



Multi-stage airline scheduling problem with stochastic passenger demand and non-cruise times

Özge Şafak, Özlem Çavuş, M. Selim Aktürk*

Department of Industrial Engineering, Bilkent University, Turkey



ARTICLE INFO

Article history:

Received 15 November 2017
Revised 2 April 2018
Accepted 17 May 2018
Available online 4 June 2018

Keywords:

Airline scheduling
Fleet assignment
Aircraft routing
Cruise speed control
Conic integer programming
Multi-stage stochastic programming

ABSTRACT

We propose a three-stage stochastic programming model which determines flight timing, fleet and routing decisions while considering the randomness of demand and non-cruise times. Our model differs from the existing two-stage stochastic models by considering not only flight timing and potential passenger demand, but also expected operational expenses, such as fuel burn and carbon emission costs. We include aircraft cruise speed decisions to compensate for non-cruise time variability so as to satisfy the time requirements of the passenger connections. We handle nonlinear functions of fuel and emission costs associated with cruise speed adjustments by utilizing mixed integer second order cone programming. Because the three-stage stochastic model leads to a large decision tree and can be very time-consuming to solve optimally, we suggest a scenario group-wise decomposition algorithm to obtain lower and upper bounds for the optimal value of the proposed model. The lower and upper bounds are obtained by solving a number of group subproblems, which are similar to proposed multi-stage stochastic model defined over a reduced number of scenarios. We suggest a cutting plane algorithm, along with improvements, to efficiently solve each group subproblem. In the numerical experiments, we provide a significant cost savings over two-stage stochastic programming and deterministic approaches.

© 2018 Elsevier Ltd. All rights reserved.

1. Introduction

At major airports, air traffic congestion may cause significant delays in flight departures and arrivals, in turn disrupting aircraft connections and passenger itineraries. According to the U.S. Department of Transportation (BTS, 2017c), around 18% of flights were delayed in 2016. Almost 29% of delays occurred due to the air carrier, 36% were due to the aircraft arriving late, and 31% were due to the National Aviation System. Only 4% of delays were due to heavy weather conditions and security issues. In 2010, FAA/Nextor (2017) reported that the cost of all US air transportation delays in 2007 totalled \$31.2 billion. \$16.7 billion, or about half of the total cost was incurred from the extra cost of the passengers' longer travel times. For airlines, increased expenses for crew, fuel, and maintenance expended \$8.3 billion. Accordingly, and especially with consideration of a highly competitive market, it is crucial for airlines to manage their flights, aircraft and crews efficiently to minimize operational costs. Airlines are willing to seek additional advanced solution techniques that allow them to make decisions jointly, thus producing solutions which provide reliable performance despite the uncertainties involved in flight operations.

* Corresponding author.

E-mail address: akturk@bilkent.edu.tr (M. Selim Aktürk).

The problem of airline scheduling has been disaggregated into different stages such as schedule design, fleet assignment, aircraft routing and crew scheduling. Accordingly, a sequential planning approach has traditionally been employed. The first step, known as schedule design, is a strategic planning problem usually requiring reconciliation at least a year in advance of departures. Schedules of flights with predetermined origins, destination airports and flight departure and arrival times are generated by considering potential passenger demand, fleet and crew resources. Following completion of this first stage, airline engineers can begin the fleet assignment problem: here, aircraft types, each having a different seat capacity, are assigned to flight legs based on their equipment capabilities, availabilities and operational expenses. However, because of the stochastic nature of passenger demand, assigning the right type of aircraft to each leg is a challenging problem to solve prior to departures. Realized demands may either exceed or fall short of the capacity of assigned aircraft type, resulting in loss of profit and customer goodwill. Following fleet assignment, a minimum cost route must be determined for each aircraft while satisfying the maintenance requirements.

As suggested by [Yan et al. \(2008\)](#), passenger demand fluctuations have to be considered in modeling the scheduling problem. In recent years, a growing literature has focused on the fleet assignment problems with uncertain demand. [Listes and Dekker \(2005\)](#) use a two-stage stochastic programming model to determine an optimal airline fleet composition (i.e., number of aircraft contained in each fleet type) to provide a re-fleeted process. In the second stage, they assign aircraft to each leg under each demand scenario by allowing swapping. One of the main drawbacks of stochastic programming is that the consideration of many scenarios simultaneously may lead to large decision trees significantly lengthening the computation time to solve the overall model. Therefore, for computational tractability, [Listes and Dekker \(2005\)](#) use a scenario aggregation approach. [Sherali and Zhu \(2008\)](#) analyze the several demand scenarios in their two stage stochastic fleet assignment model. In the first stage, they assign aircraft families having the same crew requirement with different aircraft types to flight legs. Given the assignment of such aircraft families, they subsequently assign aircraft types within the allotted family to each leg under each demand scenario. In their study, Sherali and Zhu develop a Benders' decomposition-based solution framework. However, they were unable to optimize real problem instances for United Airlines within a 25-hour time limit. [Cadarso and de Celis \(2017\)](#) propose a mathematical model to improve base schedules in terms of timetable and fleet assignments while accounting for passenger demand and operating conditions uncertainty. However, they do not consider the integration of aircraft routing problem along with the fleet assignment decisions, which may provide more economical solutions and prevent some incompatibilities between the decisions.

Aside from passenger demand fluctuations, airport congestion is another crucial uncertainty involved in flight times. These uncertainties are not included in any planning stages of a deterministic approach, despite the disruptions that flight delays propagate through the network, ultimately affecting a large scale of aircraft routes and passenger itineraries. Therefore, [Yen and Birge \(2006\)](#) consider the effect of random disruptions to develop a robust schedule which can better withstand delays. In their two-stage stochastic integer programming formulation, in the first-stage, they determine pairings of a round-trip itinerary that a crew member might fly; in the second stage, they consider several disruption scenarios to determine actual departure and arrival times. [Sohoni et al. \(2011\)](#) propose two stochastic models which incorporate block time uncertainty into the schedule development process. More recently, [Dunbar et al. \(2014\)](#) incorporate delay scenarios within the aircraft routing and crew pairing problems. In order to minimize delay propagation, they adjust flight departure times, in turn providing more slack over critical connections and drawing excess slack from remaining connections. [Chiraphadhanakul and Barnhart \(2013\)](#) desire slacks in robust schedule to absorb delays. [Ahmadbeygi et al. \(2010\)](#) propose a method to minimize the delay propagation by redistributing the existing slack in the flight schedule. [Liang et al. \(2015\)](#) also extend the robust weekly aircraft maintenance routing problem for the operational tail assignment model to minimize the total expected propagated delay.

An alternative way of ensuring on-time connections is to control aircraft speed as discussed in [Cook et al. \(2009\)](#). [Bertsimas et al. \(2010\)](#) control aircraft speed through adjustments in the time spent in each en route sector while deciding on an optimal combination of flow management actions including ground holding, rerouting and airborne holding. [Sherali et al. \(2006\)](#) emphasize the sensitivity of airline optimization decisions to fuel burn. More recently, [Şafak et al. \(2017\)](#) consider the fuel burn and CO₂ emission costs associated with cruise speed adjustments to ensure the passenger connections in their integrated aircraft-path assignment and robust scheduling problem. [Gürkan et al. \(2016\)](#) also include aircraft cruise speed decisions in an integrated airline scheduling, aircraft fleet assignment and routing problem. The major difficulty of including cruise time as a decision variable in their models is the consideration of nonlinear functions of fuel burn and carbon emissions. To handle this nonlinearity, they utilize the mixed-integer second order cone programming as discussed in [Aktürk et al. \(2014\)](#). In addition to aviation sector, [Yang et al. \(2016\)](#) and [Haahr et al. \(2017\)](#) provide significant savings in the total tractive energy consumption by optimizing train speed profiles. Furthermore, in maritime transportation, [He et al. \(2017\)](#) and [Fagerholt et al. \(2015\)](#) optimize the speed to minimize the fuel emissions. In the context of road transportation, [Bektaş and Laporte \(2011\)](#) propose a pollution routing problem to find the optimal routes and speeds to minimize the total costs of fuel consumption and emissions.

None of the discussed approaches consider the interplay between schedule design, fleet assignment and aircraft routing, and cruise time decisions in an uncertain environment. Let us consider a situation with high uncertainty in non-cruise times due to airport congestion. On the one hand, airlines may choose to set a higher cruise speed to guarantee the minimum time requirements for passenger connections. Since speeding up the aircraft results in additional fuel burn and carbon emission costs, assigning a fuel efficient but smaller aircraft may be preferable. However, such an assignment may spill some of the passengers because of the insufficient seat capacity of the aircraft. Thus, passenger demand needs to be taken into account

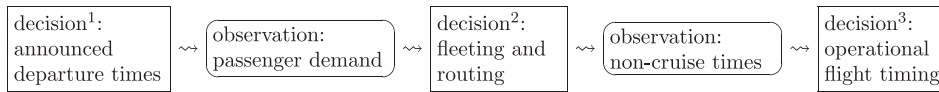


Fig. 1. Decision process.

and a fuel efficient aircraft should only be assigned when the savings from fuel conservations compensate for the cost of spilled passengers. On the other hand, if a larger aircraft is used, it may be prohibitively costly to speed up the aircraft. In that case, carefully selecting departure times helps to reduce the costs caused by the late arrival of the aircraft. Therefore, in this paper, we consider all interrelated decisions and operational cost terms in a single model.

The main contributions of this paper are:

- We propose a novel mixed-integer three-stage stochastic nonlinear programming model which determines strategic departure time decisions, tactical fleeting and routing decisions and operational flight timing decisions while considering randomness in passenger demand and non-cruise times.
- We propose valid inequalities that will allow us to handle the additional complexity of the operational level decisions of the third stage.
- We employ a scenario group-wise decomposition algorithm to handle large number of scenarios of our mixed-integer three-stage stochastic model.
- We present a cutting plane algorithm to solve scenario group subproblems. We use second order cone duality to handle the nonlinear subproblems arising in the cutting plane framework. We also utilize scenario group cuts and pareto-optimal cuts to fasten the convergence of the cutting plane algorithm.

The rest of the paper is organized as follows. In [Section 2](#), we briefly describe the framework of the problem. We formulate a mixed-integer three-stage stochastic nonlinear programming model. Then, we reformulate proposed model using second order cone programming to handle nonlinearity. We also suggest valid inequalities to make the formulation stronger. We devote [Section 3](#) to a scenario group-wise decomposition algorithm to find lower and upper bounds for the optimal value of the proposed model. [Section 4](#) presents an improved cutting plane algorithm to solve group subproblems in bounding algorithms. In [Section 5](#), we report computational results, comparing three-stage stochastic programming approach with two-stage stochastic implementations, and illustrating the high quality performances of the proposed algorithms using a sample schedule for a major U.S. airline. Finally, in [Section 6](#), we conclude with remarks and outline possible research directions arising from this study.

2. Stochastic airline scheduling

In this section, we describe a mixed-integer three-stage stochastic nonlinear programming model that generates a flight schedule by considering the randomness of passenger demand and non-cruise times. The first stage variables determine the announced departure time for each flight, but unlike the classical deterministic approach, announced departure times are selected to reduce expected operational costs while considering the uncertainty that is represented by multiple potential demand and non-cruise time scenarios. After the departure times are announced, uncertain passenger demand data are realized. For each demand realizations, the second stage variables determine the fleet assignment and aircraft routing while considering the non-cruise time uncertainty. Finally, uncertain non-cruise time data are realized. The third stage decisions determine actual flight departure times and aircraft cruise speed for each scenario. The model takes the advantage of cruise speed control to compensate for the increase in non-cruise times due to the airport congestion, to be able to ensure the connection of passengers and aircraft itineraries. The decision process is provided in [Fig. 1](#).

The objective function minimizes the expected operational costs including delay cost, passenger misconnection cost, spilled passenger cost and fuel burn cost associated with aircraft cruise speed adjustments. The resulting three-stage model assigns fleet types by considering not only passenger demand, as commonly done in the literature, but also expected operational expenses such fuel burn and carbon emission costs. The model can take the advantage of speeding up the aircraft to ensure the passengers' connections, which gives rise to a fuel burn cost. The fuel burn can be reduced through assigning a fuel efficient aircraft. However, assigning a fuel efficient, but smaller aircraft may result in spilled passengers, if the passenger demand realization yields above the seat capacity of assigned aircraft. In addition to aircraft cruise speed control, the model re-times the flight departures to satisfy the connections of passengers in response to the increase in non-cruise time variations. On the other hand, shifting the flight departures incurs an additional delay cost. Therefore, the model carefully determines the first stage announced departure times to minimize the expected costs. Because of the interplay between decisions, we propose a single model which integrates these three stage decisions using a stochastic programming where potential passenger demand and non-cruise time scenarios are included.

The notation used throughout the paper is given below:

Sets:

Ω	set of scenarios
K	set of aircraft types
J	set of flights
P_i	set of flights which have passenger connections from flight $i \in J$
C_i	set of flights which have passenger connections to flight $i \in J$
U_i	set of flights that can be followed by flight $i \in J$
D_i	set of flights that can follow flight $i \in J$
$PAIR$	set of flights $i \in J$ and $j \in J$ such that $i \neq j$ and flight i can connect flight j
J_k^e	set of flights that can be the last flight of aircraft type $k \in K$
J_k^b	set of flights that can be the first flight of aircraft type $k \in K$
$J(\omega)$	set of scenarios that have the same demand realization as in scenario $\omega \in \Omega$, $\bar{J}(\omega) = \{\bar{\omega} \in \Omega : demand_i(\omega) = demand_i(\bar{\omega}), i \in J\}$

Parameters:

$demand_i(\omega)$	demand realization for flight $i \in J$ under scenario $\omega \in \Omega$
$Ncrs_i(\omega)$	non-cruise time realization for flight $i \in J$ under scenario $\omega \in \Omega$
$p(\omega)$	probability of scenario $\omega \in \Omega$
ℓ	duration between consecutive departure time alternatives
T_i^{min}	earliest departure time unit of flight $i \in J$
T_i^{max}	latest departure time unit of flight $i \in J$
N_k	available number of aircraft type $k \in K$
TA_{ijk}	turnaround time needed to prepare aircraft type $k \in K$ between flights $(i, j) \in PAIR$
TP_{ij}	turntime needed to connect passengers between flights $i \in J, j \in P_i$
$[f_{ik}^s, f_{ik}^k]$	time window for the cruise time of flight $i \in J$ with an aircraft type $k \in K$
CAP_k	number of seats in aircraft type $k \in K$
$pass_{ij}(\omega)$	number of passengers who have a connection from flight $i \in J$ to $j \in P_i$ under scenario $\omega \in \Omega$
c_{daily}^k	daily cost of using aircraft type $k \in K$
c_{sp}^i	opportunity cost of spilled passengers of flight $i \in J$
c_{fuel}	fuel price per kilogram (kg) of fuel burned by an aircraft
c_{CO_2}	cost of emission per kg of CO_2 emitted by an aircraft
c_{idle}^k	unit idle time cost of aircraft type $k \in K$ in dollars per minute
$c_{det}^i(\omega)$	cost of delay of flight $i \in J$ under scenario $\omega \in \Omega$ in dollars per minute
c_{dist}^i	cost of disrupted passengers of flight $i \in J$ in dollars per passenger
M	a big number

Decision Variables:

x_i^1	announced departure time of flight $i \in J$ (first-stage decision variable)
τ_i^1	discrete time unit in which announced departure time of flight i is scheduled, $i \in J$ (first-stage decision variable)
$z_{ijk}^1(\omega)$	1 if both flights i and j are operated by aircraft type k and flight i is followed by flight j under scenario ω and 0 otherwise, $j \in J, i \in U_j, \omega \in \Omega, k \in K$ (second-stage decision variable)
$y_{ik}^2(\omega)$	1 if flight i is the first flight operated by aircraft type k under scenario ω and 0 otherwise, $i \in J, k \in K, \omega \in \Omega$ (second-stage decision variable)
$u_{ik}^2(\omega)$	1 if flight i is the last flight operated by aircraft type k under scenario ω and 0 otherwise, $i \in J, k \in K, \omega \in \Omega$ (second-stage decision variable)
$d_i^3(\omega)$	actual departure time of flight i under scenario $\omega, i \in J, \omega \in \Omega$ (third-stage decision variable)
$f_{ik}^3(\omega)$	cruise time of flight i operated by aircraft type k under scenario $\omega, i \in J, k \in K, \omega \in \Omega$ (third-stage decision variable)
$S_{ik}^3(\omega)$	idle time of aircraft type k after operating flight i under scenario $\omega, i \in J, k \in K, \omega \in \Omega$ (third-stage decision variable)
$del_i^3(\omega)$	delay amount of flight i under scenario $\omega, i \in J, \omega \in \Omega$ (third-stage decision variable)
$a_{ij}^3(\omega)$	1 if passengers in flight i miss flight j under scenario ω and 0 otherwise, $i \in J, j \in P_i, \omega \in \Omega$ (third-stage decision variable)

2.1. Fuel and CO₂ emission cost function

We use the idea of flying faster to compensate for the increase in non-cruise time variations to satisfy passenger connections or flying slower for conservation of fuel. To estimate fuel burn, we use the cruise stage fuel flow model developed by the Base of Aircraft Data (BADA) (EUROCONTROL, 2012). For all $i \in J, k \in K, \omega \in \Omega$, fuel burn (kg) as a function of cruise time $f_{ik}^3(\omega)$ (minutes) can be calculated as follows:

$$F_{ik}(f_{ik}^3(\omega)) = c_1^{ik} \cdot \frac{1}{f_{ik}^3(\omega)} + c_2^{ik} \cdot \frac{1}{(f_{ik}^3(\omega))^2} + c_3^{ik} \cdot (f_{ik}^3(\omega))^3 + c_4^{ik} \cdot (f_{ik}^3(\omega))^2. \quad (1)$$

The coefficients $c_s^{ik} > 0, s = 1, \dots, 4$, are expressed in terms of aircraft specific fuel consumption and drag as well as the mass of aircraft, air density and gravitational acceleration in Şafak et al. (2017). It is important to note that $F_{ik}(f_{ik}^3(\omega))$ is a convex function of $f_{ik}^3(\omega) > 0$.

