Optimization of multi-fleet aircraft routing considering passenger transiting under airline disruption

Article *in* Computers & Industrial Engineering · February 2015

DOI: 10.1016/j.cie.2014.11.026

Yuzhen Hu
Harbin Engineering University

Xu Baoguang
Chinese Academy of Sciences

Jonathan Bard
University of Texas at Austin

Hong Chi

Computers & Industrial Engineering 80 (2015) 132–144 February

Contents lists available at ScienceDirect

Computers & Industrial Engineering

journal homepage: www.elsevier.com/locate/caie

Optimization of multi-fleet aircraft routing considering passenger transiting under airline disruption ☆

Yuzhen Hu [a,c], Baoguang Xu [a,*], Jonathan F. Bard [b], Hong Chi [a], Min'gang Gao [a]

[a] Institute of Policy and Management, Chinese Academy of Sciences, Beijing 100190, China
[b] Graduate Program in Operations Research, University of Texas, Austin, TX 78712-1063, USA
[c] School of Economics and Management, Harbin Engineering University, Heilongjiang Harbin 150001, China

ARTICLE INFO

Article history:
Received 22 April 2014
Received in revised form 12 September 2014
Accepted 30 November 2014
Available online 5 December 2014

Keywords:
Airline disruption
Multi-fleet aircraft routing
Reduced time-band network
Passenger transiting

ABSTRACT

This paper proposes a new methodology for addressing the joint problems of aircraft and passenger recovery after a schedule disruption. An integrated integer programming model is presented which is based on an approximate reduced time-band network and a passenger transiting relationship. The objective is to minimize the total cost associated with reassigning aircraft and passengers to flights. A feasibility analysis for the problem is conducted to obtain the necessary conditions under which aircraft and passenger recovery is possible. Solutions are obtained to the network model with CPLEX and then, if necessary, adjusted to more accurately reflect actual costs. The effectiveness of the proposed approach is demonstrated by analyzing several scenarios that were developed using data from a big airline in China.

1. Introduction

In commercial aviation, daily uncertainties such as severe weather, crew reassignments, and maintenance problems can lead to the direct or indirect failure of regular operational schedules. These failures can be defined as disruptions, and if they are not addressed in a timely and appropriate manner, they will affect crew connections and passenger itineraries, and may result in significant damage to the airline's profitability and image. Therefore, developing plans to deal with irregular situations is essential for an airline to remain competitive.

When disruptions occur, the dispatcher will typically reschedule a number of aircraft taking flight plans, aircraft routings, maintenance schedules, crew status and other pertinent information into account. Rescheduling options include ferrying, diverting, over-flying and swapping aircraft. These adjustments must satisfy maintenance requirements, station departure curfew restrictions and aircraft balance requirements, especially at the beginning and end of a recovery period. In most cases, it is critical for the flight schedule to get back on track as soon as possible, which may be in a few hours or the end of the day at the latest. Common methods for addressing flight rescheduling include flight delays and cancellations. Either action disrupts the flow of passengers and in extreme cases may require overnight stays and complimentary meals. The corresponding costs are often significant and, for the most part, overlooked by the airlines when reacting to disruptions.

Due to the complexity of irregular operations and the cost consequences, disruption management in the airline industry has been a topic of concern for an increasing number of researchers; for example, see Etschmaier and Mathaisel (1985), Filar, Manyem, and White (2001), Kohl, Larsen, Larsen, Ross, and Tiourine (2007), Ball, Barnhart, Nemhauser, and Odoni (2007), Clause, Larsen, Larsen, and Rezanova (2010), Le, Wu, Zhan, and Sun (2011), and Castro and Oliveira (2011). These papers present the initial concepts and framework for airline disruption management along with the relevant models and algorithms for dealing with aircraft, crew, passengers, and other resources. An overview of the most common methods used to solve disruption management problems experienced by commercial airlines is given by Artigues, Bourreau, Afsar, Briant, and Boudia (2012).

