First model (Coordination + Competition)	Fourth model (No Coordination + No Competition)	Improvement (%)
Provider 1	106,100	42,054	152.29
Provider 2	64,082	45,849	39.76
Provider 3	57,431	34,387	67.01
Provider 4	34,931	42,028	-16.88

In conclusion, the modeling of the coordination between the airline and maintenance providers through the LFSG and the competition among the maintenance providers through the NG is important for airlines and maintenance providers. The airlines enjoy lower PD costs owing to the coordination and lower maintenance costs due to the competition. Moreover, the maintenance providers achieve lower labor costs due to the coordination, whereas the net profits of the majority of providers improve while considering the competition.

In this study, we propose a neural network-based algorithm to improve the forecasting of the NPD. To demonstrate the importance of this algorithm, we extend our experiments to compare the performance of the proposed algorithm with that of the expected value approach. In the experiments, we use the two methods to forecast the NPD for the collected data and then use the forecasted NPD to solve the proposed SNGM. The results are summarized in Table 10. The results show that the neural network-based algorithm is more accurate than the other method because it considers more factors that affect the NPD, including bad weather, the holiday seasons, and the maintenance station congestion factors. In addition, the neural network-based algorithm outperforms the other method by about 7.82% while handling the airline costs. Specifically, the expected value approach underestimates the NPD and, because the delay is easily propagated, this results in a higher PD cost, which in turn leads to increased airline costs. In contrast, the neural network-based algorithm provides an accurate NPD, which mitigates the problem of delay propagation. As a result, the related costs are minimized, thus resulting in reduced airline costs.

Table 10: Results obtained by the different forecasting methods

Output	Neural network-based algorithm	Expected value approach	Improvement (%)
Root mean square error (RMSE)	9.208	27.26	66.22
Airline cost	477,295	517,785	7.82

7.8. Performance of the iterative game algorithm

In the previous sections, the performance of the SNGM is presented by using the iterative game algorithm. Indeed, solving our case study that includes a single test instance is not enough to demonstrate the efficiency of the iterative game algorithm, however, by using this case study, we are able to compare the performance of the SNGM with that of the traditional models, which do not consider the coordination and competition. Therefore, to assess the scalability and applicability of the SNGM, we extend the computational experiments by using additional test instances that are characterized by different sizes. In particular, the experiments include using another three test instances for FDARP and the same MSP test instance presented in our case study. It is important to note here that, the source of the new FDARP test instances is the same airline that delivers the data for our case study. Table 11 presents more information about the test instances. For all test instances, we assumed according to the airline that T_{max} is 40 hours, MAT is 8 hours, TRT is 45 minutes, and C_{pD} is the same as presented in Table 1.

Before conducting our experiments, it is noteworthy that assessing the performance of any solution algorithm can be usually done using two criteria; the optimality gap and the computational time. Using the optimality gap is not sound in this study, as the SNGM and its iterative game algorithm seek for the equilibrium, not for optimal solution. Based on this observation, the computational time is used as a criterion for assessing the performance of the algorithm.

Table 11: Characteristics of test cases

Test case	**Airline (FDARP)**					**Maintenance providers (MSP)**
	I	FS	C_{max}	A	MT	
Case 1	160	11	7	5	4	The information is same as Table 1
Case 2	240	30	10	13	4	
Case 3 (our case study)	320	36	10	8	4	
Case 4	400	42	10	28	4	

Table 12 summarizes the results obtained from the iterative game algorithm while solving the test instances with different sizes. The results of the LFSG, in terms of Stackelberg equilibrium, are reported in the first two columns of Table 12. In addition, the results of the NG, in terms of Nash equilibrium, are represented in columns 3 to 6 of Table 12. These two equilibriums constitute the overall Nash equilibrium. The last column of Table 12, CPU (min), records the computational time taken by the algorithm to reach the overall Nash equilibrium. Note that this time is recorded as obtained by the internal calculation function of MATLAB.

Looking at Table 12, a striking observation is that, the iterative game algorithm can provide reasonable performance, in terms of computational time. For instance, it takes around 8 minutes to solve the smallest case, case 1, whereas the largest case, case 4, takes around 33 minutes to be solved. This computational time performance is acceptable in practice; therefore, the algorithm can be implemented in real industry.

Table 12: Performance characteristics of the iterative game algorithm.