Moreover, (EUROCONTROL, 2001) states that each kg of fuel burn approximately produces 3.15 kg of CO₂ emission. Therefore, we can express fuel and CO₂ emission costs as a function of cruise time as follows:

$$C_{fuel}^{ik}(f_{ik}^3(\omega)) + C_{CO_2}^{ik}(f_{ik}^3(\omega)) = \begin{cases} (c_{fuel} + c_{CO_2} \cdot \kappa) \cdot F_{ik}(f_{ik}^3(\omega)) & \text{if } m_{ik}^2(\omega) = 1 \\ 0 & \text{if } m_{ik}^2(\omega) = 0, \end{cases}$$

where c_{fuel} and c_{CO_2} are the unit prices (\$/kg) of fuel and CO₂ emission, respectively and κ is CO₂ emission constant (e.g., $\kappa = 3.15$). Here, $m_{ik}^2(\omega) := (y_{ik}^2(\omega) + \sum_{j \in U_i} z_{jik}^2(\omega))$, so that if aircraft type k is not assigned to flight i under scenario ω , then $C_{fuel}^{ik}(f_{ik}^3(\omega)) + C_{CO_2}^{ik}(f_{ik}^3(\omega)) = 0$.

2.2. Mathematical model

We provide a three-stage stochastic mixed integer nonlinear programming formulation below.

$$\begin{aligned} \min \quad & \sum_{\omega \in \Omega} \left(\sum_{i \in J} \sum_{k \in K} y_{ik}^2(\omega) \cdot c_{daily}^k \right) \cdot p(\omega) \\ & + \sum_{\omega \in \Omega} \left(\sum_{i \in J} \sum_{k \in K} \max(0, demand_i(\omega) - CAP_k) \cdot \left(\sum_{j \in U_i} z_{jik}^2(\omega) + y_{ik}^2(\omega) \right) \cdot c_{sp}^i \right) \cdot p(\omega) \\ & + \sum_{\omega \in \Omega} \left(\sum_{i \in J} \sum_{k \in K} (C_{fuel}^{ik}(f_{ik}^3(\omega)) + C_{CO_2}^{ik}(f_{ik}^3(\omega))) \right) \cdot p(\omega) \\ & + \sum_{\omega \in \Omega} \left(\sum_{i \in J} \sum_{k \in K} S_{ik}^3(\omega) \cdot c_{idle}^k \right) \cdot p(\omega) + \sum_{\omega \in \Omega} \left(\sum_{i \in J} c_{del}^i(\omega) \cdot del_i^3(\omega) \right) \cdot p(\omega) \\ & + \sum_{\omega \in \Omega} \left(\sum_{i \in J} \sum_{j \in P_i} c_{dist}^i \cdot pass_{ij}(\omega) \cdot a_{ij}^3(\omega) \right) \cdot p(\omega) \end{aligned} \tag{2}$$

subject to

$$T_i^{min} \leq \tau_i^1 \leq T_i^{max} \quad i \in J \tag{3}$$

$$x_i^1 = \ell \cdot \tau_i^1 \quad i \in J \tag{4}$$

$$\tau_i^1 \quad \text{integer} \quad i \in J \tag{5}$$

$$\sum_{k \in K} \left(\sum_{j \in U_i} z_{jik}^2(\omega) + y_{ik}^2(\omega) \right) = 1 \quad i \in J, \omega \in \Omega \tag{6}$$

$$\sum_{j \in U_i} z_{jik}^2(\omega) + y_{ik}^2(\omega) - \sum_{j \in D_i} z_{ijk}^2(\omega) - u_{ik}^2(\omega) = 0 \quad i \in J, k \in K, \omega \in \Omega \tag{7}$$

$$\sum_{i \in J} y_{ik}^2(\omega) \leq N_k \quad k \in K, \omega \in \Omega \tag{8}$$

$$y_{ik}^2(\omega) = 0 \quad i \in J \setminus J_k^s, k \in K, \omega \in \Omega \tag{9}$$

$$u_{ik}^2(\omega) = 0 \quad i \in J \setminus J_k^e, k \in K, \omega \in \Omega \tag{10}$$

$$z_{jik}^2(\omega) \in \{0, 1\} \quad i \in J, j \in U_i, k \in K, \omega \in \Omega \tag{11}$$

$$y_{ik}^2(\omega) \in \{0, 1\} \quad i \in J, k \in K, \omega \in \Omega \quad (12)$$

$$-x_i^1 + d_i^3(\omega) \geq 0 \quad i \in J, \omega \in \Omega \quad (13)$$

$$d_i^3(\omega) - x_i^1 - del_i^3(\omega) \leq 0 \quad i \in J, \omega \in \Omega \quad (14)$$

IF $z_{ijk}^2(\omega) = 1$ THEN,

$$-TA_{ijk} + d_j^3(\omega) - d_i^3(\omega) - f_{ik}^3(\omega) - S_{ik}^3(\omega) = Ncrs_i(\omega) \quad (i, j) \in PAIR, k \in K, \omega \in \Omega \quad (15)$$

$$-f_{ik}^l \left(\sum_{j \in U_i} z_{jik}^2(\omega) + y_{ik}^2(\omega) \right) + f_{ik}^3(\omega) \geq 0 \quad i \in J, k \in K, \omega \in \Omega \quad (16)$$

$$f_{ik}^u \left(\sum_{j \in U_i} z_{jik}^2(\omega) + y_{ik}^2(\omega) \right) - f_{ik}^3(\omega) \geq 0 \quad i \in J, k \in K, \omega \in \Omega \quad (17)$$

$$d_j^3(\omega) - d_i^3(\omega) - \sum_{k \in K} f_{ik}^3(\omega) + M \cdot a_{ij}^3(\omega) \geq Ncrs_i(\omega) + TP_{ij} \quad i \in J, j \in P_i, \omega \in \Omega \quad (18)$$

$$S_{ik}^3(\omega) \geq 0 \quad i \in J, k \in K, \omega \in \Omega \quad (19)$$

$$del_i^3(\omega) \geq 0 \quad i \in J, \omega \in \Omega \quad (20)$$

$$a_{ij}^3(\omega) \in \{0, 1\} \quad i \in J, j \in P_i, \omega \in \Omega \quad (21)$$

$$z_{jik}^2(\omega) = z_{jik}^2(\bar{\omega}) \quad i \in J, j \in U_i, k \in K, \omega \in \Omega, \bar{\omega} \in \bar{J}(\omega) \quad (22)$$

$$y_{ik}^2(\omega) = y_{ik}^2(\bar{\omega}) \quad i \in J, j \in U_i, k \in K, \omega \in \Omega, \bar{\omega} \in \bar{J}(\omega) \quad (23)$$

$$u_{ik}^2(\omega) = u_{ik}^2(\bar{\omega}) \quad i \in J, j \in U_i, k \in K, \omega \in \Omega, \bar{\omega} \in \bar{J}(\omega) \quad (24)$$

The objective function (2) minimizes the expected operational costs of an airline. These interrelated costs are aircraft daily usage, spilled passengers, fuel and CO₂ emission, idle times, delay, and passenger misconnection costs. In our study, flight time is integer valued since airlines usually prefer the flight times to be assigned within a set of discrete time units for practical reasons. Therefore, the first stage constraints (3), (4) and (5) ensure the departure time of flights to be within a set of discrete time units which have already been determined by the airline. Constraints (6), (7) and (8) are the second stage *fleeting and routing constraints* for each scenario. Constraint (6) assures that each scheduled flight leg is covered by exactly one aircraft type. Constraint (7) maintains the route of an aircraft throughout the entire network. Constraint (8) limits the number of employed aircraft by N_k . Constraints (9) and (10) sustain maintenance policy which determines the first and last airports for each aircraft type. Constraints (11) and (12) define the domain of the second stage variables. Constraints (13) and (14) allow departure delays with respect to announced departure times of the first stage. Delays are penalized in the objective function (2). Constraint (15) guarantees minimum aircraft turnaround time, needed between consecutive flights operated by the same aircraft, for each scenario. Constraints (16) and (17) apply cruise time upper and lower bounds for each flight. In constraint (18), if the time needed for the connection of passengers from flight i to any follow-on flight j is not satisfied under scenario ω , then passengers miss the connection with a penalty in the objective function (2). Constraints (19), (20) and (21) define the domain of the third stage variables. Constraints (22), (23) and (24) are the non-anticipativity constraints. At the second stage, fleeting decisions should have the same value for the scenarios which have the same history up to the second stage. In other words, fleeting decisions should depend on the realization of demands. Therefore, non-anticipativity constraints ensure that fleeting variables take equal values for the scenarios which share the same realization of demands.

Different than the existing two-stage models, the proposed three-stage stochastic model integrates fleet assignment and aircraft routing decisions along with operational flight timing decisions. Despite the additional complexity of the third stage, incorporating the cruise time decisions and considering potential non-cruise time scenarios result in improved solutions which can better withstand delays. In the computations, the great advantage of three-stage stochastic programming approach can be clearly seen in the significant cost savings compared to a two-stage approach. However, large number of scenarios, nonlinear fuel and carbon emission cost functions and binary fleet assignment decisions make the problem significantly more difficult to solve. To handle them, we propose valid inequalities to strengthen the formulation and we utilize a mixed integer second order cone programming to handle nonlinearity. To deal with the large number of scenarios, we employ a scenario group-wise decomposition algorithm by solving a number of group subproblems. However, even the group subproblems, defined over a reduced number of scenarios, could not be solved to optimality by CPLEX, because of the large number of binary decisions and the second order conic inequalities. Therefore, we suggest a cutting plane algorithm to mitigate the computational effort to solve group subproblems. We use second order conic duality to efficiently handle the nonlinear fuel and emission cost functions within the cutting plane algorithm. In the next section, we introduce valid inequalities.

2.3. Valid inequalities

In order to mitigate the computational difficulty, we develop a set of valid inequalities which strengthen the formulation. Propositions 1–3 present these valid inequalities.

Proposition 1. Let v_i^l and v_j^l be a lower bound for $d_i^3(\omega)$ and $d_j^3(\omega)$, respectively, and $\max_{k \in K} f_{ik}^u$ be the maximum of upper bounds f_{ik}^u for all $i \in J, j \in P_i, k \in K, \omega \in \Omega$. Then, for all $i \in J, j \in P_i, \omega \in \Omega$, the inequality

$$d_j^3(\omega) - v_i^l - \sum_{k \in K} f_{ik}^3(\omega) + (v_i^l + \max_{k \in K} f_{ik}^u + Ncrs_i(\omega) + TP_{ij} - v_j^l) \cdot a_{ij}^3(\omega) \geq Ncrs_i(\omega) + TP_{ij} \tag{25}$$

is a valid inequality for the problem (2)–(24).

Proof. For any $i \in J, j \in P_i, \omega \in \Omega$, if $a_{ij}^3(\omega) = 0$ then for feasibility, from constraint (18), we need $d_j^3(\omega) - d_i^3(\omega) - \sum_{k \in K} f_{ik}^3(\omega) \geq Ncrs_i(\omega) + TP_{ij}$. Since $d_i^3(\omega) \geq v_i^l$, the inequality (25) is satisfied.

Otherwise, if $a_{ij}^3(\omega) = 1$, the inequality (25) becomes $d_j^3(\omega) - v_j^l - \sum_{k \in K} f_{ik}^3(\omega) + \max_{k \in K} f_{ik}^u \geq 0$. Note that $\max_{k \in K} f_{ik}^u$ is an upper bound for $\sum_{k \in K} f_{ik}^3(\omega)$, since from constraint (17), $f_{ik}^3(\omega)$ would take a positive value for only one $k \in K$. Since v_j^l is a lower bound for $d_j^3(\omega)$, the inequality (25) is again satisfied. □

Proposition 2. Let v_i^l and v_j^l be a lower bound for $d_i^3(\omega)$ and $d_j^3(\omega)$, respectively, for all $(i, j) \in PAIR, k \in K, \omega \in \Omega$. Then, for all $(i, j) \in PAIR, k \in K, \omega \in \Omega$, the inequality

$$d_j^3(\omega) - v_i^l - f_{ik}^3(\omega) + (v_i^l + f_{ik}^u + Ncrs_i(\omega) + TA_{ijk} - v_j^l) \cdot (1 - z_{ijk}^2(\omega)) \geq Ncrs_i(\omega) + TA_{ijk} \tag{26}$$

is a valid inequality for the problem (2)–(24).

Proof. For any $(i, j) \in PAIR, k \in K, \omega \in \Omega$, if $z_{ijk}^2(\omega) = 1$ then for feasibility, from constraint (15), we need $d_j^3(\omega) - d_i^3(\omega) - f_{ik}^3(\omega) - S_{ik}^3(\omega) = Ncrs_i(\omega) + TA_{ijk}$. Since $d_i^3(\omega) \geq v_i^l$ and $S_{ik}^3(\omega) \geq 0$, the inequality (26) is satisfied.

Otherwise, if $z_{ijk}^2(\omega) = 0$, the inequality (26) becomes $d_j^3(\omega) - v_j^l - f_{ik}^3(\omega) + f_{ik}^u \geq 0$. Since v_j^l is a lower bound for $d_j^3(\omega)$ and f_{ik}^u is an upper bound for $f_{ik}^3(\omega)$, the inequality (26) is again satisfied. □

Proposition 3. For all $(i, j) \in PAIR, k \in K, \omega \in \Omega$, the inequality

$$z_{ijk}^2(\omega) \leq \sum_{n \in J} y_{nk}^2(\omega) \tag{27}$$

is a valid inequality for the problem (2)–(24).

Proof. For any $k \in K, \omega \in \Omega$, if $\sum_{n \in J} y_{nk}^2(\omega) = 0$, by definition of variables $y_{nk}^2(\omega)$, it implies that under scenario $\omega \in \Omega$, aircraft type k is not used in the schedule. Then, we need an aircraft type k cannot be assigned to any consecutive flights $(i, j) \in PAIR$ implying that $z_{ijk}^2(\omega) = 0$.

Otherwise, if $\sum_{n \in J} y_{nk}^2(\omega) = 1$, the valid inequality (27) becomes $z_{ijk}^2(\omega) \leq 1$. Since $z_{ijk}^2(\omega) \in \{0, 1\}$, the inequality (27) is trivially satisfied. □

2.4. Conic representation of the fuel and CO₂ cost function

In the objective function, we have nonlinear cost terms due to controllable cruise times. To handle nonlinearity, we utilize a mixed integer second order cone programming. Let us redefine fuel and CO₂ emission cost functions as:

$$K(f_{ik}^3(\omega)) =$$

$$\begin{cases} (c_{fuel} + c_{CO_2} \cdot \kappa) \cdot \left(c_1^{ik} \cdot \frac{1}{f_{ik}^3(\omega)} + c_2^{ik} \cdot \frac{1}{(f_{ik}^3(\omega))^2} + c_3^{ik} \cdot (f_{ik}^3(\omega))^3 + c_4^{ik} \cdot (f_{ik}^3(\omega))^2 \right) & \text{if } m_{ik}^2(\omega) = 1 \\ 0 & \text{if } m_{ik}^2(\omega) = 0. \end{cases}$$

Note that $K(f_{ik}^3(\omega))$ is discontinuous and therefore its epigraph

$$E_F = \{ (f_{ik}^3(\omega), t) \in \mathbb{R}^2 : K(f_{ik}^3(\omega)) \leq t \}$$

is nonconvex. In the next proposition, we describe how the convex hull of E_F is obtained. This proposition was previously proven in Şafak et al. (2017), but for the sake of completeness, here we give the proof. To simplify the presentation, we drop the indices of the variables and parameters in the next proposition. A more detailed discussion can be found in Aktürk et al. (2014), Aktürk et al. (2009) and Günlük and Linderoth (2010).

Proposition 4. Let $q, \delta, \varphi, \vartheta \geq 0$. Then the convex hull of E_F can be expressed as

$$t \geq (c_{fuel} + c_{CO_2} \cdot \kappa) \cdot (c_1 \cdot q + c_2 \cdot \delta + c_3 \cdot \varphi + c_4 \cdot \vartheta) \tag{28}$$

$$m^2 \leq q \cdot f \tag{29}$$

$$m^4 \leq f^2 \cdot \delta \cdot m \tag{30}$$

$$f^4 \leq m^2 \cdot \varphi \cdot f \tag{31}$$

$$f^2 \leq \vartheta \cdot m \tag{32}$$

in the constraint set.

Proof. According to Hiriart-Urruty and Lemaréchal (2001), perspective of a convex function $k(f)$ is $m \cdot k(f/m)$. Observe that each of the nonlinear terms $\frac{1}{f}$, $\frac{1}{f^2}$, f^3 and f^2 is a convex function of $f \geq 0$. Therefore, epigraph of the perspective of each term can be written as, $\frac{m^2}{f} \leq q$, $\frac{m^3}{f^2} \leq \delta$, $\frac{f^3}{m^2} \leq \varphi$, $\frac{f^2}{m} \leq \vartheta$, respectively. Since $m, f \geq 0$, they can be written as stated in the proposition. □

2.5. Second order conic reformulation of the model

Each inequalities (29)–(32) can be represented by conic quadratic inequalities. Observe that (29) and (32) are hyperbolic inequalities. The inequality (30) can be rewritten as the following two hyperbolic inequalities

$$(m_{ik}^2(\omega))^2 \leq \mu_{ik}^3(\omega) \cdot f_{ik}^3(\omega), \tag{33}$$

$$(\mu_{ik}^3(\omega))^2 \leq \delta_{ik}^3(\omega) \cdot m_{ik}^2(\omega), \tag{34}$$

where $\mu_{ik}^3(\omega) \geq 0$ for all $i \in J, k \in K, \omega \in \Omega$. Similarly, the inequality (31) can be restated as

$$(f_{ik}^3(\omega))^2 \leq \nu_{ik}^3(\omega) \cdot m_{ik}^2(\omega), \tag{35}$$

$$(\nu_{ik}^3(\omega))^2 \leq \varphi_{ik}^3(\omega) \cdot f_{ik}^3(\omega), \tag{36}$$

where $\nu_{ik}^3(\omega) \geq 0$ for all $i \in J, k \in K, \omega \in \Omega$. Following Ben-Tal and Nemirovski (2001), we can represent the hyperbolic inequalities (29), (32) and (33)–(36) as the second order conic inequalities (39)–(44).

