Contents lists available at GrowingScience

International Journal of Industrial Engineering Computations

homepage: www.GrowingScience.com/ijiec

Airline operational crew-aircraft planning considering revenue management: A robust optimization model under disruption

Ashkan Teymouri[a], Hadi Sahebi[a*] and Mir Saman Pishvaee[a]

[a]*School of Industrial Engineering, Iran University of Science and Technology, Tehran, Iran*

C H R O N I C L E

Article history:
Received September 18 2022
Accepted December 24 2022
Available online
December, 24 2022

Keywords:
Airline Revenue Management
Crew Scheduling
Aircraft Maintenance Routing
Airline Disruption
Efficient Metaheuristic
Adaptive Large Neighborhood Searching
Adaptive Robust Scenario-based Optimization

A B S T R A C T

Airline planning involves various issues that, in a general, can be grouped as network planning, schedule design and fleet planning, aircraft planning, and crew scheduling decisions. This study mainly aims to optimize the Crew Scheduling (CS) decisions considering the operational constraints related to Aircraft Maintenance Routing (AMR) regulations. Since, after fuel, crew costs are vital for airlines, and aircraft maintenance constraints are important operationally, the integrated Crew Scheduling and Aircraft Maintenance Routing (CS-AMR) problem is an important issue for the airlines. The present research addresses this problem using the Revenue Management (RM) approach under some disruption scenarios in the initial schedule. The proposed approach enables airlines to make more efficient decisions during disruptions to prevent flight delay/cancellation costs and recaptures an acceptable part of the spilled demand caused by disruption through the fleet stand-by capacity. This approach considers a set of disruptions in the flight schedule under different probable scenarios and provides the optimal decisions. Accordingly, airlines have two decision-making stages: Here-and-Now (HN) decisions related to the initial schedule for crew, aircraft routing and stand-by capacity to face probable disruptions and Wait-and-See (WS) decisions that determine what the executive plan of each crew and aircraft should be under each scenario, and how to use different options for flight cancellation and substitution. To this end, a novel Two-Stage Robust Scenario-based Optimization (TSRSO) model is proposed that considers the HN and WS decisions simultaneously. A numerical example is solved, and its results verify the applicability and evaluate the performance of the proposed TSRSO model. Regarding the complexity of the proposed MILP model categorized as NP-hard problems, we develop a computationally efficient solution method to solve large-scale problem instances. A single-agent local search metaheuristic algorithm, Adaptive Large Neighborhood Search (ALNS), is applied to solve the CS-AMR problem efficiently. According to the result obtained by applying the proposed revenue management approach for the CS-AMR problem, airlines can drive a robust solution under disruption scenarios that not only minimizes the total delay/cancellation costs but also increases the profit by recapturing the spilled demand.

1. Introduction

In general, revenue management, which means demand management by price or product availability through demand-based models, was created in the 1970s using IT-based profit/revenue maximization methods after the deregulation plan was implemented in the United States of America airline market. After being invented in the airline industry, the revenue management concept spread quickly in many other domains such as transportation (train, ship, freight, car rental, etc.), the hoteling/hospitality industry (tourism services), advertisement (especially audio-visual), and so forth. In the last 40 years,

* Corresponding author
E-mail: hadi_sahebi@iust.ac.ir (H. Sahebi)
ISSN 1923-2934 (Online) - ISSN 1923-2926 (Print)
2023 Growing Science Ltd.
doi: 10.5267/j.ijiec.2022.12.003

research on this field has been numerous; Chiang et al. (2007) and McGill and Van Ryzin (1999) are among the most cited, Phillips (2021) and Ruan et al. (2021) are among the most important published books. Revenue management is not only considered in strategic level decisions of airlines but also plays a crucial role in various operational issues of airlines. In this research, we intend to examine the operational issues of airlines, such as crew scheduling, maintenance and routing problems, from the point of view of the revenue management approach, as well as the disruption occurrence. The airline industry involves high operational costs, variable demands, heavy traffic and strict rules and regulations. The Airline Liberalization Law, approved in the early 1980s, made this industry face heavily competitive environments (Eltoukhy et al., 2017), wherein airlines had to manage their resources (crew-aircraft network planning) efficiently. This required solving the airline network design and scheduling problem while satisfying a large number of related constraints and regulations. The airline network design and scheduling problem can be divided into four main sub-problems as shown in Fig. 1 which are network planning, schedule planning, aircraft planning, and crew planning; as the latter is quite complicated, it is usually divided into crew pairing and crew rostering/assignment (Abdelghany & Abdelghany, 2018; Barnhart, Belobaba, & Odoni, 2003; Etschmaier & Mathaisel, 1985).

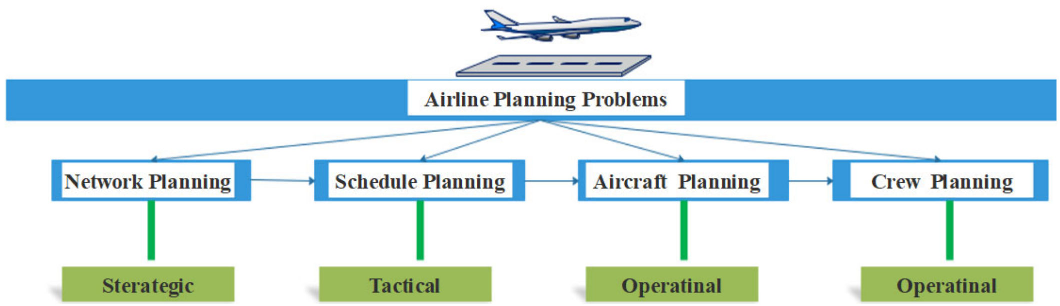


Fig. 1. Airline planning problems

All airline planning begins with network design and flight scheduling (Barnhart et al., 2009; Bazargan, 2016), where the latter is a. flight-network-based timetable that shows which city and at what time the flight should be made. Flight-service-provision decisions of an airline depend mainly on the market demand forecasts, available aircraft performance features, cockpit and crew availability, existing laws, and strategies of other airlines. This section is aimed to use the results found from the network design and flight scheduling, such as the number and type of the fleet and the flight schedule specified at the strategic level, in the optimal fleet allocation and assign them to flights or other purposes with the lowest possible costs (Barnhart et al., 2009; Gao et al., 2009; Sa et al., 2020), and determine, meanwhile or after fleet assigning, their routing based on the related laws and constraints. This stage, which forms part of the model presented in this research, requires such input information as the number and type of aircrafts, the operating costs of each, the flight network and the maintenance routing rules. Solving this sub-problem determines which aircraft should serve which flight chain so as to minimize the aircraft operating costs or achieve the company's other objectives considering the related maintenance rules (Al-Thani et al., 2016; Başdere & Bilge, 2014; Gopalan & Talluri, 1998; Safaei & Jardine, 2018; Sarac et al., 2006; Talluri, 1998). The final airline sub-problem issue is scheduling which many researchers have considered due to its high share in the operational costs (Antunes et al., 2019; Barnhart et al., 2003; Kasirzadeh et al., 2017; Schaefer et al., 2005).

Operations Research (OR) models and techniques have greatly affected airline planning, and advances in computer technology and optimization models have helped airlines find optimal/near-optimal solutions for each of the mentioned problems and improve the efficiency of their operations. The widespread role of these models has caused many airlines to save millions of dollars by establishing OR units (Barnhart, Belobaba, et al., 2003), which finally formed the prestigious "Airline Group of the International Federation of Operational Research Societies (AGIFORS)" professional association that tries to advance, develop and apply OR in the aviation industry. Although the solution of each of the four mentioned airline planning sub-problems will reduce costs and enhance the efficiency of an airline, new studies usually require two or more sub-problems to be solved together to achieve a global optimization (Cacchiani & Salazar-González, 2017; Gao et al., 2009; Papadakos, 2009). Among hierarchical solutions, integrated approaches not only avoid infeasible solutions, but also lead the airline towards a global optimal solution. Optimal network design and flight scheduling highly affect the airline revenue management because proper tactical crew scheduling, fleet allocation, and aircraft routing decisions maximize the revenue through minimized total costs, one of which is the fuel cost - the highest in an airline. As the crew, after fuel, imposes the highest cost on an airline (Bazargan, 2016), its optimal planning too can highly reduce the total cost; hence, an optimal integrated crew-aircraft schedule planning will result in the least total flight scheduling costs. This important issue has motivated many researchers to address airline planning sub-problems.

The main contributions of the present study are briefly as follows. If an airline company has designed its flight schedule at a strategic level, this research focuses on the integrated CS-AMR problem using a revenue management approach so as to reduce costs as well as control the revenue obtained from the optimal utilization of the fleet capacity and stand-by aircrafts to maximize the profit. To this end, the current research addresses the optimization of integrated CS-AMR decisions for an airline company with multiple fleets and crew bases. In this problem, we consider the disruption scenarios in the initial schedule (initial timetable) as well as flight delays, and focus on the integrated CS-AMR problem considering probable disruption scenarios using a revenue management approach. A two-stage robust scenario-based optimization model is proposed, the first stage of which yields an integrated plan for the fleet and flight crew, and the second stage of which makes decisions such as flight cancellation, delays, substitutions and so on. This research has been so organized as to review the literature and present the research gap in Section 2, fully explain the problem and its details in Section 3, present an adjustable, scenario-based, robust optimization model to solve the problem, in Section 4, use a numerical case study to evaluate the model and its outputs, as regards robustness, in Section 5 and, finally, summarize the results in Section 6.

2. Literature Review

As mentioned before, since this research addresses the integrated crew scheduling and aircraft maintenance routing problem, the effort is made next to review and explain some related innovations of the current studies, to which the reader is asked to refer (Abdelghany & Abdelghany, 2018; Barnhart, Belobaba, et al., 2003; Barnhart, Cohn, et al., 2003; Eltoukhy et al., 2018; Eltoukhy et al., 2018; Etschmaier & Mathaisel, 1985) for a thorough flight-scheduling review. Among several types of research done so far on aircraft routing and crew scheduling, that of (Cordeau et al., 2001) integrated both issues and presented a model that used the Benders decomposition approach to make simultaneously related decisions aimed to minimize the time needed, after landing, by the crew to prepare the aircraft for the next same-aircraft flight. Weide et al. (2010) studied aircraft routing and crew scheduling problems together to present a model considering such constraints as the “crew return to the first base”, and the “minimum time required to make two sequential flights”, and solved it using an iterative heuristic algorithm. (Sandhu & Klabjan, 2007) presented a model that integrated the decisions on aircraft routing, fleet allocation and crew scheduling considering a limited number of aircraft and ignoring maintenance-related issues. To solve the model, they used two approaches based on Benders decomposition and augmented Lagrange along with the column generation algorithm. Papadakos (2009) studied aircraft routing, fleet allocation and crew scheduling problems, selected a complete model after examining all those proposed in this field and presented a novel algorithm for its solution based on the Benders analysis and column generation. He omitted the constraint on the number of available crafts, included it in the model and showed that the mentioned constraint could not ensure the feasible routes. (Salazar-González, 2014) developed and implemented it to solve real aircraft routing and crew allocation/scheduling problems and checked all the mentioned items for only a one-day flight. The difference between this and previous approaches was that their model did not consider any maintenance constraints on the number of aircraft during the day.