When rescheduling flights, passenger itineraries are inevitably affected. Flight delays and especially flight cancellations are an unwelcome inconvenience to be minimized. Perhaps more damaging is the fact that the reputation and profit of the airline will unavoidably suffer from these occurrences. During the recovery process, there is a close connection between passenger recovery and aircraft recovery; therefore, it is critical to integrate the two into a single model. In this paper, our primary goal is to solve the multi-fleet aircraft recovery and passenger recovery problem (ARPRP) by considering delay costs, flight cancellations, and the

☆ Supported by the project of the major research task, Institute of Policy and Management, Chinese Academy of Sciences.
* Corresponding author. Tel.: +86 1059358817; fax: +86 1059358608.
E-mail address: xbg@mail.casipm.ac.cn (B. Xu).

http://dx.doi.org/10.1016/j.cie.2014.11.026
0360-8352/

reassignment of passengers to new itineraries in a single model. To achieve this goal, it was not only necessary to develop an extended formulation for the problem but also to analyse the problem's complexity and to determine whether or not solutions exist. The foundation for our work is a reduced time-band network and a passenger transiting relationship model. The former is designed to balance the cost of flight rescheduling, passenger delay and transiting. Two contributions stand out. First, we provide an efficient and computationally manageable way of integrating the aircraft recovery and passenger recovery problems in the form of a reduced time-band network that prevents the repetition of a large number of redundant flight arcs. Second, a necessary condition is given for the existence of feasible solutions to the network model for a given set of practical recovery options.

This paper is organized as follows. In the next section, we review the literature. This is followed in Section 3 by a formal description of the problem, including a statement of assumptions and constraints. The reduced time-band network, the passenger transiting relationship, and the full mathematical model are presented in Section 4. Section 5 contains the network model feasibility analysis and Section 6 presents the computational results. Conclusions are drawn in Section 7.

2. Literature review

The study of airline schedule recovery began in the 1980s. Teodorovic and Guberinic (1984) solved an eight-flight problem with the objective of minimizing total passenger delay. Gershkoff (1987) presented a successive shortest path method that provided a set of flight cancellations when aircraft shortages reduced airline capacity. However, he discounted the possibility of using idle aircraft for flight delays. Jarrah and Yu (1993) proposed two minimum cost network flow models for flight delays and cancellations, respectively, but they did not implement an integrated model.

Until the 1990s, the field was limited to small-scale aircraft recovery for a single fleet at a time. Yan and Yang (1996) were the first to combine flight delays, cancellations, and ferrying in a single model and to solve the problem of irregular operations using a time–space network model. Their work was extended by Yan and Lin (1997) to address airport closures for instances with up to 39 flights. Yan and Tu (1997) used the same model to address multiple fleet substitutions for a network of 24 cities, 7 fleets, and 273 flights. Thengvall, Bard, and Yu (2000) introduced the concept of a protection arc to the time–space network to reduce passenger dissatisfaction caused by reassignment to multiple flights. In a follow-up study, Thengvall, Yu, and Bard (2001) integrated flight delays, cancellations, peculiar deviations, aircraft ferrying, fleet substitution, and airport closures in a model that was considered to be the most comprehensive at that time.

A time-band is a chart that contains two dimensions that represent time intervals and airport stations respectively. Bard, Yu, and Argüello (2001) advanced the idea of a time-band network and solved the single-fleet aircraft recovery problem using this construct. Argüello, Bard, and Yu (1997) presented a greedy randomized adaptive search procedure (GRASP) to reconstruct aircraft routes in response to groundings and delays. Performance was evaluated by comparing the GRASP results with those obtained with a lower bounding optimization-based time-band network model. Their method was tested on B757 fleet data from Continental Airlines with 16 aircraft and 42 flights.

Eggenberg, Bierlaire, and Salani (2010) presented a column generation algorithm to solve the aircraft recovery problem with maintenance planning. In their paper, a time-band recovery network was constructed for each aircraft, which facilitated the inclusion of maintenance constraints through the introduction of a maintenance arc. Data from a real instance with a maximum of 10 aircraft and a maximum recovery period of 7 days produced scenarios with up to 250 flights. The running times suggested that the method is capable of recovering the proposed disruption scenarios. Hu, Xu, Gao, and Chi (2011) established an integer programming model based on a time-band network and a passenger transiting network for the combined aircraft and passenger recovery problem. Solutions for single-fleet aircraft recovery were obtained by first solving the linear programming relaxation and then applying a rounding heuristic. A data set consisting of 16 aircraft and 70 flights was used to test the approach.