Below, we provide the mixed integer second order cone programming formulation of our model, called as MISOCP. For ease of representation, from now on, we substitute $(y_{ik}^2(\omega) + \sum_{j \in U_i} z_{jik}^2(\omega))$ with $m_{ik}^2(\omega)$ for all $i \in J, k \in K, \omega \in \Omega$.

$$\begin{aligned} \min \quad & \sum_{\omega \in \Omega} \left(\sum_{i \in J} \sum_{k \in K} y_{ik}^2(\omega) \cdot c_{daily}^k \right) \cdot p(\omega) & \text{(MISOCP)} \\ & + \sum_{\omega \in \Omega} \left(\sum_{i \in J} \sum_{k \in K} \max(0, demand_i(\omega) - CAP_k) \cdot m_{ik}^2(\omega) \cdot c_{sp}^i \right) \cdot p(\omega) \end{aligned}$$

$$\begin{aligned}
 & + \sum_{\omega \in \Omega} \left(\sum_{k \in K} \sum_{i \in J} (c_{fuel} + c_{CO_2} \cdot \kappa) \cdot (c_1^{ik} \cdot q_{ik}^3(\omega) + c_2^{ik} \cdot \delta_{ik}^3(\omega) + c_3^{ik} \cdot \varphi_{ik}^3(\omega) + c_4^{ik} \cdot \vartheta_{ik}^3(\omega)) \right) \cdot p(\omega) \\
 & + \sum_{\omega \in \Omega} \left(\sum_{i \in J} \sum_{k \in K} S_{ik}^3(\omega) \cdot c_{Idle}^k \right) \cdot p(\omega) + \sum_{\omega \in \Omega} \left(\sum_{i \in J} c_{del}^i(\omega) \cdot del_i^3(\omega) \right) \cdot p(\omega) \\
 & + \sum_{\omega \in \Omega} \left(\sum_{i \in J} \sum_{j \in P_i} c_{dist}^i \cdot pass_{ij}(\omega) \cdot a_{ij}^3(\omega) \right) \cdot p(\omega)
 \end{aligned} \tag{37}$$

subject to

$$(3) - (24)$$

$$m_{ik}^2(\omega) = \left(y_{ik}^2(\omega) + \sum_{j \in U_i} z_{jik}^2(\omega) \right) \quad i \in J, k \in K, \omega \in \Omega \tag{38}$$

$$\| (2m_{ik}^2(\omega), q_{ik}^3(\omega) - f_{ik}^3(\omega)) \| \leq q_{ik}^3(\omega) + f_{ik}^3(\omega) \quad i \in J, k \in K, \omega \in \Omega \tag{39}$$

$$\| (2f_{ik}^3(\omega), \vartheta_{ik}^3(\omega) - m_{ik}^2(\omega)) \| \leq \vartheta_{ik}^3(\omega) + m_{ik}^2(\omega) \quad i \in J, k \in K, \omega \in \Omega \tag{40}$$

$$\| (2m_{ik}^2(\omega), \mu_{ik}^3(\omega) - f_{ik}^3(\omega)) \| \leq \mu_{ik}^3(\omega) + f_{ik}^3(\omega) \quad i \in J, k \in K, \omega \in \Omega \tag{41}$$

$$\| (2\mu_{ik}^3(\omega), \delta_{ik}^3(\omega) - m_{ik}^2(\omega)) \| \leq \delta_{ik}^3(\omega) + m_{ik}^2(\omega) \quad i \in J, k \in K, \omega \in \Omega \tag{42}$$

$$\| (2f_{ik}^3(\omega), v_{ik}^3(\omega) - m_{ik}^2(\omega)) \| \leq v_{ik}^3(\omega) + m_{ik}^2(\omega) \quad i \in J, k \in K, \omega \in \Omega \tag{43}$$

$$\| (2v_{ik}^3(\omega), \varphi_{ik}^3(\omega) - f_{ik}^3(\omega)) \| \leq \varphi_{ik}^3(\omega) + f_{ik}^3(\omega) \quad i \in J, k \in K, \omega \in \Omega \tag{44}$$

$$q_{ik}^3(\omega), \delta_{ik}^3(\omega), \varphi_{ik}^3(\omega), \vartheta_{ik}^3(\omega), \mu_{ik}^3(\omega), v_{ik}^3(\omega) \geq 0 \quad i \in J, k \in K, \omega \in \Omega \tag{45}$$

Here, $\| \cdot \|$ denotes Euclidean norm. Note that, the objective function (37) is slightly different than the original objective function of the proposed model. The original objective (2) is represented by the new objective and conic constraints (39)-(44).

3. Bounding algorithm by the group subproblem approach

Although incorporating the third stage provides a greater flexibility to improve the solution quality, the three-stage stochastic programming model may lead to large decision trees, that could require significant computation time to solve the overall stochastic model. In order to mitigate the computational effort, in the previous section, we provide valid inequalities and handle the nonlinearity using a mixed integer second order cone programming. However, the overall three-stage model cannot be solved for large number of scenarios. Therefore, in this section, in order to obtain lower and upper bounds for the optimal value of our problem, we suggest a scenario group-wise decomposition algorithm which is an extended version of the approach proposed by Sandikçi and Özaltın (2017) for multi-stage integer stochastic optimization problems. This approach allows us to decompose the problem into group subproblems that are smaller in size and solvable with less computational effort. In Section 3.1, we describe the scenario group subproblem approach to obtain a lower bound. Then, in Section 3.2, we give our scenario group-wise decomposition algorithm to obtain lower and upper bounds for the optimal value of the proposed model.

3.1. Lower bound from the group subproblems

In this section, our aim is to derive a lower bound by solving a number of group subproblems which are similar to our three-stage problem, but smaller in size as discussed in Sandikçi and Özaltın (2017). Each group subproblem contains the same number of stages as the original problem, but is defined over a reduced number of scenarios. The aim of this scenario group-wise decomposition is to mitigate computational effort. Let Ω_n be the scenarios considered in group subproblem

$n \in N$, where N represents the set of group subproblems. To obtain a lower bound, we ensure that $\bigcup_{n \in N} \Omega_n = \Omega$. Moreover, we adjust the probabilities of the scenarios in each group subproblem to make sure that sum of the adjusted probabilities equals to one. Let $p(\omega)$ be the probability of the scenario $\omega \in \Omega$ and $mc(\omega)$ count the number of group subproblems in which scenario ω appears. Then adjusted probability of scenario ω in group subproblem $n \in N$, represented by $\hat{p}_n(\omega)$, can be calculated as follows,

$$\hat{p}_n(\omega) = \frac{p(\omega)/mc(\omega)}{p(\Omega_n)}, \quad (46)$$

where $p(\Omega_n) = \sum_{\omega \in \Omega_n} p(\omega)/mc(\omega)$. Let $LB(N)$ denote the weighted average of the optimal values of group subproblems. It is defined as follows:

$$LB(N) = \sum_{n \in N} p(\Omega_n) \cdot obj(\Omega_n),$$

where $obj(\Omega_n)$ represents the optimal value of group subproblem $n \in N$.

The following proposition states that $LB(N)$ is a lower bound for the optimal value obj_{MISOCP} of the original problem.

Proposition 5. (Proposition 1 in Sandikçi and Özaltın (2017).) $LB(N) \leq obj_{MISOCP}$ for any set of group subproblems N satisfying $\bigcup_{n \in N} \Omega_n = \Omega$.

The steps of the lower bounding algorithm are given in Algorithm 1. The quality of the lower bound significantly depends

Algorithm 1 Lower bounding algorithm.

Initialization: Given a scenario set Ω and corresponding probabilities $p(\omega)$, $\omega \in \Omega$, define the sets Ω_n , $n \in N$.

for each group subproblem $n \in N$ **do**

for each $\omega \in \Omega_n$ **do**

 Calculate $\hat{p}_n(\omega)$ as in equation (46).

 Solve the MISOCP model in Section 2.4 by considering the scenario set Ω_n and corresponding probabilities $\hat{p}_n(\omega)$, $\omega \in \Omega_n$.

 Set $obj(\Omega_n)$ to the optimal value of the group subproblem n .

 Calculate $LB(N) = \sum_{n \in N} p(\Omega_n) \cdot obj(\Omega_n)$.

on the choice of subsets Ω_n , $n \in N$. We are interested in finding subsets Ω_n , $n \in N$ that return a tighter lower bound. We therefore devise some grouping techniques and compare their performances in the experimental study provided in Section 5. Our scenario grouping strategies can be found in Appendix A.3.

3.2. Upper bound from the group subproblems

In this section, we again utilize the scenario group subproblems, defined in the previous section, to obtain an upper bound for the optimal value of the proposed model MISOCP. In order to proceed with scenario group-wise decomposition algorithm, it is useful to discuss the decision process and scenario tree in detail. In this paper, we consider the random process (H, Z) , where H is the random demand and Z is the random non-cruise time. The uncertain data H and Z are gradually revealed over time, first H is observed and then Z is observed. The first-stage decisions are made before any uncertainty is revealed, the second-stage decisions are made after the random demand H is observed and the third-stage decisions are made after all uncertainty is resolved, in other words both H and Z are observed. Therefore, the scenario tree has three levels. At the first level, we have only one root node in which first-stage decisions are made. Let the first-stage decision variable $x^1 \in \mathbb{R}^{|J|}$ denote a vector of announced departure times x_i^1 , $i \in J$. The value of the decision vector x^1 is determined without knowing the future observations (η, ζ) of (H, Z) . The values of the second-stage decisions depend on the revealed information of H . Therefore, at the second level of scenario tree, the number of nodes is equal to the number of different demand realizations η of H . An arc connects each node of the second level with the root node. For each node of the second level (each node corresponds to a particular realization η of H), we create nodes as the number of non-cruise time realizations ζ of Z . We connect them with the node η by an arc. The values of the third-stage decisions depend on the realization (η, ζ) of (H, Z) .

Let $\hat{x}^1 \in \mathbb{R}^{|J|}$ denote a fixed value of x^1 . Solving the group subproblems proposes a number of first-stage decisions $\hat{x}_n^1 \in \mathbb{R}^{|J|}$, each is a solution of a group subproblem $n \in N$. Each first-stage decision \hat{x}_n^1 can be viewed as a candidate feasible first-stage solution to the three-stage stochastic model MISOCP. When we append one of these decisions to MISOCP, if the remaining problem is feasible, then the optimal value of it gives an upper bound for the optimal value of MISOCP. Note that for given \hat{x}_n^1 , the remaining problem (scenario tree) decomposes in each demand realization η of H . Under a demand realization η with a fixed first-stage decision \hat{x}_n^1 , the problem turns into a two-stage stochastic problem. We refer this constructed problem as *two-stage SP*(\hat{x}_n^1, η). Our aim is to solve the constructed *two-stage SP*(\hat{x}_n^1, η) models - one for each demand realization η with a fixed first-stage decision \hat{x}_n^1 , $n \in N$ to obtain an upper bound for MISOCP.

The *two-stage* $SP(\hat{x}_n^1, \eta)$ has pure binary decisions in its first-stage (i.e., second-stage decisions $z_{ijk}^2(\omega)$, $j \in J$, $i \in U_j$, $\omega \in \Omega$; $y_{ik}^2(\omega)$, $u_{ik}^2(\omega)$, $i \in J$, $k \in K$, $\omega \in \Omega$ of the original three-stage model), whereas the second stage (i.e., third-stage of the original model) involves both continuous and binary decisions. To solve each *two-stage* $SP(\hat{x}_n^1, \eta)$, we use the exact solution method suggested by Ahmed (2013). This approach is based on a simple scenario decomposition algorithm and uses the binary nature of the first-stage variables. In that approach, the single scenario subproblems are solved. However, if the problem has a large number of scenarios, many single-scenario subproblems need to be solved and therefore, a large number of proposed candidate first-stage decisions should be evaluated. For computational tractability, we suggest solving group subproblems in order to obtain fewer number of first-stage decisions. As well, the lower bound obtained by solving group subproblems would be tighter than the lower bound of the simple scenario decomposition approach suggested by Ahmed (2013). We call the group subproblems obtained from the *reduced model two-stage* $SP(\hat{x}_n^1, \eta)$ as *reduced group subproblems*. We also refer the proposed candidate first-stage decisions of *reduced group subproblem* $r \in R$ as $\hat{z}_r^2(\hat{x}_n^1, \eta)$, $\hat{y}_r^2(\hat{x}_n^1, \eta)$, and $\hat{u}_r^2(\hat{x}_n^1, \eta)$. Afterward, for each $r \in R$, we impose candidate $\hat{z}_r^2(\hat{x}_n^1, \eta)$, $\hat{y}_r^2(\hat{x}_n^1, \eta)$, $\hat{u}_r^2(\hat{x}_n^1, \eta)$ decisions upon the *two-stage* $SP(\hat{x}_n^1, \eta)$. Then, *two-stage* $SP(\hat{x}_n^1, \eta)$ decomposes in each realizations ζ of Z . Note that, each single scenario subproblem is a deterministic problem and can be referred as $detpr(\hat{z}_r^2(\hat{x}_n^1, \eta), \hat{y}_r^2(\hat{x}_n^1, \eta), \hat{u}_r^2(\hat{x}_n^1, \eta), \zeta)$. Given $\hat{z}_r^2(\hat{x}_n^1, \eta)$, $\hat{y}_r^2(\hat{x}_n^1, \eta)$, $\hat{u}_r^2(\hat{x}_n^1, \eta)$, the single scenario deterministic problems are solved to obtain an upper bound $UB^{red}(\hat{x}_n^1, \eta)$ to the reduced model *two-stage* $SP(\hat{x}_n^1, \eta)$. The crucial point of the algorithm is to cut-off explored decisions $\hat{z}_r^2(\hat{x}_n^1, \eta)$ from the feasible set $X(\hat{x}_n^1, \eta)$ of the *two-stage* $SP(\hat{x}_n^1, \eta)$. In order to accomplish this, we use the idea of *no good cuts* suggested by Ahmed (2013) and add the following no good cuts to each reduced group subproblem $r \in R$ of *two-stage* $SP(\hat{x}_n^1, \eta)$:

$$\sum_{i,j,k:\hat{z}_{ijk}^2(\hat{x}_n^1, \eta)=1} (1 - z_{ijk}^2) + \sum_{i,j,k:\hat{z}_{ijk}^2(\hat{x}_n^1, \eta)=0} z_{ijk}^2 \geq 1, \quad \hat{z}^2(\hat{x}_n^1, \eta) \in \hat{S}, \quad (47)$$

where $\hat{S} := \bigcup_{r \in R} \{\hat{z}_r^2(\hat{x}_n^1, \eta)\}$, i.e., the set of evaluated candidate decisions. We do not need to add an additional cut to eliminate the rest of the decisions $(\hat{y}_r^2(\hat{x}_n^1, \eta), \hat{u}_r^2(\hat{x}_n^1, \eta))$, since $\hat{z}_r^2(\hat{x}_n^1, \eta)$ determines both $\hat{y}_r^2(\hat{x}_n^1, \eta)$ and $\hat{u}_r^2(\hat{x}_n^1, \eta)$.

We provide a scenario group-wise decomposition algorithm in Algorithm 2 to solve the proposed three-stage stochas-

Algorithm 2 Scenario group-wise decomposition algorithm.

```

Apply Lower Bounding Algorithm
for each candidate first-stage decision  $\hat{x}_n^1, n \in N$  do
  for each demand realization  $\eta$  do
    Initialize:  $\text{improvement} \leftarrow \text{true}$ ,  $UB^{red}(\hat{x}_n^1, \eta) \leftarrow +\infty$ ,  $LB^{red}(\hat{x}_n^1, \eta) \leftarrow -\infty$ ,  $\hat{S} \leftarrow \emptyset$ 
    while  $\text{improvement}$  do
      Set  $\hat{X}(\hat{x}_n^1, \eta) \leftarrow X(\hat{x}_n^1, \eta) \setminus \hat{S}$ 
      Construct the set  $R$  of group subproblems.
      Apply Lower Bounding Algorithm with set  $R$  to solve two-stage  $SP(\hat{x}_n^1, \eta)$ 
      Set  $LB^{red}(\hat{x}_n^1, \eta)$  to the  $LB(R)$  of Lower Bounding Algorithm
      Set  $\hat{S} \leftarrow \emptyset$ 
      for each reduced group subproblem  $r \in R$  do
        Set  $\hat{S} \leftarrow \hat{S} \cup \{\hat{z}_r^2(\hat{x}_n^1, \eta)\}$ 
        Set  $\hat{S} \leftarrow \hat{S} \cup \{(\hat{z}_r^2(\hat{x}_n^1, \eta), \hat{y}_r^2(\hat{x}_n^1, \eta), \hat{u}_r^2(\hat{x}_n^1, \eta))\}$ 
      for each  $(\hat{z}^2, \hat{y}^2, \hat{u}^2) \in \hat{S}$  do
        for each realization  $\zeta$  of  $Z$  do
          Solve  $detpr(\hat{z}^2, \hat{y}^2, \hat{u}^2, \zeta)$ 
          Set  $f(\hat{z}^2, \hat{y}^2, \hat{u}^2, \zeta)$  to the optimal value of  $detpr(\hat{z}^2, \hat{y}^2, \hat{u}^2, \zeta)$ 
        Update  $UB^{red}(\hat{x}_n^1, \eta) \leftarrow \min(UB^{red}(\hat{x}_n^1, \eta), \min_{(\hat{z}^2, \hat{y}^2, \hat{u}^2) \in \hat{S}} (\sum_{\zeta \in Z} f(\hat{z}^2, \hat{y}^2, \hat{u}^2, \zeta)))$ 
      if  $LB^{red}(\hat{x}_n^1, \eta) > UB^{red}(\hat{x}_n^1, \eta)$  then
         $\text{improvement} \leftarrow \text{false}$ 
    Set  $UB \leftarrow \min_{n \in N} (\sum_{\eta} UB^{red}(\hat{x}_n^1, \eta))$ 

```

tic model MISOCP. The algorithm starts with applying the *Lower Bounding Algorithm* to propose a lower bound and a set of candidate first-stage decisions \hat{x}_n^1 , $n \in N$ for MISOCP. For each \hat{x}_n^1 , $n \in N$ and demand realization $\eta \in H$, the aim of the algorithm is to solve the reduced model *two-stage* $SP(\hat{x}_n^1, \eta)$ to obtain an upper bound UB for MISOCP. In order to find an optimal solution to *two-stage* $SP(\hat{x}_n^1, \eta)$, our algorithm uses a scenario group decomposition approach. The algorithm solves a number of *reduced group subproblems* to propose candidate solutions $\hat{z}_r^2(\hat{x}_n^1, \eta)$, $\hat{y}_r^2(\hat{x}_n^1, \eta)$, and $\hat{u}_r^2(\hat{x}_n^1, \eta)$, $r \in R$ to the reduced model. Furthermore, using the optimal values of the reduced group subproblems, a lower bound $LB^{red}(\hat{x}_n^1, \eta)$ for *two-stage* $SP(\hat{x}_n^1, \eta)$ is calculated as described in Section 3.1. However, for given \hat{x}_n^1 , if any *reduced group subproblem* is infeasible, then $LB^{red}(\hat{x}_n^1, \eta)$ is set to $+\infty$. We then iteratively append the candidate solutions $(\hat{z}_r^2(\hat{x}_n^1, \eta), \hat{y}_r^2(\hat{x}_n^1, \eta), \hat{u}_r^2(\hat{x}_n^1, \eta))$, $r \in R$ to the reduced model *two-stage* $SP(\hat{x}_n^1, \eta)$ and solve the model for the remaining decision variables, i.e., we solve single scenario $detpr(\hat{z}_r^2(\hat{x}_n^1, \eta), \hat{y}_r^2(\hat{x}_n^1, \eta), \hat{u}_r^2(\hat{x}_n^1, \eta), \zeta)$, $r \in R$, $\zeta \in Z$ models. The best of the candidate solutions, the one giving mini-

imum objective function value, provides an upper bound $UB^{red}(\hat{x}_n^1, \eta)$ for two-stage $SP(\hat{x}_n^1, \eta)$. Then, we cut-off each evaluated solution from the feasible region $X(\hat{x}_n^1, \eta)$ by adding no-good cuts (47) to the reduced group subproblems. The algorithm repeats the same steps until $LB^{red}(\hat{x}_n^1, \eta)$ becomes greater than $UB^{red}(\hat{x}_n^1, \eta)$. Since the feasible region of each reduced group subproblem gets strictly smaller after adding no-good cuts at each iteration, $LB^{red}(\hat{x}_n^1, \eta)$ is nondecreasing. When these steps are completed for each demand realization η , the algorithm computes an upper bound UB for MISOCP.