Among researchers that have recently addressed the revenue management and pricing issues, (An, Mikhaylov, & Jung, 2021) presented a linear programming approach for the mathematical modelling of a multi-fare robust revenue management in a flight network in the airline industry aiming mainly at optimizing the limited reservation policy using limited demand data. Kumar et al. (2021) proposed competitive revenue management models that focus on fully flexible and loyal customers. Stating that almost no earlier revenue management model has explicitly considered the competitive effects, and nearly all have assumed that an airline demand depends only on its fare, they point out the innovation of their work, which considers both types of customer behavior to design a model that yields more realistic competitive dynamics. Various factors such as market competition, leader-follower-based power structure, and competitiveness highly affect the fare rates and customer behavior. Machine learning-based methods can consider competitors' plans, predict the market size, estimate market share in comprehensive pricing, and determine optimal price control policy (Sahebi et al., 2022), thus play an essential role in airline revenue management. Considering the continuous demand management, (Vinod, 2021) used the adaptive robust optimization approach to solve the airline operational revenue management problem under critical COVID-19 conditions, and proposed an alternative method that changed the main revenue management criteria under the pandemic and modified the available traditional seat management programs by adjusting the supply and demand levers.

The multi-vehicle and multi-compartment inventory routing problem with stochastic demands is a fuel delivery-related issue studied by (Li & Jiao, 2022), who modelled the maximum-to-level replenishment policy problem as a two-stage stochastic programming aiming at minimizing the total cost. This model makes the inventory management and routing decisions in the first stage and implements the related resource actions in the second stage. In studying the vehicle routing problem with hard time windows, Nasri et al. (2020) tried to include two sources of uncertainties, travel and service time, both of which can be due to different reasons. In one recent research on crew pairing, mathematical optimization was used in a budget airline (Chutima & Krisanaphan, 2022). Besides cost, a focal element in conventional cockpit crew pairing, the mentioned study has many other previously ignored factors and proposed an adaptive non-dominated sorting differential algorithm for cockpit crew pairing optimization.

In a present-research-related study that did not consider fleet diversity, crew base and maintenance, Díaz-Ramírez, Huertas, & Trigos (2014) considered one fleet type and maintenance base to present an aircraft routing and crew planning model, where the crew and maintenance had one place, and solved it by the Benders decomposition approach. Using decomposition-based approaches that integrate airline decisions, Shao et al. (2017) presented a mathematical model for fleet allocation, aircraft routing, and crew scheduling integration problems considering travel demand, and used Benders decomposition approach along with several fast solution-yielding strategies to solve it. In presenting a heuristic two-stage algorithm to address the aircraft-crew rescheduling problem, some important management issues considered by Zhang and Mahadevan (2017) in their research were the delayed flights, airport capacity for aircraft reception and rest time for each crew and aircraft. In a research aimed at presenting a new integrated approach for crew pairing, aircraft routing and fleet allocation sub-problems, Cacchiani and Salazar-González (2013, 2020) presented their mathematical model based on such assumptions as “no during-night flights” and “crew return to the base when flight begins”, and used the column generation algorithm for its solution. Jalili and Manteghi (2018) used a multiple linear regression model to predict the Iranian airlines’ passenger-cargo demands; Mirabi et al. (2019) presented a multi-objective mathematical model to solve locating-allocating hub problems in the Iranian airports and suggested using differential evolution metaheuristic algorithms along with the Ɛ-constraint method for this purpose because the problem was NP-hard. In designing a numerical example for their research, they assumed that the aircraft and flight crews were equal in number.

Table 1 summarizes some of the most recent research and compares the features of this research with them as regards: *i*) aircraft maintenance and routing, *ii*) crew scheduling, *iii*) revenue management consideration, *iv*) uncertainty and disruption, *v*) robustness, and *vi*) model and methodology.

Table 1
Comparing the present study with recent studies

Research	Aircraft Decision		Crew Scheduling	Revenue Management Consideration	Uncertainty and Disruption	Robustness	Model & Method
	TAR	MP					
(Shao et al., 2017)	✓	✓	✓				BD
(Ahmed, Ghroubi, Haouari, & Sherali, 2017)	✓	✓			✓		MILP
(Safaei & Jardine, 2018)	✓	✓					MILP
(Eltoukhy, Wang, et al., 2018)	✓	✓					MILP
(Ahmed, Mansour, & Haouari, 2018)	✓		✓			✓	Nonlinear Programming
(Eltoukhy, Chan, et al., 2018)	✓	✓					MILP
(Haouari, Zeghal Mansour, & Sherali, 2019)			✓				MILP, RTL
(Antunes et al., 2019)			✓			✓	MILP
(Wen, Ma, Chung, & Khan, 2020)			✓		✓	✓	Column Generation
(Parmentier & Meunier, 2020)	✓	✓	✓				Column Generation
(Sanchez, Boyacı, & Zografos, 2020)	✓	✓					MILP
(Bulbul & Kasimbeyli, 2021)	✓	✓					MILP
(Ruan et al., 2021)	✓	✓					MILP
This Paper	✓	✓	✓	✓	✓	✓	TSRSP ALNS

TAR: Tail Assignment and Routing; MR: Maintenance Planning; BD: Benders Decomposition; MILP: Mixed Integer Linear Programming; Reformulation-Linearization Technique (RLT)

According to the published papers, research gaps in CS-AMR problem are as follows:

- Maintenance routing decisions are made sequentially along with the crew itinerary, and integration is paid less attention, but this research integrates them.
- Most of the mentioned studies have considered one center for both maintenance and crew, but this research has considered diversity for them.
- Disruption scenarios are rare in the initial schedule for network of flight, but this research considers them with different severities and uses a novel two-stage robust scenario-based optimization approach for decision robustness and its adjustability.
- Literature on the CS-AMR problem has no main consideration on applying a revenue management approach, and major part of them only deals with a classical model with a cost minimization objective function.
- Developing a computationally efficient metaheuristic algorithm to tackle the problem complexity which solves the integrated CS-AMR problem using large neighborhood search in an adaptive version.

3. Problem Description

This research has addressed the integrated crew scheduling and aircraft maintenance routing (CS-AMR) problem under disruption. The crew usually operates from its designated base, where it begins and ends its service, and its mission involves a consecutive series of flights performed in one day. Landing time is when the crew rests between two flights during its duty, while the rest time is the long break between two crew duties, which is usually directly related to its overnight stay. Each airline crew pairing is defined with the duties alternating successions and rest periods, an example of which is shown in Fig. 2 with two duties and six flights for the crew.

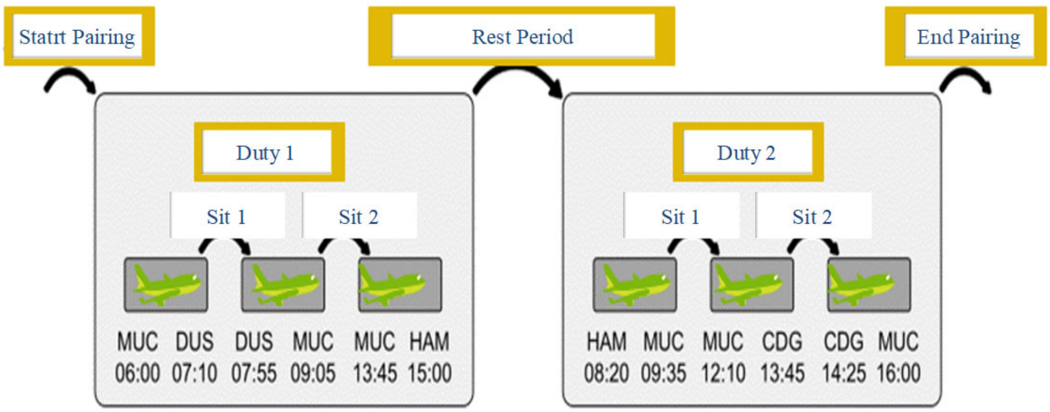


Fig. 2. Structure of an airline crew pairing (Dück, 2010)

The desired CS-AMR problem is considered for airlines with diverse crew and aircraft maintenance bases. The network design and flight schedules are assumed to be pre-specified and airline companies try to make, within a short time horizon, related decisions according to the laws and regulations applied to aircrafts and the crew itinerary so that they not only cover all the scheduled flights, but also minimize the total cost. Operating costs include that of using an aircraft (budget, fuel, etc.) and that of the crew plus that of the aircraft maintenance in the flight route; if a flight is cancelled or delayed, costs related to lost sales and penalties are included too.

Governing regulations and some airlines policies require that crew are not to fly more than the maximum time considered in the planning horizon, which is independent of the crew type, and another rule requires that all aircrafts can, regardless of their type, fly for a certain time without being subjected to maintenance operations. As mentioned before, the CS-AMR problem of the current research considers probable disruptions (climatic, technical, etc.) that cause flight delays and affect the initial schedule. One normal situation and five different flight disruption scenarios considered here are as follows:

● Scenario 0: During a specified short-term time, horizon, no disruption occurs in the initial schedule.

● Scenario 1: A very minor disruption in the initial schedule will cause flight delays of less than 10 min.

● Scenario 2: A minor disruption in the initial schedule will cause flight delays between 10 and 20 min.

● Scenario 3: A normal disruption in the initial schedule will cause flight delays of about 30 min.

● Scenario 4: A severe disruption in the initial schedule will cause flight delays of about 60 min (1h).

● Scenario 5: A very severe disruption in the initial schedule will cause flight delays of more than 1h.

Now, the desired problem of this research should be solved in such a way (aircraft maintenance routing and crew scheduling decisions should be made so) that the performance is optimal under different scenarios. Assumptions made to model and solved the problem are as follows:

• The scheduled initial timetable is the input.

• Fleet size and crew and their initial schedule including locations (maintenance and crew bases) are known.

• Each aircraft route and maintenance location during different flight legs must be determined before all scenarios.

• The Schedule of duties for each crew must be determined before all scenarios.

• Under disruptions, flight cancellation/delay is possible but incurs cost.

• In each fleet, it is assumed that all the crew members are qualified and can fly.

• When the scheduling horizon ends, each aircraft and crew must be at their initial base (because this short-term plan is assumed to repeat periodically).

4. The proposed robust optimization model

This section presents a two-stage robust optimization model for the CS-AMR problem defined in the previous section. Since it is assumed that the probable disruptions can occur under different random scenarios with known occurrence probabilities, use is made of the scenario-based robust optimization approach to control the uncertainties and the model robustness. To this end, the next section first explains this approach and then proposes the problem model based on it.

