

Article

Integrating Flight Scheduling, Fleet Assignment, and Aircraft Routing Problems with Codesharing Agreements under Stochastic Environment

Kübra Kızıloğlu *  and Ümit Sami Sakallı 

Department of Industrial Engineering, Kırıkkale University, 71451 Kırıkkale, Turkey; ssakalli@kku.edu.tr

* Correspondence: guldogankubra@gmail.com

Abstract: Airlines face the imperative of resource management to curtail costs, necessitating the solution of several optimization problems such as flight planning, fleet assignment, aircraft routing, and crew scheduling. These problems present some challenges. The first pertains to the common practice of addressing these problems independently, potentially leading to locally optimal outcomes due to their interconnected nature. The second challenge lies in the inherent uncertainty associated with parameters like demand and non-cruise time. On the other hand, airlines can employ a strategy known as codesharing, wherein they operate shared flights, in order to minimize these challenges. In this study, we introduce a novel mathematical model designed to optimize flight planning, fleet assignment, and aircraft routing decisions concurrently, while accommodating for codesharing. This model is formulated as a three-stage non-linear mixed-integer problem, with stochastic parameters representing the demand and non-cruise time. For smaller-scale problems, optimization software can effectively solve the model. However, as the number of flights increases, conventional software becomes inadequate. Moreover, considering a wide array of scenarios for stochastic parameters leads to more robust results; however, it is not enabled because of the limitations of optimization software. In this work, we introduce two new simulation-based metaheuristic algorithms for solving large-dimensional problems, collectively called “simheuristic.” These algorithms integrate the Monte Carlo simulation technique into Simulated Annealing and Cuckoo Search. We have applied these simheuristic algorithms to various problem samples of different flight sizes and scenarios. The results demonstrate the efficacy of our proposed modeling and solution approaches in efficiently addressing flight scheduling, fleet assignment, and aircraft routing problems within acceptable timeframes.

Keywords: flight scheduling; fleet assignment; aircraft routing; simheuristic; simulated annealing; cuckoo search



Citation: Kızıloğlu, K.; Sakallı, Ü.S. Integrating Flight Scheduling, Fleet Assignment, and Aircraft Routing Problems with Codesharing Agreements under Stochastic Environment. *Aerospace* **2023**, *10*, 1031. <https://doi.org/10.3390/aerospace10121031>

Academic Editor: Joost Ellerbroek

Received: 27 September 2023

Revised: 5 December 2023

Accepted: 8 December 2023

Published: 14 December 2023



Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

The airline industry is highly competitive, especially with the emergence of low-fare airlines. It is characterized by variable demand, high operating costs, heavy traffic, and strict regulations, all of which contribute to the tight competition in the market. Furthermore, an airline's sole product is its aircraft seats, which are not sold until the flight takes off. Therefore, managing supply and demand fluctuations effectively is a significant challenge for airline companies. Accurate models and solution methodologies with narrow margins of error are necessary to address this issue [1].

Airline operation planning is a complex process of making decisions with many important decisions that directly affect a company's profitability. The planning process is divided into sub-problems to manage these decisions [2]. The flight scheduling (FS) problem defines the frequency of flights to each destination and how the flights should be planned to meet this frequency. The fleet assignment (FA) problem assigns fleets to appropriate flights, while the aircraft routing (AR) problem creates flight route plans for

each aircraft, considering that maintenance constraints. The crew scheduling (CS) problem involves determining the crew members for each flight [3].

Airline companies aim to expand their airline networks to provide passengers with a wider range of destinations, regional connections, and seamless travel experiences with minimal schedule disruptions. However, achieving these goals often involves large investments in purchasing or leasing new aircraft and hiring additional crew members. Alternatively, airline companies can establish codeshare agreements with each other to reduce costs and improve their network coverage.

A codeshare agreement (CA) is a business arrangement between two airlines. One of them can market a flight that is operated by another airline on or off its network, without using their own aircraft. This airline is called a marketing carrier. The other airline operating the flight is called the operating carrier [4,5]. These agreements are made as a result of alliances between two companies, and they allow airlines to expand their flight targets without incurring the huge costs associated with acquiring new aircraft. CAs provide more flight opportunities within an airline's network, which benefits both customer satisfaction and reduces delays. Additionally, such agreements lead to increased mutual revenues and market shares for the companies [6].

This study aims to simultaneously solve the FS, FA, and AR problems considering CA. It also creates airline schedules that affect profitability while assigning the most suitable fleet types to flights. This improves the company's service quality and profitability.

The paper is organized as follows: Section 2 includes a literature review of the FA, FS, and AR problem. Section 3 defines the problem and provides a new mathematical model. Section 4 introduces the proposed solution methods and includes a small sample. Section 5 presents computational experiments. Finally, Section 6 concludes the article and discusses future studies.

2. Literature Review

Airlines aim to maximize their profits by providing the optimum combination of fleet type and passengers' demand. Assigning an aircraft that is smaller than the required passenger capacity to a flight leads to a direct loss of customers due to capacity insufficiency. Conversely, assigning an aircraft that is larger than the required passenger capacity to a flight can result in unsold seats. As a result of this, it causes an inability to sell the seats and reduce operating profits. Therefore, the FA problem is a crucial decision for an airline's planning strategy [7]. With this motivation, this study examines the FA problem, and Table 1 summarizes the features that contribute to the development of the problem and the studies that investigate these features.

There have been numerous studies on the FA problem in the literature since 1971. While the models of Abara [8] and Hane et al. [9] have contributed to the development of solutions to the problem, their applicability is limited due to certain assumptions such as fixed departure times for the same flight every day and deterministic passenger demand, which do not reflect real-life variability. Clarke et al. [10] developed a solution by incorporating the maintenance planning (MP) of aircraft at stations and crew issues into the FA model.

Berge and Hopperstad [11] suggested the re-flying concept to address demand fluctuations and used the demand-based dispatch approach to make capacity flexible allowing for fleet type assignments to occur closer to take-off. Talluri [12] extended their work in terms of opportunities and computation times. Levin [13] provided flexibility to the FA problem by allowing for variable take-off times and he was the first researcher to consider varying departure times with binary constraints by using integer programming. However, this model did not exclude the aircraft capacity and multiple fleet types. Rexing et al. [14] increased the flight connectivity possibilities by using a time window (TW) for departure times and optimizing time windows according to specific time zones. Belanger et al. [15] minimized the number of aircraft in the fleet by determining the fleet size (FSD). Thus, it aims for the utilization of the fleet (FU) optimally. ZEGHAL ET AL. [16] have addressed a

flexible aircraft fleet and routing problem, which is motivated by the Tunisian national carrier TunisAir. Flexible fleet (FF) is the ability to expand the fleet by renting an aircraft. Thus, more flights can be served.

Barnhart et al. [17] considered the passenger demand based on an itinerary (IFAP) and passenger spill and rescue costs and analyzed the profitability of the model by taking into account the effect of the flight network. In real-life scenarios, unexpected situations and disruptions can occur, so it is necessary to create robust models. Rosenberger et al. [18] developed a robust model using the hub-and-spoke network structure and produced short cycles sensitive to flight cancellations. The robustness of such an assignment and rotation has been demonstrated by using simulations of airline operations. Cadarso and Marín [19] proposed a robust model for the integrated FS and FA problem that takes into account the number of misconnected passengers. The model uses the exponential distribution of passenger connection times (PCTs) to calculate the probability of passengers missing their flight.

Smith and Johnson [20] proposed the concept of station purity (SP) by limiting the number of fleet type or crew-compatible families. Because each airport cannot serve each fleet type, Sherali et al. [21] developed a demand-driven re-flight model that dynamically reassigns the fleet type. Jacobs et al. [22], Dumas et al. [23], and Pilla et al. [24] relaxed the deterministic assumptions set by the basic FA problem. Sherali and Zhu [25] developed a two-stage stochastic programming model for the FA problem. Pilla et al. [26] explicitly evaluated the passenger demand by including direct demand scenarios in Sherali and Zhu's [25] model. Naumann et al. [27] explained the uncertainty in demand as well as the uncertainty in fuel prices. Cadarso and Celis [28] developed a stochastic model that considers demand uncertainty (SD) and discrete passenger choice models for the FS and FA problem. The model aims to maximize the airline's profits while minimizing the number of misconnected passengers. The passenger spill and rescue effects are also modeled and it focuses on the probability of passengers choosing a route. Atasoy et al. [29] proposed a similar integrated model based on discrete choice models.

Different to the common features used in the FA problem, Pita et al. [30] considered airports with the slot constrained (SC) in their study. The slot can be defined as the row/time allocation to an aircraft. The aim is to reduce large delays by controlling air traffic. Kenan et al. [31] assigned fleets to flights which have stochastic demand and stochastic fare (SF) based on fare classes. Sherali et al. [32,33] emphasized the importance of considering fuel consumption (FUEL) in airline optimization decisions. Gürkan et al. [34] integrated the cruise time controllability (CTC) of the aircraft using cruise speed adjustments into FA and AR problems. Jamili [35] integrated all three decisions into a single model, but also developed flight programs from the ground up and addressed the issue of fleet type route delays by introducing buffers. Şafak et al. [36] took into account passenger connection times, fuel consumption, and CO₂ emission costs associated with cruise speed settings by integrating the three decisions.

Kenan et al. [6] integrated CA into the FS, FA, and AR problems. It highlights that such agreements can significantly affect an airline's profit by using fewer aircraft, fewer delays, and more flights. They also added optional flight legs (OF) to the model, which can be canceled according to demand changes. Şafak et al. [37] developed a three-stage stochastic programming model for the FS, FA, and AR problems. Cacchiani and Gonzalez [38] aggregated the FS, FA, AR, and CS problems. Xu et al. [39] proposed an integrated model for the FS, FA, and AR problems that allows for spreading delays, on-demand flights, and passenger spills. The spill passenger (SPP) is the number of passengers who cannot be served for any flight.

Kenan et al. [6] and Şafak et al. [36,37,40] have made significant contributions to solving the FA problem. However, Kenan's study did not detail the costs related to passengers in the model, only considering the delay costs (DC) in the objective function. Şafak et al. [36,37] have discussed costs in detail, and CTC has been used to compensate for

the uncertainty of non-cruise time (NCTU). However, different costs have been incurred by increasing the cruise time for new flights or leasing alternatives.

The objective of this study is to maximize the profitability of airline companies, which hinges on the revenue generated from ticket sales to passengers. To efficiently serve a greater number of passengers on each flight, it is imperative to assign appropriate fleet types. Assigning aircraft with a lower capacity than the demand may result in unserved passengers, constituting a spill passenger cost. Furthermore, ensuring timely departures and arrivals is crucial to enhancing passenger satisfaction levels. Delays caused by aircraft idling at airports can lead to disruptions in consecutive flights. These delays, in turn, may cause passengers to miss their connections, especially when the required connection times cannot be met on connecting flights. The impact of these factors on the service quality translates into costs for the companies. Operational costs are incurred by companies for the use of aircraft on each flight, necessitating revenue generation above these costs to ensure profitability. For flights that may not be profitable on their own, companies can opt for less expensive CA to mitigate costs. To the best of our knowledge, there is no study in the literature that handles stochastic demand and stochastic non-cruise times on delays while also considering the impact of CA on the FA problem. This study aims to fill this gap by providing the following main contributions:

- The development of a new third-stage stochastic non-linear programming model that combines FA with operational FS decisions and AR problem using codesharing agreements.
- The consideration of stochastic demand and stochastic non-cruise times simultaneously in the model.
- The proposal of simulation-based metaheuristic (simheuristic) methods for solving the proposed model. The Monte Carlo method is used for simulation, and the Simulated Annealing (SA) and Cuckoo Search (CS) algorithms are used for metaheuristics. Two new solution methods are proposed by integrating simulation and metaheuristics.
 - i. SA + MC: Simulated Annealing + Monte Carlo
 - ii. CS + MC: Cuckoo Search + Monte Carlo

Table 1. Features added to the FA.