The Algorithm 2 terminates in a finite number of iterations, since the number of feasible second stage solutions is finite and the number of candidate first-stage decisions is as the number of group subproblems, which is also finite. Moreover, under each demand realization η and each fixed first-stage decision \hat{x}_n^1 , the algorithm detects an optimal solution for the reduced model two-stage $SP(\hat{x}_n^1, \eta)$. This is the fact that we do not discard any solution without evaluation. Therefore, an optimal solution is either discarded after evaluation or still in the feasible region in any iteration. If it is discarded, then $LB^{red}(\hat{x}_n^1, \eta)$ becomes greater than $UB^{red}(\hat{x}_n^1, \eta)$. The detailed proof can be found in Ahmed (2013).

4. Cutting plane algorithm

The group subproblem approach provided in Section 3 has been recently developed to reduce the computational effort in solving multi-stage stochastic programs. However, if group subproblems still involve a large number of decisions and constraints, then solving each group subproblem may require a significant computation time. In this section, our aim is to mitigate the computational effort by decomposing each group subproblem into problems smaller in size. We develop a cutting plane algorithm (see, Ruszczyński (2006) and references therein for details) to solve group subproblems, which are mixed-integer three-stage stochastic models with a reduced number of scenarios.

Traditionally in stochastic programming literature, two-stage stochastic problems with integer variables in the first stage and continuous decisions in the second stage are solved with the cutting plane (called *L-Shaped* for linear problems) method (see, Birge and Louveaux (1997) for details). We reformulate our proposed three-stage model to have a structure which can be efficiently solved by the cutting plane method. Therefore, we integrate integer variables in the master program (MP) and continuous ones in the subproblems. In other words, we include original first and second stages decisions in the master program. Moreover, we include the binary decisions arising in the third stage of MISOCP in the master program as well. In the subproblems, we leave the original third stage continuous decisions together with second order conic inequalities. Then, we obtain the following formulation:

$$\begin{aligned} \min \quad & \sum_{\omega \in \Omega} \left(\sum_{i \in J} \sum_{k \in K} y_{ik}^2(\omega) \cdot c_{daily}^k \right) \cdot p(\omega) & \text{MP} \\ & + \sum_{\omega \in \Omega} \left(\sum_{i \in J} \sum_{k \in K} \max(0, demand_i(\omega) - CAP_k) \cdot m_{ik}^2(\omega) \cdot c_{sp}^i \right) \cdot p(\omega) \\ & + \sum_{\omega \in \Omega} \left(\sum_{i \in J} \sum_{j \in P_i} c_{dist}^i \cdot pass_{ij}(\omega) \cdot a_{ij}^3(\omega) \right) \cdot p(\omega) \\ & + \sum_{\omega \in \Omega} Z(x^1, m^2(\omega), z^2(\omega), a^3(\omega)) \cdot p(\omega) \\ \text{subject to} \quad & (3) - (12), (21) - (24), (38), \end{aligned}$$

where $x^1 \in \mathbb{R}^{|J|}$ and for $\omega \in \Omega$, $m^2(\omega) \in \{0, 1\}^{|J||K|}$, $z^2(\omega)$ is a vector of $z_j^2(\omega) \in \{0, 1\}^{|D_j||K|}$, $j \in J$, $a^3(\omega)$ is a vector of $a_i^3(\omega) \in \{0, 1\}^{|P_i|}$, $i \in J$, and $Z(x^1, m^2(\omega), z^2(\omega), a^3(\omega))$ is the optimal value of the following subproblem.

$$\begin{aligned} \min \quad & \sum_{i \in J} \sum_{k \in K} S_{ik}^3(\omega) \cdot c_{dle}^k + \sum_{i \in J} c_{del}^i(\omega) \cdot del_i^3(\omega) \\ & + \sum_{i \in J} \sum_{k \in K} (c_{fuel} + c_{CO_2} \cdot \kappa) \cdot (c_1^{ik} \cdot q_{ik}^3(\omega) + c_2^{ik} \cdot \delta_{ik}^3(\omega) + c_3^{ik} \cdot \varphi_{ik}^3(\omega) + c_4^{ik} \cdot \vartheta_{ik}^3(\omega)) \end{aligned} \tag{48}$$

subject to

$$-x_i^1 + d_i^3(\omega) \geq 0 \quad i \in J \tag{49}$$

$$\begin{aligned} d_j^3(\omega) - d_i^3(\omega) - f_{ik}^3(\omega) - S_{ik}^3(\omega) &\leq TA_{ijk} + Ncrs_i(\omega) \\ &+ M_1 \cdot (1 - z_{ijk}^2(\omega)) \quad (i, j) \in PAIR, k \in K \end{aligned} \tag{50}$$

$$d_j^3(\omega) - d_i^3(\omega) - f_{ik}^3(\omega) - S_{ik}^3(\omega) \geq TA_{ijk} + Ncrs_i(\omega)$$

$$-M_1 \cdot (1 - z_{ijk}^2(\omega)) \quad (i, j) \in PAIR, k \in K \tag{51}$$

$$-f_{ik}^l \cdot m_{ik}^2(\omega) + f_{ik}^3(\omega) \geq 0 \quad i \in J, k \in K \tag{52}$$

$$f_{ik}^u \cdot m_{ik}^2(\omega) - f_{ik}^3(\omega) \geq 0 \quad i \in J, k \in K \tag{53}$$

$$d_i^3(\omega) - x_i^1 - del_i^3(\omega) \leq 0 \quad i \in J \tag{54}$$

$$d_j^3(\omega) - d_i^3(\omega) - \sum_{k \in K} f_{ik}^3(\omega) + M \cdot a_{ij}^3(\omega) \geq Ncrs_i(\omega) + TP_{ij} \quad i \in J, j \in P_i \tag{55}$$

$$S_{ik}^3(\omega) \geq 0 \quad i \in J, k \in K \tag{56}$$

$$del_i^3(\omega) \geq 0 \quad i \in J \tag{57}$$

$$\| (2m_{ik}^2(\omega), q_{ik}^3(\omega) - f_{ik}^3(\omega)) \| \leq q_{ik}^3(\omega) + f_{ik}^3(\omega) \quad i \in J, k \in K \tag{58}$$

$$\| (2f_{ik}^3(\omega), \vartheta_{ik}^3(\omega) - m_{ik}^2(\omega)) \| \leq \vartheta_{ik}^3(\omega) + m_{ik}^2(\omega) \quad i \in J, k \in K \tag{59}$$

$$\| (2m_{ik}^2(\omega), \mu_{ik}^3(\omega) - f_{ik}^3(\omega)) \| \leq \mu_{ik}^3(\omega) + f_{ik}^3(\omega) \quad i \in J, k \in K \tag{60}$$

$$\| (2\mu_{ik}^3(\omega), \delta_{ik}^3(\omega) - m_{ik}^2(\omega)) \| \leq \delta_{ik}^3(\omega) + m_{ik}^2(\omega) \quad i \in J, k \in K \tag{61}$$

$$\| (2f_{ik}^3(\omega), v_{ik}^3(\omega) - m_{ik}^2(\omega)) \| \leq v_{ik}^3(\omega) + m_{ik}^2(\omega) \quad i \in J, k \in K \tag{62}$$

$$\| (2v_{ik}^3(\omega), \varphi_{ik}^3(\omega) - f_{ik}^3(\omega)) \| \leq \varphi_{ik}^3(\omega) + f_{ik}^3(\omega) \quad i \in J, k \in K \tag{63}$$

$$q_{ik}^3(\omega), \delta_{ik}^3(\omega), \varphi_{ik}^3(\omega), \vartheta_{ik}^3(\omega), \mu_{ik}^3(\omega), v_{ik}^3(\omega) \geq 0 \quad i \in J, k \in K, \tag{64}$$

where M_1 is a big number. Constraints (50) and (51) are utilized to linearize constraint (15). The problem (48)–(64) is a convex problem with second order conic inequalities (58)–(63) derived in Section 2.4. Note that there exists a point which makes problem (48)–(64) strictly feasible. Since it is also bounded, according to strong duality theorem for second order cone programming problems (see, Ben-Tal and Nemirovski, 2001), we can express problem (48)–(64) in dual form by using second order cone duality. Let C be a second order (convex) cone. Since $C = epi(\|\cdot\|)$ is a convex cone, let's first compactly write the constraints (58)–(63) as:

$$\begin{aligned} & \left(\begin{bmatrix} 2m_{ik}^2(\omega) \\ q_{ik}^3(\omega) - f_{ik}^3(\omega) \end{bmatrix}, q_{ik}^3(\omega) + f_{ik}^3(\omega) \right) \in C_{ik}^1(\omega) \\ & \left(\begin{bmatrix} 2f_{ik}^3(\omega) \\ \vartheta_{ik}^3(\omega) - m_{ik}^2(\omega) \end{bmatrix}, \vartheta_{ik}^3(\omega) + m_{ik}^2(\omega) \right) \in C_{ik}^2(\omega) \\ & \quad \vdots \\ & \left(\begin{bmatrix} 2v_{ik}^3(\omega) \\ \varphi_{ik}^3(\omega) - f_{ik}^3(\omega) \end{bmatrix}, \varphi_{ik}^3(\omega) + f_{ik}^3(\omega) \right) \in C_{ik}^6(\omega). \end{aligned} \tag{65}$$

In order to formulate the dual subproblem by using conic duality, we need to define the dual cones of the second order cones in equation (65).

Definition 1. If C is a cone, then the set

$$C^* = \{y \in \mathbb{R}^n : x^T y \geq 0, \quad \forall x \in C\}$$

is the dual cone of C.

Definition 2. A cone C is called self-dual, if $C^* = C$.

The second-order cone is an example of self-dual cones. Therefore, for $i \in J, k \in K, \omega \in \Omega, s = 1, \dots, 6$, the corresponding dual cone $C_{ik}^{*s}(\omega)$ of each second-order cone $C_{ik}^s(\omega)$ in equation (65) appears as a second-order cone in the dual problem of (48)–(64). The dual cones $C_{ik}^{*s}(\omega), s = 1, \dots, 6$ with associated dual variables $\gamma_{ik}^{10}(\omega), \gamma_{ik}^{11}(\omega), \alpha_{ik}^1(\omega), \gamma_{ik}^{12}(\omega), \gamma_{ik}^{13}(\omega), \alpha_{ik}^2(\omega), \gamma_{ik}^{14}(\omega), \gamma_{ik}^{15}(\omega), \alpha_{ik}^3(\omega), \gamma_{ik}^{16}(\omega), \gamma_{ik}^{17}(\omega), \alpha_{ik}^4(\omega), \gamma_{ik}^{18}(\omega), \gamma_{ik}^{19}(\omega), \alpha_{ik}^5(\omega), \gamma_{ik}^{20}(\omega), \gamma_{ik}^{21}(\omega), \alpha_{ik}^6(\omega)$ can be expressed as follows:

$$\begin{aligned} & \left(\begin{bmatrix} \gamma_{ik}^{10}(\omega) \\ \gamma_{ik}^{11}(\omega) \end{bmatrix}, \alpha_{ik}^1(\omega) \right) \in C_{ik}^{*1}(\omega) \\ & \left(\begin{bmatrix} \gamma_{ik}^{12}(\omega) \\ \gamma_{ik}^{13}(\omega) \end{bmatrix}, \alpha_{ik}^2(\omega) \right) \in C_{ik}^{*2}(\omega) \\ & \quad \vdots \\ & \left(\begin{bmatrix} \gamma_{ik}^{20}(\omega) \\ \gamma_{ik}^{21}(\omega) \end{bmatrix}, \alpha_{ik}^6(\omega) \right) \in C_{ik}^{*6}(\omega). \end{aligned}$$

For $\omega \in \Omega, i \in J, j \in J, k \in K$, let $\gamma_i^1(\omega), \gamma_{ijk}^2(\omega), \gamma_{ijk}^3(\omega), \gamma_{ik}^4(\omega), \gamma_{ik}^5(\omega), \gamma_i^6(\omega), \gamma_{ij}^7(\omega)$ denote the dual variables associated with the constraints (49)–(55), respectively. Then, using the second order cone duality, the dual of problem (48)–(64) can be expressed as:

$$\begin{aligned} \max \quad & \sum_{i \in J} x_i^1 \cdot (\gamma_i^1(\omega) - \gamma_i^6(\omega)) \quad (DP(x^1, m^2(\omega), z^2(\omega), a^3(\omega))) \\ & + \sum_{i \in J} \sum_{j \in D_i} \sum_{k \in K} (\gamma_{ijk}^3(\omega) - \gamma_{ijk}^2(\omega)) \cdot (TA_{ijk} + Ncrs_i(\omega)) \\ & - \sum_{i \in J} \sum_{j \in D_i} \sum_{k \in K} (\gamma_{ijk}^3(\omega) + \gamma_{ijk}^2(\omega)) \cdot M_1 \cdot (1 - z_{ijk}^2(\omega)) \\ & + \sum_{i \in J} \sum_{k \in K} m_{ik}^2(\omega) \cdot (\gamma_{ik}^4(\omega) \cdot f_{ik}^l - \gamma_{ik}^5(\omega) \cdot f_{ik}^u + 2 \cdot \gamma_{ik}^{10}(\omega) - \gamma_{ik}^{13}(\omega)) \\ & + \sum_{i \in J} \sum_{k \in K} m_{ik}^2(\omega) \cdot (-\alpha_{ik}^2(\omega) + 2 \cdot \gamma_{ik}^{14}(\omega) - \gamma_{ik}^{17}(\omega)) \\ & + \sum_{i \in J} \sum_{k \in K} m_{ik}^2(\omega) \cdot (-\alpha_{ik}^4(\omega) - \gamma_{ik}^{19}(\omega) - \alpha_{ik}^5(\omega)) \\ & + \sum_{i \in J} \sum_{j \in P_i} \gamma_{ij}^7(\omega) \cdot (Ncrs_i(\omega) + TP_{ij} - M \cdot a_{ij}^3(\omega)) \end{aligned} \tag{66}$$

subject to

$$\alpha_{ik}^1(\omega) - \gamma_{ik}^{11}(\omega) \leq (c_{fuel} + c_{CO_2} \cdot \kappa) \cdot c_{ik}^1 \quad i \in J, k \in K \tag{67}$$

$$\alpha_{ik}^4(\omega) - \gamma_{ik}^{17}(\omega) \leq (c_{fuel} + c_{CO_2} \cdot \kappa) \cdot c_{ik}^2 \quad i \in J, k \in K \tag{68}$$

$$\alpha_{ik}^6(\omega) - \gamma_{ik}^{21}(\omega) \leq (c_{fuel} + c_{CO_2} \cdot \kappa) \cdot c_{ik}^3 \quad i \in J, k \in K \tag{69}$$

$$\alpha_{ik}^2(\omega) - \gamma_{ik}^{13}(\omega) \leq (c_{fuel} + c_{CO_2} \cdot \kappa) \cdot c_{ik}^4 \quad i \in J, k \in K \tag{70}$$

$$\gamma_i^6(\omega) \leq c_{del}^i(\omega) \quad i \in J \tag{71}$$

$$\begin{aligned} & \gamma_i^1(\omega) + \sum_{j \in D_i} \sum_{k \in K} \gamma_{ijk}^2(\omega) - \sum_{j \in U_i} \sum_{k \in K} \gamma_{jik}^2(\omega) - \sum_{j \in D_i} \sum_{k \in K} \gamma_{ijk}^3(\omega) + \sum_{j \in U_i} \sum_{k \in K} \gamma_{jik}^3(\omega) \\ & - \gamma_i^6(\omega) - \sum_{j \in P_i} \gamma_{ij}^7 + \sum_{j \in C_i} \gamma_{ji}^7 \leq 0 \quad i \in J \end{aligned} \tag{72}$$

$$\alpha_{ik}^5(\omega) - 2 \cdot \gamma_{ik}^{20}(\omega) - \gamma_{ik}^{19}(\omega) \leq 0 \quad i \in J, k \in K \tag{73}$$

$$\alpha_{ik}^3(\omega) - 2 \cdot \gamma_{ik}^{16}(\omega) - \gamma_{ik}^{15}(\omega) \leq 0 \quad i \in J, k \in K \tag{74}$$

$$\sum_{j \in D_i} \gamma_{ijk}^2(\omega) - \sum_{j \in D_i} \gamma_{ijk}^3(\omega) \leq c_{idte}^k \quad i \in J, k \in K \tag{75}$$