4.1. The two-stage robust optimization approach

If a problem involves some uncertain parameters in a random scenario-based manner, the robust scenario-based stochastic programming (SSP) approach can control uncertainties and ensure optimality. In general, a two-stage SSP approach is expressed as follows:

$$\begin{cases} \mathbf{max}\ \mathrm{E(Z)} = \sum_{s \in S} \pi_s . z_s \\ z_s = c_s^T . x_s + d_s^T y \quad \forall\, s \in S \\ A_s x_s + K_s y = b_s \quad \forall\, s \in S \\ Ry = q \\ y \in Y, x_s \geq 0 \end{cases} \tag{1}$$

where z_s is the objective value under scenario $s \in S$, x_s are scenario-dependent variables (usually continuous and non-negative), y are scenario-independent variables (usually binary), c_s, d_s, A_s, K_s and $b_s$are problem parameters under scenario $s \in S$, (R,q) is the deterministic parameters and π_s is the occurrence probability of scenario $s \in S$ (Salehi, Mahootchi, & Husseini, 2019; Yu & Li, 2000). In the SSP approach, the expected value of objectives is used in different probable scenarios, all of which must be feasible, and variables y^* and $x_s{}^*$ are feasible solution satisfying almost scenarios $s \in S$ so that the expected values of objective function be maximized. In developing the SSP approach, (Mulvey, Vanderbei, & Zenios, 1995) considered the sum of the model expected performance and standard deviation as the solution robustness or the optimality robustness and developed their "two-stage robust scenario-based optimization (TSRSO)" model as follows:

$$\begin{cases} \mathbf{max}\ \mathrm{E(Z)} - \mathrm{V(Z)} - \mathrm{Penalty} = \sum_{s \in S} \pi_s . z_s - \lambda \sum_{s \in S} \pi_s . \left(z_s - \sum_{s' \in S} \pi_{s'} . z_{s'} \right)^2 - \omega \sum_{s \in S} \pi_s . \xi_s^2 \\ z_s = c_s^T . x_s + d_s^T y \quad \forall\, s \in S \\ A_s x_s + K_s y = b_s + \xi_s \quad \forall\, s \in S \\ Ry = q \\ y \in Y, x_s \geq 0 \end{cases} \tag{2}$$

where ξ_s is the flexibility variable in scenario $s \in S$, ω is the model robustness coefficient and λ is the solution variance importance coefficient in different scenarios (other symbols were explained in the SSP model). Yu and Li (2000) replaced the second order function with the absolute value function in Mulvey's proposed RSSP model, which is a quadratic optimization problem as follows:

$$\begin{cases} \mathbf{max} \sum_{s \in S} \pi_s . z_s - \lambda \sum_{s \in S} \pi_s . \left| \left(z_s - \sum_{s' \in S} \pi_{s'} . z_{s'} \right) \right| - \omega \sum_{s \in S} \pi_s |\xi_s| \\ z_s = c_s^T . x_s + d_s^T y \quad \forall\, s \in S \\ A_s x_s + K_s y = b_s + \xi_s \quad \forall\, s \in S \\ Ry = q \\ y \in Y, x_s \geq 0 \end{cases} \tag{3}$$

and developed its linear mode as follows:

$$\begin{cases} \mathbf{max} \sum_{s\in S} pr_s . z_s - \lambda \sum_{s\in S} \pi_s . \left(z_s - \sum_{s'\in S} \pi_{s'} . z_{s'} + 2\theta_s \right) - \omega \sum_{s\in S} \pi_s (\xi_s^+ + \xi_s^-) \\ z_s - \sum_{s'\in S} \pi_{s'} . z_{s'} + \theta_s \geq 0 \ \forall \ s \in S \\ z_s = c_s^T . x_s + d_s^T y \ \ \forall \ s \in S \\ A_s x_s + K_s y = b_s + (\xi_s^+ - \xi_s^-) \ \ \forall \ s \in S \\ Ry = q \\ y \in Y, x_s \geq 0 \\ \xi_s^+, \xi_s^-, \theta_s \geq 0 \end{cases} \tag{4}$$

where $\theta_s \geq 0$ is an auxiliary linearization variable that calculates the deviation from the mean in each scenario. This research has used the mentioned TSRSO approach to model the CS-AMR problem.

4.2. Proposed TSRSO model for CS-AMR problem

To use the TSRSO approach in the desired research problem, the CS-AMR problem is assumed to be done in the first stage, and flights be cancelled in the second stage, considering that disruption scenarios are also probable; obviously, the actual flight commencement time and the delay in the arrival of each flight are the scenario-dependent variables. In the TSRSO approach, the problem is so modelled that the expected performance and the standard deviation of the objective function are also controlled under different scenarios.

● Nomenclature

	Sets and indices
F	Set of flights (indexed with *f*)
$f = 0$	A dummy flight (with 0 start time and 0 duration)
$F^0 = F \cup \{0\}$	Set of flights and dummy flight
N	Set of starting/ending cities (nodes) of each flight (indexed with *n*)
$FI_n \subseteq F$	Set of flights entering node *n*
$FO_n \subseteq F$	Set of flights leaving node *n*
C	Set of flight crew
$M \subseteq N$	Set of cities equipped with aircraft maintenance bases
$\Xi = \{0,1,2,\xi,\dots,\lvert\Xi\rvert = 5\}$	Set of disruption scenarios (indexed by ξ); ξ = 0 means no disruption during the desired scheduling horizon
	Parameters
dt_f	Flight *f* departure time based on the initial schedule
at_f	Flight *f* arrival time based on the initial schedule
lt_f	Flight *f* normal time period/length
o_{fn}	1, if flight *f* origin is node n, otherwise 0
d_{fn}	1, if flight *f* destination is node *n*, otherwise 0
r_f	Time required to prepare the aircraft before it starts its flight
fc_{fac}	Cost of flight *f* with aircraft *a* and crew *c*
cc_f	Flight *f* cancellation cost
pc_f	Penalty per unit excess delay to start flight *f*
sbc	Cost of each stand-by aircraft for using in disruption scenarios
$MaxS$	Maximum fleet capacity which can be considered as stand-by aircrafts
mc_a	Per unit maintenance cost of aircraft *a*
md_f	Maximum delay allowed in flight *f*
IC_{nc}	Binary matrix of the initial schedule for crew in different cities (1, if crew *c* is at node *n* when scheduling horizon begins)
IA_{na}	Binary matrix of initial schedule for aircraft in different cities (1, if aircraft *a* is at node *n* when scheduling horizon begins)
$\mathcal{MC}$	Maximum flight time of each crew during the desired time horizon
$\mathcal{MA}$	Maximum flight time of each aircraft during the desired time horizon
$\mathcal{MU}$	Maximum time each aircraft is allowed to fly without maintenance before it begins a new flight
$\mathcal{L}_{f\xi}$	Delay period before flight *f* begins under disruption scenario ξ,
π_ξ	Occurrence probability of scenario ξ

$Big\ M$	A large positive number
	Decision variables
NSA	Number of aircraft for total fleet capacity which are considered as stand-by for disruption scenarios
x_{fa}	1, if flight *f* is done by aircraft *a*, otherwise 0
y_{fc}	1, if flight *f* is in crew *c* itinerary, otherwise 0
$sub_{f\xi}$	1, if flight *f* is done by substituted aircraft *a*, otherwise 0
$g_{af'f}$	1, if flight *f'* is before flight *f* (in flight route of aircraft *a*), otherwise 0
$v_{af'f}$	1, if aircraft *a* is under maintenance operation at a specific node (from a maintenance base) between flights *f'* and *f*, otherwise 0
BC_{fc}	1, if crew *c* is available at the flight origin before flight *f* begins, otherwise 0
BA_{fa}	1, if aircraft *a* is available at the flight origin before flight *f* begins, otherwise 0
$e_{f\xi}$	1, if flight *f* is cancelled under disruption scenario ξ, otherwise 0
$st_{f\xi}$	Flight *f* commencement time under disruption scenario ξ,
$del_{f\xi}$	Delay period when flight *f* begins under disruption scenario ξ,
$cost_{\xi}$	Total cost under disruption scenario ξ,

The model proposed in this research is as follows:

$$\begin{cases} \mathbf{max} \sum_{\xi} \pi_{\xi}\, profit_{\xi} - \lambda \sum_{\xi} \pi_{\xi} \left(2\theta_{\xi} - \left(profit_{\xi} - \sum_{\xi'} \pi_{\xi'}\, profit_{\xi'} \right) \right) \\ profit_{\xi} = revenu_{\xi} - cost_{\xi} \\ revenu_{\xi} = \sum_{f} (1 - e_{f\xi}) rev_f + \sum_{f} sub_{f\xi} rev_f + \\ cost_{\xi} = \sum_{f}\sum_{a}\sum_{c} fc_{fac} \frac{(x_{fa} + y_{fc})}{2} + \sum_{a}\sum_{f}\sum_{f'} mc_a v_{af'f} + \sum_{f} cc_f\, e_{f\xi} + \sum_{f} pc_f del_{f\xi} + sbc.NSA \end{cases} \tag{5}$$

$$\theta_{\xi} \geq profit_{\xi} - \sum_{\xi'} \pi_{\xi'}\, profit_{\xi'} \quad \forall\, \xi \tag{6}$$

$$\sum_{a} x_{fa} = 1 - e_{f\xi} \quad \forall\, f, \xi \tag{7}$$

$$\sum_{c} y_{fc} = 1 - e_{f\xi} \quad \forall\, f, \xi \tag{8}$$

$$x_{fa} = \sum_{f' \in F^0} g_{af'f} \quad \forall\, f, a \tag{9}$$

$$g_{af'f} + g_{aff'} = 1 \quad \forall\, f, f', a \tag{10}$$

$$g_{af'f} + g_{af''f'} \leq g_{af''f} \quad \forall\, f, f', a \tag{11}$$

$$y_{fc} \leq BC_{fc} \quad \forall\, f, c \tag{12}$$

$$BC_{fc} = \sum_{n} IC_{nc} o_{fn} + \sum_{n}\sum_{a}\sum_{f' \in FI_n} o_{fn} y_{f'c} g_{af'f} - \sum_{n}\sum_{a}\sum_{f' \in FO_n} d_{f'n} y_{f'c} g_{aff'} \quad \forall\, f, c \tag{13}$$

$$x_{fa} \leq BA_{fa} \quad \forall\, f, a \tag{14}$$

$$BA_{fa} = \sum_{n} IA_{na} o_{fn} + \sum_{n}\sum_{a}\sum_{f' \in FI_n} o_{fn} g_{af'f} - \sum_{n}\sum_{a}\sum_{f' \in FO_n} d_{f'n} g_{aff'} \quad \forall\, f, a \tag{15}$$

$$st_{f\xi} \geq \sum_{f'} \left(lt_{f'} + r_{f'} + \mathcal{L}_{f'\xi}\right) g_{af'f} + \left(r_{f'} + \mathcal{L}_{f\xi}\right) x_{fa} \quad \forall\, f, a, \xi \tag{16}$$

$$del_{f\xi} \geq st_{f\xi} + lt_f - at_f - \; bigMe_{f\xi} \;\; \forall\, f, \xi \tag{17}$$

$$del_{f\xi} \leq md_f + bigMe_{f\xi} \quad \forall\, f, \xi \tag{18}$$

$$\sum_{f} lt_f y_{fc} \leq \; \mathcal{MU} \quad \forall\, c \tag{19}$$

$$\sum_{f} lt_f x_{fa} \leq \; \mathcal{MA} \quad \forall\, a \tag{20}$$

$$\sum_{f''} lt_{f''} g_{af'f''} - \sum_{f''} lt_{f''} g_{aff''} \leq \; \mathcal{MU} + bigM \quad \forall\, f, f', a \tag{21}$$

$$v_{af'f} \leq \sum_{n\in M} \sum_{f''} o_{f''n} g_{af'f''} - \sum_{n\in M} \sum_{f''} o_{f''n} g_{aff''} \quad \forall\, f, f', a \tag{22}$$

$$sub_{f\xi} \leq \; e_{f\xi} \quad \forall\, f\,, \xi \tag{23}$$

$$\sum_{f} sub_{f\xi} \leq \; NSA \quad \forall\, \xi \tag{24}$$

$$NSA \leq \; MaxS \tag{25}$$

$$\begin{cases} if \;\; \sum_{f'} g_{aff'} = 0 \;\; then\, IA_{na} = d_{fn} \quad \forall\, f, a, n \\ \qquad\qquad \Downarrow \text{Linear} \\ A_{na} - d_{fn} \leq bigM \sum_{f'} g_{aff'} \quad \forall\, f, a, n \end{cases} \tag{26}$$

$$\begin{cases} if \;\; \sum_{a} \sum_{f'} g_{aff'} = 0 \; and \; y_{fc} = 1 \;\; then\, IC_{nc} = d_{fn} \quad \forall\, f, c, n \\ \qquad\qquad \Downarrow \text{Linear} \\ IC_{nc} - d_{fn} \leq bigM \left(1 - y_{fc} + \sum_{f'} g_{aff'}\right) \;\forall\, f, a, n \end{cases} \tag{27}$$

$$\begin{cases} x_{fa}\,, y_{fc}, g_{af'f}, v_{af'f}, BC_{fc}, BA_{fa}, e_{f\xi}, sub_{f\xi} \in \{0{,}1\} \\ st_{f\xi}, del_{f\xi}, cost_{\xi}, \theta_{\xi} \geq 0 \\ NSA \in \mathbb{Z}_{+} \end{cases} \tag{28}$$

