Since January 2020 Elsevier has created a COVID-19 resource centre with free information in English and Mandarin on the novel coronavirus COVID-19. The COVID-19 resource centre is hosted on Elsevier Connect, the company's public news and information website.

Elsevier hereby grants permission to make all its COVID-19-related research that is available on the COVID-19 resource centre - including this research content - immediately available in PubMed Central and other publicly funded repositories, such as the WHO COVID database with rights for unrestricted research re-use and analyses in any form or by any means with acknowledgement of the original source. These permissions are granted for free by Elsevier for as long as the COVID-19 resource centre remains active.
A matheuristic approach to the air-cargo recovery problem under demand disruption

Felipe Delgado *, Julio Mora

Departamento de Ingeniería de Transporte y Logística, Pontificia Universidad Católica de Chile, Víncula Mackenna, 4860, Macul, Casilla 306, Correo 22, Santiago, Chile

ARTICLE INFO

Keywords:
Air cargo schedule recovery
Airline schedule recovery
Disruption management
Air cargo rescheduling
Column generation
Matheuristic

ABSTRACT

Air cargo transport is subject to unpredictable changes in expected demand, necessitating adjustments to itinerary planning to recover from such disruptions. We study a flight rescheduling problem to react to cargo demand disruptions in the short run. To increase flexibility, we consider two different cargo assignment policies. We propose a matheuristic approach to solve the problem that provides high-quality solutions in a short computational time, based on column generation in which each subproblem is solved using an ad-hoc heuristic. The approach is tested on demand disruption instances containing up to 75 air cargo orders with different penalty levels. The results show that the proposed method improves profit by 54% over the solution generated by a commercial MIP solver within a 1-h time limit, and by 15% over the solution with the routes fixed as in the original flight planning that only allows cargo to be re-routed. We also show that there exist incremental benefits in the range of 3-5% by allowing cargo for a given order to be transported by various aircraft.

1. Introduction

Air cargo can be carried either in dedicated freighters, that is, aircraft designed exclusively for cargo transport, or in the spare capacity in passenger aircraft bellies after baggage has been loaded. Although in recent years the annual added payload capacity in bellies has surpassed that of freighters by ever increasing margins (in 2018 it was almost 4 times greater) (IATA, 2018), freighters have certain advantages that continue to make them indispensable to any air cargo operation. They can, for example, carry types of loads that passenger aircraft cannot, such as oversize items or dangerous cargo. Furthermore, the supply characteristics of bellies on passenger flights, either because of their schedules, markets served or spare capacity, do not necessarily coordinate well with the characteristics of air cargo demand (Boeing, 2014). Moreover, freighters play a vital role in keeping functioning the supply chain in crisis as the COVID-19 pandemic, allowing transporting medicines and materials in a timely fashion. These observations lie behind the analysis offered in the present study.

One of the principal factors differentiating air cargo transport from air passenger transport is the high level of uncertainty in demand for the former. The sources of this uncertainty are the very short time windows in which cargo bookings are made (Sandhu and Klabjan, 2006) and the frequent occurrence of cargo no-shows or only partial show-ups (Wada et al., 2017). The problem typically arises just hours before a departure, causing an imbalance between planned supply and real demand.

To deal with these demand disruptions, which are difficult to anticipate, cargo airlines first intensify their sales-efforts and make focussed, last-minute adjustments to their schedule planning based on the experience of decision-makers charged with modifying schedules manually. A range of measures are employed, alone or in combination, such as adjust fleet allocation, flight cancellations, aircraft rerouting (e.g. triangle flights) and flying aircraft empty to reallocate cargo (where doing makes economic sense). These operational actions are challenging since they have to comply with the bilateral agreements between countries, frequency restrictions and must be adopted within short time frames.

Schedule design in the air transport industry has been extensively studied for the passenger business. Etschmaier and Mathaisel (1985) were the first to review the state of the art in airline scheduling. Since that time, numerous works have been published on airline schedule planning, covering topics such as incremental design of passenger schedules (Barnhart et al., 2003; Berge and Hopperstad, 1993), the simultaneous schedule design and fleet assignment problem (Levin, 1971; Rexing et al., 2000; Lohatepanont and Barnhart, 2004) and disturbances or unexpected events. The latter have been studied from two different perspectives, the first one based on the development of robust
schedule planning models (Gao et al., 2009; Lan et al., 2006; Rosenberger et al., 2004; Pita et al., 2013; Froyland et al., 2014; Jamlili, 2017) and the second one on operational schedule recovery models. In this group we find the aircraft recovery problem (Teodorović and Guberinić, 1984; Rosenberger et al., 2003; Vos et al., 2015), the integrated aircraft and passenger recovery problem (Jozefowiez et al., 2013; Hu et al., 2016; Zhang et al., 2016) and the integrated aircraft and crew schedule recovery problem (Zhang et al., 2015; Maher, 2015).

Although schedule recovery models have been analyzed in depth for passenger air transport, little work has yet been done on the recovery problem in the air cargo business. Disturbances in the passenger industry are of a different nature from those arising in the cargo market and are due mainly to weather or the propagation of delays caused by late or cancelled flights rather than demand fluctuations. Unexpected changes in demand are frequent, affecting the supply chain due to deviations in production, transportation, and storage costs, among others. For example, the COVID-19 pandemic caused an explosive increase in the demand for products such as masks and respirators. The literature has studied demand disruptions with a focus on supply chain coordination, Qi et al. (2004) consider one supplier - one retailer and propose a two-period model. In the first period, before the demand is known, a production plan is decided, which suffers modifications in the second once the actual demand is revealed. The authors study how changes to the original plan produce significant deviation costs to the system and derive conditions under which the supply chain can be coordinated. Xiao et al. (2007), study this coordination problem considering two competing retailers where the decision of one retailer affects the other since some customers may switch between them. Chen and Xiao (2009) extend it to the case of one dominant retailer and multiple fringe retailers.

Another major difference between passenger and air cargo transport lies in the type of corrective actions taken when disturbances occur. For passenger transport, the legs of a route cannot be changed in the short run due to passenger preferences for certain itineraries, but with air freight, there is greater flexibility in the legs actually flown since what counts is that the cargo arrive at its destination by the stipulated deadline regardless of the route actually taken.

One of the first published studies in a cargo context is that of Marsten and Muller (1980), who propose a mixed integer programming model to solve two strategic problems, route design and fleet planning, for a hub-and-spoke network. Lin and Chen (2003) develop an embedded multi-commodity minimum cost flow problem (MCFP) that chooses transit (stopover) airports for the design of a cargo network linking China and Taiwan using belly space on passenger flights. Yan et al. (2006) present a model that integrates the strategic processes of cargo airline route design and fleet assignment. By using a weekly planning horizon, the solutions can be applied in short-term operations. The authors formulate the problem as an MCFP assuming a single fleet. The idea is to integrate a flight network with a cargo network (O-D pairs) and design a route starting from zero independent of any base plan that considers all possible flight legs and incurs no flight cancellation penalties. The solution employs various heuristics based on the number of stops that must be made to pick up and deliver cargo to its destination. Yan and Chen (2008) extend this model to scenarios where there are alliances between airlines that require coordinated scheduling. Tang et al. (2008) devise a model that integrates route design for passenger, cargo and combi flights. The problem is formulated as an integer MCFP and a set of heuristics are supplied - one retailer.

Derigs and Friederichs (2013), after surveying the general problem of cargo flight planning and its subproblems, then turn their attention to the air cargo scheduling problem. They present a model that integrates several stages including itinerary design, fleet assignment, and aircraft and cargo routing. However, the planning generated by the model is confined to an externally predefined list of mandatory and optional flights, the latter incorporated only to the extent they generate profits. To the best of our knowledge, the only paper that deals with air cargo schedule recovery is Delgado et al. (2020). In their work, they propose a mixed integer programming model to redesign an operational schedule in an air cargo network and introduced three comparison schemes to estimate the cost of rescheduling. For a recent review for the literature regarding air cargo operations, please refer to Feng et al. (2015).