$$\alpha_{ik}^6(\omega) + \gamma_{ik}^{21}(\omega) - 2 \cdot \gamma_{ik}^{18}(\omega) + \alpha_{ik}^3(\omega) + \gamma_{ik}^{15}(\omega) - 2 \cdot \gamma_{ik}^{12}(\omega) + \alpha_{ik}^1(\omega) + \gamma_{ik}^{11}(\omega) - \sum_{j \in P_i} \gamma_{ij}^7(\omega) - \gamma_{ik}^5(\omega) + \gamma_{ik}^4(\omega) + \sum_{j \in D_i} \gamma_{ijk}^2(\omega) - \sum_{j \in D_i} \gamma_{ijk}^3(\omega) \leq 0 \quad i \in J, k \in K \tag{76}$$

$$\|(\gamma_{ik}^{10}(\omega), \gamma_{ik}^{11}(\omega))\| \leq \alpha_{ik}^1(\omega) \quad i \in J, k \in K \tag{77}$$

$$\|(\gamma_{ik}^{12}(\omega), \gamma_{ik}^{13}(\omega))\| \leq \alpha_{ik}^2(\omega) \quad i \in J, k \in K \tag{78}$$

$$\|(\gamma_{ik}^{14}(\omega), \gamma_{ik}^{15}(\omega))\| \leq \alpha_{ik}^3(\omega) \quad i \in J, k \in K \tag{79}$$

$$\|(\gamma_{ik}^{16}(\omega), \gamma_{ik}^{17}(\omega))\| \leq \alpha_{ik}^4(\omega) \quad i \in J, k \in K \tag{80}$$

$$\|(\gamma_{ik}^{18}(\omega), \gamma_{ik}^{19}(\omega))\| \leq \alpha_{ik}^5(\omega) \quad i \in J, k \in K \tag{81}$$

$$\|(\gamma_{ik}^{20}(\omega), \gamma_{ik}^{21}(\omega))\| \leq \alpha_{ik}^6(\omega) \quad i \in J, k \in K \tag{82}$$

$$\gamma_i^1(\omega), \gamma_i^6(\omega) \geq 0 \quad i \in J \tag{83}$$

$$\gamma_{ijk}^2(\omega), \gamma_{ijk}^3(\omega) \geq 0 \quad (i, j) \in PAIR, k \in K \tag{84}$$

$$\gamma_{ik}^4(\omega), \gamma_{ik}^5(\omega) \geq 0 \quad i \in J, k \in K \tag{85}$$

$$\gamma_{ij}^7(\omega) \geq 0 \quad i \in J, j \in P_i \tag{86}$$

$$\gamma_{ik}^{10}(\omega), \gamma_{ik}^{12}(\omega), \gamma_{ik}^{14}(\omega), \gamma_{ik}^{16}(\omega), \gamma_{ik}^{18}(\omega), \gamma_{ik}^{20}(\omega), \alpha_{ik}^1(\omega), \alpha_{ik}^2(\omega), \alpha_{ik}^3(\omega), \alpha_{ik}^4(\omega), \alpha_{ik}^5(\omega), \alpha_{ik}^6(\omega) \geq 0 \quad i \in J, k \in K \tag{87}$$

The feasible set of the dual subproblem (66)–(87) is nonempty and bounded. Furthermore, dual subproblem (66)–(87) and primal subproblem (48)–(64) are convex optimization problems and by strong duality, they attain the same optimal value.

It is important to note that feasible region of the dual formulation does not depend on the value of $(x^1, m^2(\omega), z^2(\omega), a^3(\omega))$, which only affects the objective function. Therefore, instead of dealing with the primal subproblem, we need to solve the dual subproblem $DP(x^1, m^2(\omega), z^2(\omega), a^3(\omega))$.

Let we have an index set \mathcal{T} of feasible solutions to MP. By an abuse of notation, for each $t \in \mathcal{T}$ and $\omega \in \Omega$, let $\alpha^t(\omega)$ and $\gamma^t(\omega)$ be an optimal solution to the dual subproblem for $(x^{t1}, m^{t2}(\omega), z^{t2}(\omega), a^{t3}(\omega))$. By introducing auxiliary variables v_ω , $\omega \in \Omega$ to approximate the third-stage objective values, we can reformulate the proposed model as the following relaxed master program (RMP):

$$\begin{aligned}
\min \quad & \sum_{\omega \in \Omega} \left(\sum_{i \in J} \sum_{k \in K} \max(0, \text{demand}_i(\omega) - \text{CAP}_k) \cdot m_{ik}^2(\omega) \cdot c_{sp}^i \right) \cdot p(\omega) & \text{(RMP)} \\
& + \sum_{\omega \in \Omega} \left(\sum_{i \in J} \sum_{k \in K} y_{ik}^2(\omega) \cdot c_{daily}^k \right) \cdot p(\omega) + \sum_{\omega \in \Omega} \left(\sum_{i \in J} \sum_{j \in P_i} c_{dist}^i \cdot \text{pass}_{ij}(\omega) \cdot a_{ij}^3(\omega) \right) \cdot p(\omega) \\
& + \sum_{\omega \in \Omega} p(\omega) \cdot v_\omega & \text{(88)}
\end{aligned}$$

subject to

$$\begin{aligned}
v_\omega \geq & \sum_{i \in J} x_i^1 \cdot (\gamma_i^{t1}(\omega) - \gamma_i^{t6}(\omega)) + \sum_{i \in J} \sum_{j \in D_i} \sum_{k \in K} (\gamma_{ijk}^{t3}(\omega) - \gamma_{ijk}^{t2}(\omega)) \cdot (TA_{ijk} + \text{Ncrs}_i(\omega)) \\
& - \sum_{i \in J} \sum_{j \in D_i} \sum_{k \in K} (\gamma_{ijk}^{t3}(\omega) + \gamma_{ijk}^{t2}(\omega)) \cdot (M_1 \cdot (1 - z_{ijk}^2(\omega))) + \sum_{i \in J} \sum_{k \in K} m_{ik}^2(\omega) \cdot (2 \cdot \gamma_{ik}^{t14}(\omega)) \\
& + \sum_{i \in J} \sum_{k \in K} m_{ik}^2(\omega) \cdot (2 \cdot \gamma_{ik}^{t10}(\omega) - \gamma_{ik}^{t13}(\omega)) + \sum_{i \in J} \sum_{k \in K} m_{ik}^2(\omega) \cdot (\gamma_{ik}^{t4}(\omega) \cdot f_{ik}^l - \gamma_{ik}^{t5}(\omega) \cdot f_{ik}^u) \\
& + \sum_{i \in J} \sum_{k \in K} m_{ik}^2(\omega) \cdot (-\gamma_{ik}^{t17}(\omega) - \alpha_{ik}^{t4}(\omega) - \gamma_{ik}^{t19}(\omega) - \alpha_{ik}^{t5}(\omega) - \alpha_{ik}^{t2}(\omega)) \\
& + \sum_{i \in J} \sum_{j \in P_i} \gamma_{ij}^{t7}(\omega) \cdot (\text{Ncrs}_i(\omega) + TP_{ij} - M \cdot a_{ij}^3(\omega)) \quad \omega \in \Omega, t \in \mathcal{T} & \text{(89)} \\
& (3) - (12), (21) - (24), (38)
\end{aligned}$$

In RMP, the exact calculation of the third stage objective value, that is $Z(x^1, m^2(\omega), z^2(\omega), a^3(\omega))$ is relaxed. The constraints (89) are called as optimality cuts. Note that, for each $t \in \mathcal{T}$ and $\omega \in \Omega$, $\alpha^t(\omega)$ and $\gamma^t(\omega)$ is a feasible solution to the dual subproblem for any feasible solution $(x^1, m^2(\omega), z^2(\omega), a^3(\omega))$ of MP. Therefore, for each $\omega \in \Omega$, the right hand side of optimality cuts provide a lower approximation of $Z(x^1, m^2(\omega), z^2(\omega), a^3(\omega))$. This implies that the auxiliary variables v_ω , $\omega \in \Omega$, together with the optimality cuts (89), denote appropriate approximations of the third stage objective values. We ensure that for each solution of RMP, every primal subproblem is feasible. Hence, we do not consider any feasibility cut.

In each iteration t of the algorithm, RMP is solved to optimality and an optimality cut for each $\omega \in \Omega$ is added. For an extensive discussion on the cutting plane method and the proofs for convergence, we refer to [Ruszczynski \(2006\)](#) and references therein, and for L-shaped method, we refer to [Birge and Louveaux \(1997\)](#) and [Van Slyke and Wets \(1969\)](#).

4.1. Improvements on cutting plane algorithm

4.1.1. Defining pareto-optimal cuts

In the cutting plane algorithm, convergence time often depends on the quality of the optimality cuts generated. If dual subproblem involves multi (alternative) optimal solutions, each multi optimal solution provides a different cut with different strength. Thus, an important question arises whether one could improve the performance of the cutting plane algorithm through defining the strong (pareto-optimal) cuts. [Magnanti and Wong \(1981\)](#) call a cut pareto optimal if no cut dominates it. Let $(x_0, m_0(\omega), z_0(\omega), a_0(\omega))$, $\omega \in \Omega$ be a core point in the relative interior of the convex hull of feasible first and second stage decision vectors and obj_ω be the optimal value of $DP(x^1, m^2(\omega), z^2(\omega), a^3(\omega))$, $\omega \in \Omega$, where $(x^1, m^2(\omega), z^2(\omega), a^3(\omega))$ is an optimal solution of RMP at iteration t . To generate a pareto-optimal cut, for each $\omega \in \Omega$, the following Magnanti-Wong problem is solved.

$$\begin{aligned}
\max \quad & \sum_{i \in J} x_0^i \cdot (\gamma_i^1(\omega) - \gamma_i^6(\omega)) + \sum_{i \in J} \sum_{j \in P_i} \gamma_{ij}^7(\omega) \cdot (\text{Ncrs}_i(\omega) + TP_{ij} - M \cdot a_{0ij}(\omega)) \\
& + \sum_{i \in J} \sum_{j \in D_i} \sum_{k \in K} (\gamma_{ijk}^3(\omega) - \gamma_{ijk}^2(\omega)) \cdot (TA_{ijk} + \text{Ncrs}_i(\omega)) \\
& - \sum_{i \in J} \sum_{j \in D_i} \sum_{k \in K} (\gamma_{ijk}^3(\omega) + \gamma_{ijk}^2(\omega)) \cdot M_1 \cdot (1 - z_{0ijk}(\omega)) \\
& + \sum_{i \in J} \sum_{k \in K} m_{0ik}(\omega) \cdot (\gamma_{ik}^4(\omega) \cdot f_{ik}^l - \gamma_{ik}^5(\omega) \cdot f_{ik}^u + 2 \cdot \gamma_{ik}^{10}(\omega) - \gamma_{ik}^{13}(\omega) - \alpha_{ik}^2(\omega)) \\
& + \sum_{i \in J} \sum_{k \in K} m_{0ik}(\omega) \cdot (2 \cdot \gamma_{ik}^{14}(\omega) - \gamma_{ik}^{17}(\omega) - \alpha_{ik}^4(\omega) - \gamma_{ik}^{19}(\omega) - \alpha_{ik}^5(\omega)) & \text{(90)}
\end{aligned}$$

subject to

$$\begin{aligned}
 & (67) - (87) \\
 obj_{\omega} = & \sum_{i \in J} x_i^{t1} \cdot (\gamma_i^1(\omega) - \gamma_i^6(\omega)) + \sum_{i \in J} \sum_{j \in P_i} \gamma_{ij}^7(\omega) \cdot (Ncrs_i(\omega) + TP_{ij} - M \cdot a_{ij}^{t3}(\omega)) \\
 & + \sum_{i \in J} \sum_{k \in K} m_{ik}^{t2}(\omega) \cdot (\gamma_{ik}^4(\omega) \cdot f_{ik}^1 - \gamma_{ik}^5(\omega) \cdot f_{ik}^u + 2 \cdot \gamma_{ik}^{10}(\omega) - \gamma_{ik}^{13}(\omega) - \alpha_{ik}^2(\omega)) \\
 & + \sum_{i \in J} \sum_{k \in K} m_{ik}^{t2}(\omega) \cdot (2 \cdot \gamma_{ik}^{14}(\omega) - \gamma_{ik}^{17}(\omega) - \alpha_{ik}^4(\omega) - \gamma_{ik}^{19}(\omega) - \alpha_{ik}^5(\omega)) \\
 & + \sum_{i \in J} \sum_{j \in D_i} \sum_{k \in K} (\gamma_{ijk}^3(\omega) - \gamma_{ijk}^2(\omega)) \cdot (TA_{ijk} + Ncrs_i(\omega)) \\
 & - \sum_{i \in J} \sum_{j \in D_i} \sum_{k \in K} (\gamma_{ijk}^3(\omega) + \gamma_{ijk}^2(\omega)) \cdot M_1 \cdot (1 - z_{ijk}^{t2}(\omega)) \tag{91}
 \end{aligned}$$

The Magnanti-Wong problem chooses the strongest cut associated with optimal dual variables chosen among the alternatives satisfying the constraint (91). However, finding a core point may not be an easy task. Papadakos (2008) shows that $(x0, m0(\omega), z0(\omega), a0(\omega))$, $\omega \in \Omega$ does not have to be a core point. Any point provides a valid optimality cut, since this only changes the objective function without altering the feasible region of Magnanti-Wong problem. Papadakos (2008) suggests following equations to obtain approximate core points:

$$\begin{aligned}
 x0^t & \leftarrow \frac{1}{2}x0^{t-1} + \frac{1}{2}x^t \\
 m0^t(\omega) & \leftarrow \frac{1}{2}m0^{t-1}(\omega) + \frac{1}{2}m^t(\omega), \quad \omega \in \Omega \\
 z0^t(\omega) & \leftarrow \frac{1}{2}z0^{t-1}(\omega) + \frac{1}{2}z^t(\omega), \quad \omega \in \Omega \\
 a0^t(\omega) & \leftarrow \frac{1}{2}a0^{t-1}(\omega) + \frac{1}{2}a^t(\omega), \quad \omega \in \Omega, \tag{92}
 \end{aligned}$$

where $(x^t, m^t(\omega), z^t(\omega), a^t(\omega))$, $\omega \in \Omega$ is an optimal solution of RMP at iteration t . In our computational experiments, for the first iteration, the point $(x0, m0(\omega), z0(\omega), a0(\omega))$, $\omega \in \Omega$ is set equal to an optimal solution of RMP, so that equation (92) gradually approaches to a core point.

However, a numerical unboundedness may occur due to equality constraint (91). Therefore, Papadakos (2008) proposes an enhanced formulation by eliminating constraint (91) from the Magnanti-Wong problem. We denote improved cutting plane algorithm with this enhancement as CP-EMW. The main drawback of CP-EMW algorithm is to solve both dual subproblem $DP(x^{t1}, m^{t2}(\omega), z^{t2}(\omega), a^{t3}(\omega))$ and the Magnanti-Wong problem for each scenario $\omega \in \Omega$ at every iteration t . To accelerate the convergence of the cutting plane method, Sherali and Lunday (2013) propose an algorithm via the generation of maximal nondominated cuts. They use a preemptive priority multiple objective programming to solve the dual subproblems $DP(x^{t1}, m^{t2}(\omega), z^{t2}(\omega), a^{t3}(\omega))$, $\omega \in \Omega$ optimally with the first priority while maximizing (90) among the alternative optimal solutions to the problem. Bayram and Yaman (2018) implement this preemptive priority multiple objective program as the following weighted-sum problem:

$$\begin{aligned}
 \max \quad & (66) - \xi \cdot (90) \quad SL(x^1, m^2(\omega), z^2(\omega), a^3(\omega)) \\
 \text{subject to} \quad & \tag{67) - (87)}.
 \end{aligned}$$

The new objective is the maximization of weighted sum of (66) and (90), where the weight vector is $(1, \xi)$. We take $\xi = 10^{-10}$. At iteration t , we update $(x0^t, m0^t(\omega), z0^t(\omega), a0^t(\omega))$, $\omega \in \Omega$ as described in (92) and replace $(x^1, m^2(\omega), z^2(\omega), a^3(\omega))$ in $SL(x^1, m^2(\omega), z^2(\omega), a^3(\omega))$ with $(x^{t1}, m^{t2}(\omega), z^{t2}(\omega), a^{t3}(\omega))$.

4.1.2. Defining scenario group cuts

In our cutting plane algorithm, each subproblem is defined for each scenario. An optimality cut associated with each scenario can be added to RMP to accelerate convergence, thus resulting in a large number of cuts at each iteration of the cutting plane algorithm. However, this multi-cut version may perform worse due to time required to solve RMP with too many cuts. On the other hand, the single cut version, e.g., aggregating optimality cuts for each scenario, could be loose to approximate the expected third stage objective value, thus resulting in increased number of iterations to converge. To overcome these trade-offs, we use the idea of group subproblems in Section 3.1. We can add reduced number of cuts, one for each group of scenarios, to RMP. Compared to the single cut version, we could obtain a tighter bound to the third stage objective value. We create subsets $\Omega_g \subset \Omega$ of scenarios such that $\bigcup_{g \in G} \Omega_g = \Omega$, where G is the set of scenario groups. It is clear that quality of the cut depends on the scenario grouping. In order to strengthen the bound, we can apply scenario selection procedures in Appendix A.3. In scenario group cut strategy, the multi cuts (89) are replaced by the following group

cuts:

$$\begin{aligned}
 v_g \geq & \sum_{\omega \in \Omega_g} \hat{p}_g(\omega) \left(\sum_{i \in J} x_i \cdot (\gamma_i^{t1}(\omega) - \gamma_i^{t6}(\omega)) + \sum_{i \in J} \sum_{j \in D_i} \sum_{k \in K} (\gamma_{ijk}^{t3}(\omega) - \gamma_{ijk}^{t2}(\omega)) \cdot (TA_{ijk} + Ncrs_i(\omega)) \right) \\
 & - \sum_{\omega \in \Omega_g} \hat{p}_g(\omega) \left(\sum_{i \in J} \sum_{j \in D_i} \sum_{k \in K} (\gamma_{ijk}^{t3}(\omega) + \gamma_{ijk}^{t2}(\omega)) \cdot (M_1 \cdot (1 - z_{ijk}^2(\omega))) \right) \\
 & + \sum_{\omega \in \Omega_g} \hat{p}_g(\omega) \left(\sum_{i \in J} \sum_{k \in K} m_{ik}^2(\omega) \cdot (\gamma_{ik}^{t4}(\omega) \cdot f_{ik}^l - \gamma_{ik}^{t5}(\omega) \cdot f_{ik}^u) + 2 \cdot \gamma_{ik}^{t10}(\omega) - \gamma_{ik}^{t13}(\omega) - \alpha_{ik}^{t2}(\omega) \right) \\
 & + \sum_{\omega \in \Omega_g} \hat{p}_g(\omega) \left(\sum_{i \in J} \sum_{k \in K} m_{ik}^2(\omega) \cdot (2 \cdot \gamma_{ik}^{t14}(\omega) - \gamma_{ik}^{t17}(\omega) - \alpha_{ik}^{t4}(\omega) - \gamma_{ik}^{t19}(\omega) - \alpha_{ik}^{t5}(\omega)) \right) \\
 & + \sum_{\omega \in \Omega_g} \hat{p}_g(\omega) \left(\sum_{i \in J} \sum_{j \in P_i} \gamma_{ij}^{t7}(\omega) \cdot (Ncrs_i(\omega) + TP_{ij} - M \cdot a_{ij}^3(\omega)) \right) \quad g \in G, t \in \mathcal{T} \tag{93}
 \end{aligned}$$

We obtain group cuts in (93) by aggregating the optimality cuts for each scenario in the set Ω_g . To make sure that the sum of probabilities in each set Ω_g is one, we use adjusted probabilities $\hat{p}_g(\omega)$ as defined in Section 3.1. Then, we replace the third stage approximate cost in the objective of RMP by $\sum_{g \in G} v_g \cdot p(\Omega_g)$, where $p(\Omega_g) = \sum_{\omega \in \Omega_g} p(\omega)/mc(\omega)$.