Authors	Maintainance Planning MP	Flight Scheduling FS	Aircraft Routing AR	Time Window TW	Stochastic Demand SD	Spill Passenger Costs SPP	Passenger Connection Time PCT	Delay Costs DC	Cruise Time Controllability CTC	Stochastic Non-Cruise Time NCTU	Codeshare Aggrements CA
Levin [13]				✓							
Berge and Hopperstad [11]					✓	✓					
Barnhart et al. [41], Sosnowska and Rolim [42]	✓		✓								
Rexing et al. [14]				✓							
Yan and Tseng [43]		✓									
Ahuja et al. [44]	✓			✓							

Table 1. Cont.

	Maintenance Planning	Flight Scheduling	Aircraft Routing	Time Window	Stochastic Demand	Spill Passenger Costs	Passenger Connection Time	Delay Costs	Cruise Time Controllability	Stochastic Non-Cruise Time	Codeshare Agreements
Lohatepanont and Barnhart [2]		✓				✓					
Li and Wang [45]	✓		✓								
Yan et al. [46]		✓									
Yan and Chen [47]			✓								✓
Pilla et al. [26], Jacobs et al. [22]					✓						
Sherali and Zhu [25], Naumann et al. [27]					✓						
Dumas et al. [23]					✓	✓					
Haouari et al. [48]	✓		✓								
Sherali et al. [49]	✓	✓									
Zeghal et al. [16]	✓		✓	✓							
Sherali et al. [33]		✓		✓		✓					
Liang and Chaovalitwongse [50]	✓		✓								
Pita et al. [30]		✓		✓		✓	✓	✓			
Cadarso and Marin [19]		✓					✓				
Atasoy et al. [29]		✓				✓					
Shao et al. [51]	✓		✓								
Gürkan et al. [34]	✓	✓	✓						✓		
Dong et al. [52]		✓									
Liu et al. [53]	✓		✓								
Cacchiani and Salazar-González [54]	✓		✓								
Cadarso and Celis [28]		✓			✓		✓				
Jamili et al. [35]	✓		✓				✓				
Şafak et al. [36]				✓					✓		
Kenan et al. (a) [6]		✓	✓		✓			✓			✓
Kenan et al. (b) [31]		✓			✓						
Şafak et al. [37]	✓	✓	✓	✓	✓		✓	✓	✓	✓	
Şafak et al. [38]	✓	✓	✓	✓	✓		✓	✓	✓		
Wei et al. [55]	✓	✓	✓	✓							
Cacchiani and Salazar-González [38]	✓	✓	✓	✓							
Xu et al. [39]	✓	✓	✓			✓	✓	✓			
Ahmed et al. [56]	✓		✓			✓					
This study	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓

✓: Access.

3. Problem Description and Formulation

3.1. Terminology

- Flight leg: Describes a flight of an aircraft from the departure airport to the destination airport.
- Path (itinerary): A sequence of one or more flight legs between a specific origin and destination.
- Fleet type (aircraft type): A certain model of aircraft. All the aircrafts of the same type have the same cockpit configuration, crew qualification requirements, maintenance requirements, and capacity.
- Fleet family (aircraft family): A set of aircraft types, each having the same cockpit configuration and crew qualification requirements. Thus, the same crew can fly any fleet type of the same family.
- Fare class (FC): Available seats on aircrafts are divided into classes according to their fares. The cost of seats in each class is the same. Seat capacities in these classes may be fixed or variable.
- Turnaround time: It is the time required for a fleet type to prepare for flight. This period includes the cleaning of the aircraft, passenger, and baggage movements.
- Cruise time is the time of flight that falls between climb and descent.
- Non-cruise time: It is the sum of the taxi in and out times between an aircraft's landing and take-off. In other words, it is the time that the aircraft moves on the ground.
- Tax in: It is the duration of the aircraft moving towards the apron after landing.
- Taxi out It is the duration of the aircraft moving from the apron to the take-off field.

3.2. Problem Description

In order to effectively tackle the complex FA problem, it is imperative to consider the interrelated decisions pertaining to FS and AR while also accounting for codesharing agreements. To address the uncertainty associated with demand and non-cruise time, this study presents a novel and integrated approach that models the problem as a three-stage stochastic mixed-integer non-linear programming (TSS_MINLP) framework.

In line with union regulations, the airline industry necessitates operational decisions to be made 2–3 months prior to the scheduled flight. Similarly, codesharing agreements are strategic decisions that require meticulous planning well in advance. However, the uncertainty surrounding the demand for flights during this period poses a significant challenge (as highlighted by Kenan et al. [6]). Thus, this approach first focuses on making decisions regarding FA, routing, and CA for the previously scheduled flights, followed by determining the number of passengers to be accommodated as the flight date approaches, based on these decisions. Consequently, the fleet type route schedule is adjusted accordingly. Thus, the proposed three-stage decision-making model, illustrated in Figure 1, integrates the FA, FS, and AR decisions in a cohesive framework that effectively addresses the uncertainty associated with demand and non-cruise time.

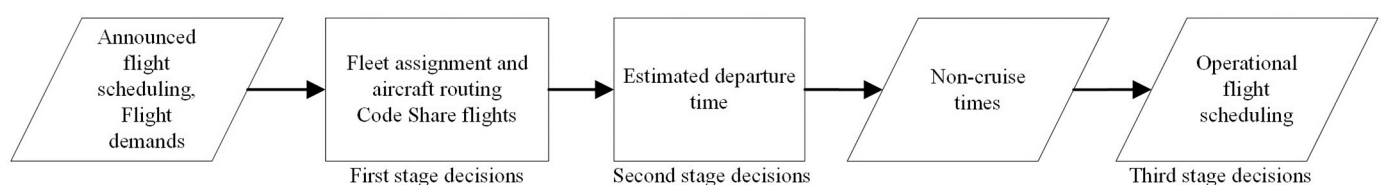


Figure 1. The inputs and outputs of the model stages.

The problem at hand can be divided into three main steps. The first step involves determining the flights that will be operated by the airline and those that will be operated under CA. Based on this, an appropriate fleet type is assigned to each flight, and the routes are created while accounting for the maintenance planning required for the fleet type. Once consecutive flights are completed, the starting and final stations for each aircraft are determined.

Moving on to the second step, the estimated departure time for the airline's flights is selected from a predetermined time window. This ensures that the departure time is optimized, keeping in mind the operational constraints and other factors that may influence the timing of the flight.

In the final step, the timetable of the routes assigned to each aircraft is updated, and the number of passengers to be carried by the aircraft is determined. This step is critical in ensuring that the aircraft is utilized optimally while adhering to safety standards and operational efficiency. By determining the number of passengers to be carried, the airline can ensure that the aircraft is optimally utilized, and that the entire operation runs smoothly.

Overall, this three-step approach, which involves determining flights and fleet types, optimizing departure times, and updating the timetable while considering passenger count, enables the airline to operate efficiently and effectively.

The problem has several assumptions which are given as follows:

- The analysis of data obtained from the BTS [57] revealed that non-cruise times follows a normal distribution.
- It is assumed that demand follows a uniform distribution with lower and upper parameters covering the minimum and maximum fleet type capacities.
- The departure and arrival times of flights are also pre-determined.
- A time window is used for departure times, and delays that exceed this time window result in additional costs that are included in the objective function.
- Maintenance planning is conducted at either the first or the last airport.
- Codeshare flights are not included in the airline's routes, and a specific budget is allocated for CA.
- The cruising speed of the fleet type is allowed to vary at certain rates.
- If there are connecting flights for passengers, a minimum time is given for passengers to switch to the next flight, and if this time is not sufficient, passengers miss their connections.

3.3. Mathematical Model

TSS_MINLP model is below:

Sets	
S	: Scenarios $s \in S$
K	: Fleet types $k \in K$
F	: Flights $i \in F$
YB _i	: Flights which have passenger connections from flight $i \in F$
BU	: Connected flights sets $i, j \in F$
O _i	: Flights before flight $i \in F$
H _i	: Flights after flight $i \in F$
SU _k	: Last flights of fleet type $k \in K$
IU _k	: First flights of fleet type $k \in K$
Y	: Different allied airlines $v \in Y$
P	: Codeshare agreements $p \in P$

Parameters	
$[T_i^{\min}, T_i^{\max}]$	: Earliest and latest departure time of flight $i \quad i \in F$
pr_s	: Probability of scenario $s \quad s \in S$
AN_k	: Number of current aircraft of fleet type $k \quad k \in K$
Kap_k	: Number of seats in fleet type $k \quad k \in K$
DY_i	: Opportunity cost of spilled passengers of flight $i \quad i \in F$
BOS_k	: Unit idle time cost of fleet type k (per minute) $k \in K$
KY_i	: Cost per passenger for miss-connected passengers on flight $i \quad i \in F$
DC_i	: Per minute delay cost of flight $i \quad i \in F$
$fare_i$	: Flight revenue per passenger $i \in F$
$cost_i$	: Flight operating cost per passenger $i \in F$
μ_{is}	: Demand of flight i in scenario $s \quad s \in S, i \in F$, stochastic
η_{is}	: Non-cruise time of flight i in scenario $s \quad s \in S, i \in F$, stochastic
$[CT_{ik}^l, CT_{ik}^u]$	: Time window for cruise time of flight i of fleet type $k \quad i \in F, k \in K$
TA_{ik}	: Turnaround time required to prepare fleet type k after flight i $i \in F, k \in K$
CP_{ij}	: Transit time for connected passengers between flights $i, j \quad i \in J, j \in YB_i$
$pass_{ij}$	: Number of passengers from flight i connected to flight $j \quad i \in J, j \in YB_i$
$code_{vpi}$	: The cost per passenger in contract type $p \quad p \in P, v \in Y, i \in F$
Cap_{vpi}	: Capacity provided under contract type p on flight leg i , $p \in P, v \in Y, i \in F$
BUD	: Available budget for codeshare agreements
θ	: The ratio of total codeshare flight capacities to the total capacity of the airline
M	: A big number