What emerges from the above review is that the problem of cargo route adjustment in response to last-minute demand disruptions has been given little attention in the literature at the operational level. This problem has a number of special characteristics: time is a limited resource, there is no predefined flight list and minor adjustments to the base planning scheme must be doable. The principal contributions of the present article that distinguish it from prior studies may be summed up as follows:

a) We develop a model that can react to cargo demand disruptions in the short run. The proposed formulation is not comparable to reactive models of passenger air transport where demand disruptions are of a different nature, caused typically by weather and flight delays or cancellations.

b) The proposed model simultaneously handles aircraft routing, cargo routing and changes in the base plan, while attempting to deviate as little as possible from the base plan through the use of penalties.

c) We develop a fast matheuristic solution procedure that can provide high quality solutions in short computation time.

d) We explore the potential benefits to the recovery process obtained by allowing the cargo for a given order to be transported by various aircraft.

The remainder of this paper is organized into four sections. Section 2 models the problem to be solved, setting out the notation, the main assumptions and the cargo assignment policies. Section 3 describes the proposed solution method, including the column generation technique and the heuristic developed to solve each subproblem. Section 4 reports the computational results for a set of air cargo instances with demand disruption used to test the model. Finally, Section 5 presents our conclusions.

2. Model development

In this section, we define and formulate the flight rescheduling problem to react to cargo demand disruptions in the short run. This problem simultaneously addresses two types of decisions: flight scheduling and cargo routing. Flight scheduling decisions determine which new flights to add to the base plan, and which to cancel. In cargo routing, the model must decide the orders to transport, how much of each order to accept, and their corresponding routing. In what follows, we provide a formal definition, give a mathematical formulation for the problem and present two cargo assignment policies.

2.1. Problem definition

An airline operates a fleet of \( K \) in \( K \) aircraft. Each aircraft has a subnetwork it can operate over and which is delimited by various operating restrictions or landing rights that prevent it from operating freely over the entire airport network. The fleet is heterogeneous, with
each plane having a given load capacity (CAP), initial and final locations \((0_0^+, 0_0^-)\) at either end of the recovery horizon and a base plan \(FB\) to which adjustments must be as few as possible. During the recovery horizon, there is a set of orders \(P = \{1, \ldots, |P|\}\) to be fulfilled. An order \(i\) is defined by a freight rate \(r_i\) a strategic factor \(\phi_i\), a pickup node \(i^-\) where the cargo specified in the order is loaded and a delivery node \(i^+\) where it is unloaded, both nodes located at an airport and with its respective time window and loading/unloading time. The start of the pickup time window represents the order release time while the end of the delivery time window represents the delivery deadline. The strategic factor can be used to give priority to original orders or certain specific clients. Each node in the cargo network is an event representing a take-off, landing or stopover at a given airport with its corresponding loading and unloading of an order or parts thereof.

The arcs connecting the nodes are ground arcs (representing airport stopovers) if both are located at the same airport, and flight legs if they are located at different airports. Besides, there is a set of macronodes which consolidates all the pickup and/or delivery nodes for the orders to be transported by a given aircraft to or from a single airport and thus defines an origin or destination node for the corresponding flight leg in the base plan.

Fig. 1 shows an illustrative example of the aircraft routing using macronode representation. In the figure, the aircraft first picks up orders (1,2,3) which constitutes the first macronode. Then it flies to the second macronode at airport B, where it delivers order 2 and picks up order 4. Finally, the aircraft continues to airport C where orders (1,3,4) are delivered.

The proposed problem is a variant of the pickup and delivery problem with time windows (PDPTW) (Dumas et al., 1991; Cordeau et al., 2002; Parragh et al., 2008). Adjustments and constraints have been added to the classic formulation to adapt it to the case of air cargo.

In what follows we set forth the main assumptions of the model, its notation and mathematical formulation, and then introduce the decomposition method for solving it.

### 2.2. Assumptions

The proposed model makes the following assumptions:

- **a)** Cargo cannot be transferred between aircraft (i.e., no transhipments).
- **b)** Load capacity restrictions are expressed in terms of weight, in keeping with industry practice.
- **c)** No penalties are applied for orders in the base plan that are partially or fully undelivered within the recovery horizon. However, the strategic factor \(\phi_i\) can be used to give priority to original orders or to strategic clients as was explained at the beginning of section 2.

### 2.3. Notation

#### Sets

| Set   | Description                                      |
|-------|--------------------------------------------------|
| \(K\) | Set of aircraft, \(k = \{1, \ldots, |K|\}\)     |
| \(P\) | Set of orders, \(P = \{1, \ldots, |P|\}\)       |
| \(P^k\) | Set of orders that can be carried by aircraft \(k\), \(P^k \subseteq P\) |
| \(N^k\) | Set of pickup and delivery arcs that can be served by aircraft \(k\), including the initial and final location nodes \((0_0^+, 0_0^-)\) at the start and end of the recovery horizon, respectively. |
| \(A^k\) | Set of arcs that can be served by aircraft \(k\), \((i,j) \in A^k\), where \(i,j \in N^k\). |
| \(NA^k\) | Set of predecessor arcs that can be served by aircraft \(k\) from arc \((i,j)\). |
| \(NS^k\) | Set of successor arcs that can be served by aircraft \(k\), \(NS^k = N^k \setminus \{(i,j)\}\). |
| \(S\) | Set of operating arcs and/or delivery arcs constituting a macrone \(i \in S\). |
| \(S\) | Set of macronodes. |
| \(FB^k\) | Set of flight legs in the base plan for aircraft \(k\), indexed as \((n,m)\) where \(n, m \in S\). |

#### Parameters

| Parameter | Description                                      |
|-----------|--------------------------------------------------|
| \(\psi_i\) | Operating cost of arc \((i,j) \in A^k\) by aircraft \(k\) [US$/arc]. |
| \(\omega_{n,m}\) | Penalty cost imposed for modification to flight \((n,m) \in FB^k\) of aircraft \(k\). [US$/flight]. |
| \(\phi_i\) | Strategic factor associated with order \(i \in P\). |
| \(r_i\) | Freight rate for transporting order \(i \in P\) [US$/tonnes]. |
| \(t_{ij}\) | Travel time between node \(i\) and \(j\) for aircraft \(k\) [hrs]. |
| \(d_i\) | Loading or unloading time at node \(i\) [hrs]. |
| \(\epsilon_i\) | Earliest arrival time at which node \(i\) can be visited [hrs]. |
| \(\lambda_i\) | Latest arrival time at which node \(i\) can be visited [hrs]. |
| \(CAP_k\) | Aircraft load capacity \(k\) [tonnes]. |

#### Decision variables

| Decision variable | Description                                      |
|-------------------|--------------------------------------------------|
| \(x_{ij}^k\) | Binary variable, equal to 1 if aircraft \(k\) traverse arc \((i,j)\). |
| \(\beta^k\) | Time at which aircraft \(k\) begins its visit to node \(i \in K\) [hrs]. |
| \(Q_f^k\) | Amount of cargo assigned to node \(i \in K^N\). Represents the cargo picked up \((Q_f^k > 0)\) if node \(i\) is a pickup node, and the cargo delivered \((Q_f^k < 0)\) if node \(i\) is a delivery node [tonnes]. |
| \(\delta_{n,m}\) | Amount of cargo carried by aircraft \(k\) when departing node \(i \in K^N\) [tonnes]. |
| \(\mu_{n,m}\) | Auxiliary variable, equal to 1 if flight leg \((n,m)\) is cancelled, otherwise 0. |

### 2.4. Mathematical formulation

The mathematical formulation of the proposed model is set out below.