4.1.3. Improved cutting plane algorithm with lazy constraints

Classical cutting plane algorithm involves sequentially solving RMP and subproblems. An optimality cut is added to RMP before it is re-optimized. One of the drawbacks of the classical cutting plane algorithm is that optimally solving RMP with too many optimality cuts may require a significant computational effort. Besides, whenever an optimality cut is added, a new search tree on RMP is constructed, thereby resulting in revisiting candidate solutions that had been already eliminated earlier. Therefore, we utilize the recent development that processes cutting plane algorithm on a single search tree by using a lazy constraint callback feature of CPLEX. In contrast to the classical cutting plane algorithm, we solve subproblems and generate optimality cuts whenever a new candidate incumbent on RMP is proposed by lazy constraint callback feature. Lazy constraints are initially put into a pool, they are not initially a part of the active model. As solutions are generated, the solver checks whether any lazy constraint is violated or not. Then, violated lazy constraints are added to active constraints set. This is a great advantage to reduce the computation time per iteration as the number of optimality cuts generated increases significantly. More detailed information on lazy constraints can be found in Rubin (2011).

We define our improved cutting plane algorithm in Algorithm 3. We invoke CPLEX to solve RMP using the branch-and-cut procedure. In our implementation, whenever CPLEX identifies a candidate incumbent in the search tree, the lazy constraint callback routine is invoked. The goal of the callback routine is to identify the optimality cuts violated by the candidate incumbent and introduce the optimality cuts as lazy constraints or to approve the candidate incumbent as the current incumbent. Let $(\bar{x}^1, \bar{m}^2(\omega), \bar{z}^2(\omega), \bar{a}^3(\omega))$ be the candidate incumbent to RMP. In the callback routine, to generate the optimality cuts, the dual subproblem $DP(\bar{x}^1, \bar{m}^2(\omega), \bar{z}^2(\omega), \bar{a}^3(\omega))$ for each $\omega \in \Omega$ is solved to optimality. If the difference between the optimal value of any subproblem and the current optimal value of v_ω , that is \bar{v}_ω , is greater than ϵ , then we append the optimality cuts (89) associated with the candidate incumbent to RMP as lazy constraints. Here, ϵ is a very small number, which is used in order to handle the numerical precision in computations. Once all lazy constraints appended are satisfied in RMP, we accept the candidate incumbent as the current incumbent. The algorithm continues until the CPLEX determines the optimality of the current incumbent or any user defined termination criteria is satisfied. In the experimental study, we set a time limit as a termination criteria.

Algorithm 3 Improved cutting plane algorithm.

Invoke CPLEX to solve RMP and initiate branch-and-cut procedure

while CPLEX determines the optimality of the current incumbent or user defined termination criteria is satisfied **do**

Identify a candidate incumbent $\bar{x}^1, \bar{m}^2(\omega), \bar{z}^2(\omega), \bar{a}^3(\omega), \bar{v}_\omega, \omega \in \Omega$

for each $\omega \in \Omega$ **do**

Solve dual subproblem $DP(\bar{x}^1, \bar{m}^2(\omega), \bar{z}^2(\omega), \bar{a}^3(\omega))$ **to optimality and let** obj_ω **denote the optimal value**

if $\bar{v}_\omega < obj_\omega - \epsilon$ **then**

Add corresponding optimality cuts of the form (89) associated with the candidate incumbent to RMP as lazy constraints

else

Accept the candidate incumbent as the current incumbent with objective value of RMP

Table 1
Factor values.

Factor Description	Levels	
	Low	High
Base Spill Cost (\$/passenger)	60	200
C_{fuel} (\$/kg)	0.6	1.2

5. Computational results

In this study, we propose a novel three-stage stochastic programming model, which generates a schedule by incorporating operational flight timing decisions along with the fleet assignment and aircraft routing problem. Compared to the existing two stage stochastic fleet assignment problems which only consider passenger demand uncertainty, we additionally incorporate operational flight timing decisions and non-cruise time uncertainty in the third stage of our proposed model. In the experimental study, we first analyze the cost-wise benefits of including the third stage. On the other hand, the third stage brings an additional complexity along with a larger decision tree and nonlinear functions of fuel and emission costs. Therefore, we then analyze the impact of valid inequalities derived in Section 2.3 to efficiently handle the additional complexity of the third stage. Moreover, we test the ability of our scenario group-wise decomposition algorithm to handle the large number of scenarios arising in our three-stage stochastic programming model. Finally, we analyze the effect of improvements in Section 4.1 on the convergence of our cutting plane algorithm.

We use three different schedules generated from the work of Aktürk et al. (2014). The flight information is extracted from database “Airline On-Time Performance Data,” provided by the Bureau of Transportation Statistics of the U.S. Department of Transportation, BTS (2010). We perform the numerical experiments in Sections 5.1 and 5.2 on a OS X Yosemite computer with 8 GB memory and 2.6GHz Intel Core i5 processor. Then, we complete experiments on larger problem instances in Section 5.3 using a workstation with a 3.60GHz Intel Xeon R CPU E5-1650 and 32 GB main memory. All experiments are carried out by implementing proposed MISOCP model, scenario group-wise decomposition and cutting plane algorithms in JAVA programming language with a connection to commercial solver IBM ILOG CPLEX Optimization Studio 12.7.1. In order to analyze the effects of problem parameters, we conduct a 2^k full-factorial experimental design. We provide levels of experimental factors in Table 1.

Base Spill Cost represents the opportunity cost per each spilled passengers due to under capacity of assigned aircraft. We adjust the cost of spilled passengers for each flight using airport congestion coefficients, e.g., favoring the populated markets, as follows:

$$Csp_i = BaseSpillCost \cdot e_{o_i} \cdot e_{Dn_i}, \quad i \in J, \tag{94}$$

where e_{o_i} and e_{Dn_i} represent the congestion coefficients for origin and destination airports of flight $i \in J$ as discussed in Şafak et al. (2017).

C_{fuel} is the price of jet fuel per kilogram (kg) of fuel burned by an aircraft. According to IATA fuel price monitor IATA (2017), the price of fuel per kg is fluctuating between \$0.6 and \$1.2 between years 2008–2017. In this study, we consider fuel prices as \$0.6/kg and \$1.2/kg for lower and higher settings, respectively.

Delay cost represents the cost of passengers’ goodwill due to the flight delays. We calculate at-gate delay cost as a function of passenger demand, as follows:

$$c_{del}^i(\omega) = \beta \cdot demand_i(\omega), \quad i \in J, \omega \in \Omega. \tag{95}$$

In the experimental design, we set β to \$0.08 per minute of delay per passenger. Eurocontrol of Transport Studies University of Westminster (2011) reports that delay costs 0.18 € /minute per passenger for the base case scenario by considering not only flight delays but also passengers disrupted due to misconnection. Since we penalize the misconnections in a separate cost term in the objective function, we set β to a value relatively smaller than 0.18 €. In the computations, we incur \$50 per passenger, when the passenger misses the next flight in his/her itinerary.

We include aircraft daily usage cost by adjusting the payment of purchasing a new aircraft on a daily basis under the assumption that it is regularly used over twenty years. We incorporate additional costs of insurance, hangar rental, taxes, etc., into daily usage cost parameter. We express the fuel burn and carbon emission costs as in equation 2.1. We calculate the coefficients c_s^{ik} , $i \in J, k \in K, s = 1, \dots, 4$ using the corresponding values of fuel burn related parameters as well as the mass and surface in Table 2, as expressed in Şafak et al. (2017). We consider five different aircraft types and list the fuel burn related parameters, corresponding maximum range cruise (MRC) speed, seat capacity, idle time and daily usage cost parameters in Table 2.

We estimate the aircraft turnaround time, TA_{ijk} , needed for an aircraft at an airport using the expression (96):

$$TA_{ijk} = TA_k^{base} \cdot e_{Dn_i} \quad k \in K, (i, j) \in PAIR, \tag{96}$$

where e_{Dn_i} represents the congestion level of destination airport for flight $i \in J$. Therefore, turntime of an aircraft visiting a congested airport takes longer.

Table 2
Aircraft parameters.

Aircraft type	B737 500	MD 83	A320 111	A320 212	B767 300
Capacity (CAP_k)	122	148	172	180	218
Mass (kgs)	50,000	61200	62,000	64,000	135000
Surface (m^2)	105.4	118	122.4	122.6	283.3
$C_{D0, CR}$	0.018	0.0211	0.024	0.024	0.021
$C_{D2, CR}$	0.055	0.0468	0.0375	0.0375	0.049
Cf_1	0.46	0.7462	0.94	0.94	0.763
Cf_2	300	638.59	50,000	100,000	1430
cf_{CR}	1.079	0.9505	1.095	1.06	1.0347
MRC speed (km/h)	859.2	867.6	855.15	868.79	876.70
Base Turntime (min) (TA_k^{base})	36	26	28	30	40
Idle Time Cost (\$)/min (c_{idle}^k)	140	142	136	144	147
Daily Usage Cost (\$) (c_{daily}^k)	25,000	27,300	28,250	29,000	58,550

In this study, we select the announced departure times from a time interval which has already been determined by an airline. We extracted departure time information from database “Airline On-Time Performance Data,” provided by [BTS \(2010\)](#). We determine the lower departure time for a flight by subtracting 10 min from the departure time reported in the database ([BTS, 2010](#)). Similarly, for the upper departure times, we added 10 min to the departure times reported in the database ([BTS, 2010](#)). Thus, we select an announced departure time for a flight from the 20 min time interval determined based on the history of flight departures. We only allow a flight to depart from one of the five discrete points within this time interval. Duration between two consecutive points is taken as 5 min.

We generate a network of passenger connections using the flight departure information of sample schedule used in the computations. We only allow passenger connections between two flights $i \in J$ and $j \in J$, if the departure time of flight j in a sample schedule is within 45 min or 180 min of the arrival time of flight i and destination airport of flight i is same as the origin airport of the flight j . We take passenger connection times, TP_{ij} , $i \in J$, $j \in P_i$ uniformly between 25 and 40 min. We calculate upper bound f_{ik}^u , $i \in J$, $k \in K$ for cruise times by dividing the cruise length of flight i with MRC speed of aircraft type k . Max range cruise (MRC) speed is the minimizer of fuel burn of an aircraft. Although it is ideal to fly at MRC speed, we may prefer to accelerate the aircraft to satisfy the minimum time requirements of passenger connections. In our model, we allow compression on cruise times by 15% of f_{ik}^u .

5.1. Numerical example

We provide a numerical example to show the great advantage of incorporating cruise time decisions, passenger demand and non-cruise time uncertainties within an integrated schedule design, fleet assignment and aircraft routing problem. We use a small sample schedule with 9 flights operated by two aircraft. We extract flight information from the schedule provided by [Şafak et al. \(2017\)](#). We consider randomness for passenger demands for two round way trips {Chicago O'Hare International Airport (ORD)-Dallas/Fort Worth International Airport (DFW)} and {ORD-LaGuardia Airport (LGA)}. The demand realization of a trip (ORD-DFW) in the morning would be different than the demand realization of a flight at noon of the same trip (ORD-DFW). Therefore, for each trip, we segment the passenger demand into three different levels according to the flight departure times of the sample schedule. This segmentation based on flight departure times is explained in detail in [Appendix A.1](#). We assume that the demand levels on all segments belonging to same trip are positively correlated, taking either some “high” or “low” level of demand simultaneously. On the other hand, demands of different round trips are independent. Therefore, such a demand pattern would yield a total of $2^2 = 4$ demand realizations. In addition, we select two key cities (ORD, DFW), e.g., hubs in a hub-and-spoke network. We consider four data points for ORD airport and two points for DFW airport to generate non-cruise time scenarios. Similarly, we assume that “taxi-out”, “departure delay”, “taxi-in” and “arrival delay”, which are the components of non-cruise time of a flight, are positively correlated for the same airport. On the other hand, non-cruise times at different airports are independent. Thus, this pattern yields in $4 \cdot 2 = 8$ non-cruise time realizations. In conclusion, total number of scenarios, which is characterized by multiple demand and non-cruise time realizations, would yield $4 \cdot 8 = 32$ scenarios. More detailed information on scenario generation procedure along with the realization values can be found in [Appendices A.1](#) and [A.2](#).

In this example, we initially start with analyzing what might be the impact of including uncertain passenger demand and non-cruise times in our proposed model. First, we solve the proposed MISOCP model by replacing the passenger demand and non-cruise time data with their expected values. We call it as Deterministic Approach. Then, we solve the proposed three-stage stochastic MISCOP model. It is important to note that we use the same demand and non-cruise time data whose mean values are equal to the values used in the deterministic approach. In [Table 3](#), we compare expected costs for the deterministic and three-stage stochastic approach. We calculate the expected cost of the Deterministic Approach by imposing the first and second stage decisions upon the MISOCP model and optimizing the third stage decisions. In the first column of Deterministic Approach, we consider cruise time as a decision variable as we propose. To analyze the impact of cruise time

Table 3
Comparison of expected costs for the deterministic and three-stage stochastic approach.

Factor		Deterministic				Three-Stage
Spill	C_{fuel}	Speed Control		No Speed Control		Stochastic
		Cost	Incr.(%)	Cost	Incr.(%)	
(\$/pass.)	(\$/kg)					Cost
60	0.6	137,736	0.70	138,312	1.12	136,776
	1.2	180,588	0.65	180,957	0.86	179,417
200	0.6	140,268	2.55	140,844	2.97	136,776
	1.2	183,120	1.92	183,488	2.13	179,670
	Avg:		1.46		1.77	

Table 4
Comparison of expected costs for the deterministic approach with buffer time and three-stage stochastic approach.

Factor		Deterministic with buffer time		Three-stage
Spill	C_{fuel}	No speed control		Stochastic
		Cost	Incr.(%)	
(\$/pass.)	(\$/kg)			Cost
60	0.6	138,731	1.43	136,776
	1.2	181,375	1.09	179,417
200	0.6	141,262	3.28	136,776
	1.2	183,907	2.36	179,670
	Avg:		2.04	

controllability, in the second column, we take the cruise time as a fixed parameter. For each flight $i \in J$, we set the cruise time to f_{ik}^u , $k \in K$ which is the ideal duration when an aircraft type k flies at a speed with minimum fuel burn (MRC speed).

Considering the uncertainty of multiple demand and non-cruise time scenarios may lead to large decisions trees, potentially resulting in significant computation time. Despite the complexity of considering many scenarios, the three-stage stochastic approach leads to an average of 1.77% in cost saving over the deterministic approach. More importantly, cost savings increase for high spill cost setting, since the deterministic approaches do not consider the fluctuations in passenger demand during the planning process. The cost saving could reach to 2.97%, when the deterministic model does not utilize the advantage of cruise time controllability to compensate for the non-cruise time uncertainty.

In the second experiment, we extend the deterministic approach by putting 30 min buffer time into the flight durations. In the current practical solutions of airlines, instead of accelerating the aircraft, airlines may add an additional 30 min to announced flight duration to absorb the non-cruise time uncertainty. In this experiment, we want to see the difference between the deterministic approach with buffer times and our three-stage stochastic approach with controllable cruise times. In Table 4, we compare the resulting schedules.

Although nonlinear function of fuel burn together with many scenarios make the three-stage stochastic approach difficult to solve, it pays off since the average cost saving reaches to 2.04%. If the fuel price is low, the three-stage stochastic approach takes the advantage of cruise time controllability more, then the cost saving increases. It is important to mention, the current deterministic approaches put buffer times without considering the potential scenarios so that some buffer times may be unnecessary for some realizations of non-cruise times, which gives rise to an additional idle time cost. Therefore, it is more cost efficient to determine the amount of idle times and cruise time compressions simultaneously for each scenario as in the three-stage stochastic approach. Furthermore, adding an additional 30 min of buffer time in flight durations adversely affects the aircraft utilization, and could reduce the number of flights that can be assigned to a single aircraft.

In the third experiment, we analyze the impact of incorporating third stage decisions and non-cruise time uncertainty. For this analysis, we compare our proposed model with a two-stage stochastic model. The two-stage model corresponds to the first two stages of our proposed model. i.e., two stage model considers first and second stage decisions and the demand uncertainty, but third level operational flight timing decisions as well as the non-cruise time uncertainty are not taken into account. We calculate the expected cost for the two-stage approach by using optimal solution of two-stage model as an input to the original three stage MISOC model and then optimizing the third stage decisions for each scenario under two settings; one calculates the expected cost by considering cruise times as a third stage decision and the other takes the cruise times as fixed parameters.