Decision Variables

x_{ijk}^1	: $\begin{cases} 1 & \text{If flight } j \text{ comes after flight } i \text{ and both are performed with fleet type } k \\ & j \in F, i \in O_j, k \in K \\ 0 & \text{otherwise} \end{cases}$
f_{ik}^1	: $\begin{cases} 1 & \text{If flight } i \text{ is the first flight with fleet type } k \\ & i \in F, k \in K \\ 0 & \text{otherwise} \end{cases}$
e_{ik}^1	: $\begin{cases} 1 & \text{If flight } i \text{ is the last flight with fleet type } k \\ & i \in F, k \in K \\ 0 & \text{otherwise} \end{cases}$
q_{vpi}^1	: $\begin{cases} 1 & \text{If the flight leg } i \text{ is performed with the aircraft in the contract } p \text{ of company } v \\ & v \in Y, i \in F, p \in P \\ 0 & \text{otherwise} \end{cases}$
a_i^2	: Announced departure time of flight $i \quad i \in F$
b_{is}^3	: Actual departure time of flight i in scenario $s \quad i \in F, s \in S$
c_{is}^3	: Actual arrival time of flight i in scenario $s \quad i \in F, s \in S$
d_{iks}^3	: Cruise time of flight i with fleet type k in scenario $s \quad i \in F, s \in S, k \in K$
IT_{iks}^3	: Idle time of flight i with fleet type k in scenario $s \quad i \in F, s \in S, k \in K$
del_{is}^3	: Delay time of flight i in scenario $s \quad i \in F, s \in S$
π_{is}^3	: Number of accepted passengers in flight leg i under scenario s $i \in F, s \in S$
πc_{is}^3	: Number of accepted passengers in flight leg i under scenario s in codeshare $i \in F, s \in S$
w_{ijs}^3	: $\begin{cases} 1 & \text{In scenario } s, \text{ if passenger on flight } i \text{ miss flight } j \\ & i \in J, j \in YB_i, s \in S, \\ 0 & \text{otherwise} \end{cases}$

Maximize

$$\begin{aligned}
&= \sum_{s \in S} pr_s \left[\begin{aligned} &\left(\sum_{i \in F} (\pi_{is}^3 + \pi c_{is}^3) * fare_i \right) - \left(\sum_{i \in F} \pi_{is}^3 * cost_i \right) \\ &- \left(\sum_{i \in F} \sum_{v \in Y} \sum_{p \in P} code_{vpi} * Cap_{vpi} * q_{vpi}^1 \right) \\ &- \left(\sum_{i \in F} \sum_{v \in Y} \sum_{p \in P} \max(0, \mu_{is} - Cap_{vpi}) * q_{vpi}^1 * (fare_i - code_{vpi}) \right) \\ &- \left(\sum_{i \in F} \sum_{k \in K} \max(0, \mu_{is} - Kap_k) * \left(\sum_{j \in O_i} x_{jik}^1 + f_{ik}^1 \right) * DY_i \right) \\ &- \left(\sum_{i \in F} \sum_{k \in K} IT_{iks}^3 * BOS_k \right) - \left(\sum_{i \in F} DC_i * del_{is}^3 \right) - \left(\sum_{i \in F} \sum_{j \in YB_i} KY_j * pass_{ij} * w_{ijs}^3 \right) \end{aligned} \right] \quad (1)
\end{aligned}$$

Subject to

$$\sum_{k \in K} \left(\sum_{j \in O_i} x_{jik}^1 + f_{ik}^1 \right) + \sum_{v \in Y} \sum_{p \in P} q_{vpi}^1 = 1 \quad \forall i \in F \quad (2)$$

$$\sum_{j \in O_i} x_{jik}^1 + f_{ik}^1 - \sum_{j \in H_i} x_{ijk}^1 - e_{ik}^1 = 0 \quad \forall i \in F, \forall k \in K \quad (3)$$

$$\sum_{i \in F} f_{ik}^1 \leq AN_k \quad \forall k \in K \quad (4)$$

$$f_{ik}^1 = 0 \quad \forall i \in F \setminus IU_k, \forall k \in K \quad (5)$$

$$e_{ik}^1 = 0 \quad \forall i \in F \setminus SU_k, \forall k \in K \quad (6)$$

$$\sum_{i \in F} \sum_{v \in Y} \sum_{p \in P} code_{vpi} * Cap_{vpi} * q_{vpi}^1 \leq BUD \quad (7)$$

$$\sum_{i \in F} \sum_{v \in Y} \sum_{p \in P} Cap_{vpi} * q_{vpi}^1 \leq \theta * \sum_{k \in K} AN_k * Kap_k \quad (8)$$

$$\sum_{i \in F} \sum_{v \in Y} \sum_{p \in P} q_{vpi}^1 \leq \sum_{i \in F} \sum_{k \in K} \left(\sum_{j \in O_i} x_{jik}^1 + f_{ik}^1 \right) \quad (9)$$

$$\left[1 - \sum_{v \in Y} \sum_{p \in P} q_{vpi}^1 \right] * T_i^{min} \leq a_i^2 \leq \left[1 - \sum_{v \in Y} \sum_{p \in P} q_{vpi}^1 \right] * T_i^{max} \quad \forall i \in F \quad (10)$$

$$-a_i^2 + b_{is}^3 \geq 0 \quad \forall i \in F, \forall s \in S \quad (11)$$

$$b_{is}^3 - a_i^2 - del_{is}^3 \leq 0 \quad \forall i \in F, \forall s \in S \quad (12)$$

$$b_{is}^3 + \sum_{k \in K} \left(\sum_{j \in O_i} x_{jik}^1 + f_{ik}^1 \right) * \eta_{is} + \sum_{k \in K} d_{iks}^3 = c_{is}^3 \quad \forall i \in F, \forall s \in S \quad (13)$$

$$\text{If } x_{ijk}^1 = 1, \quad b_{js}^3 - b_{is}^3 = \eta_{is} + TA_{ik} + d_{iks}^3 + IT_{iks}^3 \quad \forall (i, j) \in BU, \forall s \in S, \forall k \in K \quad (14)$$

$$-CT_{ik}^l \left(\sum_{j \in O_i} x_{jik}^1 + f_{ik}^1 \right) + d_{iks}^3 \geq 0 \quad \forall i \in F, \forall s \in S, \forall k \in K \quad (15)$$

$$CT_{ik}^u \left(\sum_{j \in O_i} x_{jik}^1 + f_{ik}^1 \right) - d_{iks}^3 \geq 0 \quad \forall i \in F, \forall s \in S, \forall k \in K \quad (16)$$

$$b_{js}^3 - b_{is}^3 - \sum_{k \in K} d_{iks}^3 + M * w_{ijs}^3 \geq \left[1 - \sum_{v \in Y} \sum_{p \in P} q_{vpi}^1 \right] * (\eta_{is} + CP_{ij}) \quad \forall i \in F, \forall j \in YB_i, \forall s \in S \quad (17)$$

$$\pi_{is}^3 \leq \sum_{k \in K} Kap_k * \left(\sum_{j \in O_i} x_{jik}^1 + f_{ik}^1 \right) \quad \forall i \in F, \forall s \in S \quad (18)$$

$$\pi c_{is}^3 \leq \sum_{v \in Y} \sum_{p \in P} Cap_{vpi} * q_{vpi}^1 \quad \forall i \in F, \forall s \in S \quad (19)$$

$$\pi_{is}^3 \leq \mu_{is} \quad \forall i \in F, \forall s \in S \quad (20)$$

$$\pi c_{is}^3 \leq \mu_{is} \quad \forall i \in F, \forall s \in S \quad (21)$$

$$x_{ijk}^1 \in \{0, 1\} \quad \forall i \in F, \forall j \in O_i, \forall k \in K \quad (22)$$

$$f_{ik}^1 \in \{0, 1\} \quad \forall i \in F, \forall k \in K$$

$$e_{ik}^1 \in \{0, 1\} \quad \forall i \in F, \forall k \in K$$

$$q_{vpi}^1 \in \{0, 1\} \quad \forall i \in F, \forall v \in Y, \forall p \in P$$

$$\begin{aligned} a_i^2 &\geq 0 \quad \forall i \in F \\ \pi_{is}^3 &\geq 0 \quad \forall i \in F, \forall s \in S \\ \pi c_{is}^3 &\geq 0 \quad \forall i \in F, \forall s \in S \\ IT_{iks}^3 &\geq 0 \quad \forall i \in F, \forall k \in K, \forall s \in S \end{aligned}$$

$$del_{is}^3 \geq 0 \quad \forall i \in F, \forall s \in S, sw_{ijs}^3 \in \{0, 1\} \quad \forall i \in F, \forall j \in YB_i, \forall s \in S$$

Equation (1) aims to maximize the expected profit of an airline, which is calculated as the difference between the income from passengers and the associated costs. The costs include flight cost per passenger, codeshare costs, cost of spilled passengers, fleet type idle time cost, delay cost, and passenger misconnection costs. Equations (2)–(22) are constraints. The decisions of the first stage are determined by Equations (2)–(9). Equation (2) ensures that each planned flight leg is covered by exactly one fleet type, while Equation (3) maintains the route structure of the network. Equation (4) specifies that the total number of aircraft which is used should not exceed the number of aircraft available. Equations (5) and (6) identify the first and last airports of each fleet type. These airports serve as maintenance stations for the aircraft. Equation (7) ensures that the total investment in CA between operators and flight legs does not exceed the allocated budget. The airline being considered is a marketing carrier that operates its own flights and may also serve as the operator for other flights. Hence, Equation (8) specifies that the ratio of the total number of seats purchased to the total capacity of all available aircraft should not exceed the value of (θ). Equation (9) specifies that the number of codeshare flights should not exceed the number of the company's own flights. The second-stage decision is determined by Equation (10), which ensures that the announced departure time of the airline's own flights falls between determined lower and upper limits. Equations (11)–(21) determine the third-stage decisions of the model. Equations (11) and (12) allow for delays between the announced departure times of the first stage and the actual departure times. Since these delays can impact other flights on the route, they are considered as costs in the objective function. Equation (13) calculates the arrival times for each flight based on the actual departure times. Equation (14) determines the departure times of successive flights on a route, as well as the minimum idle times of fleet type. Equations (15) and (16) impose upper and lower cruise time limits for each flight. Equation (17) accounts for passengers who miss their connecting flights due to the absence of sufficient transit time, which creates a cost for the company. Equations (18) and (19) specify that the number of passengers admitted to a flight leg must not exceed the capacity of the aircraft or the capacity in the selected contract. Similarly, Equations (20) and (21) ensure that the number of passengers accepted on a flight leg does not exceed the demand for the flight leg. Finally, Equation (22) defines the domain of all variables.

The proposed model is subjected to a range of diverse scenarios, characterized by distinct levels of passenger demand and non-cruise time of flights. The resulting third-stage decisions and corresponding objective function values exhibit significant variability across these scenarios. To effectively account for these real-world complexities, the model

employs a probabilistic weighting approach. Specifically, the outcomes for each scenario are multiplied by their respective probabilities, and subsequently summed to derive the expected value of the objective function. This approach enables the model to incorporate and adapt to the diverse conditions encountered in practical settings.

4. Solution Methodology

Optimization packages are limited in their ability to solve the TSS_MINLP, especially when the number of flights and scenarios is increased. Therefore, the development of heuristic or metaheuristic methods is necessary [34–36,38,39,55].

Simheuristics are an extension of metaheuristics for solving optimization problems under uncertainty. They have been widely applied to different problems such as production planning, supply chain design, flexible flow shop scheduling, resource allocation, transportation, and healthcare scheduling [58–63]. Simheuristics combine metaheuristics with simulation methods, allowing for stochastic variables to be considered in mathematical models [64]. This means that a simheuristic algorithm includes a special simulation for stochastic variables, making them efficient and effective in overcoming stochastic optimization problems [65]. In the proposed solution method, scenarios with varying demand and non-cruise times are generated using the Monte Carlo simulation method.