In recent years, a number of heuristics and metaheuristics have been developed to find solutions to the recovery problem. Babić, Kalić, Pavković, Dožić, and Čangalović (2010) presented a mathematical formulation for the airline schedule disturbance problem and developed a heuristic for generating new daily operational flight schedules. In their algorithm, priority flights are defined and are taken into account by adding an unplanned flight followed by the priority flights. The idea is to offer a list of different feasible solutions that are ordered according to the value of an objective function. The dispatcher can then select and implement one of the solutions.

Optimization methods based on local search operating on a network model are presented by Løve, Sørensen, Larsen, and Clausen (2005) following from the work of Løve and Sørensen (2001). In the network, aircraft and flights nodes are defined separately. The assignment of an aircraft to a given flight is performed by selecting an edge that connects the two nodes. Using this approach, the existing solution is altered by interchanging flight nodes between two aircraft. The data used in Løve et al. (2005) were randomly generated.

Liu, Chen, and Chou (2010) employed a hybrid multi-objective genetic algorithm to find solutions for a daily short-haul aircraft schedule recovery problem. Their method only considered a single fleet with 7 aircraft and 39 flights, and without cancellations or ferrying. Ryerson and Churchill (2013) described a deterministic routing model for airport outages. In the model, flight arrival times are reassigned at the diversion airports to avoid fuel emergencies. Testing showed that under suitable assumptions, all flights could be diverted before reaching a fuel-critical state within 20 min after a disaster occurs.

Integrating recovery of several resources is a difficult task due to the size of the resultant problem, and only a few attempts have appeared in the literature. Lettovsky (1997) was one of the first to take an approach that integrated three resources: aircraft, crews and passengers. However, only a portion of his model was implemented. With respect to passenger recovery, Bratu and Barnhart (2006) proposed two models, abbreviated DPM and PDM, which are based on the time-band network developed by Bard et al. (2001). In the first, DPM, only approximate delay costs are considered along with passenger disruption costs, whereas in PDM, delay costs are more accurately computed by explicitly modeling passenger disruptions and recovery options. Their test data set included 4 aircraft types with a total of 302 aircraft, 74 airports, and three hubs, as well as 83,869 passengers following 9925 different itineraries per day. Three scenarios each with different levels of disruptions were analysed with DPM only since PDM was not suitable for operational use.

Zhang and Hansen (2008) introduced ground transportation modes as an alternative to passenger recovery by air during disruptions in hub-and-spoke networks. An integer programming model was formulated with a nonlinear objective function aimed at minimizing passenger costs due to delay, cancellation or substitution, plus operating costs for transportation. A numerical example for a 4-h recovery period with 40 flights and 7360 passengers was the only scenario investigated. After transportation substitution, the results indicated a huge decrease in cost as the number of disrupted passengers went from a high of 90 down to a low of 14. Jafari and Zegordi (2010) presented a model that simultaneously recovered aircraft and passengers. Integrated with the model was the length of the recovery period and the use of aircraft rotations and passenger itineraries. Their data set only contained 13 aircraft that were divided into 2 fleets, 100 flights, and 2236 passengers with 8 itineraries and 55 different connections. The results were compared with those obtained by Andersson and Varbrand (2004) who only focused on aircraft recovery. Neither set of authors, however, demonstrated that their method was computational efficient, nor were they able to deal with disruptions that reflected the operations of more than a small scale airline.

Bisaillon, Cordeau, Laporte, and Pasin (2011) developed a large neighborhood search heuristic for an airline recovery problem combining aircraft routing and passenger reassignment with the objective of minimizing operating costs and impacts on passengers. The core components of the heuristic were a construction phase, a repair phrase and an improvement phase. In each iteration, priority is given to constructing locally optimal aircraft route, and only then are passenger itineraries updated based on those routes. The work won the first prize of the ROADEF 2009 Challenge that was organized by Artigues et al. (2012). Sinclair, Cordeau, and Laporte (2014) improved the heuristic by adding some additional steps in each phase. Their algorithm found 17 best solutions for the 22 instances within a 10-min time limit.