Test case	**LFSG**		**NG**				**Computational time** $CPU\ (min)$
	Airline costs	Maintenance providers costs	Profit of Maintenance provider 1	Profit of Maintenance provider 2	Profit of Maintenance provider 3	Profit of Maintenance provider 4	
Case 1	186,156	127,000	20,533	15,029.64706	14,921	8,670	7.78
Case 2	329,294	224,700	32,054	29,904	29,688	12,948	14.32
Case 3	477,295	207,000	106,100	64,082	57,431	34,931	19.45
Case 4	500,840	372,000	46,340	37,692	37,368	7,438	32.63

7.9. Managerial implications

Our findings have a number of managerial implications, which are outlined below:

- Airlines can benefit from the leader-follower Stackelberg game by obtaining lower propagated delay costs due to the dominant position and being the leader in the game. The maintenance providers mainly benefit from the leader-follower Stackelberg game in terms of having lower labor costs, but with less advantage if compared with airline, owing to their subordinate position as the followers in the game.
- The competition among the maintenance providers captured by the Nash game should favor the airlines because it provides opportunities to select cheaper providers, resulting in lower maintenance costs for the airline. Moreover, this game also benefits the majority of the providers because it helps them attract more demand from the airlines, and thus increases their net profit. However, this game does not help the providers that cannot reduce their prices because they suffer from lower demand and lower net profit.
- The leader-follower Stackelberg game is more beneficial to the maintenance providers than the Nash game because all the providers enjoy lower labor costs owing to the leader-follower Stackelberg game, as shown in Table 7, whereas some providers suffer lower profits due to the Nash game, as shown in Table 5.
- The change in the service prices of the maintenance providers affects the decisions of the airlines. This is apparent because the airlines enjoy an approximately 8% reduction in maintenance costs with improved routing plans. This change has a significant impact on the net profits of the maintenance providers, as demonstrated by the significant increase in the net profits of the first and third providers in the study. These results are shown in Table 5.
- The iterative game algorithm can find the overall Nash equilibrium for the model within at maximum 33 minutes. This computational time is acceptable in practice, and therefore, the algorithm can be implemented in practice.
- Data analytics is an important tool for airlines because it can consider massive amounts of information, which in turn results in accurate non-propagated delay forecasting.

8. Conclusions and future directions

In this study, we discuss how airline and maintenance providers interact to maximize their own profits. This is captured by our proposed SNGM, which consists of the LFSG between the airline and the maintenance providers and the NG between the maintenance providers. An iterative game algorithm is developed to determine the overall Nash equilibrium for the proposed model.

To verify the superior performance of the proposed model, we present a case study based on real data acquired from a major Middle Eastern airline and four maintenance providers located in the same region. The case study requires forecasting the NPD for the airline and the demand-price function for each maintenance provider. To achieve this, we develop a neural network-based algorithm to forecast an accurate NPD based on a one-year dataset that contains information on the historical flight delays and other external factors, such as bad weather and maintenance station congestion. We also use a data analytics tool, called the multiple linear regression algorithm, to predict the relationship between the demand and price for each maintenance provider. The results reveal significant savings for the airline and the maintenance providers owing to the LFSG, whereas the NG improves the net profits for the majority of the maintenance providers.

Although this study presents a formulation of a unique problem in the literature, there are some limitations that could be addressed in future research. First, the scope of the proposed FDARP is limited to a 4-day planning horizon, thus it would be interesting to solve the FDARP with a weekly planning horizon in which the size of the problem increases significantly. Second, we assume the workforce capacity for the MSP is deterministic. Another research direction would be to solve this model with a stochastic workforce capacity. In addition, the proposed model considers a single airline and multiple maintenance providers. Thus, it would be beneficial to extend this model to consider multiple airlines and multiple maintenance providers. Finally, in this study, the proposed game is limited to the price competition, thus it would be a fruitful idea to involve the tendering (auction mechanism) in the scope of the game.

Acknowledgements

The work described in this paper was supported by grants from the Research Grants Council of the Hong Kong Special Administrative Region, China (Project No. PolyU 15201414); The Natural Science Foundation of China (Grant No. 71471158); The Research Committee of Hong Kong Polytechnic University (Project Numbers G-YBFD; G-YBN1); and The Hong Kong Polytechnic University under student account code RTYN.