Table 5 shows that by incorporating the third stage, the three-stage approach provides an average of 2.97% in cost saving over the two-stage stochastic approach. Even if the fuel price is high, cost saving could reach to 3.21%. Since the existing two-stage stochastic approaches for solving the fleet assignment problem do not take the cruise time decisions and non-cruise time uncertainty into consideration, they fail to adjust the fleeting decisions to reflect the increase in fuel price. However, fuel expenses account for around 30% of the total operating cost. In 2008, the share for fuel costs reached 50%

Table 5
Comparison of expected costs for two-stage and three-stage stochastic approach.

Factor		Two-stage stochastic				Three-Stage
Spill (\$/pass.)	C_{fuel} (\$/kg)	Speed control		No speed control		Stochastic Cost
		Cost	Incr.(%)	Cost	Incr.(%)	
60	0.6	139,779	2.20	140,618	2.81	136,776
	1.2	184,599	2.89	185,172	3.21	179,417
200	0.6	139,779	2.20	140,618	2.81	136,776
	1.2	184,599	2.74	185,172	3.06	179,670
	Avg		2.51		2.97	

Table 6
Comparison of strength of lower bounds.

Factor		LB Algorithm				Cutting Plane Algorithm				CPLEX	
Spill (\$/pass.)	C_{fuel} (\$/kg)	Procedure P1		Procedure P2		Group Cut		Multi Cut		Solver	
		Gap(%)	Cpu(sec)	Gap(%)	Cpu(sec)	Gap(%)	Cpu(sec)	Gap(%)	Cpu(sec)	Gap(%)	Cpu(sec)
60	0.6	0.27	337	0.15	3170	3.04	3170	0.75	3170	2.91	3170
	1.2	0.49	396	0.35	6695	3.05	6695	1.37	6695	3.92	6695
200	0.6	0.50	109	0.42	226	3.83	226	2.85	226	2.06	226
	1.2	0.45	143	0.36	390	4.24	390	2.01	390	1.77	390
	Avg:	0.43		0.32		3.54		1.74		2.66	

due to the sharp increase in the fuel prices (ICAO, 2009). Thus, 3.21% cost saving is significantly important for airlines since this may translate into billions of dollars.

5.2. Analysis on the lower bounds

In this section, we use a sample schedule generated from the study (Aktürk et al., 2014). The schedule has 15 flights operated by 4 different aircraft. We use the same scenario set as in the numerical example in Section 5.1. In this experiment, we compare the lower bounds achieved by solving group subproblems with the lower bounds achieved by the cutting plane (CP) algorithm and CPLEX solver. To be fair in comparison, we set the time limit of CP algorithm and CPLEX solver as much as the time consumed to solve the group subproblems. In Table 6, we report the relative gaps between the lower bound achieved by each approach and the optimal value of the problem. As it is stated, the quality of the lower bound clearly depends on the choice of the group subproblems. For simple case, we start with combining two scenarios in each group as provided in Procedure P1. For each group subproblem, two scenarios are selected in a way that they do not have common demand or non-cruise time realizations. This diversity gives a better representation of overall MISOCP model, thus has a potential to improve the lower bound compared to any scenario selection procedure with common demand or non-cruise time realizations. Then, to increase the variety of scenarios, we implement Procedure P2 in which we extend the group size to four by randomly combining two scenario groups of Procedure P1. Solving the group subproblems larger in size brings an additional complexity, which in turn leads to higher CPU times. Despite the longer CPU times, increasing the number of scenarios in each group leads to smaller optimality gaps, which can be attributed entirely to tighter lower bounds.

In Table 6, we also observe that the optimality gap between the optimal value and the lower bound at termination is higher for both CP algorithm and CPLEX solver. For group subproblem approach, the optimality gap is less than 0.5% for all instances. On the other hand, group cut version of CP algorithm yields an average of 3.5% optimality gap and the optimality gap for CPLEX solver reaches to 4% at termination when fuel price is high. Moreover, CPLEX solver cannot provide an incumbent (feasible) solution for some instances within the time limit even though this experiment is designed with small instances. This experiment clearly demonstrates the effectiveness of the group subproblem approach to obtain a lower bound for the proposed MISOCP model.

We take ten replications to observe if randomly combining scenario groups in Procedure P2 has any impact on the lower bound values. For each replication, we conduct a 2² full-factorial experimental design. In Table 7, for each replication, we report the minimum, average and maximum relative gaps with the lower bound and the optimal value. We could conclude that there is no statistically significant randomization effect on the lower bound values.

To see the effect of each factor on the computation time of the LB algorithm with Procedure P2, we report minimum, average and maximum values of CPU's in seconds over five replications in Table 8. Increase in fuel price makes the fleet assignment decisions more crucial. As expected, CPU time to deal with the nonlinear fuel and emission cost functions increases. If the base spill cost increases, then the solution has a tendency to closely match the seat capacity of aircraft and potential passenger demands with a less consideration of the fuel efficiency of the aircraft. Thus, solution time decreases.

Table 7
Comparison for different replications.

Replications	Optimality Gap(%)		
	Min	Avg	Max
1	0.16	0.33	0.40
2	0.14	0.31	0.38
3	0.19	0.34	0.42
4	0.14	0.30	0.40
5	0.19	0.33	0.41
6	0.15	0.32	0.42
7	0.15	0.32	0.42
8	0.15	0.31	0.37
9	0.15	0.32	0.40
10	0.14	0.31	0.40

Table 8
CPU time analysis of the LB algorithm.

Factor	Level	CPU Time (s)		
		Min	Avg	Max
C_{fuel} (\$/kg)	0.6	102	1384	7886
	1.2	117	3111	8585
Spill (\$/passenger)	60	291	4276	8585
	200	102	219	694

5.3. Analysis on a schedule with 48 flights

In this section, we select the first 48 flights of the sample schedule provided in Şafak et al. (2017). There are 13 aircraft with 5 different types. We consider the same scenario set as in the previous section. The corresponding three-stage stochastic MISOCP formulation has 188,080 rows, 258,839 columns and 924,860 non-zeros after pre-solve is completed. This is a large problem even with 32,584 binary decisions and 70,880 quadratic constraints.

In this experiment, we compare the scenario group-wise decomposition algorithm in Section 3.2 with cutting plane algorithm in Section 4. We implement different improvements suggested in Section 4 within the cutting plane algorithm. We test the impact of each valid inequality derived in Section 2.3 separately. We summarize the proposed algorithms below:

Scenario group-wise decomposition algorithm (SGWD): We implement the scenario group-wise decomposition algorithm (Algorithm 2). First, Algorithm 1 is implemented to obtain a lower bound by solving group subproblems. We apply the scenario grouping Procedure P2. We solve each group subproblems using our cutting plane algorithm (Algorithm 3). Instead of solving each group subproblem to optimality, we set a time limit to 10 min and consider the lower bounds at the termination for each group subproblem. To obtain an upper bound, we evaluate randomly selected two candidate first-stage decisions $\hat{x}_n^1, n \in N$ within the Algorithm 2. For each given \hat{x}_n^1 , we solve reduced group subproblems obtained from the reduced model *two-stage SP* (\hat{x}_n^1, η) using the cutting plane algorithm (Algorithm 3) within 4 min of time limit. Then, single scenario *detpr* ($\hat{z}_r^2(\hat{x}_n^1, \eta), \hat{y}_r^2(\hat{x}_n^1, \eta), \hat{u}_r^2(\hat{x}_n^1, \eta), \zeta$) models are solved to optimality using CPLEX solver with an option to solve a quadratically constrained mixed integer program. In the first column of Table 9, we report the relative gap between the objective function value of the best incumbent solution and the lower bound derived within the scenario group-wise decomposition along with CPU hours.

Cutting plane algorithm (CP): We solve three-stage MISOCP model by implementing cutting plane algorithm (Algorithm 3). We use the optimality cuts derived in (89) by solving the dual subproblem $DP(x^1, m^2(\omega), z^2(\omega), a^3(\omega))$. We test both scenario group cut and multi-cut strategies. Scenarios are grouped using Procedure P1. For consistency, we set the time limit of the cutting plane algorithm to the time consumed by scenario group-wise decomposition algorithm. In the second column of Table 9, we report the relative gaps between the lower and upper bounds provided by the cutting plane algorithm.

Cutting plane algorithm with the Magnanti-Wong enhancement (CP-EMW): In this version of cutting plane algorithm (Algorithm 3), as suggested by Papadakos (2008), we solve an enhanced Magnanti–Wong auxiliary problem which is obtained by eliminating constraint (91) from the Magnanti–Wong problem. At each iteration, we add two types of optimality cuts (89), one is derived by solving the original dual subproblem $DP(x^1, m^2(\omega), z^2(\omega), a^3(\omega))$ and the other is derived by solving the enhanced Magnanti-Wong problem. We update the points $(x0^t, m0^t(\omega), z0^t(\omega), a0^t(\omega))$ in enhanced Magnanti–Wong problem as in equation (92) to approximate core points. We use multi-cut strategy.

Cutting plane algorithm with preemptive priority multiple objective dual subproblem (CP-SL): In this version of cutting plane algorithm, we solve the weighted-sum problem $SL(x^1, m^2(\omega), z^2(\omega), a^3(\omega))$ instead of solving $DP(x^1, m^2(\omega), z^2(\omega), a^3(\omega))$. We give the first priority to the objective of $DP(x^1, m^2(\omega), z^2(\omega), a^3(\omega))$ (66) and consider the objective of Magnanti-

Table 9
Comparison on relative gaps.

Factor			SGWD	CP		CP-EMW		CP-SL		CPU(hr)	
Spill (\$/pass.)	C_{fuel} (\$/kg)	Valid Ineq.	Gap(%)	Group Cut Gap(%)	Multi Cut Gap(%)	Group Cut Gap(%)	Multi Cut Gap(%)	Group Cut Gap(%)	Multi Cut Gap(%)		
60	0.6	No	5.23	5.81	6.55	7.07	6.09	6.47	6.09	6.4	
		(25)	5.30	6.77	5.92	6.71	6.32	6.73	6.06	6.5	
		(26)	5.32	6.55	6.31	6.94	6.40	6.36	6.30	6.4	
		(27)	5.10	6.52	6.72	6.61	6.64	6.01	6.45	6.5	
		1.2	No	5.86	7.30	7.20	6.87	7.56	7.41	7.42	6.4
		(25)	5.72	7.31	7.42	8.01	7.48	7.47	6.92	6.4	
	200	0.6	(26)	5.84	7.34	7.30	7.60	7.43	7.22	7.56	6.4
			(27)	5.73	7.84	7.22	7.31	7.35	7.24	6.66	6.4
			No	5.14	6.45	6.70	6.62	6.76	6.06	6.25	6.4
			(25)	5.15	6.87	7.25	6.84	6.51	6.33	6.53	6.4
			(26)	5.26	6.40	6.66	6.89	6.91	6.23	6.21	6.4
			(27)	5.31	6.61	6.31	7.18	7.41	6.58	6.37	6.5
200	1.2	No	5.98	7.21	7.58	7.31	7.47	7.11	6.84	6.4	
		(25)	5.92	7.13	6.85	7.03	7.80	7.87	7.13	6.4	
		(26)	5.97	7.70	7.61	7.13	7.34	7.00	6.92	6.4	
		(27)	5.95	7.33	7.45	7.35	7.68	7.21	6.82	6.4	
		Avg:	5.55	6.95	6.94	7.09	7.07	6.83	6.66	6.4	

Wong problem (90) with a weight ξ . We take $\xi = 10^{-10}$. We update the points $(x0^t, m0^t(\omega), z0^t(\omega), a0^t(\omega))$ as in equation (92) to approximate core points. We use multi-cut strategy.

The results in Table 9 indicate that our scenario group-wise decomposition algorithm performs the best over all versions of the cutting plane algorithm for each problem instance. While the scenario group-wise decomposition algorithm provides an average of 5.55% optimality gap, cutting plane algorithm yields an optimality gap around 6.94% on the average. It is also clear that the performance of the cutting plane algorithm could be improved by the optimality cuts provided in CP-SL. Furthermore, the results show that including either valid inequality (25) or (27) decreases the optimality gap for many instances.

5.4. Analysis on a schedule with 102 flights

For this computational study, we utilize the sample schedule provided in Şafak et al. (2017). We extract the flight information of the first 102 flights of the schedule. These flights are operated by 26 aircraft with five different types. We again perform 2^2 full-factorial experimental design.

In this section, we use the scenario group-wise decomposition algorithm in Section 3.2 to solve the instances. We obtain a lower bound by solving group subproblems. We employ scenario grouping Procedure P2 so that each group subproblem has four scenarios. We set the time limit for solving each group subproblem to 15 min using the cutting plane algorithm (Algorithm 3). To obtain an upper bound, we evaluate randomly selected two candidate first-stage decisions $\hat{x}_n^1, n \in N$ within the Algorithm 2. For each given \hat{x}_n^1 , we solve reduced group subproblems obtained from the reduced model two-stage $SP(\hat{x}_n^1, \eta)$ within a time limit of 6 min using the cutting plane algorithm (Algorithm 3). Then, we solve the single scenario models $detpr(\hat{z}_r^2(\hat{x}_n^1, \eta), \hat{y}_r^2(\hat{x}_n^1, \eta), \hat{u}_r^2(\hat{x}_n^1, \eta), \zeta)$ to optimality using CPLEX solver. Within the scenario group-wise decomposition algorithm, we employ different versions of cutting plane algorithm for solving group subproblems and reduced group subproblems obtained from the reduced models two-stage $SP(\hat{x}_n^1, \eta)$. The details are provided below.

Scenario group-wise decomposition algorithm with simple cutting plane (CP): In this version, we use the optimality cut (89) derived from solving the original dual subproblem $DP(x^1, m^2(\omega), z^2(\omega), a^3(\omega))$ within the cutting plane algorithm (Algorithm 3).

Scenario group-wise decomposition algorithm with CP-SL: This version differs from the previous one by solving the weighted-sum problem $SL(x^1, m^2(\omega), z^2(\omega), a^3(\omega))$ instead of solving the original dual subproblem $DP(x^1, m^2(\omega), z^2(\omega), a^3(\omega))$ within the cutting plane algorithm (Algorithm 3). We derive the optimality cut (89) associated with the optimal solutions of $SL(x^1, m^2(\omega), z^2(\omega), a^3(\omega))$. We take weight vector as $(1, \xi)$, where $\xi = 10^{-10}$.

In both version, we use the multi cut strategy for the optimality cuts. In Table 10, we report the relative gaps between the lower and upper bounds provided by the scenario group-wise decomposition algorithm. The results indicate that on the average, SGWD with CP-SL can improve the solutions by slightly increasing the CPU times.

6. Conclusion

We propose a novel three-stage stochastic programming model which determines strategic departure time decisions, tactical fleet and routing decisions and the more operational flight timing decisions while examining potential demand and non-cruise time scenarios. The model utilizes the great advantage of cruise time controllability to compensate for non-

Table 10
Comparison on relative gaps.

Factor			Scenario Group-Wise Decomposition Alg.				
Spill (\$/pass.)	C_{fuel} (\$/kg)	Valid Ineq.	with				
			CP		CP-SL		
			Gap(%)	CPU(hr)	Gap(%)	CPU(hr)	
60	0.6	No	6.58	17.7	6.67	18.2	
		(25)	6.68	18.9	6.68	20.7	
		(26)	6.75	18.9	6.54	19.2	
		(27)	6.59	17.2	6.69	19.3	
		1.2	No	7.72	17.8	7.66	18.5
			(25)	7.57	19.3	7.65	18.4
	(26)		7.77	19.2	7.71	18.9	
	200	0.6	(27)	7.87	19.4	7.70	19.1
			No	6.71	20.2	6.66	19.2
			(25)	6.52	18.8	6.56	19.3
			(26)	6.67	19.3	6.55	20.5
			(27)	6.97	18.8	6.70	18.8
1.2			No	7.66	21.2	7.67	20.3
		(25)	7.69	19.7	7.78	18.7	
		(26)	7.86	20.3	7.69	20.1	
		(27)	7.59	19.1	7.89	20.1	
		Avg:	7.20	19.1	7.17	19.3	

cruise time uncertainty. To achieve a minimum cost schedule, the model assigns fleet types through consideration of not only passenger demand, as commonly done in the literature, but also operational expenses such as fuel burn and carbon emission costs. Thus, the major contribution of this paper is to incorporate operational flight timing decisions and different levels of uncertainty into fleet assignment and aircraft routing decisions. In turn, as indicated by our computational results, the resulting three-stage stochastic model yields substantial savings compared to the deterministic and two-stage stochastic programming approaches.

In this paper, we propose different techniques to handle the additional complexity of the third stage decisions in airline scheduling. We provide valid inequalities to strengthen the formulation. We suggest a scenario group-wise decomposition algorithm to handle the large number of scenarios. The algorithm separately solves group subproblems similar to the original three-stage model, but are defined over a reduced number of scenarios. Even the group subproblems can be very time-consuming to solve because of large number of decisions and constraints. Therefore, we implement cutting plane algorithm to efficiently solve each group subproblem. We handle the nonlinear subproblems using the second order cone duality. It is finally worth mentioning that the proposed method can be generalized to solve three-stage stochastic programming models, which include pure binary decisions in their second stages.

There are some future research directions arising from this study. We have demonstrated the importance of scenario selection procedures, but it is still open to evaluate the impact of various selection procedures. Moreover, we currently provide pareto optimality and scenario-group cuts to improve the performance of cutting plane algorithm. Analysing the effect of different cut strategies is of another future interest. Finally, given a novel three-stage stochastic programming approach to airline scheduling problem, one can take advantages of parallel computing technology within the framework of the scenario group-wise decomposition algorithm so that an increased number of scenarios could be explored.

Acknowledgment

The authors thank the editor and two anonymous referees for their constructive comments and suggestions that significantly improved this paper. This research was partially supported by TÜBİTAK under grant 116M542.

Appendix A

A1. Generate potential demand realizations

The fleet assignment problem is typically modeled by considering expected demand values based on historical data and forecasts. However, because of the stochastic nature of passenger demand, certain realizations may result in either spilled passengers, who cannot be accommodated due to under capacity, or unsold seats and presumably higher operational costs. On the other hand, as the competition increases and the profit margins of airlines get in threat, they have to produce solutions that perform well relative to this uncertainty in demand. Hence, we explicitly consider several potential demand scenarios in making fleet and routing decisions.