Arıkan, Sinan Gürel, and Aktürk (2013) derived a mathematical model for the joint problem by superimposing the passenger itinerary network on the aircraft network. Cruise speed control was considered when trying to balance fuel consumption costs and delay propagation. The problem was formulated as a mixed-integer nonlinear program and solved with CPLEX. Several simultaneous disruptions were optimized on a four-hub network of a major U.S. airline in less than a minute on average. Similarly, Chan, Chung, Chow, and Wong (2013) proposed a formulation that integrates the recovery of aircraft and passengers with the objective of minimizing the sum of passenger delay costs and airline operating costs. In their model, the latter included the cost of late arrivals and an inconvenience cost due to transshipment of passengers to other airlines. No solution methods were offered.

3. Problem description

The problem investigated in this paper reflects the practical concerns of an airline when irregularities force its operating schedule to be altered. When an aircraft is grounded due to unforeseen events, its flights as well as the passengers on those flights will be disrupted. To recover the schedule as soon as possible, available aircraft can be rerouted, passengers can be reassigned to other flights, delays can be imposed, and surplus aircraft can be called into service. Several objectives have been proposed to guide the recovery, depending on the data available at the time of the disruption and the priorities of the airline. In our case, the objective is to minimize the cost of compensating passengers for delays, transferring them to other flights, and reimbursing them for their ticket when they cannot be rerouted. The flight delay cost refers to the direct costs such as overtime payment and meal expenses. It is a function of the length of the delay and the number of passengers. The transiting cost is a combination of the service cost and dissatisfaction cost that result when passengers are transferred from their original flight to another. It is related to the difference between the actual departure time of the new flight and the scheduled departure time of the cancelled flight, as well as the costs associated with baggage transfer and rebooking. The cost of refunding a ticket has both a monetary component and a qualitative component associated with the damage to the reputation of the airline due to the disruption.

In defining the problem, the following assumptions and rules are imposed. When a disruption occurs, three recovery options are considered: (i) swapping aircraft (exchanging aircraft that have fewer remaining legs with the grounded aircraft of either the same type or different type), (ii) delays (providing new departure times for the flights according to the estimated available times of the delayed aircraft or the newly assigned aircraft), and (iii) cancellations (cancelling flights and reassigning their passengers). Other recovery options, such as ferrying aircraft (transporting an empty aircraft between airports) or calling reserve aircraft (using spare aircraft), are not considered here because they are rarely if ever available. In practice, ferrying an aircraft is the last choice when a disruption occurs. Also, each flight can be delayed a maximum number of hours before it must be cancelled.

During the recovery period, we do not consider externally imposed changes to an aircraft's original flight plan, which might result, for example, from a change in its maintenance schedule or something similar. In addition, crew requirements are not taken into account so modifications to aircraft routes are not limited by crew availability. When an aircraft routing is selected, it is deemed acceptable as long as appropriate crew members can be found to fly the aircraft. However, we do consider multiple fleet types and the fact that each fleet has different characteristics that limit cross-assignments. In particular, different fleets have different passenger carrying capacities. Substitution is allowed between some fleets in accordance with airline policies and aircraft configurations.

Suitable arrangements for passengers who are affected by flight delays and cancellations present a critical problem. A maximum allowed delay time is defined for each flight. As a rule, all passengers should travel on their original flights if the flights are not cancelled; otherwise, they should be reassigned as best as possible. It is assumed that all passengers accept the arrangements made by the airline. Three options exist for accommodating passengers whose flights are cancelled: the first is to reassign them to flights associated with the same airline and with the same destination as their original flight; the second is to negotiate with other carriers to allow them to transfer to flights also with the same destination as the cancelled one; the third is to return the ticket or provide them with overnight accommodations and an itinerary for the next day. Because the problem we address is limited to a single airline, the second and third options are combined in our analysis. Considering the cost of delay and the complexity of transitioning between different flights, the passengers who receive compensation for flight delays do not include those who transfer from other cancelled flights; only passengers who were on the delayed flight are included. The delay cost for the passengers who transfer from other cancelled flights is included in the transiting cost.