4.1. Simulated Annealing

Simulated annealing (SA) is a search algorithm that is inspired by the physical annealing process of a solid material. Physical annealing involves gradually cooling a heated solid to its minimum energy state [66]. SA begins by performing a broad search of the solution space, accepting even suboptimal solutions with a high probability. As the algorithm progresses, it performs a more focused search around the current solution, with the goal of finding the global optimum. At each iteration, SA compares the current solution to a neighboring solution. If the neighboring solution is better, it is accepted as the new current solution. If the neighboring solution is worse, the algorithm may still accept it with a certain probability, to avoid getting stuck at a local minimum. The best available solution is also retained. The acceptance probability is based on the change in profit and the temperature parameters. This process is repeated at each temperature level, with the temperature gradually decreasing according to a predetermined cooling schedule [67].

SA is a stochastic search algorithm used to reach the global optimum hidden among local optima. It has the great advantage of having a diverse and deep neighborhood structure. It is also a simple and fast algorithm with few parameters [68]. In the literature, strong results have been obtained in most applications [69]. Therefore, this study uses SA as a metaheuristic method, integrated with the Monte Carlo simulation (MC). The flow diagram of the simheuristic method (SA + MC) is shown in Figure 2, and the detailed pseudocode is given in Algorithm 1. The flowchart is divided into smaller rectangles, numbered, and drawn with dashed lines to aid in explanation. The integration of SA and MC allows for the efficient and effective optimization of complex problems with stochastic variables.

Figure 2 provides an overview of the proposed methodology, which comprises several iterative stages. In the first red rectangle, an initial solution is generated by creating routes that are suitable for flights. Next, the fleet type with the closest capacity to the flight with the highest expected demand on each route is assigned to that route. The resulting assignment and routing decisions are evaluated using a short Monte Carlo simulation (e.g., 100 runs) to calculate the expected objective function value of the problem, as shown in the second red rectangle.

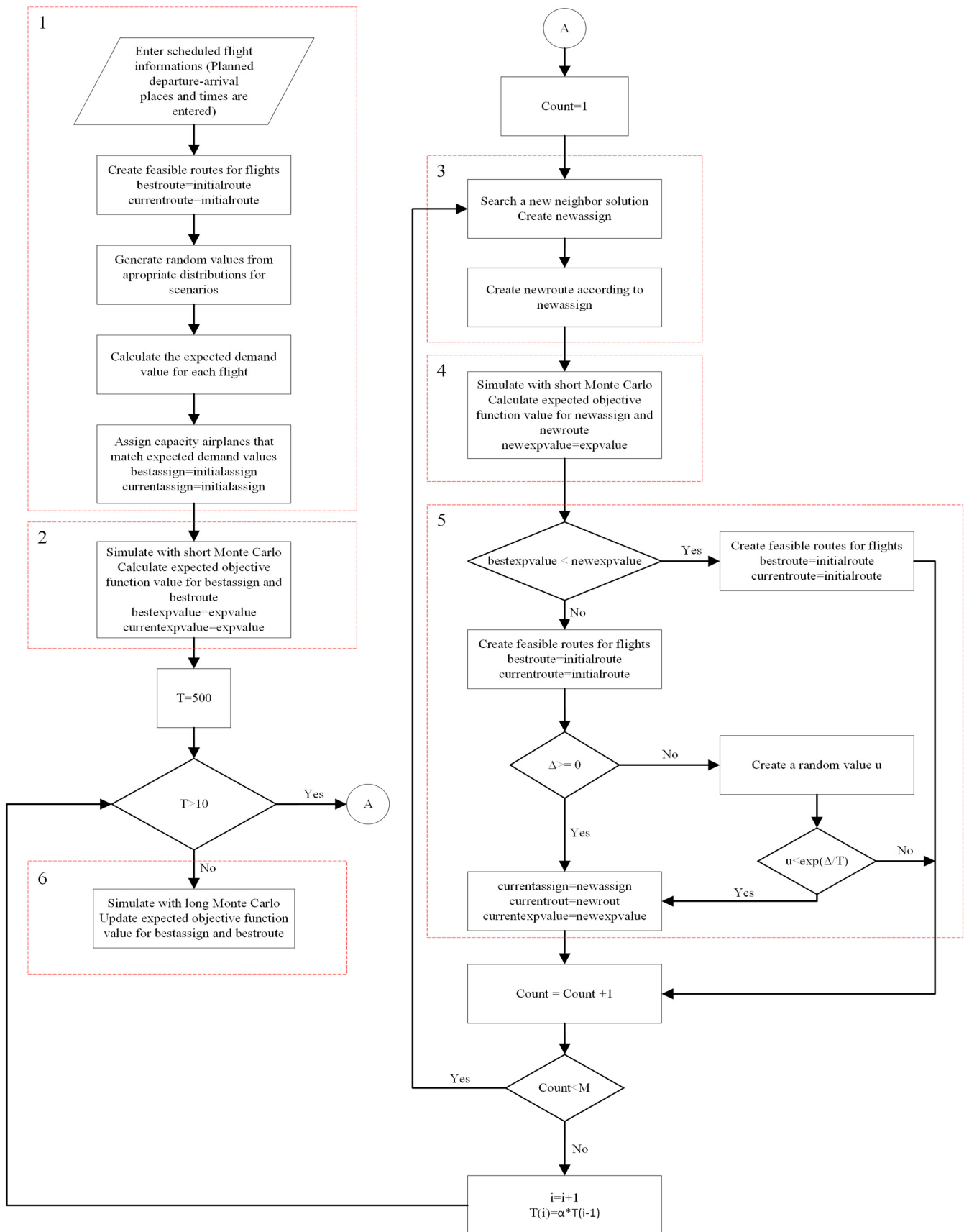


Figure 2. The flow diagram of SA + MC algorithm.

Subsequently, the SA algorithm is initiated with the initial solution obtained in the first red rectangle. At each iteration, a neighboring solution is generated using the algorithm presented in Algorithm 2, as shown in the third red rectangle. The neighboring solution assigns different fleet types to some flights and creates new routes based on this assignment. The expected objective function value is then calculated in a short Monte Carlo simulation of the newly generated neighboring solution, as shown in the fourth red rectangle.

In the fifth red rectangle, the SA algorithm updates the best solution and best objective function value based on the newly generated neighboring solution. Once the stopping criteria are met, the expected objective function value of the best solution in the long Monte Carlo simulation (e.g., 1000 runs) is updated, as shown in the sixth red rectangle.

Algorithm 1: SA + MC Algorithm

Input flight informations

Create initial solution $s_0 \in S$:

 Create feasible routes for flights

Generate random values (μ_{is}, η_{is}) from appropriate distributions for scenarios

 Calculate the expected demand value for each flight

Assign capacity airplanes that match these demand values

Calculate the objective function value $f(s_0)$ with short Monte Carlo simulation of the initial solution

Set an initial temperature: $T > 0$;

$$s = s_0; f(s) = f(s_0);$$

$$s_{iyi} = s_0; f(s_{iyi}) = f(s_0);$$

Repeat

 Count = 1

Repeat

 Generate a neighbor solution s' of S

 Create routes based on new assignment

 Calculate the objective function value $f(s')$ with a short Monte Carlo simulation of s' solution

$$\Delta = f(s') - f(s);$$

 If $\Delta \geq 0$ then $s = s'$;

 If $\Delta < 0$ then generate a random number u in the range $(0,1)$

 If $u < e^{\frac{\Delta}{T}}$ then $s = s'$;

 If $f(s') < f(s_{iyi})$ then $s_{iyi} = s'$

 Count = Count + 1;

Until count > M (number of neighbors to search)

$i = i + 1$;

$T(i) = \alpha * T(i - 1)$;

Until (until the stopping criteria is met)

s_{iyi} best solution

 Calculate objective function value $f(s_{iyi})$ with long Monte Carlo simulation

Algorithm 2: Neighbor Solution Search Algorithm

flt = 0,

y, generate a random value between 0–1

If $y \leq 0.3$ then

 For $(i = 0; i < 3; i++)$

 Assign fleet type assigned to randomly selected flight in the same solution to

 flight $(flt + i)$ in solution

 End for

If not

 For $(i = 0; i < 3; i++)$

 Assign random fleet type to flight $(flt + i)$ in solution

 End for

End if

flt = flt + 1

4.2. Cuckoo Search Algorithm (CS)

The CS algorithm draws inspiration from the aggressive breeding techniques employed by cuckoos. Cuckoos lay their eggs in the nests of other birds that are randomly selected. If the host bird discovers the cuckoo egg, it may either remove the egg or abandon the nest entirely. The eggs that survive form the next generation [70].

The study by Rakesh and Mahesh [71] describes the successful application of the nature-inspired cuckoo algorithm, with various variations, to comprehensive global optimization problems. CS has been shown to outperform other metaheuristics in terms of speed and accuracy in solving the traveling salesman problem (TSP) [72]. Additionally, it has demonstrated a superior ability to find higher quality solutions for constrained optimization problems than genetic algorithm and particle swarm optimization. Its stochastic nature makes it particularly effective in exploring the search space, allowing for its successful application to highly constrained optimization problems such as job scheduling, multi-objective scheduling, flow shop scheduling, and TSP. However, in such problems, some modifications may be required to produce new solutions with Levy flights [73–75]. New solutions can also be produced with the help of heuristics without using the Levy distribution. Gao et al. [76]’s study contains many examples. In this study, improved new solutions with local search heuristics (Algorithm 3) are produced without using levy flights. The local search (Algorithm 4) is also used in the SA + MC algorithm.

The CS algorithm initially starts with a certain number of candidate solutions, then evaluates new solutions and repeats a certain number of bad solutions. The search process in this solution space shows that it has potential to better balance exploitation and exploration. The algorithm has also fewer adjustable parameters [77]. The reason for using CS is that it has these advantages along with its performance in the literature.

The flow diagram of the resulting simheuristic method (CS + MC) is shown in Figure 3. The detailed pseudo code of this flowchart is given in Algorithm 3. The flowchart is divided into small rectangles drawn with dashed lines and numbered to aid in explanation.

Figure 3 illustrates the multi-stage process of the proposed methodology for solving the airline profit maximization problem using the CS algorithm. In the first red rectangle, an initial solution equal in size to the population is generated, following the same approach as is depicted in Figure 2.

In the second red rectangle, the expected objective function value of each initial solution is calculated using a short Monte Carlo simulation (e.g., 100 runs). The initial population is then fed into the CS algorithm, as shown in the third red rectangle.

At each iteration, a randomly selected solution is updated using the local search algorithm outlined in Algorithm 4, as shown in the third red rectangle. The updated solution is then compared to another randomly selected solution from the population, and the best solution and objective function value are updated accordingly. This process is iteratively repeated until the stopping criterion is satisfied.

Finally, the objective function value of the best solution is updated using a long Monte Carlo simulation (e.g., 1000 runs), as shown in the fourth red rectangle. This comprehensive approach ensures that the optimal solution is robust and reliable, accounting for the inherent complexities and uncertainties of real-world airline operations.