References

[1] K. T. Talluri, "The Four-Day Aircraft Maintenance Routing Problem," *Transportation Science,* vol. 32, pp. 43-53, 1998.
[2] Z. Liang, W. A. Chaovalitwongse, H. C. Huang, and E. L. Johnson, "On a New Rotation Tour Network Model for Aircraft Maintenance Routing Problem," *Transportation Science,* vol. 45, pp. 109-120, 2011.
[3] A. Sarac, R. Batta, and C. M. Rump, "A branch-and-price approach for operational aircraft maintenance routing," *European Journal of Operational Research,* vol. 175, pp. 1850-1869, 2006.
[4] M. Başdere and Ü. Bilge, "Operational aircraft maintenance routing problem with remaining time consideration," *European Journal of Operational Research*, vol. 235, pp. 315-328, 2014.
[5] Z. Liang, Y. Feng, X. Zhang, T. Wu, and W. A. Chaovalitwongse, "Robust weekly aircraft maintenance routing problem and the extension to the tail assignment problem," *Transportation Research Part B: Methodological*, vol. 78, pp. 238-259, 2015.
[6] N. M. Kabbani and B. W. Patty, "Aircraft routing at American airlines," in *In Proceedings of the 32nd annual symposium of AGIFORS*, Budapest, Hungary, 1992.

[7] R. Gopalan and K. T. Talluri, "The Aircraft Maintenance Routing Problem," *Operations Research,* vol. 46, pp. 260-271, 1998.
[8] L. Clarke, E. Johnson, G. Nemhauser, and Z. Zhu, "The aircraft rotation problem," *Annals of Operations Research,* vol. 69, pp. 33-46, 1997.
[9] C. Sriram and A. Haghani, "An optimization model for aircraft maintenance scheduling and re-assignment," *Transportation Research Part A: Policy and Practice,* vol. 37, pp. 29-48, 2003.
[10] M. Haouari, S. Shao, and H. D. Sherali, "A Lifted Compact Formulation for the Daily Aircraft Maintenance Routing Problem," *Transportation Science,* vol. 47, pp. 508-525, 2012.
[11] S. Lan, J.-P. Clarke, and C. Barnhart, "Planning for Robust Airline Operations: Optimizing Aircraft Routings and Flight Departure Times to Minimize Passenger Disruptions," *Transportation Science,* vol. 40, pp. 15-28, 2006.
[12] M. Dunbar, G. Froyland, and C.-L. Wu, "An integrated scenario-based approach for robust aircraft routing, crew pairing and re-timing," *Computers & Operations Research,* vol. 45, pp. 68-86, 2014.
[13] C. S.H., T. Ying Kei, and C. T.M., "Managing disruption risk in express logistics via proactive planning," *Industrial Management & Data Systems,* vol. 115, pp. 1481-1509, 2015.
[14] D. C. Dietz and M. Rosenshine "Optimal specialization of a maintenance workforce," *IIE Transactions,* vol. 29, pp. 423-433, 1997.
[15] N. Beaumont, "Scheduling staff using mixed integer programming," *European Journal of Operational Research,* vol. 98, pp. 473-484, 1997.
[16] M. J. Brusco and L. W. Jacobs, "Eliminating redundant columns in continuous tour scheduling problems," *European Journal of Operational Research,* vol. 111, pp. 518-525, 1998.
[17] M. J. Brusco, "Solving personnel tour scheduling problems using the dual all-integer cutting plane," *IIE Transactions,* vol. 30, pp. 835-844, 1998.
[18] H. K. Alfares, "An efficient two-phase algorithm for cyclic days-off scheduling," *Computers & Operations Research,* vol. 25, pp. 913-923, 1998.
[19] T.-H. Yang, S. Yan, and H.-H. Chen, "An airline maintenance manpower planning model with flexible strategies," *Journal of Air Transport Management,* vol. 9, pp. 233-239, 2003.
[20] S. Yan, T.-H. Yang, and H.-H. Chen, "Airline short-term maintenance manpower supply planning," *Transportation Research Part A: Policy and Practice,* vol. 38, pp. 615-642, 2004.
[21] J. Beliën, B. Cardoen, and E. Demeulemeester, "Improving Workforce Scheduling of Aircraft Line Maintenance at Sabena Technics," *Interfaces,* vol. 42, pp. 352-364, 2011.
[22] J. Van den Bergh, P. De Bruecker, J. Beliën, L. De Boeck, and E. Demeulemeester, "A three-stage approach for aircraft line maintenance personnel rostering using MIP, discrete event simulation and DEA," *Expert Systems with Applications,* vol. 40, pp. 2659-2668, 2013.
[23] J. Beliën, E. Demeulemeester, P. De Bruecker, J. Van den Bergh, and B. Cardoen, "Integrated staffing and scheduling for an aircraft line maintenance problem," *Computers & Operations Research,* vol. 40, pp. 1023-1033, 2013.
[24] P. De Bruecker, J. Beliën, J. Van den Bergh, and E. Demeulemeester, "A three-stage mixed integer programming approach for optimizing the skill mix and training schedules for aircraft maintenance," *European Journal of Operational Research,* vol. 267, pp. 439-452, 2018.
[25] D. Yang, J. Jiao, Y. Ji, G. Du, P. Helo, and A. Valente, "Joint optimization for coordinated configuration of product families and supply chains by a leader-follower Stackelberg game," *European Journal of Operational Research,* vol. 246, pp. 263-280, 2015.
[26] M. Esmaeili, M.-B. Aryanezhad, and P. Zeephongsekul, "A game theory approach in seller–buyer supply chain," *European Journal of Operational Research,* vol. 195, pp. 442-448, 2009.
[27] T. Xiao, T.-M. Choi, and T. C. E. Cheng, "Product variety and channel structure strategy for a retailer-Stackelberg supply chain," *European Journal of Operational Research,* vol. 233, pp. 114-124, 2014.
[28] D. Wang, G. Du, R. J. Jiao, R. Wu, J. Yu, and D. Yang, "A Stackelberg game theoretic model for optimizing product family architecting with supply chain consideration," *International Journal of Production Economics,* vol. 172, pp. 1-18, 2016.