Eq. (5) indicates the objective function by maximizing expected profit and minimizing the standard deviation. Profit is calculated as the difference between cost and revenue ($profit_{\xi} = revenu_{\xi} - cost_{\xi}$). We should note that cost includes those of the aircraft and crew in each flight, maintenance, cancelled flights and flight delays. Eq. (6) calculates the standard deviation in each disruption scenario. Eqs. (7-25) are the problem constraints explained below; Eq. (7) and Eq. (8) guarantee that each non-cancelled flight needs one aircraft and one crew, respectively. Eq. (9) presents that an aircraft should have already been executed before each flight. Eq. (10) and Eq. (11) guarantee the ordering and transitive relation in the sequence of each aircraft route. Eq. (12) states that assigning a flight to a crew duty requires that the crew be ready in the origin city. Eq. (13) shows how a crew is ready in the origin city of a flight. Eq. (14) and Eq. (15) verify that one aircraft exists in the origin city of the flight, which is required for a flight to be executed. Eq. (16) determines each flight start time under each disruption scenario. Eq. (17) calculates the delay in each flight under different scenarios, while Eq. (18) ensures that a flight delay is not to exceed a certain amount; otherwise, it will be cancelled. Eq. (19) and Eq. (20) show the maximum duty time of a crew and an aircraft during the planning time horizon, respectively. Eq. (21) ensures that the total time between two different flights cannot exceed a certain limit provided for maintenance operation. Eq. (22) ensures that aircraft maintenance is possible only in cities with related bases. Eq. (23) enforces that flight substitutions require cancellation. In other words, if a flight is not cancelled, the substitution is not allowed. Eq. (24) shows that the number of all flight substitutions cannot exceed the number of stand-by aircrafts. Eq. (25) guarantees that a specified upper bound limits the number of aircraft which can be stand-by. Eq. (26) ensures that when an aircraft ends its pairing according to the desired time horizon, it should be present in the initial schedule for the origin city. In the same way, Eq. (27) guarantees that the flight crew should be rotated to the origin city of the initial schedule; these two conditions are represented in the linear formulation to reduce the model complexity and keep its tractability. Finally, Eq. (28) shows the domain of decision variables.

The proposed TSRSO for the CS-AMR problem, is a mixed linear programming model which can be efficiently solved by a CPLEX solver.

5. The Proposed Computationally Efficient Solution Method

In the previous section, the TSRSO model is proposed to solve the desired CS-AMR problem; however, due to the TSRSO formulation is MILP, which involves binary variables related to the assigning the flights to aircraft and crew, routing and flight scheduling, categorized as NP-hard problems (Lenstra & Rinnooy Kan, 1978; Mnich & van Bevern, 2018). Although the proposed model is efficient and can be solved in the small and medium-sized instances, it is not tractable for large-scale instances, and requires high time and cost to be solved by MILP solvers such as CPLEX. Thus, this paper presents a solution method which is more computationally efficient. The proposed method is based on an adaptive version of large neighborhood searching as a common local search metaheuristic for complex optimization problems. In the following, we first explain the general mechanism of the proposed algorithm, and then we apply it to solve the CS-AMR problem.

5-1- Adaptive Large Neighborhood Search

In the combinatorial optimization and related problems such as the scheduling and routing, several researches have used the Adaptive Large Neighborhood Search (ALNS) algorithm (Abreu & Nagano, 2022; Chen et al., 2021; Schaap et al., 2022), which is a improved version of Large Neighborhood Search (LNS) algorithm first introduced in (Ropke & Pisinger, 2006). Local search (LS) algorithms are improving metaheuristic algorithms that usually start with an initial solution, improve it by searching among the close neighboring ones, and repeat the process until no improvement occurs in the solution. In the developed local search metaheuristic algorithms such as the simulated annealing (SA), if the neighborhood is capable of searching for a very large number of new solutions in each iteration, they are considered as very large-scale neighborhood algorithms (VLSN), which involve different techniques. In fact, LNS algorithm, too, follows the neighborhood search procedure to improve the initial solution, but the difference is that it provides this ability by two operators, Destroy ($\mathcal{D}$), and Repair ($\mathscr{R}$) to enable a large number of neighborhood searches in different iterations.

The LNS algorithm generally starts with an initial feasible solution, destroys part of it with operator $\mathcal{D}$, repairs this pert with new values with operator $\mathscr{R}$ and creates a new neighbor solution in different iterations. In the LNS algorithm, one main parameter is the destruction rate $q \in (0,1)$ that can vary in the model either dynamically and randomly or be predetermined; classically, the stop condition of the LNS algorithm is either a maximum iteration or run-time limitation. In different iterations of the LNS algorithm, $x' = \mathscr{R}(\mathcal{D}(x))$ is created as a new neighborhood solution after the destroy-repair procedure, and can in, various ways, replace the current solution x. The acceptance of the x' substitution for x is shown with $\mathcal{A}(x', x)$; its value is *0* if replacement is denied and is *1* if it is accepted. The fitness function $\mathscr{F}(.)$ explains different acceptance states and determines the value of the fitness of each solution; the higher is this value, the better is the solution.

The acceptance operator $\mathcal{A}$ works based on the "*Greedy Acceptance*" rule, which means x' can replace x only if it is improved, in other words:

$$\textbf{Gready Acceptance}: \mathcal{A}(x', x) = \begin{cases} 1, & \mathscr{F}(x') \geq \mathscr{F}(x) \\ 0, & o.w. \end{cases} \tag{29}$$

More efficiently, $\mathcal{A}$ works based on the Boltzmann's proposed probabilistic acceptance rule whereby x' replaces x when it either improves it or is accepted by the Boltzmann's probability, in other words:

$$\textbf{SA Boltzmann Acceptance}: \mathcal{A}(x', x) = \begin{cases} 1, & \mathscr{F}(x') \geq \mathscr{F}(x) \\ 1, & \bigwedge\left(\mathscr{F}(x') < \mathscr{F}(x),\ v \leq \exp\left(\dfrac{\mathscr{F}(x') - \mathscr{F}(x)}{T}\right)\right) \\ 0, & \bigwedge\left(\mathscr{F}(x') < \mathscr{F}(x),\ v > \exp\left(\dfrac{\mathscr{F}(x') - \mathscr{F}(x)}{T}\right)\right) \end{cases} \tag{30}$$

Here, if the first condition is not met, $v \in (0,1)$ is generated to check the possible acceptability. In the SA algorithm, T shows the temperature; it is first T_0, and then the temperature is reduced with an annealing rate α in iteration k as $T = \alpha^k T_0$.

In NS algorithms, $\mathcal{D}$ and $\mathscr{R}$ operators that, respectively, destroy the current solution and then repair it, are quite important and influential in the algorithm performance. In general, the destroy-repair mechanisms include the random removal (*RR*), worst removal (*WR*), random insertion (*RI*) and best insertion (*BI*). In the ALNS, the main idea is the concurrent use of several destroy-repair operators to adapt the search process dynamically and automatically to the best ones for more intelligent and efficient searches. This makes the ALNS algorithm more adaptable, causes the search to adapt to the best destroy-repair operations, enables more efficient, intelligent searches and ensures optimal/near optimal solutions.

In the ALNS, the destroy-repair operators are first defined with different strategies and then assigned equal weights $\omega_{\mathcal{D}}^{-}$ and $\omega_{\mathscr{R}}^{+}$. Next, one strategy is selected for $\mathcal{D}$ and one for $\mathscr{R}$ by the roulette wheel (*RW*) mechanism and, after applying the combined operator $\mathscr{R}(\mathcal{D}(.))$, the $x' = \mathscr{R}(\mathcal{D}(x))$ neighboring solution is generated; the selected operators are given fitness based on the optimality assessment and their values are calculated as follows:

$$\Psi' = \begin{cases} \Psi, & \mathscr{F}(x') \geq \mathscr{F}(X_{Best}) \\ \dfrac{\Psi}{2}, & \mathscr{F}(x') \geq \mathscr{F}(x) \text{ AND } \mathscr{F}(x') < \mathscr{F}(X_{Best}) \\ \dfrac{\Psi}{4}, & \mathcal{A}(x',x) = 1 \text{ AND } \mathscr{F}(x') < \mathscr{F}(x) \\ 0, & \mathcal{A}(x',x) = 0 \end{cases} \tag{31}$$

where $\Psi \geq 0$. Next, the weight and probability of the selection are updated as follows:

$$\omega_{\mathscr{R}}^{+} \leftarrow \lambda\omega_{\mathscr{R}}^{+} + (1-\lambda)\Psi' \tag{32}$$

$$\omega_{\mathscr{R}}^{+} \leftarrow \lambda\omega_{\mathscr{R}}^{+} + (1-\lambda)\Psi' \tag{33}$$

$$\pi_{\mathcal{D}} = \frac{\omega_{\mathcal{D}}^{-}}{\sum_{\mathcal{D}\in \mathcal{S}^{-}} \omega_{\mathcal{D}}^{-}} \tag{34}$$

$$\pi_{\mathscr{R}} = \frac{\omega_{\mathscr{R}}^{+}}{\sum_{\mathscr{R}\in \mathcal{S}^{+}} \omega_{\mathscr{R}}^{+}} \tag{35}$$

In updating the weight of the destroy-repair operators, $\lambda \in (0,1)$ is the coefficient of the initial weight decay. Here, the standards of q, λ and Ψ (main parameters of the ALNS method), as well as T_0 and α (used in the Boltzmann Acceptance Rule, which is based on such parameter tuning techniques as the *Taguchi Method*) are tuned, and the initial weights of the operators are taken equal to parameters that have equal weights ($\omega_{\mathscr{R}}^{+} = \omega_{\mathcal{D}}^{-} = 1$); Fig. 3 shows the pseudocode of proposed ALNS algorithm.

5.2. Applying ALNS for Solving the CS-AMR Problem

To apply the ALNS algorithm to solve the desired CS-AMR problem, the solution representation is first encoded and structured, then the sets of destroy-repair operators are defined to find the neighboring solutions and the solution repair mechanisms or penalties for infeasible solutions are explained. For encoding the solutions of the CS-AMR problem, it is to be noted that each solution consists of five main parts from X_1 to X_5 that relate, respectively, to assigning each flight to aircraft and crew, sequence of flights in the route of each aircraft, pairing the crew to the flight assigned to, percentage of stand-by fleet capacity, and cancelled or substituted flights in each disruption scenario. Each problem solution is defined as $X = [X_1; X_2; X_3; X_4; X_5]$ and each one is explained as follows.

In X_1, an array is defined with the length of the number of flights and two rows, wherein the first and second rows specify, respectively, which aircraft and which crew are allocated to each flight. In X_2, another array is defined with the length of the number of flights for each selected aircraft and its assigned flights, and the route of each aircraft is determined with a permutation of different flight numbers. We should note that in the structure of solution X, rows of array X_2 equal the number of the selected aircrafts and its columns equal their assigned flights. In X_3, similar to X_2, it is determined with what sequence each selected flight lies in the duty of each crew. In X_4, the simplest part of the structure of solution X, a *1-D* array is defined that represents the percent stand-by of the fleet capacity and, finally, in X_5, a matrix is defined where the rows show the disruption scenarios and columns show the flights. Arrays of this matrix are coded with numbers *1* for no flight cancellation and substitution, *2* for cancellation, and finally *3* for substitution with a stand-by aircraft.