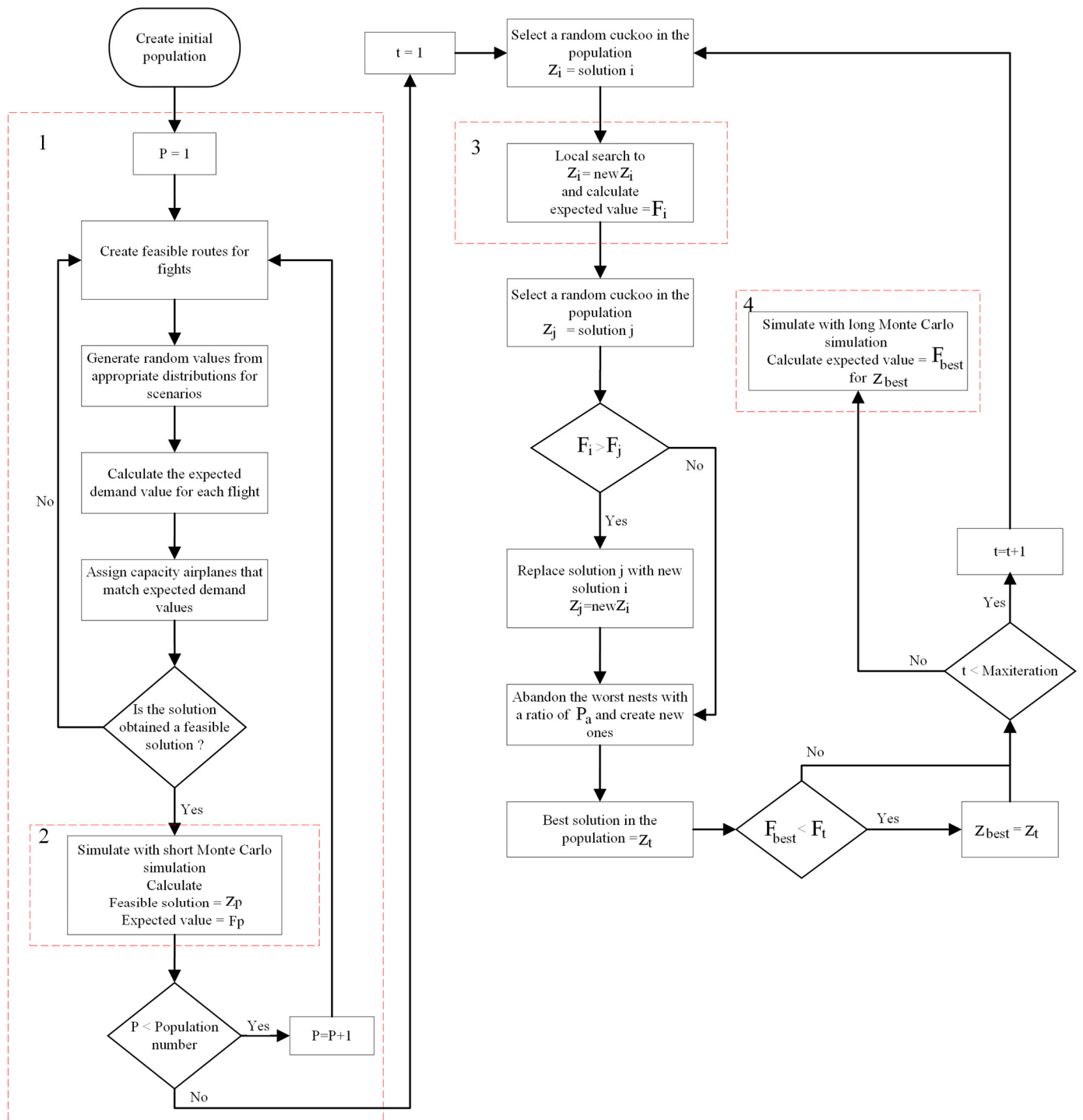


Figure 3. The flow diagram of CS + MC algorithm.

Algorithm 3: CS + MC Algorithm

```

Input flight informations
Generate P initial populations  $x_i$  ( $i = 1, \dots, P$ )
    Create feasible routes for flights
Generate random values  $(\mu_{is}, \eta_{is})$  from appropriate distributions for scenarios
    Calculate the expected demand value for each flight
Assign capacity airplanes that match these demand values
Calculate the objective function value  $f(x_i)$  with short Monte carlo simulation of  $x_i$  solution
While ( $t < \text{Maximum iteration}$ ) do
    Take a random cuckoo  $x_i$ 
    Local search  $x'_i$  and  $f(x'_i)$ 
    Randomly select a nest (let it be  $x_j$ ) in P nests
    If ( $f(x'_i) > f(x_j)$ ) then
        Replace  $x_j$  with new solution
    end if
    Abandon the worst nests with  $\alpha$  ratio of  $p_a$  and create new ones
        Create routes for flights
        Assign a random fleet type to each route
    Calculate the objective function value with a short Monte Carlo simulation of each new solution
    Keep best solutions (nests with quality solution)
    Sort the solutions and find the current best solution
end while
Calculate objective function value with long Monte Carlo simulation of best solution

```

Algorithm 4: Local Search Algorithm

```

For ( $fl = 0$ ;  $fl < \text{flightnumber} - 2$ ;  $fl++$ )
    y, generate a random value between 0–1
    If  $y \leq 0.3$  then
        For ( $i = 0$ ;  $i < 3$ ;  $i++$ )
            Assign fleet type assigned to randomly selected flight in the same solution to
            flight ( $fl + i$ ) in solution  $x_i$ 
            Create routes based on new assignment
            Calculate the objective function value  $f(x'_i)$  with short Monte Carlo simulation of new solution ( $x'_i$ )
            End for
        If not
            For ( $i = 0$ ;  $i < 3$ ;  $i++$ )
                Assign random fleet type to flight ( $fl + i$ ) in solution  $x_i$ 
                Create routes based on new assignment
                Calculate the objective function value  $f(x'_i)$  with short Monte Carlo simulation of new solution ( $x'_i$ )
                End for
            End if
        End for
    End for

```

4.3. Small Example

This section aims to demonstrate the validation and verification of the proposed solution methods through a small sample problem and its corresponding solution. The sample problem includes six flights, two fleet types, two codeshare agreements, and five airports. These airports are Los Angeles (LAX), Dallas Fort Worth (DFW), Atlanta (ATL), Miami (MIA), and Kahului (OGG). The parameters of the flights are given in Table 2. There are two fleet types in the problem: B737-800 and A321-200. The parameters for the fleet type are also given in Table 3.

Table 2. Flight parameters of small example.

Flights (i)	Departure Airports	Arrival Airports	Departure Time	Arrival Time	Min. Departure Time	Max. Departure Time
					$T_i^{\min}$ (min)	$T_i^{\max}$ (min)
1	LAX	ATL	08:00	12:30	470	490
2	ATL	LAX	13:10	17:50	780	800
3	LAX	DFW	18:40	21:55	1110	1130
4	LAX	MIA	05:00	10:10	290	310
5	MIA	LAX	11:30	16:50	680	700
6	LAX	OGG	18:00	23:40	1070	1090

Flights (i)	Spilled Passengers Cost	Misconnected Passenger Cost	Aircraft Delay Cost	Flight Fare per Passenger	Flight Operating Cost per Passenger
	DY_i (USD)	KY_i (USD)	DC_i (USD)	$fare_i$ (USD)	$cost_i$ (USD)
1	102	50	4.85	255	153
2	102	50	7.03	255	153
3	88	50	4.29	220	132
4	163.6	50	3.84	409	245.4
5	124	50	7.21	310	186
6	89.6	50	5.51	224	134

Flights (i)	Codesharing Costs per Passenger		Codesharing Capacity		Min. Cruise Time		Max. Cruise Time		Turnaround Time	
	$code_{vpi}$ (USD)		Cap_{vpi}		CT_{ik}^l (min)		CT_{ik}^u (min)		TA_{ik} (min)	
	$p = 1$	$p = 2$	$p = 1$	$p = 2$	$k = 1$	$k = 2$	$k = 1$	$k = 2$	$k = 1$	$k = 2$
1	76.5	65.03	80	110	213	213	250	250	42	39
2	77	65	80	110	221	221	260	260	46	43
3	66	56.1	80	110	149	149	175	175	44	42
4	122.7	104.3	80	110	247	247	290	290	50	47
5	93	79.05	80	110	255	255	300	300	46	43
6	67.2	57.12	80	110	272	272	320	320	26	25

Table 3. Fleet type parameters.

Fleet type (k)	Number of Aircrafts	Capacity of Fleet Type	Aircraft Idle Time Costs
	AN_k	Kap_k	BOS_k (USD/min)
1	1	172	40
2	1	187	55

The problem has been solved for two scenarios by using the GAMS/BARON optimization package and the proposed solution methods. The solutions, presented in Table 4, show that the GAMS/BARON and proposed solution methods have obtained the same assignments and routes for each scenario. According to the solution, Flights 1, 2, and 3 are codeshare flights and are operated with aircraft under the second type contract. Therefore, these flights are not included in the airport's route plan. Flights 4, 5, and 6 are operated

by the airline's second type of aircraft, specifically, the A321-200. The expected objective function value has been obtained by averaging the objective functions of the scenarios. The results of the SA + MC and CS + MC are consistent with GAMS result. Therefore, the validation and verification of the proposed methods have been successfully completed. The detailed plan for each scenario that includes the actual departure time, actual arrival time, cruise time, idle time, delay time, and non-cruise time are listed in Table 5.

Table 4. Solution results of the sample problem.

	GAMs/BARON	SA + MC	CS + MC
Objective function values (USD)	89,422.62	89,422.62	89,422.62
Assignments	1-codeshare 2-codeshare 3-codeshare	4-A321-200 5-A321-200 6-A321-200	
Route	4-5-6		
Stations	Codeshare: LAX-ATL Codeshare: ATL-LAX Codeshare: LAX-DFW Route: LAX-MIA-LAX-OGG		

Table 5. Solutions of the scenarios.

	Flights	Actual Departure Time	Actual Arrival Time	Cruise Time (min)	Idle Time (min)	Delay Time (min)	Non-Cruise Time (min)	Demand	Passenger Number
First scenario results	1							147	110
	2							74	74
	3							220	110
	4	05:44	10:58	290	0	34	24	129	129
	5	11:45	17:07	300	0	5	22	201	187
	6	17:50	22:46	272	0		24	172	172
Second scenario results	1							83	83
	2							136	110
	3							65	65
	4	05:45	10:57	290	0	35	22	138	138
	5	11:44	17:07	300	0	4	23	159	159
	6	17:50	22:43	272	0		21	170	170

5. Computational Experiments

5.1. Data Descriptions

Four test problems have been generated using data obtained from the “Airline On-Time Performance Data” open access database (BTS) [57]. The flight information has been filtered to include only flights departing and landing at Los Angeles City airport (LAX) from American Airlines. The resulting test problems have 30, 78, 127, and 180 flights, respectively, and include six fleet types. Table 6 provides general information about the fleet types used in the test problems.

Table 6. Properties of fleet types.

Fleet Type	B737-800	A321-200	A319-100	A321-NEO	B787-8	ERJ-175
Capacity	172	187	128	196	234	76
Base Turntime	25.47	25.47	23.87	31.05	42.39	16.47
Idle Time Costs (USD/min)	140	142	136	144	147	125

In the context of codeshare flights, the model assumes the existence of two contract types, which entail the operating carrier receiving 30% and 23% of the ticket price, respectively. The passenger capacities associated with these contracts are set at 56 and 80. To obtain the codeshare ticket prices, the model leverages data from the airline's website. It should be noted that the total passenger capacity on codeshare flights is subject to the constraint that it cannot exceed one third of the airline's overall aircraft capacity.

It is worth mentioning that the model relies on a number of parameters that have been previously determined by Şafak et al. [36,37]. These parameters are critical to ensuring the validity and reliability of the model's outputs:

- A 10 min tolerance period is added to the departure times.
- The turnaround time of each fleet type is multiplied by the complexity coefficient of the airport for each flight and fleet type to obtain the turnaround time for that flight.
- The connection time for the passengers is set at 30 min.
- The highest and lowest cruise times are calculated by subtracting 20 min from the total flight time. The lowest cruise times are then compressed by 15% of the highest cruise time.
- The profit from one passenger is defined as the basic spill cost of a lost passenger. The passenger spill cost for each flight has been calculated by multiplying the basic spill cost by the airport complexity coefficient.
- The cost of a missed connection is set at USD 50 per passenger.