[29] Y. Yu, G. Q. Huang, and L. Liang, "Stackelberg game-theoretic model for optimizing advertising, pricing and inventory policies in vendor managed inventory (VMI) production supply chains," *Computers & Industrial Engineering,* vol. 57, pp. 368-382, 2009.
[30] S. van Hoesel, "An overview of Stackelberg pricing in networks," *European Journal of Operational Research,* vol. 189, pp. 1393-1402, 2008.
[31] Y. Yu and G. Q. Huang, "Nash game model for optimizing market strategies, configuration of platform products in a Vendor Managed Inventory (VMI) supply chain for a product family," *European Journal of Operational Research,* vol. 206, pp. 361-373, 2010.
[32] S. C. Choi, "Price Competition in a Channel Structure with a Common Retailer," *Marketing Science*, vol. 10, pp. 271-296, 1991.
[33] S. Sinha and S. P. Sarmah, "Coordination and price competition in a duopoly common retailer supply chain," *Computers & Industrial Engineering*, vol. 59, pp. 280-295, 2010.
[34] G. Cai, Z. G. Zhang, and M. Zhang, "Game theoretical perspectives on dual-channel supply chain competition with price discounts and pricing schemes," *International Journal of Production Economics,* vol. 117, pp. 80-96, 2009.
[35] C.-W. Hsu, Y. Lee, and C.-H. Liao, "Competition between high-speed and conventional rail systems: A game theoretical approach," *Expert Systems with Applications,* vol. 37, pp. 3162-3170, 2010.
[36] S. Tiwari, H. M. Wee, and Y. Daryanto, "Big data analytics in supply chain management between 2010 and 2016: Insights to industries," *Computers & Industrial Engineering,* vol. 115, pp. 319-330, 2018.
[37] F. Ghofrani, Q. He, R. M. P. Goverde, and X. Liu, "Recent applications of big data analytics in railway transportation systems: A survey," *Transportation Research Part C: Emerging Technologies,* vol. 90, pp. 226-246, 2018.
[38] T.-H. Tsai, C.-K. Lee, and C.-H. Wei, "Neural network based temporal feature models for short-term railway passenger demand forecasting," *Expert Systems with Applications,* vol. 36, pp. 3728-3736, 2009.
[39] M. Tavana, A.-R. Abtahi, D. Di Caprio, and M. Poortarigh, "An Artificial Neural Network and Bayesian Network model for liquidity risk assessment in banking," *Neurocomputing,* vol. 275, pp. 2525-2554, 2018.
[40] N. Misiunas, A. Oztekin, Y. Chen, and K. Chandra, "DEANN: A healthcare analytic methodology of data envelopment analysis and artificial neural networks for the prediction of organ recipient functional status," *Omega,* vol. 58, pp. 46-54, 2016.
[41] J. J. Ruiz-Aguilar, I. J. Turias, and M. J. Jiménez-Come, "Hybrid approaches based on SARIMA and artificial neural networks for inspection time series forecasting," *Transportation Research Part E: Logistics and Transportation Review*, vol. 67, pp. 1-13, 2014.
[42] D. A. Hensher and T. T. Ton, "A comparison of the predictive potential of artificial neural networks and nested logit models for commuter mode choice," *Transportation Research Part E: Logistics and Transportation Review,* vol. 36, pp. 155-172, 2000.
[43] P. Murali, M. Dessouky, F. Ordóñez, and K. Palmer, "A delay estimation technique for single and double-track railroads," *Transportation Research Part E: Logistics and Transportation Review,* vol. 46, pp. 483-495, 2010.
[44] R. Batley, J. Dargay, and M. Wardman, "The impact of lateness and reliability on passenger rail demand," *Transportation Research Part E: Logistics and Transportation Review,* vol. 47, pp. 61-72, 2011.
[45] S. Tabrizi, S. H. Ghodsypour, and A. Ahmadi, "Modelling three-echelon warm-water fish supply chain: A bi-level optimization approach under Nash–Cournot equilibrium," *Applied Soft Computing,* 2017.
[46] G. M. Shafiullah, A. B. M. S. Ali, A. Thompson, and P. J. Wolfs, "Predicting vertical acceleration of railway wagons using regression algorithms," *Trans. Intell. Transport. Sys.,* vol. 11, pp. 290-299, 2010.