\[
\begin{align*}
\text{max} & \quad \sum_{k \in K} \sum_{(i,j) \in A^k} \psi_i \cdot q_{ij}^k - \sum_{k \in K} \sum_{(i,j) \in A^k} t_{ij}^k \cdot x_{ij}^k - \sum_{k \in K} \sum_{(n,m) \in F^k} \omega_{n,m} \cdot \alpha_{n,m}^k \quad (1) \\
\text{subject to:} & \\
\mu_{n,m}^k & \geq 1 - \sum_{j \in K} \sum_{(i,j) \in A^k} x_{ij}^k \quad \forall (n,m) \in F^k, k \in K \quad (2) \\
\rho_{n,m}^k & \geq 0 \quad \forall (n,m) \in F^k, k \in K \quad (3) \\
\sum_{j \in K} x_{ij}^k & = 1 \quad \forall k \in K \quad (4) \\
\sum_{j \in K} x_{ij}^k & = 1 \quad \forall k \in K \quad (5) \\
\sum_{j \in K} x_{ij}^k - \sum_{j \in K} x_{ij}^k & = 0 \quad \forall i \in N^k, k \in K \quad (6) \\
B_i^k & \geq (B_i^k + d_i + t_{ij}^k) \cdot x_{ij}^k \quad \forall (i,j) \in A^k, k \in K \quad (7) \\
Q_f^k & \geq (Q_f^k + q_f^k) \cdot x_{ij}^k \quad \forall (i,j) \in A^k, k \in K \quad (8)
\end{align*}
\]
practice in the industry where separating customers load is considered a poor service and often not accepted by clients.

The idea behind the study of these two policies is to quantify how much an airline loses by preventing the same order to be transported in different aircraft or by accepting partial orders.

Note that under policy P2, if an order is accepted, the complete cargo must be transported by a single aircraft if the capacity constraint is satisfied. If the weight exceeds the remaining capacity, the full order is offloaded.

3. Solution method

Since the problem as set out above in (1)-(17) is NP-hard (Savelsbergh and Sol, 1995), solving it exactly would be very costly in computation time. This could well mean that the exact solutions the model eventually finds in response to last-minute changes in cargo demand would not be generated quickly enough to be useful in the real-world application the model is intended for. Some other approach is therefore required.

Various solution methods have been developed for solving PDPTW problems, most of which have been either exact (Sigurd et al. 2004; Ropke et al., 2007; Ropke and Cordeau, 2009) or heuristics (Ropke and Pisinger, 2006; Qu and Bard, 2012; Cherkesly et al., 2015). For our PDPTW-variant problem we propose to combine the two methods in the form of a heuristic technique, a technique that combines the use of heuristics with mathematical programming models (Maniezzo et al., 2010). Since the problem has a block structure, we can solve it using the Dantzig-Wolfe decomposition (Dantzig and Wolfe, 1960). In the present case, this means the formulation is first decomposed into a restricted master problem (RMP) and a set of KSPs, each of which generates feasible routes for a given aircraft. The linear relaxation of the RMP is then solved using the column generation approach (Wolsey, 1998; Barnhart et al., 1998; Desrosiers and Lübbecke, 2005; Feillet, 2010) while the KSPs are tackled under a heuristic approach. The definitive solution employs an approach which consists in solving a restricted version of the RMP that includes only those columns generated by solving the linear relaxation.

3.1. General framework

The formulations of the restricted master problem and the kth subproblem are set out below.

3.1.1. Restricted master problem (RMP)

The sets, parameters and additional variables in the RMP are as follows:

3.1.1.1. Sets. \( R_i \): Set of feasible routes for aircraft \( r \).
\( N_r \): Set of nodes visited on route \( r \).
\( P_r \): Set of orders corresponding to the nodes visited on route \( r \).

3.1.1.2. Parameters. \( Y_{b,i}^r \): Represents the route generated by the \( k \)th subproblem; a vector defined as

\[
Y_{b,i}^r = \left\{ \left( x_{i,j}^r \right)_{i,j \in N_r} \right\} \in P_r \cap \left( \{ Q_i \}_{i \in N_r} \cap \{ B_i \}_{i \in N_r} \cap \{ \mu_{a_n} \}_{a_n \in \{ 0,1 \}} \right) \frac{P^T}{FB}
\]

\( C_b^r \): The profit associated with route \( r \) of aircraft \( k \); a vector whose elements are the coefficients of each family of variables \( Y_{b,i}^r \), defined as

\[
C_b^r = \left\{ -c_{i,j}^r \right\}_{i,j \in N_r} \frac{P_r}{FB}
\]

The dot product \( C_b^r \cdot Y_{b,i}^r \) is therefore the value of the profit for aircraft \( k \) in the original objective function, evaluated in the solution or route generated by the kth subproblem.

3.2. Cargo assignment policies

In this paper, we analyze two different cargo assignment policies:

i) P1 where orders can be partially accepted or continuously divisible, thus allowing the cargo for a given order to be transported by various aircraft, each one picking up a partial order.

ii) P2, under this policy orders must be delivered completely or not delivered at all, thus avoiding partial orders. The latter is a common

\[
Q_{b,i}^r = \begin{cases} 0 & \forall k \in K \\ \min \{ CAP^b, CAP^b + q_{i}^r \} & \forall i \in N_r, \forall k \in K \end{cases}
\]

\[
q_{i}^r \leq CAP^b \sum_{j \in N_r} x_{i,j}^r, \forall i \in P_r, k \in K
\]

\[
\sum_{i \in N_r} q_{i}^r \leq D_i, \forall i \in P
\]

\[
q_{i}^r + q_{j}^r = 0, \forall i \in P_r, k \in K
\]

\[
\sum_{i \in N_r} x_{i,j}^r - \sum_{i \in N_r} x_{i,j}^r = 0, \forall i \in P_r, k \in K
\]

\[
0 \leq e_i \leq D_i, \forall i \in N_r, k \in K
\]

\[
x_{i,j}^r \in \{ 0,1 \}, \forall (i,j) \in A^r, k \in K
\]

\[
q_{i}^r \in \mathbb{R}, \forall i \in N_r, k \in K
\]
part of the family \( \{ q_i \} \) in vector \( Y_k \).

### 3.1.1.3. Variables.

\( x_k^r \): A binary variable, equal to 1 if route \( r \) is assigned to aircraft \( k \), otherwise 0.

The RMP can now be formulated as follows:

\[
\text{(RMP)} \quad \max \sum_{k \in K} \sum_{r \in R_k} (C_k^r - Y_k^r) x_k^r \\
\text{subject to:}
\sum_{k \in K} \sum_{r \in R_k} x_k^r q_i^r - \pi_i \leq D_i, \forall i \in P
\]

\[
\sum_{r \in R_k} x_k^r = 1, \forall k \in K
\]

\[
x_k^r \in \{0, 1\}, \forall r \in R_k, \forall k \in K
\]

The objective (18) of the RMP can be described as a convex linear combination of the values of the solutions or routes generated by the subproblems. The \( |R_k| \) subproblem solutions are used to generate a feasible solution of the original problem. Constraint (19) is a linking condition imposing the interdependence of the different subproblems, restricting cargo demand at each pickup node for order in analogous fashion to constraint (12) in the original problem. Constraint (20) is known as the convexity constraint and may be interpreted as a restriction that ensures each aircraft \( k \) is assigned only one route \( r \) from the set of routes it can feasibly fly. Constraint (21) expresses the binary nature of the decision variable, which must be relaxed to facilitate the solution of the problem and thus obtain the dual values that are then entered into the \( k \) subproblems.

Each time the relaxed RMP is solved, we obtain the dual values \( \pi_i \) for each pickup node and \( \gamma_k \) for each subproblem. \( \pi_i \) may be interpreted as the opportunity cost of transporting an additional unit of order \( i \) before a unit of any other order while \( \gamma_k \) may be interpreted as the profit on the optimal route generated for aircraft \( k \) up to that point.