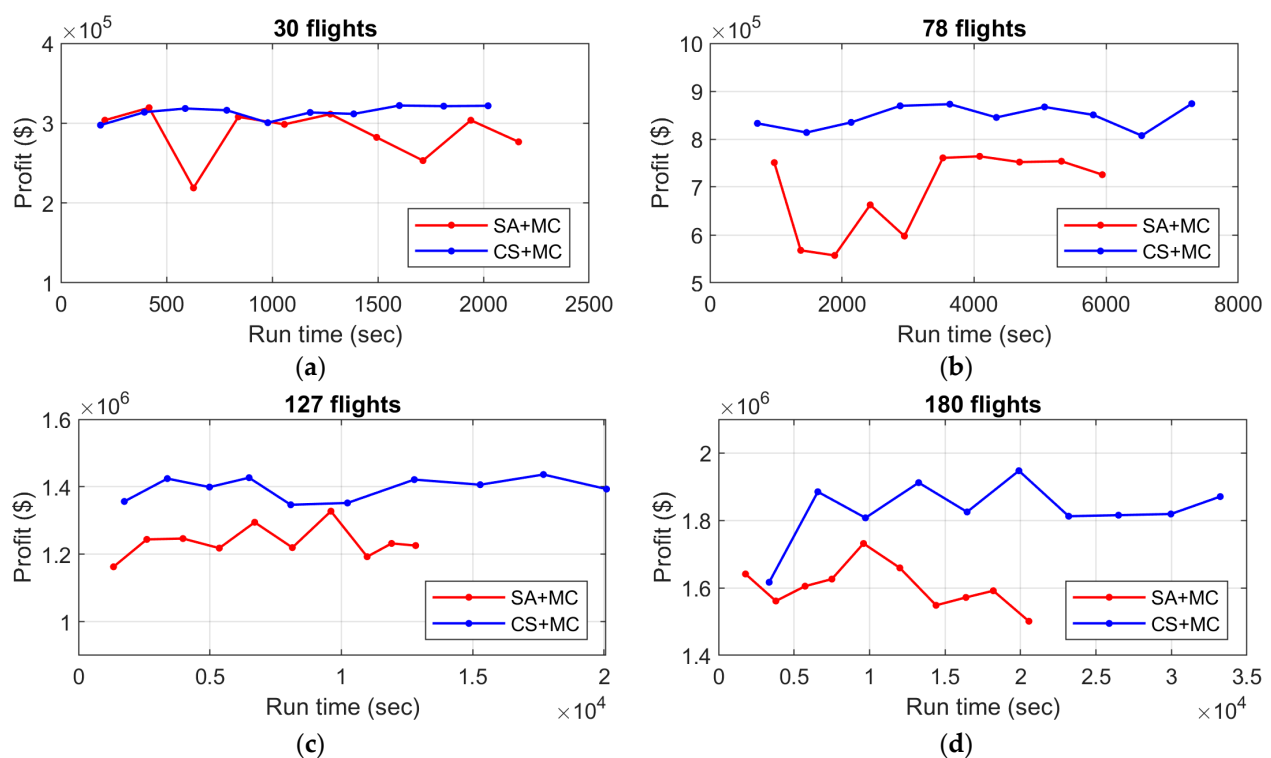
In order to solve the problem with 1000 scenarios, 30 flights, and 6 fleet types, the corresponding TSS_MINLP formulation has 704,532 constraints, 518,577 variables, and 3,122,383 non zeroes after the pre-solve is completed. This calculation also describes the complexity of the algorithm. As the size of the problem increases, it becomes more difficult to solve. Therefore, simheuristic solution methods have been used to solve the test problems. Based on experimental studies, the following parameters have been used for the SA algorithm: an initial temperature of 500, a final temperature of 10, a search for 100 neighbors in each step, and a temperature reduction ratio of 0.99. For the CS algorithm, the population size is set to 20, the rate of changing bad solutions is set to 0.2, and the maximum number of iterations is set to 500.

5.2. Computational Results

The proposed methods for solving large-scale problems have been implemented using a C# program and utilized to solve the test problems. Their performances have been compared by using the same stochastic values. Each test problem has been executed with 100 scenarios in a short simulation and 1000 scenarios in a long simulation, with the number of aircraft and flights per fleet type varying based on the problem size. The test problems have been solved using the proposed solution methods for the same total run times, and the best results have been given in Table 7. The number of algorithm runs during these periods and the results obtained in each run are presented in Figure 4a–d. The results indicate that CS + MC achieved better profits than SA + MC, possibly due to the CS algorithm's population-based approach, which expands the search space with more diverse solutions. Furthermore, unlike in Şafak's study [37], where scenarios have been grouped and tested, this study tested many scenarios using metaheuristics and obtained fast and effective results.

Table 7. Comparison of results of test problems.

Problems	Scenario Numbers	Flight Numbers	Aircraft Numbers	Obj. Function Values (USD)	SA + MC			CS + MC		
					Code Flight Numbers	Times (s)	Obj. Function Values (USD)	Code Flight Numbers	Times (s)	Obj. Function Values (USD)
1	100	30	6	319,314.3	0	209.26	322,102.7	0	216.01	
2	100	78	14	767,161.46	25	665.96	874,041.44	2	758.32	
3	100	127	20	1,327,546.52	28	1467.69	1,436,292.3	12	2406.63	
4	100	180	25	1,731,738.91	56	2097.55	1,947,504.96	12	3432.55	

**Figure 4.** Comparison of profits of algorithms (a) for 30 flights, (b) for 78 flights, (c) for 127 flights, (d) for 180 flights.

5.3. Effect of Codeshare Flights

To analyze the impact of the codesharing strategy, test problem 4 (180 flights) has been solved again without the codesharing option, and the obtained solutions are presented in Table 8. In these solutions, the number of single-flight routes represents the number of dead flights, which incur additional costs for the airline since the aircraft has to fly to the starting city of the other route without passengers. The solutions obtained by considering the CA for the same problem are given in Table 9, including the number of routes and the number of flights made with the CA. The effect of codesharing on the profits in each algorithm is shown in Figure 5a,b. These results confirm that codesharing has a significant contribution to increasing the airline's profit.

Table 8. Results of proposed methods for 180 flights without codeshare flights.

Run	SA + MC			CS + MC		
	Profit (USD)	Route Number	Deadhead Flights	Profit (USD)	Route Number	Deadhead Flights
1	1,270,180.8	44	21	1,605,883.39	46	14
2	1,048,609.36	44	21	1,553,091.89	45	15
3	1,138,399.14	44	21	1,572,453.71	46	10
4	1,000,286.11	45	17	1,477,181.86	46	12
5	1,240,760.95	45	17	1,618,113.63	45	17
6	1,282,708.31	44	20	1,479,553.91	45	15
7	1,183,632.79	45	13	1,542,911.92	46	10
8	1,173,466.25	43	26	1,623,962.93	46	10
9	1,015,405.99	46	10	1,485,155.01	46	10
10	1,137,581.98	43	25	1,591,569.03	46	10

Table 9. Results of proposed methods for 180 flights with codeshare flights.

Run	SA + MC			CS + MC		
	Profit (USD)	Route Number	Codeshare Flights	Profit (USD)	Route Number	Codeshare Flights
1	1,641,346.89	38	52	1,616,784.05	44	24
2	1,561,180.42	36	62	1,885,131.11	42	28
3	1,605,142.62	36	62	1,807,812.28	43	30
4	1,626,210.07	36	54	1,912,243.16	45	14
5	1,731,738.91	37	56	1,825,287.15	40	37
6	1,659,333.69	39	50	1,947,504.96	46	12
7	1,548,412.45	36	54	1,812,634.21	43	33
8	1,571,768.08	37	55	1,815,737.61	43	26
9	1,591,618.95	36	57	1,819,234.24	44	20
10	1,501,147.29	35	64	1,870,894.85	42	28

Since airline operations have a complex structure, there are differences between the planned and actual schedule. Evaluating scenarios containing uncertainty can minimize these differences. The results obtained in this study were obtained by using many scenarios and two algorithms. Looking at the literature, software programs are available to simulate airline operations. For example, open data and open source simulation programs are available for air traffic management (AirTrafficSim, BlueSky). The programs quickly simulate operations and take into account a large number of scenarios. They are easy-to-use and accessible programs [78,79]. The effectiveness of the results in the scenarios we produced in our study can be seen by testing them with such simulators. Additionally, better results can be achieved with more scenarios.

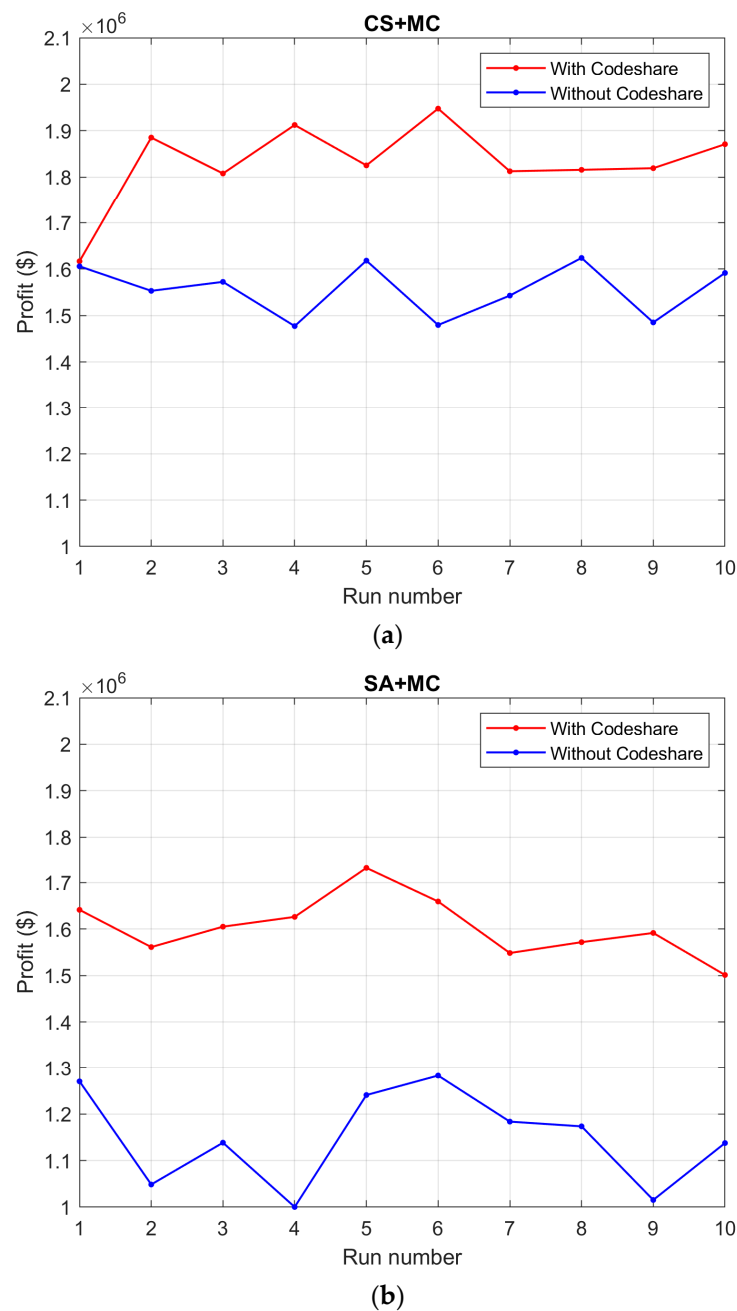


Figure 5. The effect of codesharing on profits with (a) SA + MC simheuristic algorithm and (b) CS + MC simheuristic algorithm.