- **Input**
  - Initial feasible solution (x)
  - ALNS parameters ($q, \lambda, \Psi, \alpha, T_0$); Tuned using Taguchi method
  - Destroy operators (Set of $\mathcal{S}^-$)
  - Repair operators (Set of $\mathcal{S}^+$)
  - Stop criterion (maximum iteration or run-time limitation)
- $X_{Best} = x$;
- **Repeat**
  - Calculate $\pi_{\mathcal{R}}$ and $\pi_{\mathcal{D}}$ for all $\mathcal{R} \in \mathcal{S}^+$ and $\mathcal{D} \in \mathcal{S}^-$
    - $\pi_{\mathcal{D}} = \frac{\omega_{\mathcal{D}}^-}{\sum_{\mathcal{D} \in \mathcal{S}^-} \omega_{\mathcal{D}}^-}$
    - $\pi_{\mathcal{R}} = \frac{\omega_{\mathcal{R}}^+}{\sum_{\mathcal{R} \in \mathcal{S}^+} \omega_{\mathcal{R}}^+}$
  - Apply RW selection mechanism to select $\mathcal{D}$ and $\mathcal{R}$ operators
  - Apply selected $\mathcal{D}$ and $\mathcal{R}$ operators to search new neighborhood solution
    - $x' = \mathcal{R}(\mathcal{D}(x))$
  - Apply Boltzmann acceptance rule
    - **If** $\mathcal{A}(x', x) = 1$ **Then**
    - $x = x'$;
    - **End If**
  - Update best solution
    - **If** $\mathcal{F}(x') > \mathcal{F}(X_{Best})$ **Then**
    - $X_{Best} = x'$;
    - **End If**
  - Update the weight of selected $\mathcal{D}$ and $\mathcal{R}$ operators
    - $\omega_{\mathcal{R}}^+ \leftarrow \lambda\omega_{\mathcal{R}}^+ + (1-\lambda)\Psi$
    - $\omega_{\mathcal{D}}^- \leftarrow \lambda\omega_{\mathcal{D}}^- + (1-\lambda)\Psi$
- **Until the stop criterion is met**
- **Return** X_{Best} and $\mathcal{F}(X_{Best})$.

Fig. 3. Pseudocode of Proposed ALNS Algorithm

Part X1 of solution encoding: Assigning flight to airplane and crew

X1	F1	F2	F3	F4	F5	F6	F7	F8	F9	F10
Airplane	1	2	1	3	3	1	2	1	2	1
Crew	3	1	2	1	2	2	3	3	1	1

Part X2 of solution encoding: Determining the sequence of flights in the route of each airplane

X2	F1	F2	F3	F4	F5	F6	F7	F8	F9	F10
A1	1	0	4	0	0	2	0	5	0	3
A2	0	1	0	0	0	0	2	0	3	0
A3	0	0	0	2	1	0	0	0	0	0

Part X3 of solution encoding: Pairing the crew to the flight assigned to

X3	F1	F2	F3	F4	F5	F6	F7	F8	F9	F10
C1	0	1	0	2	0	0	0	0	3	4
C2	0	0	1	0	2	3	0	0	0	0
C3	1	0	0	0	0	0	2	3	0	0

Part X4 of solution encoding: Determining the percentage of stand-by fleet capacity

X4	25

Part X5 Determining cancelled or substituted flights in each disruption scenario

X5	F1	F2	F3	F4	F5	F6	F7	F8	F9	F10
ξ1	1	1	1	1	1	2	1	3	1	1
ξ2	1	2	1	1	3	1	2	1	1	1
ξ3	1	1	1	1	2	1	3	1	1	1

Fig. 4. Proposed solution encoding (Structure X) to represent the solution of CS-AMR problem

Fig. 4 shows a schematic view of the numerical example of the solution structure of the desired problem of the present study with 10 flights, *4* aircrafts, *3* crews and *3* flight disruption scenarios. Part X_1 shows each flight is assigned to which aircraft and which crew; for instance, F_4 should be flown with aircraft A_3 and crew C_1, and since the fleet capacity is *4*, one aircraft is obviously in the stand-by mode. Part X_2 shows the flight route; for instance, row A_1 states that aircraft A_1 first takes flight *1*, then *6*, and then *10*, *3*, and *9*, respectively. Part X_3 shows the crew-flight pairing; for instance, crew C_1 directs flight *2* first and then flights *4*, *9*, and *10*, respectively. Part X_4 shows that *25%* of the fleet capacity is in the stand-by mode. And, part X_5 that is related to the RM section of the problem, determines which flights should be canceled or replaced under each disruption scenario; for instance, in scenario ξ_2, flights *2* and *7* are canceled and flight *5* is done with a stand-by replacement aircraft.

After the solution representation, the question is how to search the neighborhood solution. In the ALNS algorithm, a neighborhood search requires specifying the destroy and repair; here two of each are defined as follows, the combination of which creates four operators for the neighborhood search and the ALNS algorithm adapts to them.

- **Destroy with Random Removal:** For this destroy operator, a part of the solution structure is randomly removed in X_1; in other words, some flights are randomly removed to be inserted again with the repair operator (insertion *j*).
- **Destroy with Worst Removal:** For this destroy operator, flights in undesirable conditions are removed from the initial schedule; in other words, the worst flights that have delays, are canceled or are replaced are temporarily removed to be replaced through the repair operator.
- **Repair with Random Insertion:** In this repair operator, flights removed with the destroy operator are replaced and randomly assigned aircraft and crew.
- **Repair with Best Insertion:** In this repair operator, the duty of flights removed by the destroy operator is assigned to the best aircraft and crew, meaning the one with the least added costs, the one with the highest unused capacity.

Current Solution

X1	F1	F2	F3	F4	F5	F6	F7	F8	F9	F10
Airplane	1	2	1	3	3	1	2	1	2	1
Crew	3	1	2	1	2	2	3	3	1	1

X2	F1	F2	F3	F4	F5	F6	F7	F8	F9	F10
A1	1	0	4	0	0	2	0	5	0	3
A2	0	1	0	0	0	0	2	0	3	0
A3	0	0	0	2	1	0	0	0	0	0

X3	F1	F2	F3	F4	F5	F6	F7	F8	F9	F10
C1	0	1	0	2	0	0	0	0	3	4
C2	0	0	1	0	2	3	0	0	0	0
C3	1	0	0	0	0	0	2	3	0	0

X4	25

X5	F1	F2	F3	F4	F5	F6	F7	F8	F9	F10
ξ1	1	1	1	1	1	2	1	3	1	1
ξ2	1	2	1	1	3	1	2	1	1	1
ξ3	1	1	1	1	2	1	3	1	1	1

Detsroy with Worst Removal

X1	F1	F2	F3	F4	F5	F6	F7	F8	F9	F10
Airplane	1	2	1	3	3	1	2	1	2	1
Crew	3	1	2	1	2	2	3	3	1	1

X2	F1	F2	F3	F4	F5	F6	F7	F8	F9	F10
A1	1	0	4	0	0	2	0	5	0	3
A2	0	1	0	0	0	0	2	0	3	0
A3	0	0	0	2	1	0	0	0	0	0

X3	F1	F2	F3	F4	F5	F6	F7	F8	F9	F10
C1	0	1	0	2	0	0	0	0	3	4
C2	0	0	1	0	2	3	0	0	0	0
C3	1	0	0	0	0	0	2	3	0	0

X4	25

X5	F1	F2	F3	F4	F5	F6	F7	F8	F9	F10
ξ1	1	1	1	1	1	2	1	3	1	1
ξ2	1	2	1	1	3	1	2	1	1	1
ξ3	1	1	1	1	2	1	3	1	1	1

Repair with Random Inseriton

X1	F1	F2	F3	F4	F5	F6	F7	F8	F9	F10
Airplane	1	3	1	3	2	2	1	2	2	1
Crew	3	2	2	1	1	2	2	3	1	1

X2	F1	F2	F3	F4	F5	F6	F7	F8	F9	F10
A1	1	0	2	0	0	0	3	0	0	4
A2	0	0	0	0	1	3	0	4	2	0
A3	0	1	0	2	0	0	0	0	0	0

X3	F1	F2	F3	F4	F5	F6	F7	F8	F9	F10
C1	0	0	0	1	2	0	0	0	3	4
C2	0	1	2	0	0	3	4	0	0	0
C3	1	0	0	0	0	0	0	2	0	0

X4	25

X5	F1	F2	F3	F4	F5	F6	F7	F8	F9	F10
ξ1	1	1	1	1	2	1	2	1	1	1
ξ2	1	1	2	1	1	1	1	3	1	1
ξ3	1	3	1	1	1	2	1	1	1	1

New Solution

Fig. 5. Schematic of the proposed destroy and insertion operators

Fig. 5 illustrates an example of a combined destroy-repair operators for neighborhood search. The worst destroy operator is selected from the initial solution in Fig. 4 to remove the canceled and replaced flights, another aircraft and crew is substituted for them by the random Insertion operator, and the repaired solution is generated as a neighborhood solution. To start the ALNS, the initial solution is randomly generated based on the proposed solution encoding; if the generated initial solution or the created neighborhood one is unjustified, the re-repair mechanism or the penalty strategy is used to reach justified solutions. Here, the maximum number of iterations is the stop criterion, which can also be controlled with a time constraint.

6. Model implementation and numerical result

To evaluate the proposed model, this section first conducts a moderate-scale numerical study of the CS-AMR problem and solves the TSRSO model.

To conduct a numerical study of the CS-AMR problem in some provinces of Iran, it is assumed that the air transport network is part of an airline company. As shown in Table 2, this network involves ten origin-destination cities (N: flight nodes), 42 out of 90 feasible flights (N × (N-1) = 90, flight arc = 42), five flight crew, six fleets (fleet size) and three maintenance bases. Fig. 6 presents a schematic view of the flight network for the numerical example. In addition to the location of the flight nodes, the flight arcs that connect the maintenance bases are also drawn in this map.

Table 2
The scale of flight network in the numerical study of the CS-AMR problem

Flight Node	Flight Arc	Fleet Size	Crew	Maintenance Base
10	42	6	5	3

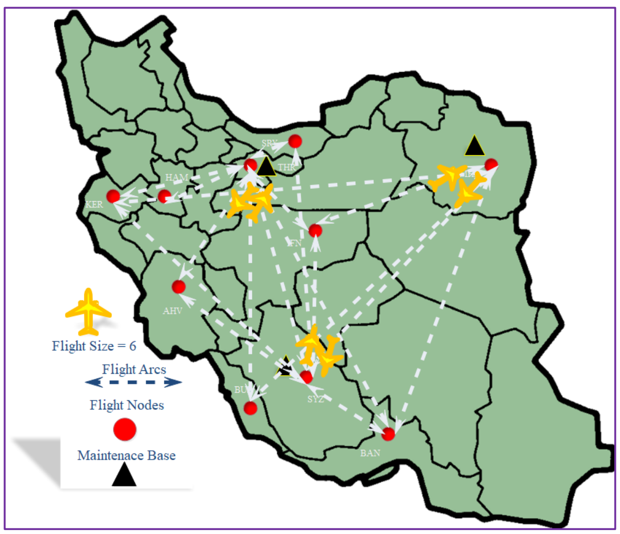


Fig. 6. Schematic of flight network in the numerical study

Table 3 shows the total number of flights scheduled in this network, and Table 4 depicts the initial schedule of the crew and aircraft.