We segment passenger demand volume with respect to the flight departure times. Passengers may not want to waste their valuable time on a flight during business hours, e.g., business travellers would prefer to depart early in the morning

and going back late at night. Hence, in this segmentation, we consider different demand volumes for different departure time intervals to better reflect reality. We segment three different levels of passenger demand by the flight departure times as follows:

1 st segment:	6:00am - 8:59am and after 5:00pm
2 nd segment:	9:00am - 11:59am and 3:00pm-4:59pm
3 rd segment:	12:00 noon - 2:59pm.

In this manner, we refer time interval (6:00 - 8:59am) and time after 5pm as peak times which are mostly preferred by passengers. After 9am, passenger demand slightly decreases and departures at noon are unfavourable. Afterward, passenger demand increases again and reaches a pick point after 5pm.

Moreover, we consider stochastic passenger demand for three different origin-destination points, ORD-DFW and ORD-LGA. To estimate potential passenger demand realizations, we analyze the seat capacities of assigned aircraft to these flights for a major U.S. airline. We also analyze passenger demand data of [Arikan et al. \(2016\)](#) which are constructed by extracting 10% of the number of passengers provided by airline origin and destination survey (DB1B) ([BTS, 2017b](#)). By combining these analyses, we generate two levels of demand realizations for two origin-destination points, ORD-DFW and ORD-LGA as in [Table A.1](#).

It is important to clarify that, under low realization for ORD-DFW flight, if the departure time is within 6am - 08:59am or after 5pm, we would consider passenger demand as 161. If departure time is within 9am - 11:59am or 3pm-4:59pm, then realization would be 121.

A2. Generate potential non-cruise time realizations

In this study, we separate flight block times into two components as cruise and non-cruise times. In non-cruise times, we include departure and arrival delays as well as the taxi-in and taxi-out times, which are all highly uncertain due to airport congestions. To handle these uncertainties, we analyze potential non-cruise time scenarios, and then set a cruise speed for each non-cruise time realization. Cruise time for each flight is a decision variable. We may set a higher cruise speed to compensate for variations in non-cruise time so as to ensure passenger connections on the subsequent flights and set maximum range cruise (MRC) speed for conservation of fuel.

In this study, we consider uncertain non-cruise times for flights departing from and arriving to Chicago O'Hare International Airport (ORD) and Dallas/Fort Worth International Airport (DFW), which are used as hub airports of major U.S. airlines. To generate potential non-cruise time scenarios, we extracted flight information from the database "Airline On-Time Statistic," provided by the Bureau of Transportation Statistics of the US Department of Transportation, ([BTS, 2017a](#)). We queried the "departure delays," "taxi-in times," "taxi-out times" and "arrival delays" of the flights of a commercial airline for three months. We then clustered data with respect to the observed values queried and generated a value to represent each cluster. Representative value for each cluster is generated by taking weighted average of all observed values in a cluster, where the weights correspond to observation frequencies over three months. We assign probabilities for each representative value by taking proportion of total observation frequencies in a cluster to the number of all observations in the data so that the sum of the probabilities, each one is associated with one cluster, is equal to one.

In the experimental study, we consider four realizations for departure delay, taxi-out time, taxi-in time and arrival delay for ORD airport. In other words, we make four clusters and get one representative value for each of the clusters. These representative values can be viewed as potential realizations. We provide these four realizations and their associated probabilities in [Table A.2](#). It is important to note that we assume that the realization levels for all departure delay, taxi-out, arrival delay and taxi-in times at the same airport are positively correlated, taking some "lowest", "low", "high" or "highest" frequency level simultaneously. Because, these components of non-cruise times are exposed to the same weather condition and airport congestion.

For DFW airport, we consider two different realizations. [Table A.3](#) shows realizations and associated probabilities for DFW airport.

To clarify how we generate a non-cruise time of a flight, let's consider a flight departing from Chicago O'Hare International Airport (ORD) and arriving to Dallas/Fort Worth International Airport (DFW). Under a scenario ω^* involving realization

Table A.1
Demand realizations & associated probabilities.

	ORD - DFW		ORD - LGA	
	Realization		Realization	
	Low (L)	High (H)	Low (L)	High (H)
1st Seg.	161	200	121	140
2nd Seg.	121	160	101	120
3rd Seg.	81	120	81	100
Probability	0.5	0.5	0.4	0.6

Table A.2
Non-cruise time scenarios for ORD.

	Realization with			
	highest frequency (HH)	high frequency (H)	low frequency (L)	lowest frequency (LL)
Dep. delay (min)	0	10	30	50
Taxi-out (min)	15	25	10	35
Arr. delay (min)	0	10	30	50
Taxi-in (min)	15	25	10	35
Probability	0.65	0.20	0.1	0.05

Table A.3
Non-cruise time scenarios for DFW.

	Realization with	
	high frequency (H)	low frequency (L)
Dep. delay (min)	5	40
Taxi-out (min)	15	35
Arr. delay (min)	5	40
Taxi-in (min)	15	35
Probability	0.85	0.15

with low frequency for ORD and realization with high frequency for DFW, observed values of departure delay, taxi-out, arrival delay, taxi-in times are equal to 30, 10, 30 and 10 for ORD and 5, 15, 5 and 15 for DFW, respectively. Then, non-cruise time of this flight is calculated as follows:

$$\begin{aligned}
 Ncrs_i(\omega^*) &= \text{dep. delay}_{ORD}^L + \text{taxi-out}_{ORD}^L + \text{taxi-in}_{DFW}^H + \text{arr.delay}_{DFW}^H \\
 &= 30 + 10 + 15 + 5 = 60.
 \end{aligned}
 \tag{A.1}$$

The associated probability for scenario ω^* can be calculated as follows:

$$\begin{aligned}
 p(\omega^*) &= (\text{prob}_{ORD}^L) \cdot (\text{prob}_{DFW}^H) \\
 &= (0.1) \cdot (0.85) = 0.085.
 \end{aligned}
 \tag{A.2}$$

A3. Scenario generation and selection procedures

We use the following notation through the scenario generation procedure.

J' :	set of flights that have random demand realizations
$D(j)$:	set of demand realizations for each flight $j \in J'$, indexed by $d_1(j), d_2(j), \dots, d_{ D(j) }(j)$
B' :	set of airports that have random non-cruise times
$A(b)$:	set of arrival delay, departure delay, taxi-out and taxi-in time realizations for each airport $b \in B'$, indexed by $a_1(b), a_2(b), \dots, a_{ A(b) }(b)$

Let demand and non-cruise time realizations for each set $D(j)$, $j \in J'$ and $A(b)$, $b \in B'$, respectively, be in increasing order with respect to indices. Then, the following pseudocode generates the scenarios.

Algorithm 4 Scenario generation procedure.

```

Initialization: index  $\leftarrow$  0;
for  $i = 1$  to  $|D(1)|$  do
  for  $j = 1$  to  $|D(2)|$  do
    .
    .
  for  $k = 1$  to  $|D(\lfloor J' \rfloor)|$  do
    for  $l = 1$  to  $|A(1)|$  do
      for  $m = 1$  to  $|A(2)|$  do
        .
        .
      for  $r = 1$  to  $|A(|B'|)|$  do
        index  $\leftarrow$  index + 1;
         $\omega_{index} \leftarrow (d_i(1), d_j(2), \dots, d_k(\lfloor J' \rfloor), a_l(1), a_m(2), \dots, a_r(|B'|));$ 

```

Since, the realizations are in increasing order with respect to indices, the first scenario generated (ω_1) represents lowest demand realizations for each flight $j \in J'$ and non-cruise time (delay) realizations with lowest frequency for each airport $b \in B'$. Similarly, the last scenario generated represents the highest realizations.

Let Ω_n denote the set of scenarios in group subproblem $n \in N$. We generate subsets $\Omega_n \subset \Omega$ of scenarios such that $\bigcup_{n \in N} \Omega_n = \Omega$. In Procedure P1, we consider two scenarios in each set Ω_n for all $n \in N$. We put the first and last scenarios ($\omega_1, \omega_{|\Omega|}$) in the first set Ω_1 , then put the second first and last scenarios ($\omega_2, \omega_{|\Omega|-1}$) in the second set Ω_2 . We repeat the same selection procedure to construct the rest of group subproblems. We summarize the steps to construct Ω_n , $n \in N$ in Procedure P1. Scenario sets Ω_n form a complete partition of scenarios Ω . With this selection procedure, we ensure that Ω_n has no common demand or non-cruise time realizations among two scenarios, i.e., each group subproblem with Ω_n has no deterministic demand or non-cruise time. Therefore, considering the uncertain demand and non-cruise times in each group subproblem may better reflect the original stochastic problem. Procedure P1 is a simple way of obtaining this variety in each Ω_n as a result of generating scenarios in an increasing order of indices. Note that the number of demand realizations we consider for each flight $i \in J'$ is an even number, similarly, the number of non-cruise time realizations for each airport $b \in B'$ is even. This is another reason that Procedure P1 provides stochastic passenger demand and non-cruise time in each group subproblem.

Algorithm 5 Combine first-last scenarios procedure (P1).

Initialization: Given set of scenarios Ω , $n \leftarrow 0$;

for $i = 1$ **to** $|\Omega|/2$ **do**
 $n \leftarrow n + 1$;
 $\Omega_n \leftarrow \{\omega_i, \omega_{|\Omega|-i+1}\}$;

Another approach to strengthen the lower bound is to increase the size of the subsets Ω_n , $n \in N$. We provide another scenario selection procedure, called as Procedure P2, which includes four scenarios in each group subproblem. We first call Procedure P1 to construct group subproblems of size two. Then, in each iteration of Procedure P2, we randomly combine two sets Ω_n and construct a group subproblem with $\hat{\Omega}_n$ including four scenarios. Note that new sets of scenarios $\hat{\Omega}_n$ form the complete partition of scenarios Ω . The detailed steps are given in Algorithm 6.

Algorithm 6 Combine randomly chosen group subproblems from first-last scenarios procedure (P2).

Initialization: Given set of scenarios Ω ;

Call Procedure P1 to construct Ω_n , $n \in N$;

Randomly order numbers $\{1, 2, \dots, |\Omega|/2\}$ represented by $random_i$, $i = 1, \dots, |\Omega|/2$, respectively
 $n \leftarrow 0$

for $i = 1$ **to** $|\Omega|/2$ **by 2 do**
 $n \leftarrow n + 1$;
 $\hat{\Omega}_n \leftarrow \{\Omega_{random_i}, \Omega_{random_{i+1}}\}$;

References

- Ahmadbeygi, S., Cohn, A., Lapp, M., 2010. Decreasing airline delay propagation by re-allocating scheduled slack. *IIE Trans.* 42, 478–489.
- Ahmed, S., 2013. A scenario decomposition algorithm for 0–1 stochastic programs. *Oper. Res. Lett.* 41, 565–569.
- Aktürk, M.S., Atamtürk, A., Gürel, S., 2009. A strong conic quadratic reformulation for machine-job assignment with controllable processing times. *Op. Res. Lett.* 37, 187–191.
- Aktürk, M.S., Atamtürk, A., Gürel, S., 2014. Aircraft rescheduling with cruise speed control. *Oper. Res.* 62, 829–845.
- Ankan, U., Gürel, S., Aktürk, M.S., 2016. Integrated aircraft and passenger recovery with cruise time controllability. *Ann. Oper. Res.* 236 (2), 295–317.
- Bayram, V., Yaman, H., 2018. Shelter location and evacuation route assignment under uncertainty: a benders decomposition approach. *Transp. Sci.* 52, 416–436.
- Bektaş, T., Laporte, G., 2011. The pollution-routing problem. *Transp. Res. Part B: Methodol.* 45 (8), 1232–1250.
- Ben-Tal, A., Nemirovski, A., 2001. Lectures on modern convex optimization: analysis, algorithms, and engineering applications. *MPS-SIAM Series on Optimization*. SIAM.
- Bertsimas, D., Lulli, G., Odoni, A., 2010. An integer optimization approach to large-scale air traffic flow management. *Oper. Res.* 59, 211–227.
- Birge, J., Louveaux, F., 1997. *Introduction to Stochastic Programming*. Springer.
- BTS, 2017a. Airline on-time statistics. <https://www.transtats.bts.gov/ONTIME/Index.aspx> Visited March 2017.
- BTS, 2017b. Airline origin and destination survey (DB1B). https://www.transtats.bts.gov/Tables.asp?DB_ID=125 Visited February 2017.
- BTS, 2017c. On-time arrival performance. https://www.transtats.bts.gov/OT_Delay/OT_DelayCause1.asp?pn=1 Visited February 2017.
- BTS, 2010. Airline on-time performance data Visited April 2016.
- Cadarso, L., de Celis, R., 2017. Integrated airline planning: robust update of scheduling and fleet balancing under demand uncertainty. *Transp. Res. Part C: Emerg. Technol.* 87, 227–245.
- Chiraphadhanakul, V., Barnhart, C., 2013. Robust flight schedules through slack re-allocation. *EURO J. Transp. Logist.* 2, 277–306.
- Cook, A., Tanner, G., Williams, G., Meise, G., 2009. Dynamic cost indexing-managing airline delay costs. *J. Air Transp. Manag.* 15, 26–35.
- Şafak, Ö., Gürel, S., Aktürk, M.S., 2017. Integrated aircraft-path assignment and robust schedule design with cruise speed control. *Comput. Oper. Res.* 84, 127–145.

- Dunbar, M., Froyland, G., Cheng-Lung, W., 2014. An integrated scenario-based approach for robust aircraft routing, crew pairing and re-timing. *Comput. Oper. Res.* 45, 68–86.
- EUROCONTROL, 2001. Forecasting civil aviation fuel burn and emissions in europe. Technical Report. EEC Technical/Scientific, Eurocontrol, Eurocontrol Experimental Centre, B.P. 15, F-91222 Bretigny-sur-Orge, France.
- EUROCONTROL, 2012. User manual for the base of aircraft data (BADA) revision 3.10.. Technical Report. EEC Technical/Scientific, Eurocontrol, Eurocontrol Experimental Centre, B.P. 15, F-91222 Bretigny-sur-Orge, France.
- FAA/Nextor. Total delay impact study. http://www.nextor.org/pubs/TDI_Report_Final_11_03_10.pdf Visited March 2017.
- Fagerholt, K., Gausel, N., Rakke, J., Psaraftis, H., 2015. Maritime routing and speed optimization with emission control areas. *Transp. Res. Part C: Emerg. Technol.* 52, 57–73.
- Günlük, O., Linderöth, J., 2010. Perspective reformulations of mixed integer nonlinear programs with indicator variables. *Math. Program* 124, 183–205.
- Gürkan, H., Gürel, S., Aktürk, M.S., 2016. An integrated approach for airline scheduling, aircraft fleet and routing with cruise speed control. *Transp. Res. Part C: Emerg. Technol.* 68, 38–57.
- Haahr, J.T., Pisinger, D., Sabbaghian, M., 2017. A dynamic programming approach for optimizing train speed profiles with speed restrictions and passage points. *Transp. Res. Part B: Methodol.* 99, 167–182.
- He, Q., Zhang, X., Nip, K., 2017. Speed optimization over a path with heterogeneous arc costs. *Transp. Res. Part B: Methodol.* 104, 198–214.
- Hiriart-Urruty, J., Lemaréchal, C., 2001. *Fundamentals of Convex Analysis*. Springer, Berlin.
- IATA, 2017. Fuel price analysis. <http://www.iata.org/publications/economics/fuel-monitor/Pages/price-analysis.aspx> Visited April 2017.
- ICAO, 2009. ICAO fuel factor. Technical Report. International Civil Aviation Organization (ICAO).
- Liang, Z., Feng, Y., Zhang, X., Wu, T., Chaovalitwongse, W.A., 2015. Robust weekly aircraft maintenance routing problem and the extension to the tail assignment problem. *Transp. Res. Part B: Methodol.* 78, 238–259.
- Listes, O., Dekker, R., 2005. A scenario aggregation-based approach for determining a robust airline fleet composition for dynamic capacity allocation. *Transp. Sci.* 39 (3), 367–382.
- Magnanti, T.L., Wong, R.T., 1981. Accelerating benders decomposition: algorithmic enhancement and model selection criteria. *Oper. Res.* 29, 464–484.
- Papadakos, N., 2008. Practical enhancements to the Magnanti-Wong method. *Oper. Res. Lett.* 36, 444–449.
- Rubin, P., 2011. Benders decomposition then and now. <http://orinanobworld.blogspot.com.tr/2011/10/benders-decomposition-then-and-now.html> Visited May 2016.
- Ruszczynski, A., 2006. *Nonlinear Optimization*. Princeton University Press.
- Sandikçi, B., Özaltın, O.Y., 2017. A scalable bounding method for multi-stage stochastic integer programs. *SIAM J. Optim.* 27, 1772–1800.
- Sherali, H., Lunday, B., 2013. On generating maximal nondominated benders cuts. *Ann. Oper. Res.* 210 (1), 52–72.
- Sherali, H., Zhu, X., 2008. Two-stage fleet assignment model considering stochastic passenger demands. *Oper. Res.* 56 (2), 383–399.
- Sherali, H.D., Staats, R.W., Trani, A.A., 2006. An airspace-planning and collaborative decision-making model: part ii-cost model, data considerations, and computations. *Transp. Sci.* 40, 147–164.
- Sohoni, M., Lee, Y., Klabjan, D., 2011. Robust airline scheduling under block-time uncertainty. *Transp. Sci.* 45, 451–464.
- Van Slyke, R.M., Wets, R., 1969. L-Shaped linear programs with applications to optimal control and stochastic programming. *SIAM J. Appl. Math.* 17 (4), 638–663.
- D. of Transport Studies University of Westminster, 2011. European airline delay cost reference values Final Report (Version 3.2). Technical Report. Performance Review Unit EUROCONTROL.
- Yan, S., Tang, C.-H., Fu, T.-C., 2008. An airline scheduling model and solution algorithms under stochastic demands. *Eur. J. Oper. Res.* 190 (1), 22–39.
- Yang, X., Chen, A., Ning, B., Tang, T., 2016. A stochastic model for the integrated optimization on metro timetable and speed profile with uncertain train mass. *Transp. Res. Part B: Methodol.* 91, 424–445.
- Yen, J., Birge, J., 2006. A stochastic programming approach to the airline crew scheduling problem. *Transp. Sci.* 40 (1), 3–14.