6. Conclusions

The aviation industry has witnessed a steady increase in demand owing to its speed, convenience, and reliability. Airlines face numerous challenges to serve a vast network of destinations. Foremost among these, there are the high costs of investing in a large fleet, personnel, and equipment. To mitigate these costs, airlines have resorted to forming strategic partnerships through codesharing, enabling them to leverage each other's flights.

In addition to this, airlines face the challenge of optimizing their operations across multiple domains such as FS, FA, and AR. While solving each of these problems independently may seem like an option, doing so may lead to local optima and increase operational costs. To overcome this, it is imperative to simultaneously address all these problems in a model that yields a more effective solution.

To address these challenges, this study proposes a novel model that integrates the FS, FA, and AR problems with codesharing options. Specifically, a three-stage stochastic non-linear mixed-integer mathematical model is developed. The first step involves determining the flights that will be operated by the airline and those that will be operated under the CA. Based on this, an appropriate fleet type is assigned to each flight, and the routes are created. In terms of the second stage, the estimated departure time for the airline's own flights is selected from a predetermined time window. In the final step, the timetable of the routes assigned to each aircraft is updated, and the number of passengers to be carried by the aircraft is determined.

However, another critical challenge that makes solving the aforementioned optimization problems complex is the uncertainty surrounding customer demand and non-cruise times. Capturing these uncertainties accurately is crucial in modeling and decision-making problems. The developed model has the ability to model uncertainties in demand and non-cruise time parameters stochastically. The variation in the values of stochastic parameters will have an impact on the decisions to be made. Therefore, evaluating a wide range of scenarios reflecting the different values of stochastic parameters will ensure robust results. In this study, the Monte Carlo technique is used to analyze a vast number of scenarios for stochastic parameters.

On the other hand, the increase in the number of flights and scenarios makes it impractical to solve the proposed model using optimization software. Therefore, in this study, the SA and CS algorithms are proposed for solving the non-linear mixed-integer model by integrating the Monte Carlo technique. Testing these algorithms on real-life data reveals that the proposed model accurately represents the problem and produces efficient and robust results with the proposed solution algorithms. The CS algorithm is shown to have a better performance than SA for the integrated FS, FA, and AR problems.

The proposed model assumes that all fleet types can use all airports. However, in reality, certain fleets may not be able to use certain airports for technical reasons, and this restriction is referred to as station purity. This limitation serves as a constraint for the proposed model. Additionally, the developed model neglects the option of overbooking, where airlines sell more tickets than the airplane's capacity. The model considers a single fare class but does not account for fare classes such as business class and first class. All these limitations present research potential for future studies. Finally, techniques such as Sample Average Approximation could be implemented to the metaheuristics to analyze stochastic parameters.

Author Contributions: K.K.: Methodology; Software; Data Curation; Writing—Original Draft Preparation. Ü.S.S.: Conceptualization; Validation; Investigation; Writing—Review & Editing; Supervision. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The dataset generated for the test problems is available in the Airline On-Time Performance Data, open access database, "https://www.transtats.bts.gov/Fields.asp?gnoyr_VQ=FGK (accessed on 10 March 2020)".

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Kalafatoğlu, Y. Effects of Pricing and Fleet Structure on the Airline Fleet Assignment Problem. Master's Thesis, The Graduate School of Engineering and Science of Boğaziçi University, İstanbul, Türkiye, 2014.
2. Lohatepanont, M.; Barnhart, C. Airline schedule planning: Integrated models and algorithms for schedule design and fleet assignment. *Transp. Sci.* **2004**, *38*, 19–32. [CrossRef]
3. Abdelghany, A.; Abdelghany, K. *Modeling Applications in the Airline Industry*; Asghate: Burlington, VT, USA, 2009; pp. 53–129.
4. Yimga, J. Code-Sharing Agreements and Path Quality in the US Airline Industry. *Transp. Policy* **2022**, *116*, 369–385. [CrossRef]
5. Zou, L.; Chen, X. The Effect of Code-Sharing Alliances on Airline Profitability. *J. Air Transp. Manag.* **2017**, *58*, 50–57. [CrossRef]
6. Kenan, N.; Diabat, A.; Jebali, A. Codeshare agreements in the integrated aircraft routing problem. *Transp. Res. Part B Methodol.* **2018**, *117*, 272–295. [CrossRef]

7. Sarsenov, B. Fleet Assignment Problem and a Case Study for Turkish Airlines. Master's Thesis, Yıldız Technical University, İstanbul, Türkiye, 2011.
8. Abara, J. Applying integer linear programming to the fleet assignment problem. *Interfaces* **1989**, *19*, 20–28. [\[CrossRef\]](#)
9. Hane, C.A.; Barnhart, C.; Johnson, E.L.; Marsten, R.E.; Nemhauser, G.L.; Sigismondi, G. The fleet assignment problem: Solving a large-scale integer program. *Math. Program.* **1995**, *70*, 211–232. [\[CrossRef\]](#)
10. Clarke, L.W.; Hane, C.A.; Johnson, E.L.; Nemhauser, G.L. Maintenance and crew considerations in fleet assignment. *Transp. Sci.* **1996**, *30*, 249–260. [\[CrossRef\]](#)
11. Berge, M.E.; Hopperstad, C.A. Demand driven dispatch: A method for dynamic aircraft capacity assignment, models and algorithms. *Oper. Res.* **1993**, *41*, 153–168. [\[CrossRef\]](#)
12. Talluri, K.T. Swapping applications in a daily airline fleet assignment. *Transp. Sci.* **1996**, *30*, 237–248. [\[CrossRef\]](#)
13. Levin, A. Scheduling and fleet routing models for transportation systems. *Transp. Sci.* **1971**, *5*, 232–255. [\[CrossRef\]](#)
14. Rexing, B.; Barnhart, C.; Kniker, T.; Jarrah, A.; Krishnamurthy, N. Airline fleet assignment with time windows. *Transp. Sci.* **2000**, *34*, 1–130. [\[CrossRef\]](#)
15. Belanger, N.; Desautniers, G.; Soumis, F.; Desrosiers, J. Periodic airline fleet assignment with time windows, spacing constraints, and time dependent revenues. *Eur. J. Oper. Res.* **2006**, *175*, 1754–1766. [\[CrossRef\]](#)
16. Zeghal, F.M.; Haouari, M.; Sherali, H.D.; Aissaoui, N. Flexible aircraft fleet and routing at Tunis Air. *J. Oper. Res. Soc.* **2011**, *62*, 368–380. [\[CrossRef\]](#)
17. Barnhart, C.; Kniker, T.S.; Lohatepanont, M. Itinerary-based airline fleet assignment. *Transp. Sci.* **2002**, *36*, 199–217. [\[CrossRef\]](#)
18. Rosenberger, J.M.; Johnson, E.L.; Nemhauser, G.L. A robust fleet-assignment model with hub isolation and short cycles. *Transp. Sci.* **2004**, *38*, 357–368. [\[CrossRef\]](#)
19. Cadarso, L.; Marín, Á. Robust passenger-oriented timetable and fleet assignment integration in airline planning. *J. Air Transp. Manag.* **2013**, *26*, 44–49. [\[CrossRef\]](#)
20. Smith, B.C.; Johnson, E.L. Robust airline fleet assignment: Imposing station purity using station decomposition. *Transp. Sci.* **2006**, *40*, 497–516. [\[CrossRef\]](#)
21. Sherali, H.D.; Bish, E.K.; Zhu, X. Polyhedral analysis and algorithms for a demand-driven re-fleet model for aircraft assignment. *Transp. Sci.* **2005**, *39*, 349–366. [\[CrossRef\]](#)
22. Jacobs, T.L.; Smith, B.C.; Johnson, E.L. Incorporating network flow effects into the airline fleet assignment process. *Transp. Sci.* **2008**, *42*, 514–529. [\[CrossRef\]](#)
23. Dumas, J.; Aithard, F.; Soumis, F. Improving the objective function of the fleet assignment problem. *Transp. Res. Part B Methodol.* **2009**, *43*, 466–475. [\[CrossRef\]](#)
24. Pilla, V.L.; Rosenberger, J.M.; Chen, V.; Engsuwan, N.; Siddappa, S. A multivariate adaptive regression splines cutting plane approach for solving a two-stage stochastic programming fleet assignment model. *Eur. J. Oper. Res.* **2012**, *216*, 162–171. [\[CrossRef\]](#)
25. Sherali, H.D.; Zhu, X. Two-stage fleet assignment model considering stochastic passenger demands. *Oper. Res.* **2008**, *56*, 383–399. [\[CrossRef\]](#)
26. Pilla, V.L.; Rosenberger, J.M.; Chen, V.C.; Smith, B. A statistical computer experiments approach to airline fleet assignment. *IIE Trans.* **2008**, *40*, 524–537. [\[CrossRef\]](#)
27. Naumann, M.; Suhl, L.; Friedemann, M. A stochastic programming model for integrated planning of re-fleet and financial hedging under fuel price and demand uncertainty. *Procedia-Soc. Behav. Sci.* **2012**, *54*, 47–55. [\[CrossRef\]](#)
28. Cadarso, L.; Celis, R. Integrated airline planning: Robust update of scheduling and fleet balancing under demand uncertainty. *Transp. Res. Part C Emerg. Technol.* **2017**, *81*, 227–245. [\[CrossRef\]](#)
29. Atasoy, B.; Salani, M.; Bierlaire, M. An integrated airline scheduling, fleet, and pricing model for a monopolized market. *Comput.-Aided Civ. Infrastruct. Eng.* **2014**, *29*, 76–90. [\[CrossRef\]](#)
30. Pita, J.P.; Barnhart, C.; Antunes, A.P. Integrated flight scheduling and fleet assignment under airport congestion. *Transp. Sci.* **2013**, *47*, 477–492. [\[CrossRef\]](#)
31. Kenan, N.; Jebali, A.; Diabat, A. An integrated flight scheduling and fleet assignment problem under uncertainty. *Comput. Oper. Res.* **2018**, *100*, 333–342. [\[CrossRef\]](#)
32. Sherali, H.D.; Bish, E.K.; Zhu, X. Airline fleet assignment concepts, models, and algorithms. *Eur. J. Oper. Res.* **2006**, *172*, 1–30. [\[CrossRef\]](#)
33. Sherali, H.D.; Bae, K.H.; Haouari, M. A benders decomposition approach for an integrated airline schedule design and fleet assignment problem with flight retiming, schedule balance, and demand recapture. *Ann. Oper. Res.* **2013**, *210*, 213–244. [\[CrossRef\]](#)
34. Gürkan, H.; Gürel, S.; Aktürk, M.S. An integrated approach for airline scheduling, aircraft fleet and routing with cruise speed control. *Transp. Res. Part C Emerg. Technol.* **2016**, *68*, 38–57. [\[CrossRef\]](#)
35. Jamili, A. A robust mathematical model and heuristic algorithms for integrated aircraft routing and scheduling, with consideration of fleet assignment problem. *J. Air Transp. Manag.* **2017**, *58*, 21–30. [\[CrossRef\]](#)
36. Şafak, Ö.; Gürel, S.; Aktürk, M.S. Integrated aircraft-path assignment and robust schedule design with cruise speed control. *Comput. Oper. Res.* **2017**, *84*, 127–145. [\[CrossRef\]](#)
37. Şafak, Ö.; Çavuş, Ö.; Aktürk, M.S. Multi-stage airline scheduling problem with stochastic passenger demand and non-cruise times. *Transp. Res. Part B Methodol.* **2018**, *114*, 39–67. [\[CrossRef\]](#)