Table 3
Scheduled flights in the numerical study of the CS-AMR problem

	THR	IFN	SYZ	MHD	SRY	KER	AHV	HAM	BAN	BUR
THR	-	☑	☑	☑	☑	☑	☑	☑	☑	☑
IFN	☑	-	☑	☑		☑	☑			
SYZ	☑	☑	-		☑	☑	☑		☑	
MHD	☑	☑	☑	-		☑			☑	☑
SRY	☑	☑	☑	☑	-					
KER	☑	☑	☑			-	☑			
AHV	☑	☑	☑	☑			-			
HAM	☑	☑	☑	☑		☑		-	☑	
BAN	☑	☑	☑	☑		☑	☑		-	
BUR	☑	☑	☑	☑	☑		☑		☑	-

This numerical study considers a 2-day schedule horizon, during which the maximum allowable no-maintenance flight time, maximum allowable crew flight time, and maximum allowable flight time for each aircraft are 480, 500 and 600 minutes, respectively, and at the end of which each crew and aircraft should be rotated at the initial base.

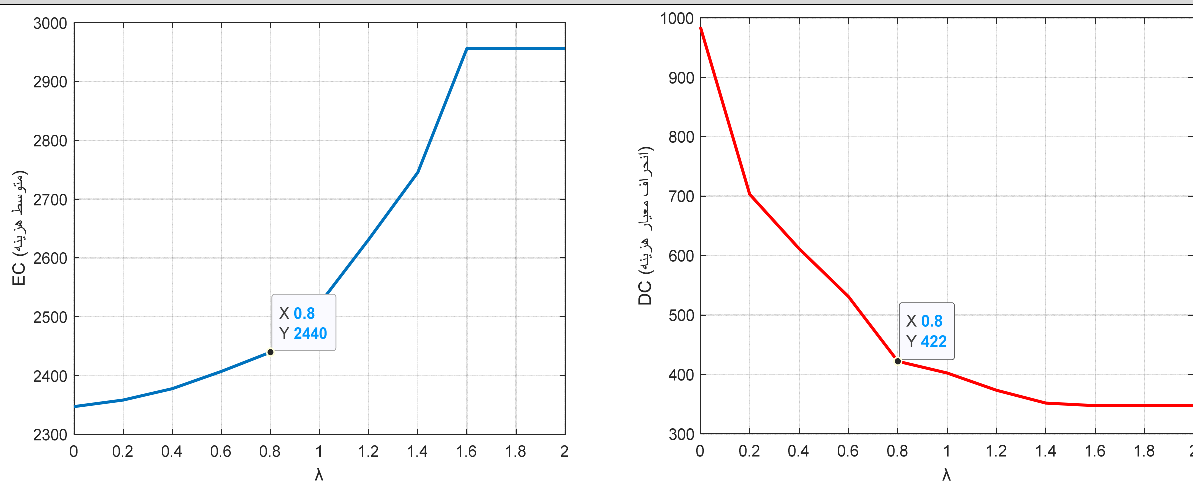
Table 4
Initial schedule for crew and aircraft in numerical study of the CS-AMR problem

	THR	IFN	SYZ	MHD	SRY	KER	AHV	HAM	BAN	BUR
aircraft	a1,a3,a4	-	a2	a5,a6	-	-	-	-	-	-
Crew	c1,c3,c4	-	c2	c5	-	-	-	-	-	-

Before implementing the TSRSO model, it is necessary first to set the optimality robustness coefficient and control the objective function fluctuations under different scenarios (λ). In Table 10, for $\lambda \geq 1.6$, the standard deviation value of the solutions and the expected system performance (mean cost) do not change; mean cost is 2956.12 (highest), and standard deviation of cost is 347.54 (lowest), but at λ =0, mean cost=2347.23 (minimum) and standard deviation of cost=984.54 (quite high) under different scenarios. Increasing λ to 0.8 will cause standard deviation of cost fluctuations to reduce by about 58% and mean cost to increase slightly (< 4% For $\lambda > 0.8$) fluctuations do not decrease considerably and, the mean cost increases significantly, concluding that $\lambda = 0.8$ is the best setting in the proposed TSRSO model; this is clearly shown in Figs. 7 and 8.

Table 5
Variations of the mean and standard deviation of cost relative to the model robustness coefficients

Optimality robustness coefficient (λ)	mean cost	standard deviation of cost	Percentage of increase in expected cost	Percentage of decrease in standard deviation of cost
0	2347.23	984.54	0.00	0.00
0.2	2358.36	703.19	0.47	28.58
0.4	2377.81	611.58	1.30	37.88
0.6	2407.12	531.19	2.55	46.05
0.8	2439.63	422.03	3.94	57.13
1	2522.03	402.45	7.45	59.12
1.2	2631.41	373.18	12.11	62.10
1.4	2745.54	351.78	16.97	64.27
1.6	2956.12	347.54	25.94	64.70
1.8	2956.12	347.54	25.94	64.70
2	2956.12	347.54	25.94	64.70



(a) (b)

Fig. 7. Cost sensitivity analyses: mean (a), standard deviation (b) with respect to the model robustness coefficients

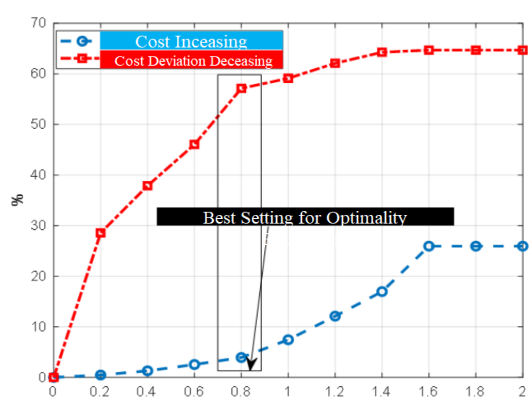


Fig. 8. The best setting for model robustness coefficient

Comparing the solution of the proposed TSRSO approach and that of the nominal approach (that does not consider probable disruptions in the initial schedule) reveals that the TSRSO approach allows some flights to be cancelled under some scenarios or aircraft routing and aircraft/crew assignment to flights are such that the least delay from the original flight schedule occurs under different disruption scenarios. To validate the proposed TSRSO model, use can be made of two criteria defined at the end of this section. The created flight delay parameter is first realized under different scenarios; in the CS-AMR problem of this research. For the validation of the TSRSO model, assume that the problem is solved by the "M" approach, z_{ξ}^{M} is the value of the objective function corresponding to this approach under scenario $\xi \in \Xi$ and z_{ξ}^{*} is the optimal value of each scenario. Approach M is more valuable when z_{ξ}^{M} has less deviation than z_{ξ}^{*}. In other words, the less than value of $\left|z_{\xi}^{M} - z_{\xi}^{*}\right|$ indicates higher value of approach M. Therefore, the validation criteria can be presented as follows:

$$Cr1 = \sum_{\xi \in \Xi} \pi_{\xi}\left(\left|z_{\xi}^{M} - z_{\xi}^{*}\right|\right) + \Psi \tag{36}$$

$$Cr2 = \max_{\xi \in \Xi} \pi_{\xi}\left(\left|z_{\xi}^{M} - z_{\xi}^{*}\right|\right) + \Psi \tag{37}$$

where Ψ is the constraint violation penalty in each scenario (in the model proposed in this research, Ψ is the flight delay/cancellation penalty equal to ($\sum_f cc_f \, e_{f\xi} + \sum_f pc_f del_{f\xi}$). Table 6 shows the values of these criteria calculated for both nominal and proposed TSRSO approaches, revealing that the solution of the latter is much more valuable than the former because both its expected and worst-case performances are much better equaling 35.76 and 39.38%, respectively, which, compared to the nominal case, are more robust.

Table 6
Comparing the proposed TSRSO approach vs Nominal model in measures of expected and worst-case performance

Approach	(Cr1)		(Cr2)	
	Value	improvement (%)	Value	improvement (%)
Nominal	21.08	-	29.12	-
Proposed TSRSO	13.54	35.76	17.65	39.38

Besides the above criteria, another well-known criterion is Value of Stochastic Solution (VSS), which is usually defined as the difference between stochastic and nominal solutions. It is used to validate the proposed TSRSO approach through the earlier performed comparison; a higher VSS means a more valuable stochastic approach. It is defined as follows:

$$VSS = |E(Z_{stochstic}) - E(Z_{nominal})| \tag{38}$$

As mentioned earlier, the CS-AMR problem was solved under five disruption and one non-disruption scenarios. Table 7 compares the results of the nominal (non-robust), and the proposed TSRSO (robust) approaches under different scenarios. Considering the probability assumed for each scenario, the VSS of the proposed approach was found to be: VSS= 2756.45-2539.67= 216.78.

Table 7
Comparing the nominal and proposed TSRSO approaches under different disruption scenarios

Scenario	Occurrence chance (π_{ξ})	Proposed TSRSO approach	Nominal Approach
$\xi = 0$	$\pi_{\xi} = \frac{1}{2}$	2395.19	2013.91
$\xi = 1$	$\pi_{\xi} = \frac{1}{10}$	2467.49	2521.67
$\xi = 2$	$\pi_{\xi} = \frac{1}{10}$	2565.42	3016.61
$\xi = 3$	$\pi_{\xi} = \frac{1}{10}$	2690.54	3412.53
$\xi = 4$	$\pi_{\xi} = \frac{1}{10}$	2781.45	3931.43
$\xi = 5$	$\pi_{\xi} = \frac{1}{10}$	2915.89	4612.79
Mean		2539.67	2756.45

7. Computational analysis and evaluation of the proposed ALNS method

This section analyzes the computational efficiency of the proposed model and the solution algorithm. To this end, a TSRSO model, with the MILP formulation, and an ALNS algorithm are first presented for the proposed CS-AMR problem, then the parameter tuning of the ALNS algorithm is done with the Taguchi method, then several different-scale experimental instances are randomly generated and solved with the proposed model and solution algorithm and the results are compared and, finally, the Box-plot is used to test the stability of the proposed ALNS algorithm.

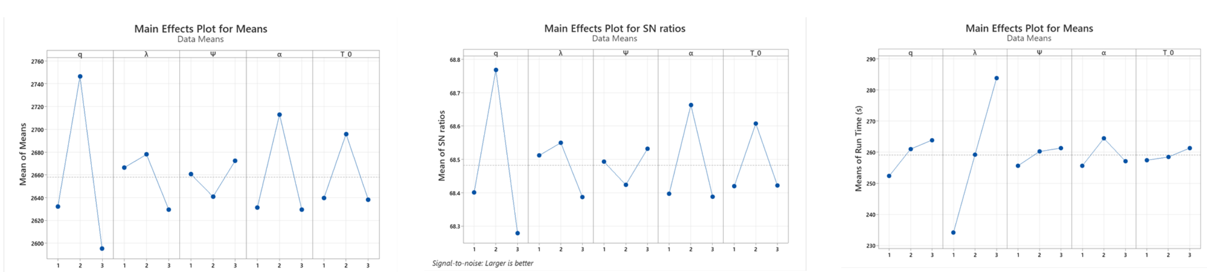
Techniques of Design of Experiments (DOE) are among effective methods for the parameter tuning of metaheuristic algorithms, which can be classified as active statistical methods (pre-execution decisions) for the quality control (QC). It is possible, through DOE techniques, to optimally/near-optimally tune the level of the parameters (factors) that influence the results of a process, or a metaheuristic algorithm such as ALNS. These techniques are generally divided into Full Factorial Experiments and Fractional Factorial Experiments (FFE). While one weakness of the former is the large number of experiments that consume much time and money, FFE methods conduct and analyze only a certain fraction of the total possible experiments to investigate the related parameters/factors. In DOE techniques, the Taguchi method is a very important and common FFE that instead of doing all the experiments performs only a fraction (few) of them to tune the required parameters optimally and much faster (Karna & Sahai, 2012). Taguchi method works based on the signal-to-noise ratio (SNR or S/N) and factors affecting the test results are either controllable (signal (S)) or uncontrollable (noise (N)). These factors/parameters are tuned at levels where the signal-to-noise ratio (S/N) criterion is maximized.