The following flow balance and operational constraints must also be taken into account when rerouting aircraft: (i) flights for each aircraft can only be adjusted once or cancelled; (ii) each flight in a new schedule should not depart earlier than its originally scheduled departure time prior to the disruption; (ii) the delay time for each flight should not be greater than its prespecified maximum; (iv) subsequent departures for each aircraft should be scheduled only after the previous flight arrives and only after a minimum turnaround time, that is, aircraft flow should be continuous; and (v) the airport arrival curfew time should be observed.

At the end of the recovery period, enough number of appropriate aircraft types should be positioned at each airport in the network to ensure that the published schedule can be executed. If this is not the case, the disruption will extend into the following day – a situation that we wish to avoid. Finally, when assigning passengers from cancelled flights to other flights, the flights that accept passengers must not depart earlier than when the roll-out flight departs, and the number of passengers on each flight cannot be greater than the seating capacity of the aircraft assigned to this flight.

4. Network model

In this section, we develop a reduced time-band network and a passenger transiting relationship to represent the ARPRP. The model is an integer program with an objective of minimizing the sum of flight delay costs, passenger transiting costs, and ticket refund costs.

4.1. Reduced time-band network (RTBN)

RTBN can be depicted by a two-dimensional grid with horizontal and vertical labels denoted by s and t. The s-axis represents airport stations and the t-axis represents time which is divided into intervals of equal length. Three types of nodes and two types of arcs are used in the construction, as described below; nodes are denoted by the pair (s, t), where t refers to the beginning time of the time interval.

- Origin node (s, t_0): represents the initial state of aircraft at the beginning of the recovery period at station s.
- Intermediate node (s, t): represents aggregate activity (landing, turnaround, and perhaps taking off) for a specific time interval at station s.
- Sink node (s, T): represents the end of the recovery period at station s.
- Flight arc $((s, t), (s', t', f))$: links different airport stations in the network and represents scheduled flight or delayed flight f. The arc includes actual block time and minimum turnaround time at station s'. The delay time of the flight arc is denoted by $del_{((s,t),\ (s',\ t'),\ f)}$.
- Ground arc $((s, t), (s, t'))$: links the adjoining nodes at the same station s in the network and corresponds to holding an aircraft at the same station s within the period from t to t'.

In constructing the model, each aircraft fleet type $e \in E$ has a corresponding recovery subnetwork. As such, let $F(e)$ be the subset of flights that can be covered by aircraft in the fleet, P_e be the subset of aircraft that belong to the fleet type e, and let S_e be the subset of stations where the aircraft in fleet e can terminate at the end of the recovery period.

We also define the following symbols.

ON	set of origin nodes in *RTBN*
TN	set of intermediate nodes in *RTBN*
J	set of sink nodes in *RTBN*
$Farc$	set of flight arcs in *RTBN*
$Garc$	set of ground arcs in *RTBN*
N	ordered list of nodes (including origin nodes, intermediate nodes and sink nodes)
$Maxd$	maximum amount of time a flight can be delayed
std_f	scheduled departure time of flight f
dur_f	block time of flight f
dep_f	departure airport of flight f
arr_f	arrival airport of light f
$Tura$	minimal turnaround time
$curf_s$	arrival curfew time in airport station s

We now present the algorithm for generating the *RTBN*.