38. Cacchiani, V.; Salazar-González, J.J. Heuristic approaches for flight retiming in an integrated airline scheduling problem of a regional carrier. *Omega* **2020**, *91*, 102028. [\[CrossRef\]](#)
39. Xu, Y.; Wandelt, S.; Sun, X. Airline integrated robust scheduling with a variable neighborhood search based heuristic. *Transp. Res. Part B Methodol.* **2021**, *149*, 181–203. [\[CrossRef\]](#)
40. Şafak, Ö.; Atamtürk, A.; Aktürk, M.S. Accommodating new flights into an existing airline flight schedule. *Transp. Res. Part C Emerg. Technol.* **2019**, *104*, 265–286. [\[CrossRef\]](#)
41. Barnhart, C.; Boland, N.L.; Clarke, L.W.; Johnson, E.L.; Nemhauser, G.L.; Shenoi, R.G. Flight string models for aircraft fleet and routing. *Transp. Sci.* **1998**, *32*, 208–220. [\[CrossRef\]](#)
42. Sosnowska, D.; Rolim, J. Fleet scheduling optimization: A simulated annealing approach. In *International Conference on the Practice and Theory of Automated Timetabling*; Springer: Berlin/Heidelberg, Germany, 2000. [\[CrossRef\]](#)
43. Yan, S.; Tseng, C.H. A passenger demand model for airline flight scheduling and fleet routing. *Comput. Oper. Res.* **2002**, *29*, 1559–1581. [\[CrossRef\]](#)
44. Ahuja, R.K.; Liu, J.; Orlin, J.B.; Goodstein, J.; Mukherjee, A. A neighborhood search algorithm for the combined through and fleet assignment model with time windows. *Netw. Int. J.* **2004**, *44*, 160–171. [\[CrossRef\]](#)
45. Li, Y.; Wang, X. Integration of fleet assignment and aircraft routing. *Transp. Res. Rec.* **2005**, *1915*, 79–84. [\[CrossRef\]](#)
46. Yan, S.; Chen, S.C.; Chen, C.H. Air cargo fleet routing and timetable setting with multiple on-time demands. *Transp. Res. Part E Logist. Transp. Rev.* **2006**, *42*, 409–430. [\[CrossRef\]](#)
47. Yan, S.; Chen, C.H. Coordinated scheduling models for allied airlines. *Transp. Res. Part C Emerg. Technol.* **2007**, *15*, 246–264. [\[CrossRef\]](#)
48. Haouari, M.; Aissaoui, N.; Mansour, F.Z. Network flow-based approaches for integrated aircraft fleet and routing. *Eur. J. Oper. Res.* **2009**, *193*, 591–599. [\[CrossRef\]](#)
49. Sherali, H.D.; Bae, K.H.; Haouari, M. Integrated airline schedule design and fleet assignment: Polyhedral analysis and benders' decomposition approach. *INFORMS J. Comput.* **2010**, *22*, 500–513. [\[CrossRef\]](#)
50. Liang, Z.; Chaovalitwongse, W.A. A network-based model for the integrated weekly aircraft maintenance routing and fleet assignment problem. *Transp. Sci.* **2013**, *47*, 493–507. [\[CrossRef\]](#)
51. Shao, S.; Sherali, H.D.; Haouari, M. A novel model and decomposition approach for the integrated airline fleet assignment, aircraft routing, and crew pairing problem. *Transp. Sci.* **2015**, *51*, 233–249. [\[CrossRef\]](#)
52. Dong, Z.; Chuhan, Y.; Lau, H.H. An integrated flight scheduling and fleet assignment method based on a discrete choice model. *Comput. Ind. Eng.* **2016**, *98*, 195–210. [\[CrossRef\]](#)
53. Liu, W.M.; Zhu, X.H.; Qi, Y.L. Integrated fleet assignment and aircraft routing based on delay propagation. *Sādhanā* **2016**, *41*, 713–719. [\[CrossRef\]](#)
54. Cacchiani, V.; Salazar-González, J.J. Optimal solutions to a real-world integrated airline scheduling problem. *Transp. Sci.* **2017**, *51*, 250–268. [\[CrossRef\]](#)
55. Wei, M.; Zhao, L.; Ye, Z.; Jing, B. An integrated optimization mode for multi-type aircraft flight scheduling and routing problem. *Math. Biosci. Eng.* **2020**, *17*, 4990–5004. [\[CrossRef\]](#) [\[PubMed\]](#)
56. Ahmed, M.B.; Hryhoryeva, M.; Hvattum, L.M.; Haouari, M. A matheuristic for the robust integrated airline fleet assignment, aircraft routing, and crew pairing problem. *Comput. Oper. Res.* **2022**, *137*, 105551. [\[CrossRef\]](#)
57. BTS. Airline on-Time Statistics. Available online: <https://www.transtats.bts.gov/ONTIME/Index.aspx> (accessed on 10 March 2020).
58. Byrne, M.D.; Hossain, M.M. Production planning: An improved hybrid approach. *Int. J. Prod. Econ.* **2005**, *93*, 225–229. [\[CrossRef\]](#)
59. Jung, J.Y.; Blau, G.; Pekny, J.F.; Reklaitis, G.V.; Eversdyk, D. Integrated safety stock management for multi-stage supply chains under production capacity constraints. *Comput. Chem. Eng.* **2008**, *32*, 2570–2581. [\[CrossRef\]](#)
60. Stanciu, A.; Vargas, L.; May, J. A revenue management approach for managing operating room capacity. In *Proceedings of the 2010 Winter Simulation Conference*, Baltimore, MD, USA, 5–8 December 2010; IEEE: Piscataway, NJ, USA, 2010. [\[CrossRef\]](#)
61. Juan, A.A.; Faulin, J.; Grasman, S.E.; Rabe, M.; Figueira, G. A review of simheuristics: Extending metaheuristics to deal with stochastic combinatorial optimization problems. *Oper. Res. Perspect.* **2015**, *2*, 62–72. [\[CrossRef\]](#)
62. Yazdani, M.; Mojtahedi, M.; Loosemore, M. Enhancing evacuation response to extreme weather disasters using public transportation systems: A novel simheuristic approach. *J. Comput. Des. Eng.* **2020**, *7*, 195–210. [\[CrossRef\]](#)
63. Latorre-Biel, J.I.; Ferone, D.; Juan, A.A.; Faulin, J. Combining simheuristics with Petri nets for solving the stochastic vehicle routing problem with correlated demands. *Expert Syst. Appl.* **2021**, *168*, 114240. [\[CrossRef\]](#)
64. Gruler, A.; Panadero, J.; Armas, J.; Pérez, J.A.M.; Juan, A.A. A variable neighborhood search simheuristic for the multiperiod inventory routing problem with stochastic demands. *Int. Trans. Oper. Res.* **2020**, *27*, 314–335. [\[CrossRef\]](#)
65. Chica, M.; Juan Pérez, A.A.; Cordon, O.; Kelton, D. Why simheuristics? Benefits, limitations, and best practices when combining metaheuristics with simulation. *Soc. Sci. Res. Netw.* **2017**. [\[CrossRef\]](#)
66. Kirkpatrick, S.; Gelatt, C.D., Jr.; Vecchi, M.P. Optimization by simulated annealing. *Science* **1983**, *220*, 671–680. [\[CrossRef\]](#)
67. Gendreau, M.; Potvin, J.Y. Metaheuristics in combinatorial optimization. *Ann. Oper. Res.* **2005**, *140*, 189–213. [\[CrossRef\]](#)
68. Suman, B.; Kumar, P. A Survey of Simulated Annealing as a Tool for Single and Multiobjective Optimization. *J. Oper. Res. Soc.* **2006**, *57*, 1143–1160. [\[CrossRef\]](#)
69. Amine, K. Multiobjective Simulated Annealing: Principles and Algorithm Variants. *Adv. Oper. Res.* **2019**, *2019*, 8134674. [\[CrossRef\]](#)

70. Yang, X.S.; Deb, S. Cuckoo search via Lévy flights. In Proceedings of the 2009 World Congress on Nature & Biologically Inspired Computing, Coimbatore, India, 9–11 December 2009; IEEE: Piscataway, NJ, USA, 2009. [\[CrossRef\]](#)
71. Rakesh, S.; Mahesh, S. A comprehensive overview on variants of cuckoo search algorithm and applications. In Proceedings of the 2017 International Conference on Electrical, Electronics, Communication, Computer, and Optimization Techniques (ICEECCOT), Mysuru, India, 15–16 December 2017; IEEE: Piscataway, NJ, USA, 2017. [\[CrossRef\]](#)
72. Ouaraab, A.; Ahiod, B.; Yang, X.S. Discrete cuckoo search algorithm for the travelling salesman problem. *Neural Comput. Appl.* **2014**, *24*, 1659–1669. [\[CrossRef\]](#)
73. Sadeghi, M.; Aghayan, I.; Ghaznavi, M. A Cuckoo Search Based Approach to Design Sustainable Transit Network. *Transp. Lett. Int. J. Transp. Res.* **2020**, *13*, 635–648. [\[CrossRef\]](#)
74. Alssager, M.; Othman, Z.A.; Ayob, M.; Mohamad, R.; Yuliansyah, H. Hybrid cuckoo search for the capacitated vehicle routing problem. *Symmetry* **2020**, *12*, 2088. [\[CrossRef\]](#)
75. Sangeetha, R.; Srinivasan, A. Mutated Cuckoo Search Algorithm for Dynamic Vehicle Routing Problem and Synchronization Occurs within the Time Slots in Home Healthcare. *Int. J. Syst. Assur. Eng. Manag.* **2021**. [\[CrossRef\]](#)
76. Gao, S.; Gao, Y.; Zhang, Y.; Xu, L. Multi-Strategy Adaptive Cuckoo Search Algorithm. *IEEE Access* **2019**, *7*, 137642–137655. [\[CrossRef\]](#)
77. Tarhini, A.; Danach, K.; Harfouche, A. Swarm Intelligence-Based Hyper-Heuristic for the Vehicle Routing Problem with Prioritized Customers. *Ann. Oper. Res.* **2020**, *308*, 549–570. [\[CrossRef\]](#)
78. Hui, K.Y.; Nguyen, C.; Lui, G.N.; Liem, R.P. AirTrafficSim: An Open-Source Web-Based Air Traffic Simulation Platform. *J. Open Source Softw.* **2023**, *8*, 4916. [\[CrossRef\]](#)
79. Hoekstra, J.M.; Ellerbroek, J. BlueSky ATC simulator project: An open data and open source approach. In Proceedings of the 7th International Conference on Research in Air Transportation, Philadelphia, PA, USA, 20–24 June 2006; FAA/Eurocontrol: Washington, DC, USA, 2016; Volume 131, p. 132.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.