Using the Taguchi method to tune the parameters of metaheuristic algorithms requires knowing whether it is a minimization or a maximization problem because the former uses the "smaller is better" formula while the latter utilizes the "larger is better" formula to calculate the S/N ratio. In this research, since the objective function of the CS-AMR problem maximizes the profit by the revenue management, it uses the "larger is better" formula $S/_N = -10Log_{10}\left(\frac{\sum_{e=1}^{E} Z_e}{E}\right)$ where Z_e is the value of the objective function (response variable) in each experiment and E is the number of experiments. To tune the parameters of the proposed ALNS algorithm, first the numerical instance of this research, explained in the previous section, is solved for different ALN parameters (q, λ, Ψ, α, T_0) at various levels and then the level that yields a higher S/N for each parameter is chosen as the optimal level; if a parameter has close S/N at two levels, the one with lower run-time is selected, and if two parameters have no significant difference, the one that leads to less run-time is selected.

Table 8
The values of ALNS parameters/factors in different levels for Taguchi method

ALNS Parameter	Small (Level 1)	Medium (Level 2)	Big (Level 2)
q	0.20	0.35	0.60
λ	0.30	0.50	0.70
Ψ	1	3	5
α	0.25	0.50	0.75
T_0	1000	5000	10000

Table 8 considers three values at three levels (small, medium and big) for each of the five parameters of the ALNS algorithm, which means the total number of experiments is 3^5. However, Taguchi method needs only 27 experiments because it is among the FFE methods. Table 9 shows these experiments along with their objective function values and run times. Implementing the Taguchi method in Minitab and the result in tuning ALNS parameters leads to the graphs in Fig. 9 and analyzing them based on the S/N criterion reaches the levels and optimal values of the ALNS parameters shown in Table 10.



a - Means of profit in Taguchi experiments

b- S/N ratio in Taguchi experiments

c - Means of run-time in Taguchi experiments

Fig. 9. The result of Taguchi experiments in tuning ALNS parameters

Table 9
Objective function and run-time of solving problem using ALNS in Taguchi experiments

Experiments	Level of q	Level of λ	Level of Ψ	Level of α	Level of T_0	Objective	Time (s)
1	1	1	1	1	1	2581.43	209
2	1	1	1	1	2	2674.54	221
3	1	1	1	1	3	2594.41	232
4	1	2	2	2	1	2743.51	239
5	1	2	2	2	2	2781.43	264
6	1	2	2	2	3	2545.53	274
7	1	3	3	3	1	2459.41	287
8	1	3	3	3	2	2634.51	267
9	1	3	3	3	3	2675.73	278
10	2	1	2	3	1	2509.43	234
11	2	1	2	3	2	2874.43	241
12	2	1	2	3	3	2743.74	231
13	2	2	3	1	1	2693.64	253
14	2	2	3	1	2	2804.47	256
15	2	2	3	1	3	2764.04	271
16	2	3	1	2	1	2897.21	285
17	2	3	1	2	2	2680.94	294
18	2	3	1	2	3	2748.79	284
19	3	1	3	2	1	2780.43	265
20	3	1	3	2	2	2641.47	211
21	3	1	3	2	3	2597.41	264
22	3	2	1	3	1	2498.94	256
23	3	2	1	3	2	2675.83	278
24	3	2	1	3	3	2594.04	242
25	3	3	2	1	1	2593.31	289
26	3	3	2	1	2	2494.84	294
27	3	3	2	1	3	2481.54	276

Table 10
Optimal tuning of ALNS parameters resulting from Taguchi method

q	Λ	Ψ	α	T_0
Level 2 = 0.35	Level 1 = 0.30	Level 3 = 5	Level 2 = 0.50	Level 2 = 5000

To evaluate the performance and show the computational efficiency of the proposed TSRSO model and ALNS algorithm, Table 11 shows the results of 20 different-size experimental CS-AMR problems, for which the maximum solution time by the proposed MILP model was 3600 sec and the ALNS algorithm used maximum 100 and 250 iterations for the first and the second 10 instances, respectively. The numerical results show that although the TSRSO model is formulated as MILP, it is tractable and can be solved by the CPLEX Solver, but its computational efficiency is not acceptable in large-scale problems. The proposed ALNS algorithm yields optimal/near optimal solutions in small-scale instances, but in large-scale problems where MILP is not applicable, it cannot ensure a computationally efficient performance by consuming less time.

Table 11
Comparison of the proposed MILP model and ALNS algorithm to solve CS-AMR problem

Instances	Problem Scale						MILP - CPLEX		ALNS		
	Node	Flight	Fleet	Crew	Maintenance base	Scenario	OF	RT (s)	OF ($\bar{Z}$)	RT (s)	Gap (%)
1	5	10	2	2	1	3	1234.51	46	1234.51	73	0.00
2	6	15	3	3	1	3	1344.54	84	1344.54	103	0.00
3	7	20	4	3	2	3	1679.54	178	1679.54	157	0.00
4	8	25	5	4	2	5	1840.57	355	1794.43	187	2.51
5	9	30	6	4	2	5	2355.83	578	2195.51	214	6.81
6	10	35	7	5	3	5	2894.01	749	2654.39	267	4.82
7	12	40	8	5	3	5	3575.75	983	3575.75	304	0.00
8	14	45	9	6	3	10	4075.41	1349	3758.41	398	5.32
9	16	50	10	7	4	10	4678.53	1593	4385.67	495	6.26
10	18	60	11	8	4	10	5239.41	1683	5087.31	583	2.90
11	20	70	13	9	4	10	5893.75	1946	5897.41	645	0.06
12	25	80	15	10	5	15	6456.21	+3600	6678.04	798	-3.44*
13	30	90	17	12	5	15	7054.51	+3600	7376.29	875	-4.56
14	35	100	20	14	5	15	8544.51	+3600	9284.78	984	-8.66
15	40	110	25	16	6	15	9435.71	+3600	10343.12	1249	-9.62
16	50	120	30	18	6	30	NS	+3600	11984.65	1575	-
17	60	130	35	20	7	30	NS	+3600	13675.98	1984	-
18	70	150	40	25	8	30	NS	+3600	16530.04	2638	-
19	80	150	45	30	9	30	NS	+3600	18931.82	2984	-
20	90	150	50	30	10	30	NS	+3600	22874.67	3481	-

OF = objective function (profit); RT = run-time (maximum CPLEX run-time is set 3600 sec); No Solution; * = Negative value means that ALNS improves CPLEX

Since metaheuristic algorithms, including the ALNS algorithm proposed in this research, involve random search and may yield different objective-function values for a particular instance in several algorithm run times, each example problem is solved 10 times and the average value of the objective function is considered as $\left(\bar{\mathcal{Z}} = \left[\frac{1}{10}\sum_{i=1}^{10}\mathcal{Z}_i\right]\right)$, where $\mathcal{Z}_i$ is the objective value in run i. Table 12 shows the Output (Profit) of various runs of the proposed ALNS algorithm for large-scale instances to show its stability in 10 times implementation of ALNS for each large-scale instances, and Fig. 10 shows the box-plot related to 10 large-scale test problems, each of which is run 10 times; as shown, the proposed ALNS algorithm not only yields optimal/near-optimal solutions, but also has low output deviations in different runs and has acceptable stability.

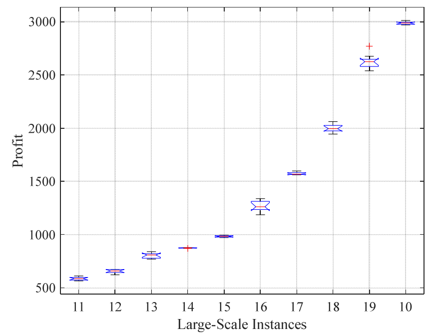


Fig. 10. Box plot of ALNS implementations to verify its stability

Table 12
Output (Profit) in 10 times implementation of ALNS for each large-scale instances to show stability

# Run	Instance 11	Instance 12	Instance 13	Instance 14	Instance 15	Instance 16	Instance 17	Instance 18	Instance 19	Instance 20
1	597.94	659.08	839.53	869.56	983.72	1204.68	1567.80	1975.26	2578.42	2997.63
2	564.60	643.78	810.45	877.66	982.46	1270.39	1564.70	1974.34	2579.34	2969.60
3	590.02	658.87	824.36	875.60	989.87	1236.01	1597.81	2053.95	2643.36	2985.62
4	585.60	641.02	816.52	873.61	994.22	1313.34	1569.03	1998.37	2674.91	2975.17
5	572.35	642.59	790.27	877.53	994.36	1186.29	1582.26	1993.75	2770.13	2988.92
6	579.47	673.92	805.49	871.84	975.93	1250.87	1573.13	2062.25	2603.92	2974.26
7	585.79	672.35	777.94	874.81	984.72	1280.82	1583.62	1944.35	2647.57	2991.95
8	612.15	672.51	820.63	874.55	972.66	1312.38	1567.58	2018.33	2633.78	2995.99
9	605.56	658.04	768.78	875.59	973.60	1337.92	1561.39	2025.16	2539.21	3011.76
10	572.00	621.56	770.77	875.58	983.94	1253.95	1561.20	1971.99	2615.57	2980.85

8. Conclusions and future studies

Crew scheduling and aircraft maintenance routing- are two important operational problems in the airline industry, which, after fuel, account for the highest costs of an airline. Integrating decisions related to these two issues, from the point of view of the revenue management approach, as well as controlling probable disruptions in the initial schedule, are among the serious challenges facing airlines in solving this problem. This study has addressed the CS-AMR problem, wherein, aircraft maintenance routing and crew scheduling decisions are made simultaneously, moreover, rules/regulations governing the flight route of each aircraft/crew such as limited maximum no-maintenance flight, maximum activity of each crew in a known time horizon, etc. are considered in a flight network consisting of several maintenance/crew bases. To solve the CS-AMR problem, we use a TSRSO model, wherein the itinerary of each aircraft and crew is specified in the first stage based on the initial schedule and the available fleet and crew, in compliance with the rules and regulations, and decisions regarding the substation of aircrafts and cancellation of flights are made in the second stage, according to the variables of the first stage and the desired disruption scenario (which causes flight delays). The proposed TSRSO model has been finally solved by the CPLEX solver aiming at maximizing expected profit and controlling cost standard deviation. Numerical results, found based on two criteria, expected performance and worst-case performance, show that the solution of the proposed TSRSO approach method is much more reliable than that the nominal model because both performances are much better; under the two mentioned criteria they are, respectively, 35.76% and 39.38% and are more robust than the nominal case. Positivity of VSS shows that the value of the proposed TSRSO approach is quite tangible compared to cases that do not consider disruption scenarios. In order to tackle the complexity of the proposed model, a solution method based on ALNS algorithm has been developed. The results of solving several experimental instances in different scales indicate that the proposed ALNS algorithm is able to solve large-scale problems in an acceptable run-time, and it is a suitable alternative to CPLEX solver which can provide an optimality guarantee.