Step 1. Set $ON = \emptyset$, $TN = \emptyset$, $J = \emptyset$, Farc = $\emptyset$ and $Garc = \emptyset$
Step 2. For $e \in E$ do
(2.1) Set $N = \emptyset$
For $p \in P_e$ do
– Create an origin node (s, t_0) and put $N \leftarrow N \cup \{(s, t_0)\}$, $ON \leftarrow ON \cup \{(s, t_0)\}$
(2.2) For $f \in F(e)$ do
– Create a node (s, t), set $t = std_f$, $s = dep_f$, and put $N \leftarrow N \cup \{(s, t)\}$, $TN \leftarrow TN \cup \{(s, t)\}$
(2.3) Sort N in non-decreasing order of t
(2.4) While $N \neq \emptyset$ do
– Select the first node (s, t) in N and put $N \leftarrow N/(s, t)$
– If $\exists f \in F(e)$ such that $dep_f = s$, $t + dur_f \leqslant curf_s$ and $0 \leqslant t - std_f \leqslant maxd$, then put $s' = arr_f$, $t' = t + dur_f + tura$, $N \leftarrow N \cup \{(s', t')\}$, $TN \leftarrow TN \cup \{(s', t')\}$, and sort N in nondecreasing order of t. Also, put $Farc \leftarrow Farc \cup \{((s, t), (s', t'), f)\}$, $del_{((s,t),\ (s',\ t'),\ f)} = t - std_f$
(2.5) For $s \in S_e$ do
Create a sink node (s, T) and put $J \leftarrow J \cup \{(s, T)\}$
Step 3. Set $Garc = \emptyset$
For all $(s, t) \in ON \cup TN \cup J$ and $(s', t') \in ON \cup TN \cup J$ do
if $s' = s$, $t < t'$ and $\nexists\ (s'', t'') \in ON \cup TN \cup J$ such that $t < t'' < t'$, then put $Garc \leftarrow Garc \cup \{(s,t),\ (s',\ t')\}$
Step 4. For $(s, t) \in TN$ do
If $\nexists\ ((s', t'), (s, t), f) \in Farc$ and $\nexists\ ((s', t'), (s, t)) \in Garc$, then $TN \leftarrow TN/(s, t)$
Step 5. Output ON, TN, J, $Farc$ and $Garc$

At Step 1, all node sets and arc sets are initialized. At Step 2, flight arcs are created for each aircraft fleet. In (2.1), origin nodes are created for each aircraft in the fleet and used to initialize the set N. In (2.2), time-station nodes for all flights in the set $F(e)$ are created. Each node for the flight represents the departure of the flight as originally scheduled. This step allows each flight to depart without delay. The set N is sorted in non-decreasing order of the time t for each node in (2.3). In (2.4), all flight arcs are added to *Farc* iteratively. First, the node (s, t) with the earliest time t is chosen and deleted from N. Then for each flight that departs from s, and arrive before curfew time and its scheduled departure time is earlier than t, if the difference between t and the scheduled departure time of the flight is not greater than the maximal delay time, a flight arc is added to *Farc*. Also, the destination node is added to N and TN. In (2.5), sink nodes are created for each station where aircraft in the fleet are originally scheduled to be at the end of the recovery period.

At Step 3, ground arcs are created between each pair of adjacent nodes associated with the same station. At Step 4, some time-station nodes with no incoming arcs are deleted from TN. For some time-station nodes representing the original flight schedule departure times, if no aircraft are available before those times, there will be no flight arc into the node. If the time corresponding to the node is the earliest time at the station, there will be no ground arc into the node. Therefore, the node is not required and is deleted from the network. At Step 5, all nodes and all arcs are outputted.

The complexity of the generation algorithm is $O(mr\alpha^2\tau)$, where m is the number of fleet types, r is the maximum number of flights that can be covered by any aircraft fleet, α is the number of airport stations, and τ is the number of time intervals. To see this, note that at Step 2, for each fleet e, the generation of origin nodes in (1) requires $O(p(\alpha + \tau))$ time where p is the maximum number of aircraft in any fleet. Steps in (2) can be done in $O(r(\alpha + \tau))$ time.

Sorting N in (3) can be done in $O((r+p)\log(r+p))$ time. For generating flight arcs iteratively in (4), the number of flight arcs is no more than $O(r\alpha\tau)$, and the destination node search can be done in $O(\alpha+\tau)$ time. Placing the node into an ordered list requires $O(\log(\alpha+\tau))$ time. Therefore, the combined effort takes $O(r\alpha\tau(\alpha+\tau+\log(\alpha+\tau)))$ time, which is dominated by $O(r\alpha^2\tau)$, the amount of time to create all flight arcs in (4). Creating sink nodes in (5) can be finished in $O(\alpha^2)$ time. In total, Step 2 requires $O(m(p(\alpha+\tau)+r(\alpha+\tau)+(r+p)\log(r+p)+r\alpha^2\tau+\alpha^2))$ time, or $O(mr\alpha^2\tau)$. At Step 3, ground arcs are all created in $O(\alpha^2\tau^2)$ time. At Step 4, the number of ground arcs is no more than $O(\alpha\tau)$, so deleting the nodes can be done in $O(\alpha\tau(mr(\alpha+\tau)+\alpha\tau))$ time, which is dominated by $O(r\alpha^2\tau)$.