### 3.1.2. The \( k \)-th subproblem (k-SP)

Each satellite problem, hereafter denoted \( k \)-SP, attempts to find the optimal route for a given aircraft \( k \) and consists of all of the constraints in the original problem except linking condition (12).

![Solution algorithm flow diagram](image-url)
subject to:

\[(2)-(11) \text{ y (13)-(17)}\]

Objective function (22) represents the reduced costs \((\bar{c}^k)\) in the original problem. The solution obtained is a feasible route for aircraft \(k\) without considering the demand constraints. If the OF value or reduced cost given by the solution to (22) is greater than 0, the route is beneficial for the problem and is incorporated into the RMP.

Since the search for an exact solution of the RMP would involve an exponential number of columns (every feasible route for each aircraft), we propose to solve it using a column generation approach in which the RMP has a limited number of initial columns and new columns are added as the k-SP subproblems are solved heuristically. Then for any k-SP, if the heuristic solution for the final route generated is a positive reduced cost, the corresponding column is added to the RMP and the contrary case is discarded. The column generation algorithm is iterated until no more subproblems are found that have generated columns with positive reduced cost. Once this procedure is completed, the RMP is solved (imposing the binary variable constraint) for a restricted version of the problem that includes only those columns generated up to that point. This solution methodology is summarized in Fig. 2.

Since at the first iteration no columns have yet been added to the RMP, a set of initial columns must be generated. Three such sets are in fact generated for each aircraft as follows:

1) **Base Plan.** This set consists of three routes (columns) for each aircraft. One of them is the base plan itself while the other two are obtained by applying to it respectively two local search operators, one denoted \(p\) and add, the other swap, as will be described below in Section 3.2.2. If the base plan is no longer feasible after disruptions have occurred, it is made feasible by adjusting the time windows and loads.

2) **Deadheading.** This set of columns is for the case in which an aircraft flies directly from the initial to the final airport \((i_0, j_0)\) without taking any orders.

3) **Additional Columns.** This set consists of three columns for each aircraft. One of them is a route specified by a construction heuristic while the other two are obtained by applying to the route respectively the same two operators applied for the base plan column set.

The construction heuristic generates routes for all of the aircraft in parallel. This is done to get routes that are more balanced (Labadie et al., 2016). The heuristic proceeds by first adding an order to each of the routes, then adding a second one, and continuing iteratively until all orders have been routed. Each new order \(i\) so added is inserted at the end of the route, its pickup node \(i^*\) first and its delivery node \(i\) immediately thereafter. In each case, the new order is the one with the lowest insertion cost. In formal terms, this procedure is as follows: Let \(j\) be the last order inserted into route \(k\) so far, and \(l_k^j\) be the number of pending orders that can be carried by aircraft \(k\). Then the next order \(i^*\) to be inserted will be the one that satisfies \(i^* = \arg \min_{i \in L_k} \{ \phi_{k,i} + c_{k,i} \} + t_w \) and the time window and load capacity constraints. When the procedure terminates, each aircraft \(k\) will have a route of type \((i_0, j_0, i^*, p^*, r^*, \ldots, j_n)\). There may, however, be some orders that have not been routed, in which case the procedure attempts to insert both their pickup and delivery nodes into the positions that generates the greatest profit.

\[(k - \text{SP}) \max \left( \sum_{i \in k} \phi_{k,i} q_{k,i} - \sum_{i,j \in A_k} c_{ij} x_{ij} - \sum_{i \in \{1, m\} : j \in \{1, r\}} \omega_{k,m} p_{k,m} \right) - \sum_{i \in k} E_i - \gamma \]  

\[(22)\]

Seven initial columns are thus generated for each aircraft. In other words, the RMP begins with 7k routes.

The heuristic for solving the subproblems is described in detail in what follows.

3.2. Description of k-SP heuristic

The k-SP heuristic proceeds in two major sequential stages: initialization and local search. In the initialization stage, an initial feasible route is created for each subproblem. This route is then subjected to various modifications in the local search stage by different operators whose probability of being chosen for the task is updated over time. The local search stage terminates when either the maximum number of iterations \(\Omega\) is reached or all of the operators have been executed with no further improvements in the objective function.

A simplified flow diagram of the heuristic algorithm is shown in Fig. 3. Below we explain in detail the two stages, the operators used in the local search stage and the updating of each operator’s probability of being chosen.

3.2.1. Initialization

In the initialization stage, the heuristic randomly chooses an initial route for each k-SP from among the feasible routes for the corresponding aircraft \(k\) inserted up to that point into the RMP (i.e., the RMP columns) and recalculates the chosen route’s profit as given by the objective function for k-SP (Eq. (22)). Thus, as the heuristic advances through the iterations, the set of candidate routes for the initial solution increases in number.

3.2.2. Local search

In the local search stage, certain routes are modified without taking account of the existence of routes created for other aircraft up to that point. Interaction with other subproblems takes place in the process of solving RMP, which supplies the dual values for the subproblem’s objective function.

Seven different operators are utilized to modify the initial route (column) added to the RMP. To be accepted, a modification must improve the level of profit as given by the OF. Four of the operators change the positions of orders on a single route, the fifth exchanges an order with an unassigned one and the remaining two change the number of orders either by eliminating one already on the route or inserting a new one. This is explained in greater detail below in Section 3.2.2.1.

The choice of operators is made randomly on the basis of their efficiency. For each operator \(h\), we initialize the efficiency \(E_h^1\) = 1. If operator \(h\) is used at a certain iteration and the objective function improves, then \(E_h^{new} = 2 \times E_h^1\), while \(E_h^{new} = E_h^1 - 1\) in other case. Finally, the probability of choosing an operator \(h\) is defined as

\[P_h = \frac{E_h^1}{\sum_{i \in H} E_i^1} \quad \forall h \in \{1, \ldots, H\}\]  

\[(24)\]

This formulation increases the probability of choosing operators that have achieved high levels of efficiency in improving the OF value.

3.2.2.1. Operators. The operators are described in detail below.

**Intra-route operators:**

These four order operators are the intra-route operators used in Cherkesly et al. (2015). The *intra-route request exchange operator*
modifies the positions of two orders \( i \) and \( j \) on a given route by exchanging pickup position \( i' \) with \( j' \) and delivery position \( i \) with \( j \).

*Intra-route request relocate* attempts to relocate order \( i \) to its best position on the route. The remaining two operators, *intra-route multiple request exchange* and *intra-route multiple request relocate*, exchange and reposition truncated routes, i.e., subroutes of an order. The truncated route for an order \( i \) is the subset of a route whose initial and final nodes are that order’s pickup and delivery nodes. It also satisfies the conditions that the aircraft departs the initial node with no other cargo than that for order \( i \) and departs the final node with no cargo at all.

*Swap:* The swap operator, based on the swap routed and covered customers operator developed by Sharif and Salari (2015), exchanges an order belonging to route \( k \) with another order belonging to the list \( L_p \) of pending orders. These latter are orders that have not yet been incorporated into a route, and are ordered from higher to lower unit profit. For an order \( i, \) unit profit is given by \( g_i = \phi_i - \pi_i \), where \( \phi_i \) is the dual of the cost derived from Eq. (19) in the RMP problem. For a given route \( k \), this operator proceeds through the entire list \( L_p \) searching, for each order \( i \in L_p \), its best exchange option with order \( j \) on route \( k \), where “best” means the option that most improves total route profit. Note that in any swap of orders between \( k \) and \( L_p \), neither of the two change in size.

*Drop and Add:* This operator, also proposed by Sharif and Salari (2015), increases the length of route \( k \) by inserting orders into route \( k \) from the pending order list \( L_p \). The operator inserts each order \( i \in L_p \) in its best position, where “best” means the position that most improves total route profit.