[47] M. d. C. Bas, J. Ortiz, L. Ballesteros, and S. Martorell, "Evaluation of a multiple linear regression model and SARIMA model in forecasting 7Be air concentrations," *Chemosphere,* vol. 177, pp. 326-333, 2017.
[48] T. Fang and R. Lahdelma, "Evaluation of a multiple linear regression model and SARIMA model in forecasting heat demand for district heating system," *Applied Energy,* vol. 179, pp. 544-552, 2016.
[49] P.-Y. Yin and J.-Y. Wang, "Ant colony optimization for the nonlinear resource allocation problem," *Applied Mathematics and Computation,* vol. 174, pp. 1438-1453, 2006.
[50] P. Samuelson, *Foundations of Economic Analysis*: Foundations of Economic Analysis, Cambridge, MA, 1947.
[51] J. F. Bard and J. E. Falk, "An explicit solution to the multi-level programming problem," *Computers & Operations Research*, vol. 9, pp. 77-100, 1982.
[52] J. Fortuny-Amat, xe, and B. McCarl, "A Representation and Economic Interpretation of a Two-Level Programming Problem," *The Journal of the Operational Research Society,* vol. 32, pp. 783-792, 1981.
[53] G. Anandalingam and D. J. White, "A solution method for the linear static Stackelberg problem using penalty functions," *IEEE Transactions on Automatic Control,* vol. 35, pp. 1170-1173, 1990.
[54] M. M. Muñoz and F. B. Abdelaziz, "Satisfactory solution concepts and their relations for Stochastic Multiobjective Programming problems," *European Journal of Operational Research,* vol. 220, pp. 430-442, 2012.
[55] H. T. Ozdemir and C. K. Mohan, "Flight graph based genetic algorithm for crew scheduling in airlines," *Information Sciences,* vol. 133, pp. 165-173, 2001.
[56] C. Wang, D. Mu, F. Zhao, and J. W. Sutherland, "A parallel simulated annealing method for the vehicle routing problem with simultaneous pickup–delivery and time windows," *Computers & Industrial Engineering,* vol. 83, pp. 111-122, 2015.
[57] P. Lučic and D. Teodorovic, "Simulated annealing for the multi-objective aircrew rostering problem," *Transportation Research Part A: Policy and Practice,* vol. 33, pp. 19-45, 1999.
[58] H. A. Hashim, S. El-Ferik, and M. A. Abido, "A fuzzy logic feedback filter design tuned with PSO for adaptive controller," *Expert Systems with Applications,* vol. 42, pp. 9077-9085, 2015.
[59] S.-H. Huang, Y.-H. Huang, C. A. Blazquez, and G. Paredes-Belmar, "Application of the Ant Colony Optimization in the Resolution of the Bridge Inspection Routing Problem," *Applied Soft Computing,* 2018.
[60] D. P. Mahato, R. S. Singh, A. K. Tripathi, and A. K. Maurya, "On scheduling transactions in a grid processing system considering load through Ant Colony Optimization," *Applied Soft Computing,* vol. 61, pp. 875-891, 2017.
[61] R. Skinderowicz, "An improved Ant Colony System for the Sequential Ordering Problem," *Computers & Operations Research,* vol. 86, pp. 1-17, 2017.
[62] S. R. Balseiro, I. Loiseau, and J. Ramonet, "An Ant Colony algorithm hybridized with insertion heuristics for the Time Dependent Vehicle Routing Problem with Time Windows," *Computers & Operations Research,* vol. 38, pp. 954-966, 2011.
[63] B. Liu, "Stackelberg-Nash equilibrium for multilevel programming with multiple followers using genetic algorithms," *Computers & Mathematics with Applications,* vol. 36, pp. 79-89, 1998.
[64] C. Sai Ho, M. Hoi Lam, and C. Hing Kai, "Cascading Delay Risk of Airline Workforce Deployments with Crew Pairing and Schedule Optimization," *Risk Analysis,* vol. 37, pp. 1443-1458, 2017.