Fig. 1 provides an example of an *RTBN* based on the data in Bard et al. (2001), where the network has 66 flight arcs. There are 34 flight arcs in the figure, which is less than that in the network given in Bard et al. (2001). Note that the point of origin of a flight arc represents the actual departure time, so it prevents repeating a large number of redundant flight arcs, compare to the generation algorithm in Bard et al. (2001). The definition and construction of the *RTBN* is one of the primary contributions of this paper.

4.2. Passenger transiting relationship (PTR)

We now introduce a passenger transiting relationship that takes into account the options that are available to passengers on disrupted flights. The *PTR* is derived from the *RTBN* and is a set whose elements consist of three components f, k and λ; that is, $(f, k, \lambda) \in PTR$, where f is a cancelled flight whose passengers need to be transferred, k is a flight that can accept the passengers, and λ is the flight arc in *Farc* that represents the flight k. The following principle is used to construct *PTR*: passengers can be transferred between flights with the same departure and arrival airports as long as there is no conflict in transit times. For example, in Fig. 2, there are three flights 1831, 1501 and 1519 with the same route from PEK to SHA; they depart at 7:30, 8:30, and 9:30 and arrive at 9:40, 10:40, and 11:40, respectively. Assume that flight 1501 has to be cancelled due to mechanical problems and flights 1831 and 1519 can depart on time. If there are seats available, passengers on flight 1501 can be transferred to flight 1519. If flight 1831 is delayed until 8:30 and flight 1519 departs on time, then flights 1831 and 1519 can both provide seats to passengers on flight 1501. In our model, priority is given to the flight with the lowest transfer cost.

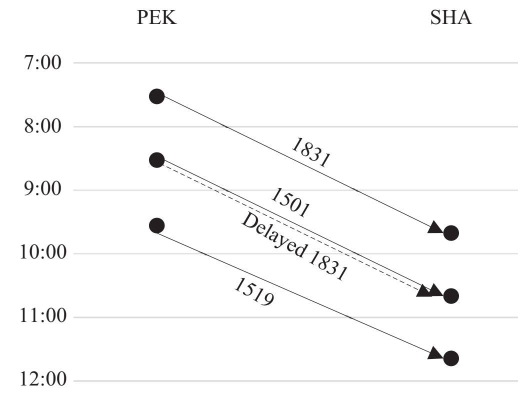

Fig. 2. Example of passenger transiting.

Note that disrupted flights can be covered by aircraft with different capacities. Because the number of surplus seats may be different for transferred passengers, the recovered solution may bear little resemblance to the original schedule.