*Delete:* This operator proceeds through each route \( k \), eliminating orders one by one to test whether doing so improves the route’s total profit.

Note that the operators described above represent cargo assignments under policy P2 since they only modify the sequence of nodes that are visited, but they do not optimize the nodes’ cargo assignments.

### 4. Computational experiments

In this section we describe the computational experiments we carried out to test the proposed matheuristic under the two cargo assignment policies. We start describing the benchmark used to compare our results, then introduce the experimental design and ends this section by analysing the results.

#### 4.1. Benchmarks

To test the performance of the proposed matheuristic under the two cargo assignment policies, we compared its results with two benchmarks:

i) **MIP**, the solution of the optimization model (1)–(17) using Gurobi’s MIP solver;

ii) **FIX Route**, the solution of the model with the routes fixed as in the base plan. The latter benchmark in effect measures the impact of disruptions on the base plan when only the routing and amount of each order to be carried can be optimized.

#### 4.2. Experimental design

We study the performance of our proposed matheuristic and the two benchmarks in two sets of experiments. The first experiment is inspired by the real operation of an airline. The second experiment is presented to test our model in a broader set of instances.

##### 4.2.1. Base experiment

Our base experiment involves a recovery horizon of 3 days. In the network under study operates a fleet of 4 Boeing 777, with capacity of 100 tons each, and an operating cost of 10,000 US $/hr. The fleet size considered is consistent to the freighter fleet of carriers as KLM and Singapore airlines (Royal Dutch Airlines, 2018; Singapore Airlines, 2018). Each aircraft is available to fly across the network, i.e., there are no flight restrictions. The initial and final locations for each aircraft, obtained from the base plan, are presented in Table 1.

The base plan is obtained from data provided by an airline that has preferred to remain anonymous. Due to the lack of detailed information for the base plan, we assumed that each order travels between the origin and destination of a flight segment. This means that each flight segment of the base plan only carries one order. In this way, the base plan has 32 flights in total that visit 15 airports, shown in blue in Fig. 4. The average flight time in the network is 5.3 h, with a range between 45 min and 14 h. To create the disruption scenario, we added 15 new orders (47 in total), equivalent to a 30% increase in the total demand of the system. These new orders request the incorporation of an additional airport to those visited in the base plan: Montevideo (MVD), shown in yellow in Fig. 4.

Based on the information provided by the airline about loads

**Table 1** Aircraft fleet characteristics for the base plan.

| Aircraft | Origin | Destination |
|----------|--------|-------------|
| 1        | BSB    | VCP         |
| 2        | ASU    | MIA         |
| 3        | VCP    | MIA         |
| 4        | MIA    | MIA         |
transported on itineraries of different weeks, we create a histogram with 20 buckets (ranges) of possible weight for each order and built the corresponding cumulative distribution function. Thus, to generate the weight of each order, first, a random variable between 0 and 1 is obtained, which represents the probability of selecting a specific bucket. After having chosen the bucket, another uniform random variable is used between the minimum and maximum value of that bucket to establish the weight associated with that order. A time window of 12 h was defined by both pick-up and delivery.

We test our base experiment with 4 alternative penalty values \(\omega_{\text{pen}} = 2500, 5000, 10,000, 20,000\). These penalty values represent between 5% and 40% of the operating cost of the average flight leg.

4.2.2. Extended experiments

To test our model under a wider range of conditions, we carried out several experiments using a modification of the AA instances in Ropke and Cordeau (2009) for the PDPTW. A total of 10 demand disruption instances were tested, the number of orders varying between 30 and 75. Each instance has a planning horizon of \(T = 600\), a time window width of \(W = 60\) and an aircraft load capacity of \(\text{CAP} = 15\). For each instance a square space is assumed measuring \([0,50] \times [0,50]\) and the initial and final locations are chosen randomly. Thus, the average distance between two different location is approximately \(26\) (0.52-50). The base plan is then drawn up reflecting the initial condition of the orders and the aircraft locations at the start and end of the planning horizon. The instances with disruptions are constructed from the changes to the initial condition, which may be either of two types: i) a modification in the demand (weight) of the original orders, and ii) the appearance of new orders. The modifications in the original orders are determined by a factor \(\varphi \sim U[0,1]\). If \(\varphi \leq \lambda_1\), the order is unchanged; if \(\lambda_1 < \varphi \leq \lambda_2\), the order changes and if \(\varphi > \lambda_2\), the order disappears, that is, falls to zero (in the wake of the disruption). In the case of a change, the new demand is given by \(D_i' = \zeta \cdot D_i\), where \(\zeta \sim U[z_1, z_2]\).

For each instance, \(\alpha\) new orders are generated where \(\alpha \sim U[0, \eta]\) and \(\eta < 1\). For each new order, both its pickup and delivery locations are also generated according to a uniform distribution over the square space and demand is \(D_i \sim U[0, \text{CAP}]\). To define the time windows for order \(i\), the earliest time \(e_i\) is first set for the pickup node by a uniform distribution on the interval \([0,T-\alpha_i]\), where \(\alpha_i\) is the direct travel time between the pickup and delivery nodes, and the latest time \(l_i\) is then calculated as \(l_i = e_i + W\). The time windows for the delivery node are then given by \(e_i = e_i + 2 \cdot l_i\) and \(l_i = e_i + W\).

Since in the case of a disruption, the cargo airline has the option of not taking an order or taking it only partially if it is not profitable, a separate freight rate \(r_i\) is generated for each one. For this purpose it is assumed that the cost of carrying a complete order \(i\) from its pickup node to its delivery node \(c_{i,j}\) is a percentage \(\theta\) of the total customer freight charge \(F_i = \frac{c_i \cdot t_i}{\theta}\), where \(\theta \sim U[0, g_2]\). Thus, the freight rate for \(i\) is given by \(r_i = F_i/D_i\). The values for the parameters \((l_1, l_2, z_2, \eta, g_1, g_2)\) in our tests were set at \((0,3,0.85,0.7,1,0,2,0.05,0.2)\).

Note that even if an order disappears (demand falls to zero), it must still be taken into account given that it is part of the base plan. This may occur in situations where the flight crew of the aircraft go off duty at a
specific airport or the aircraft must undergo maintenance. Finally, the size of the fleet of aircraft was set at 7.

A summary of the 10 instances is given in Table 2, showing for each one the numbers of original, new, disappeared and modified orders.

In this experiment, we consider penalty values that are higher in percentage terms than the one used in our base experiment. In particular, each of the 10 instances was tested with 3 alternative penalty values \( (\alpha_{\text{penalty}}=10, 15, 20) \) representing between 38% and 77% of the operating cost of the average flight leg. Thus, for the 10 instances there were 30 test cases.

### 4.3. Computational results

The proposed matheuristic algorithm was coded in Python 2.7, and Gurobi 8.0 was used to solve the optimization models (RMP) at each iteration. The different instances were run on a computer with an Intel Core i5 2.6 GHz processor and 8 GB of RAM.

For the proposed matheuristic, given the random nature of the choice of initial columns and of operators in the local search stage of the heuristic, it is highly likely that successive runs of the same instance will produce different results. Also, in a certain period of time – say, 1 h – the heuristic may be executed more than once. A stopping criterion was therefore imposed, consisting of whichever of the following three conditions is satisfied first: i) \( G_n \), the maximum number of global runs, reaches 10; ii) \( \Delta \), the number of global iterations with no improvement in the objective function value, reaches 3; iii) \( \tau \), the maximum run time, reaches 3600 s. We also set a solution time limit of 1 h for both benchmarks.