Some airlines may consider part of their aircraft and crew capacities in the backup mode to use under disruption conditions; it is suggested here that future studies consider cancellations and substitution based on fleet backup capacity. Considering online pricing in the CS-AMR problem can be considered a development of the proposed model, which improves the applicability of the model for more real cases.

References

Abdelghany, A., & Abdelghany, K. (2018). *Airline network planning and scheduling*: John Wiley & Sons.

Abreu, L. R., & Nagano, M. S. (2022). A new hybridization of adaptive large neighborhood search with constraint programming for open shop scheduling with sequence-dependent setup times. *Computers & Industrial Engineering, 168*, 108128.

Ahmed, M. B., Ghroubi, W., Haouari, M., & Sherali, H. D. (2017). A hybrid optimization-simulation approach for robust weekly aircraft routing and retiming. *Transportation Research Part C: Emerging Technologies, 84*, 1-20.

Ahmed, M. B., Mansour, F. Z., & Haouari, M. (2018). Robust integrated maintenance aircraft routing and crew pairing. *Journal of Air Transport Management, 73*, 15-31.

Al-Thani, N. A., Ahmed, M. B., & Haouari, M. (2016). A model and optimization-based heuristic for the operational aircraft maintenance routing problem. *Transportation Research Part C: Emerging Technologies, 72*, 29-44.

An, J., Mikhaylov, A., & Jung, S.-U. (2021). A linear programming approach for robust network revenue management in the airline industry. *Journal of Air Transport Management, 91*, 101979.

Antunes, D., Vaze, V., & Antunes, A. P. (2019). A robust pairing model for airline crew scheduling. *Transportation science, 53*(6), 1751-1771.

Barnhart, C., Belobaba, P., & Odoni, A. R. (2003). Applications of operations research in the air transport industry. *Transportation science, 37*(4), 368-391.

Barnhart, C., Cohn, A. M., Johnson, E. L., Klabjan, D., Nemhauser, G. L., & Vance, P. H. (2003). Airline crew scheduling. In *Handbook of transportation science* (pp. 517-560): Springer.

Barnhart, C., Farahat, A., & Lohatepanont, M. (2009). Airline fleet assignment with enhanced revenue modeling. *Operations Research, 57*(1), 231-244.

Başdere, M., & Bilge, Ü. (2014). Operational aircraft maintenance routing problem with remaining time consideration. *European Journal of Operational Research, 235*(1), 315-328.

Bazargan, M. (2016). *Airline operations and scheduling*: Routledge.

Bulbul, K. G., & Kasimbeyli, R. (2021). Augmented Lagrangian based hybrid subgradient method for solving aircraft maintenance routing problem. *Computers & Operations Research, 132*, 105294.

Cacchiani, V., & Salazar-González, J.-J. (2013). A heuristic approach for an integrated fleet-assignment, aircraft-routing and crew-pairing problem. *Electronic Notes in Discrete Mathematics, 41*, 391-398.

Cacchiani, V., & Salazar-González, J.-J. (2017). Optimal solutions to a real-world integrated airline scheduling problem. *Transportation science, 51*(1), 250-268.

Cacchiani, V., & Salazar-González, J.-J. (2020). Heuristic approaches for flight retiming in an integrated airline scheduling problem of a regional carrier. *Omega, 91*, 102028.

Chen, C., Demir, E., & Huang, Y. (2021). An adaptive large neighborhood search heuristic for the vehicle routing problem with time windows and delivery robots. *European journal of operational research, 294*(3), 1164-1180.

Chiang, W.-C., Chen, J. C. H., & Xu, X. (2007). An overview of research on revenue management: current issues and future research. *International Journal of Revenue Management, 1*(1), 97-128.

Chutima, P., & Krisanaphan, N. (2022). Cockpit crew pairing Pareto optimisation in a budget airline. *International Journal of Industrial Engineering Computations, 13*(1), 67-80.

Cordeau, J.-F., Stojković, G., Soumis, F., & Desrosiers, J. (2001). Benders decomposition for simultaneous aircraft routing and crew scheduling. *Transportation science, 35*(4), 375-388.

Díaz-Ramírez, J., Huertas, J. I., & Trigos, F. (2014). Aircraft maintenance, routing, and crew scheduling planning for airlines with a single fleet and a single maintenance and crew base. *Computers & Industrial Engineering, 75*, 68-78.

Dück, V. (2010). Increasing stability of aircraft and crew schedules. *Framework*(July), 1-157.

Eltoukhy, A. E. E., Chan, F. T. S., & Chung, S. H. (2017). Airline schedule planning: a review and future directions. *Industrial Management & Data Systems*.

Eltoukhy, A. E. E., Chan, F. T. S., Chung, S. H., & Niu, B. (2018). A model with a solution algorithm for the operational aircraft maintenance routing problem. *Computers & Industrial Engineering, 120*, 346-359.

Eltoukhy, A. E. E., Wang, Z. X., Chan, F. T. S., & Chung, S. H. (2018). Joint optimization using a leader–follower Stackelberg game for coordinated configuration of stochastic operational aircraft maintenance routing and maintenance staffing. *Computers & Industrial Engineering, 125*, 46-68.

Etschmaier, M. M., & Mathaisel, D. F. X. (1985). Airline scheduling: An overview. *Transportation science, 19*(2), 127-138.

Gao, C., Johnson, E., & Smith, B. (2009). Integrated airline fleet and crew robust planning. *Transportation science, 43*(1), 2-16.

Gopalan, R., & Talluri, K. T. (1998). The aircraft maintenance routing problem. *Operations Research, 46*(2), 260-271.

Haouari, M., Zeghal Mansour, F., & Sherali, H. D. (2019). A new compact formulation for the daily crew pairing problem. *Transportation science, 53*(3), 811-828.

Jalili, M., & Manteghi, M. (2018). Analysis of forecasting the demand for passenger and cargo transport in Iran aviation industry. *Quarterly Journal of Transportation Engineering, 10*(1), 75-101.

Karna, S. K., & Sahai, R. (2012). An overview on Taguchi method. *International journal of engineering and mathematical sciences, 1*(1), 1-7.

Kasirzadeh, A., Saddoune, M., & Soumis, F. (2017). Airline crew scheduling: models, algorithms, and data sets. *EURO Journal on Transportation and Logistics, 6*(2), 111-137.

Kumar, R., Wang, W., Simrin, A., Arunachalam, S. K., Guntreddy, B. R., & Walczak, D. (2021). Competitive revenue management models with loyal and fully flexible customers. *Journal of Revenue and Pricing Management, 20*(3), 256-275.

Lenstra, J. K., & Rinnooy Kan, A. H. G. (1978). Complexity of scheduling under precedence constraints. *Operations Research, 26*(1), 22-35.

Li, Z., & Jiao, P. (2022). Two-stage stochastic programming for the inventory routing problem with stochastic demands in fuel delivery. *International Journal of Industrial Engineering Computations, 13*(4), 507-522.

McGill, J. I., & Van Ryzin, G. J. (1999). Revenue management: Research overview and prospects. *Transportation science, 33*(2), 233-256.

Mirabi, M., Ghiyasvand-Mohammadkhani, H., & Tavakkoli-Moghaddam, R. (2019). Mathematical modeling and solving a hub location-allocation problem in Iranian airports. *Quarterly Journal of Transportation Engineering*.

Mnich, M., & van Bevern, R. (2018). Parameterized complexity of machine scheduling: 15 open problems. *Computers & Operations Research, 100*, 254-261.

Mulvey, J. M., Vanderbei, R. J., & Zenios, S. A. (1995). Robust optimization of large-scale systems. *Operations Research, 43*(2), 264-281.

Nasri, M., Metrane, A., Hafidi, I., & Jamali, A. (2020). A robust approach for solving a vehicle routing problem with time windows with uncertain service and travel times. *International Journal of Industrial Engineering Computations, 11*(1), 1-16.

Papadakos, N. (2009). Integrated airline scheduling. *Computers & Operations Research, 36*(1), 176-195.

Parmentier, A., & Meunier, F. (2020). Aircraft routing and crew pairing: Updated algorithms at Air France. *Omega, 93*, 102073.

Phillips, R. L. (2021). Pricing and revenue optimization. In *Pricing and Revenue Optimization*: Stanford university press.

Ropke, S., & Pisinger, D. (2006). An adaptive large neighborhood search heuristic for the pickup and delivery problem with time windows. *Transportation Science, 40*(4), 455-472.

Ruan, J. H., Wang, Z. X., Chan, F. T. S., Patnaik, S., & Tiwari, M. K. (2021). A reinforcement learning-based algorithm for the aircraft maintenance routing problem. *Expert Systems with Applications, 169*, 114399.

Sa, C. A. A., Santos, B. F., & Clarke, J.-P. B. (2020). Portfolio-based airline fleet planning under stochastic demand. *Omega, 97*, 102101.

Safaei, N., & Jardine, A. K. S. (2018). Aircraft routing with generalized maintenance constraints. *Omega, 80*, 111-122.

Sahebi, H., Ranjbar, S., & Teymouri, A. (2022). Investigating different reverse channels in a closed-loop supply chain: A power perspective. *Operational Research, 22*(3), 1939-1985.

Salazar-González, J.-J. (2014). Approaches to solve the fleet-assignment, aircraft-routing, crew-pairing and crew-rostering problems of a regional carrier. *Omega, 43*, 71-82.

Salehi, F., Mahootchi, M., & Husseini, S. M. M. (2019). Developing a robust stochastic model for designing a blood supply chain network in a crisis: A possible earthquake in Tehran. *Annals of Operations Research, 283*(1), 679-703.

Sanchez, D. T., Boyacı, B., & Zografos, K. G. (2020). An optimisation framework for airline fleet maintenance scheduling with tail assignment considerations. *Transportation Research Part B: Methodological, 133*, 142-164.

Sandhu, R., & Klabjan, D. (2007). Integrated airline fleeting and crew-pairing decisions. *Operations Research, 55*(3), 439-456.

Sarac, A., Batta, R., & Rump, C. M. (2006). A branch-and-price approach for operational aircraft maintenance routing. *European Journal of Operational Research, 175*(3), 1850-1869.

Schaap, H., Schiffer, M., Schneider, M., & Walther, G. (2022). A Large Neighborhood Search for the Vehicle Routing Problem with Multiple Time Windows. *Transportation Science*.

Schaefer, A. J., Johnson, E. L., Kleywegt, A. J., & Nemhauser, G. L. (2005). Airline crew scheduling under uncertainty. *Transportation science, 39*(3), 340-348.

Shao, S., Sherali, H. D., & Haouari, M. (2017). A novel model and decomposition approach for the integrated airline fleet assignment, aircraft routing, and crew pairing problem. *Transportation science, 51*(1), 233-249.

Talluri, K. T. (1998). The four-day aircraft maintenance routing problem. *Transportation science, 32*(1), 43-53.

Vinod, B. (2021). An approach to adaptive robust revenue management with continuous demand management in a COVID-19 era. *Journal of Revenue and Pricing Management, 20*(1), 10-14.

Weide, O., Ryan, D., & Ehrgott, M. (2010). An iterative approach to robust and integrated aircraft routing and crew scheduling. *Computers & Operations Research, 37*(5), 833-844.

Wen, X., Ma, H.-L., Chung, S.-H., & Khan, W. A. (2020). Robust airline crew scheduling with flight flying time variability. *Transportation Research Part E: Logistics and Transportation Review, 144*, 102132.

Yu, C.-S., & Li, H.-L. (2000). A robust optimization model for stochastic logistic problems. *International Journal of Production Economics, 64*(1-3), 385-397.

Zhang, X., & Mahadevan, S. (2017). Aircraft re-routing optimization and performance assessment under uncertainty. *Decision Support Systems, 96*, 67-82.