#### 4.3.1. Base experiment

Table 3 shows the results in terms of revenue, operating and cancellation costs for the proposed matheuristic under the two cargo assignment policies (P1 and P2) and the two benchmarks under different penalty values. The table also shows the percentage increase in profit compared with the MIP. From the table, we first observe that independent of the penalty for flight cancelation, any of the three recovery strategies (MIP, P1 and P2) achieve total profits that are more than three times higher than those obtained by the FIX Route where the flight legs executed are the same as in the base plan. If we compare between recovery strategies, both P1 and P2 achieve higher profits than MIP with the exception of P2 when the penalty for flight cancelation equals 20,000. The matheuristic under cargo assignment policy P1 is the one that achieves the higher profits, with an average increase compared with the MIP of a 26%-4%, while under P2 this profit increase reach 19.3%. The extra profits obtained by P1 can be explained by the added flexibility of allowing to transport partial orders, where the policy can choose to make a mix of non-complete orders that maximize the profit obtained.

In terms of the number of orders transported, we observe from Table 4, on the one hand, that the FIX Route is the strategy that carries fewer orders due to its inability to adapt its routes to the new disrupted scenario, resulting in many orders not served. On the other hand, the MIP benchmark is the strategy that transports a greater number of orders. However, near 17% of all the orders transported under the MIP benchmark correspond to partial orders compared to an average of 5% for P1. Transporting more orders does not always traduce to higher revenues when compared with the two proposed policies. For example, under penalty factor 20,000 the MIP strategy transport a total of 41 orders (35 complete and 6 partials) while P1 40 and P2 38, but the revenue obtained by these two proposed strategies are 14% and 5% higher than that achieved by the MIP. As expected, P1 transport the same or more orders than P2 due to the flexibility of transporting partial orders. Note than in general, the acceptance of partial orders under P1 rarely occurs.

In terms of computation time, both P1 and P2 are orders of magnitude faster than the MIP, with no instance taking more than 1 min for both proposed strategies. Moreover, under P2, the average solution time is less than half a second, showing the potential of the proposed matheuristic in helping practitioners by delivering recommendations in a short time. The longer solution times presented by P1 are explained because during the matheuristic, every time the capacity of the aircraft is exceeded, an optimization model must be resolved, which, despite being fast to solve, can be called several times during the process.

These results demonstrate that allowing flexibility in carrying partial orders allows for incremental profits for an airline. However, these extra profits come at a cost: solving a more complex model during a disruption, with longer computational times and the negative impact it can have on the relationship with the client.

### Table 2

Characteristics of instances in model tests.

| Original | New | Disappeared | Modified (weight) |
|----------|-----|-------------|------------------|
| AA30     | 30  | 5           | 4                | 12               |
| AA35     | 35  | 6           | 2                | 18               |
| AA40     | 40  | 5           | 8                | 13               |
| AA45     | 45  | 2           | 9                | 20               |
| AA50     | 50  | 2           | 7                | 24               |
| AA55     | 55  | 3           | 7                | 26               |
| AA60     | 60  | 9           | 8                | 25               |
| AA65     | 65  | 11          | 10               | 26               |
| AA70     | 70  | 3           | 9                | 26               |
| AA75     | 75  | 3           | 9                | 30               |

#### Table 3

Comparison of objective function values for the different strategies and penalties.

| Penalty | Revenue | Operating Cost | Cancelation Cost | Total Profit | %increase |
|---------|---------|----------------|------------------|--------------|-----------|
| 2500    | 1,725,927 | 1,509,284 | 1,128,610 | 3,543,821 | –78.03%   |
| 5000    | 1,568,407 | 1,509,284 | 1,128,610 | 3,206,301 | –72.07%   |
| 10,000  | 1,856,814 | 1,404,800 | 1,404,040 | 5,705,658 | –71.54%   |

#### Table 4

Number of orders transported for the different strategies and penalties.

| orders (complete-partial) | 2500 | 5000 | 10,000 | 20,000 |
|---------------------------|------|------|--------|--------|
| MIP | 31.8 | 31.6 | 37.7   | 35.6   |
| FIX | 30.2 | 30.2 | 30.2   | 30.2   |
| P1  | 35.1 | 34.4 | 37.1   | 39.1   |
| P2  | 35.0 | 38.0 | 38.0   | 38.0   |

For the proposed matheuristic, given the random nature of the choice of initial columns and of operators in the local search stage of the heuristic, it is highly likely that successive runs of the same instance will produce different results. Also, in a certain period of time – say, 1 h – the heuristic may be executed more than once. A stopping criterion was therefore imposed, consisting of whichever of the following three conditions is satisfied first: i) \( G_n \), the maximum number of global runs, reaches 10; ii) \( \Delta \), the number of global iterations with no improvement in the objective function value, reaches 3; iii) \( \tau \), the maximum run time, reaches 3600 s. We also set a solution time limit of 1 h for both benchmarks.

4.3.1. Base experiment

Table 3 shows the results in terms of revenue, operating and cancellation costs for the proposed matheuristic under the two cargo assignment policies (P1 and P2) and the two benchmarks under different penalty values. The table also shows the percentage increase in profit compared with the MIP. From the table, we first observe that independent of the penalty for flight cancelation, any of the three recovery strategies (MIP, P1 and P2) achieve total profits that are more than three times higher than those obtained by the FIX Route where the flight legs executed are the same as in the base plan. If we compare between recovery strategies, both P1 and P2 achieve higher profits than MIP with the exception of P2 when the penalty for flight cancelation equals 20,000. The matheuristic under cargo assignment policy P1 is the one that achieves the higher profits, with an average increase compared with the MIP of a 26%-4%, while under P2 this profit increase reach 19.3%. The extra profits obtained by P1 can be explained by the added flexibility of allowing to transport partial orders, where the policy can choose to make a mix of non-complete orders that maximize the profit obtained.

In terms of the number of orders transported, we observe from Table 4, on the one hand, that the FIX Route is the strategy that carries fewer orders due to its inability to adapt its routes to the new disrupted scenario, resulting in many orders not served. On the other hand, the MIP benchmark is the strategy that transports a greater number of orders. However, near 17% of all the orders transported under the MIP benchmark correspond to partial orders compared to an average of 5% for P1. Transporting more orders does not always traduce to higher revenues when compared with the two proposed policies. For example, under penalty factor 20,000 the MIP strategy transport a total of 41 orders (35 complete and 6 partials) while P1 40 and P2 38, but the revenue obtained by these two proposed strategies are 14% and 5% higher than that achieved by the MIP. As expected, P1 transport the same or more orders than P2 due to the flexibility of transporting partial orders. Note than in general, the acceptance of partial orders under P1 rarely occurs.

In terms of computation time, both P1 and P2 are orders of magnitude faster than the MIP, with no instance taking more than 1 min for both proposed strategies. Moreover, under P2, the average solution time is less than half a second, showing the potential of the proposed matheuristic in helping practitioners by delivering recommendations in a short time. The longer solution times presented by P1 are explained because during the matheuristic, every time the capacity of the aircraft is exceeded, an optimization model must be resolved, which, despite being fast to solve, can be called several times during the process.

These results demonstrate that allowing flexibility in carrying partial orders allows for incremental profits for an airline. However, these extra profits come at a cost: solving a more complex model during a disruption, with longer computational times and the negative impact it can have on the relationship with the client.
4.3.2. Extended experiments

The results for the test cases are set out in Table 5. For each one, the first column lists the type of instance (AA), the number of orders and the penalty value ($\omega = 20, 15, 10$). Columns 2 and 3 give the results for the MIP benchmark, columns 4 and 5 for the FIX Route benchmark, while columns 6 through 8 and 9 through 11 do the same respectively for the proposed matheuristic under policy P1 and P2. In each case, the values reported are the objective function value (i.e., profit) and the solution time. The best solution for each instance appears if the solution found is optimal. Indeed, there were cases such as AA45_ and AA65_ where MIP simply did not identify a feasible integer solution at all within the time limit.

As for solution times, the matheuristic under P2 produces the fastest computation among all the strategies. While the average solution time under P2 reach 12 s, under strategy P1 almost reach 1000 s. As regards the benchmarks, MIP managed to solve to optimality only in four cases within the time limit while FIX Route achieved the best times compared with the MIP simply because it solved a restricted version of the original problem in which cargo was routed across a network with fixed flights.

These test outcomes taken together demonstrate that both of the proposed matheuristic delivered better results than either of the two benchmarks within the imposed time limit. Furthermore, incremental benefits in the order of 3% can be obtained by allowing flexibility in carrying partial orders.

5. Conclusions

We propose a solution approach for the problem of last-minute adjustments to cargo flight schedules made to recover from disruptions in short-run demand. It is built around a reactive model based on the pickup and delivery problem with time windows (PDPTW) that is designed to be used with real-world air cargo operations. The formulation simultaneously solves the flight scheduling, aircraft routing and cargo routing stages while attempting to deviate as little as possible from the base plan through the use of penalties for modifications to flight leg planning.

We develop a matheuristic algorithm to solve the model that combines heuristics with mathematical programming techniques. It employs the column generation method in which subproblems are solved...
heuristically in order to determine the optimal route for each cargo aircraft. Based on this matheuristic algorithm, we study two cargo assignment policies. The P1 policy allows orders to be continuously divisible, thus allowing the cargo for a given order to be transported by various aircraft. The second policy (P2) only permits delivering complete orders if they are accepted.

The performance of the matheuristic was measured on two sets of experiments: The first inspired by the real operation of an airline and the second experiment involving 10 instances with disruption of demand that vary from 30 to 75 cargo orders. Each instance was tested at three different penalty levels for departures from the base flight planning, giving a total of 30 test cases. The results of the matheuristic were compared to those generated by two benchmarks, one being the direct solution of the model using a commercial MIP solver and the other giving a total of 30 test cases. The results of the matheuristic were shown below is executed to redistribute the cargo among the nodes. The method thus has a hybrid function. This sequential approach, first applying the greedy algorithm and then the CR program only if capacity constraints are violated, is much faster than simply applying the LP to every case.

The cargo assignment (CA) method must assign an amount of cargo (\( q_{ijr} \)) to each node \( i \) on each route \( r \) generated by the operators while satisfying the load capacity restrictions (8)–(11) in the original problem. It does this by greedily assigning each of the nodes its full demand sequentially, in the order in which the aircraft visits them. If this results in the aircraft’s load capacity being exceeded, the cargo reassignment (CR) linear optimization problem shown below is executed to redistribute the cargo among the nodes. The method thus has a hybrid function. This sequential approach, first applying the greedy algorithm and then the CR program only if capacity constraints are violated, is much faster than simply applying the LP to every case.

\[
\text{max } \sum_{i \in P, r} (p_i r_i - s_i) q_{ir} \\
\text{subject to:} \\
q_{ir} = q_{0r} = 0 \\
q_{jm} + q_{jm} = 0 \quad \forall i \in P \\
q_{ir} \geq 0 \quad \forall i \in P \\
q_{ir} \leq D_{ir} \quad \forall i \in P \\
Q_r + q_i \leq Q_{0r} \quad \forall (i, f) \in r \\
Q_r = 0 \\
Q_i \leq \text{CAP} \quad \forall i \in P \\
q_{ir} \in \mathbb{R} \\
Q_i \in \mathbb{R}^+ 
\]

where \( P \) is the set of orders that correspond to the nodes visited on route \( r \). The objective of this problem is to maximize profit on the transport of the set of orders associated with the route. Constraint (A2) prevents any cargo from being assigned to the initial and final nodes on the route associated with the start and end of the planning horizon. Constraint (A3) ensures that the cargo at the pickup and delivery nodes for each order is the same. Constraints (A4) and (A5) impose that the cargo associated with a pickup be non-negative and less than the order’s total demand. Constraints (A6) and (A7) restrict the continuity of the cargo. Constraint (A8) is the aircraft load capacity restriction. Finally, (A9) and (A10) define the nature of the variables.

Appendix A. Cargo assignment model under P1

Under policy P1 the proposed model allows aircraft to pick up part of an order from a node and the nodes can be visited by more than one aircraft. The cargo assignment (CA) method must assign an amount of cargo (\( q_{ijr} \)) to each node \( i \) on each route \( r \) generated by the operators while satisfying the load capacity restrictions (8)–(11) in the original problem. It does this by greedily assigning each of the nodes its full demand sequentially, in the order in which the aircraft visits them. If this results in the aircraft’s load capacity being exceeded, the cargo reassignment (CR) linear optimization problem shown below is executed to redistribute the cargo among the nodes. The method thus has a hybrid function. This sequential approach, first applying the greedy algorithm and then the CR program only if capacity constraints are violated, is much faster than simply applying the LP to every case.

\[
\text{max } \sum_{i \in P, r} (p_i r_i - s_i) q_{ir} \\
\text{subject to:} \\
q_{ir} = q_{0r} = 0 \\
q_{jm} + q_{jm} = 0 \quad \forall i \in P \\
q_{ir} \geq 0 \quad \forall i \in P \\
q_{ir} \leq D_{ir} \quad \forall i \in P \\
Q_r + q_i \leq Q_{0r} \quad \forall (i, f) \in r \\
Q_r = 0 \\
Q_i \leq \text{CAP} \quad \forall i \in P \\
q_{ir} \in \mathbb{R} \\
Q_i \in \mathbb{R}^+ 
\]

where \( P \) is the set of orders that correspond to the nodes visited on route \( r \). The objective of this problem is to maximize profit on the transport of the set of orders associated with the route. Constraint (A2) prevents any cargo from being assigned to the initial and final nodes on the route associated with the start and end of the planning horizon. Constraint (A3) ensures that the cargo at the pickup and delivery nodes for each order is the same. Constraints (A4) and (A5) impose that the cargo associated with a pickup be non-negative and less than the order’s total demand. Constraints (A6) and (A7) restrict the continuity of the cargo. Constraint (A8) is the aircraft load capacity restriction. Finally, (A9) and (A10) define the nature of the variables.

Appendix B. Supplementary data

Supplementary data to this article can be found online at https://doi.org/10.1016/j.jairtraman.2020.101939.
References

Singapore Airlines, 2018. Annual Report FY2017/18.
Barnhart, C., Johnson, E., Nemhauser, G., Savelsbergh, M., Vance, P., 1998. Branch-and-price: column generation for solving huge integer programs. Oper. Res. 46 (3), 316–329.
Barnhart, C., Belobaba, P., Odoni, A., 2003. Applications of operations research in the air transport industry. Transport. Sci. 37 (4), 368–391.
Berge, M., Hopperstad, C., 1993. Demand driven dispatch: a method for dynamic aircraft capacity assignment, models and algorithms. Oper. Res. 41 (1), 153–168.
Boeing, 2014. World Air Cargo Forecast 2014-2015.
Chen, F., Xiao, T., 2009. Demand disruption and coordination of the supply chain with a dominant retailer. Eur. J. Oper. Res. 197 (1), 225–234.
Cherkesly, M., Desaulniers, G., Laporte, G., 2015. A population-based metaheuristic for the pickup and delivery problem with time windows and LIFO loading. Comput. Oper. Res. 62, 23–35.
Cordeau, J.-F., 2006. A branch-and-cut algorithm for the dial-a-ride problem. Oper. Res. 54 (3), 573–586.
Cordeau, J.-F., Desaulniers, G., Desroijers, J., Solomon, M.M., Soumis, F., 2002. VRP with Time Windows. In: Toth, P., Vigo, D. (Eds.), The Vehicle Routing Problem, SIAM Monographs on Discrete Mathematics and Applications. SIAM, Philadelphia, pp. 157–193.
Dantzig, G., Wolfe, P., 1960. Decomposition principle for linear programs. Oper. Res. 8 (1), 101–111.
Delgado, F., Sirhan, C., Katscher, M., Larrain, H., 2020. Recovering from demand disruptions on an air cargo network. J. Air Transport. Manag. 85, 101779.
Derigs, U., Friederichs, S., 2013. Air cargo scheduling: integrated models and solution procedures. Spectrum 35 (2), 325–362.
Derigs, U., Friederichs, S., Schäfer, S., 2009. A new approach for air cargo network planning. Transport. Sci. 43 (3), 370–380.
Desroijers, J., Lübbecke, M.E., 2005. In: Ira (Ed.), Column Generation. Springer, New York.
Dumas, Y., Desroijers, J., Soumis, F., 1991. The pickup and delivery problem with time windows. Eur. J. Oper. Res. 54, 7–22.
Ehmon, M., Mathaisel, D., 1985. Airline scheduling: an overview. Transport. Sci. 19 (2), 127–138.
Feillet, D., 2010. A tutorial on column generation and branch-and-price for vehicle routing problems. 4OR 8 (4), 407–424.
Feng, B., Li, Y., Shen, Z.J.M., 2015. Air cargo operations: literature review and comparison with practices. Transport. Res. C Emerg. Technol. 56, 263–280.
Foyaland, G., Maher, S., Wu, C.L., 2014. The recoverable robust tail assignment problem. Transport. Sci. 48 (3), 351–372.
Gao, C., Johnson, E., Smith, R., 2009. Integrated airline fleet and crew robust planning. Transport. Sci. 43 (1), 2–16.
Hu, Y., Song, Y., Zhao, K., Xu, B., 2016. Integrated recovery of aircraft and passengers after airline operation disruption based on a GRASP algorithm. Transport. Res. E Logist. Transportation. Rev. 87, 97–112.
IATA, 2018. Cargo eChartbook Q1.
Jamili, A., 2017. A robust mathematical model and heuristic algorithms for integrated aircraft routing and scheduling, with consideration of fleet assignment problem. J. Air Transport. Manag. 58, 21–30.
Jozefowiez, N., Mancel, C., Mora-Camino, F., 2013. A heuristic approach based on shortest path problems for integrated flight, aircraft, and passenger rescheduling under disruptions. J. Oper. Res. Soc. 64 (3), 384–395.
Labadie, N., Prins, C., Prodhon, C., 2016. Metaheuristics for Vehicle Routing Problems. John Wiley & Sons.
Lan, S., Clarke, J.-P., Barnhart, C., 2006. Planning for robust airline operations: optimizing aircraft routings and flight departure times to minimize passenger disruptions. Transport. Sci. 40 (1), 15–28.
Levin, A., 1971. Scheduling and fleet routing models for transportation systems. Transport. Sci. 5 (3), 232–255.
Lin, C., Chen, Y., 2003. The integration of Taiwanese and Chinese air networks for direct air cargo services. Transport. Res. Pol. Pract. 37 (7), 629–647.
Lohataganont, M., Barnhart, C., 2004. Airline schedule planning: integrated models and algorithms for schedule design and fleet assignment. Transport. Sci. 38 (1), 19–32.
Maher, S.J., 2015. Solving the integrated airline recovery problem using column-and-row generation. Transport. Sci. 50 (1), 216–239.
Mathematics – Hybridizing Metaheuristics and Mathematical Programming. Annals of Information Systems. In: Maniezzo, V., Stützle, T., Voß, S. (Eds.), 2010, vol. 10. Springer.
Marsten, R., Muller, M., 1980. A mixed-integer programming approach to air cargo fleet planning. Manag. Sci. 26 (11), 1906–1107.
Parragh, S.N., Doerner, K.F., Hartl, R.F., 2008. A survey on pickup and delivery problems. J. Betriebswirtschaft 58 (1), 21–51.
Pita, J., Barnhart, C., Antunes, A., 2013. Integrated flight scheduling and fleet assignment under airport congestion. Transport. Sci. 47 (4), 477–492.
Qi, X., Bard, J.F., Yu, G., 2004. Supply chain coordination with demand disruptions. Omega 32 (4), 301–312.
Qi, Y., Bard, J., 2012. A GRASP with adaptive large neighborhood search for pickup and delivery problems with transshipment. Comput. Oper. Res. 39 (10), 2439–2456.
Rexing, B., Barnhart, C., Kniker, T., Jarrah, A., Krishnamurthi, N., 2000. Airline fleet assignment with time windows. Transport. Sci. 34 (1), 1–20.
Rople, S., Cordeau, J.-F., 2009. Branch and cut and price for the pickup and delivery problem with time windows. Transport. Sci. 43 (3), 267–286.
Rople, S., Pisinger, D., 2006. An adaptive large neighborhood search heuristic for the pickup and delivery problem with time windows. Transport. Sci. 40 (4), 455–472.
Rople, S., Cordeau, J.-F., Laporte, G., 2007. Models and branch-and-cut algorithms for pickup and delivery problems with time windows. Networks 49 (4), 258–272.
Rosenberger, J., Johnson, E., Nemhauser, G., 2003. Rerouting aircraft for airline recovery. Transport. Sci. 37 (4), 408–421.
Rosenberger, J., Johnson, E., Nemhauser, G., 2004. A robust fleet-assignment model with hub isolation and short cycles. Transport. Sci. 38 (3), 357–368.
Royal Dutch Airlines, 2018. Annual Report 2018.
Sandhu, R., Klabjan, D., 2006. Fleeting with passenger and cargo origin-destination booking control. Transport. Sci. 40 (4), 517–529.
Savelbergh, M., Sol, M., 1995. General pickup and delivery problem. Transport. Sci. 29 (1), 17.
Sharif, M.T., Salami, M., 2015. A GRASP algorithm for a humanitarian relief transportation problem. Eng. Appl. Artif. Intell. 41, 259–269.
Sigurd, M., Pisinger, D., Sig, M., 2004. Scheduling transportation of live animals to avoid the spread of diseases. Transport. Sci. 38, 197–209.
Tang, C.-H., Yan, S., Chen, Y.-H., 2008. An integrated model and solution algorithms for passenger, cargo, and combi flight scheduling. Transport. Res. E Logist. Transportation. Rev. 44 (6), 1004–1024.
Teodorovic, D., Guberinic, S., 1984. Optimal dispatching strategy on an airline network after a schedule perturbation. Eur. J. Oper. Res. 15 (2), 178–182.
Vos, H.W.M., Santos, B.F., Omondi, T., 2015. Aircraft schedule recovery problem—a dynamic modeling framework for daily operations. Trans. Res. Proc. 10, 931–940.
Wada, M., Delgado, F., Pagnoncelli, B., 2017. A risk averse approach to the capacity allocation problem in the airline cargo industry. J. Oper. Res. Soc. 68 (6), 643–651.
Wolsey, L.A., 1998. In: Ira (Ed.), Integer Programming, Wiley, New York.
Xiao, T., Qi, X., Yu, G., 2007. Coordination of supply chain after demand disruptions when retailers compete. Int. J. Prod. Econ. 109 (1–2), 162–179.
Yan, S., Chen, C.-H., 2008. Optimal flight scheduling models for cargo airlines under alliances. J. Sched. 11 (3), 175–186.
Yan, S., Chen, S., Chen, C.-H., 2006. Air cargo fleet routing and timetable setting with multiple on-time demands. Transport. Res. E Logist. Transportation. Rev. 42 (5), 409–430.
Zhang, D., Lai, H.I., Yu, C., 2015. A two stage heuristic algorithm for the integrated aircraft and crew schedule recovery problems. Comput. Ind. Eng. 87, 436–453.
Zheng, D., Yu, C., Desai, J., Lai, H.H., 2016. A math-heuristic algorithm for the integrated air service recovery. Transp. Res. Part B Methodol. 84, 211–236.

Journal of Air Transport Management 90 (2021) 101939