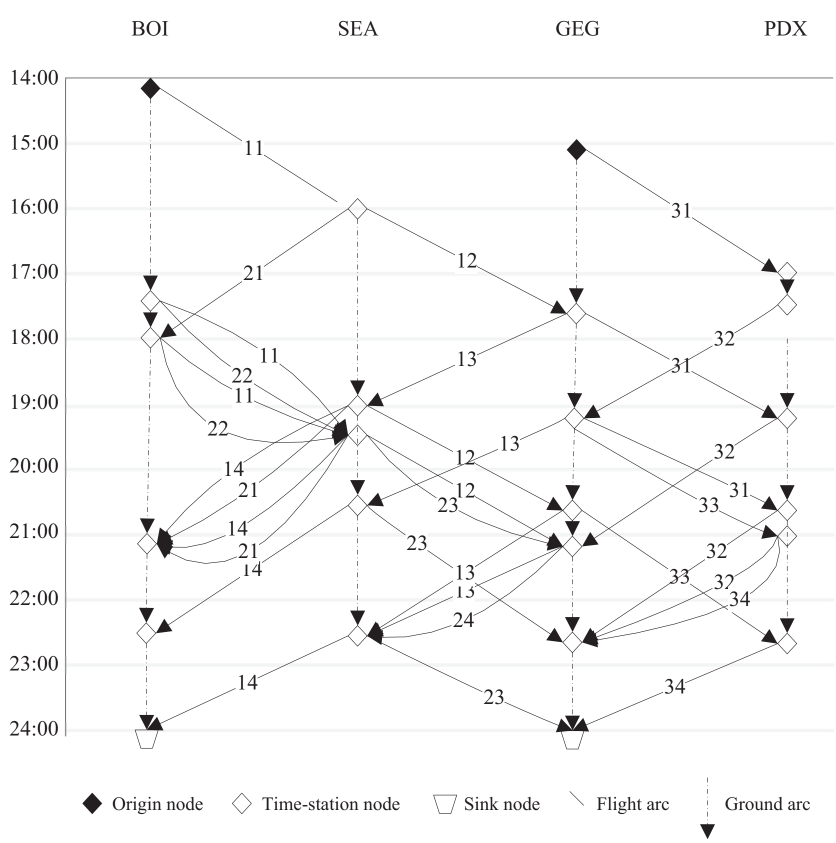

Fig. 1. Sample of reduced time-band network.

The *PTR* generation algorithm is described as follows.

Step 1. Set $PTR = \emptyset$.
Step 2. For $f \in F$ do
For $\lambda \in Farc$ do
Set $\lambda = ((s, t), (s', t'), k)$
If $dep_f = dep_k$, $arr_f = arr_k$, $f \neq k$, and $t \geqslant std_f$, then put
$PTR \leftarrow PTR \cup \{(f, k, \lambda)\}$
Step 3. Output *PTR*.

Steps 1 and 3 are self explanatory. At Step 2, two nested loops are used to add transiting connection to *PTR*. The outer loop is for each flight in *F* and the inner loop is for each flight arc in *RTBN*. The goal is to find specific flight arcs that have the same origin and destination as the flight *f* being considered in the outer loop, and with departure times later than the scheduled departure time of *f*. The complexity of the algorithm is $O(mn^2\alpha\tau)$, where *n* is the total number of flights. To see this, note that at Step 2, the number of flights is $O(n)$, and the number of flight arcs is not more than $O(mr\alpha\tau)$. Therefore, two levels of circles need to finished in $O(mnr\alpha\tau)$, which is dominated by $O(mn^2\alpha\tau)$.

4.3. Model formulation

In this section, we develop an integer programming model for the ARPRP based on the *RTBN* and *PTR*. The following notation is used.

Sets and indices	
E	Set of fleets; $e \in E$
F	Set of flights; $f, k \in F$
$F(e)$	Set of flights that can be flown by aircraft in fleet e
$I = ON \cup TN$	Set of time-station nodes, including origin nodes and intermediate nodes; $i, j \in I$
J	Set of sink nodes; $i, j \in J$
$D(e, i)$	Set of flights originating at time-station node $i \in I$ for aircraft in fleet e
$G(e, f, i)$	Set of time-station nodes for flight f with aircraft fleet e originating from time-station node i
$H(e, i)$	Set of flights terminating at node $i \in I, J$ for aircraft in fleet e
$K(e, f, i)$	Set of time-station nodes for flight f using aircraft in fleet e terminating at time-station node $i \in I$
$K'(e, f, i)$	Set of time-station nodes for flight f using aircraft in fleet e terminating at time-station node $i \in J$
$M(e, f)$	Set of time-station nodes from which flight f originates using aircraft in fleet e
$O(e, i)$	Set of nodes originating from node $i \in I$ representing aircraft in fleet e stay at the same station
$S(e, i)$	Set of nodes to node i representing aircraft in fleet e stay at the same station
$P(f)$	Set of flights with the same departure and destination airport as flight f
$Q(e, f, k)$	Set of time-station nodes for flight k with aircraft fleet e that can accept passengers from flight f
$R(e, k, i)$	Set of flights originating at time-station node $i \in I$ from which passengers can be transferred to flight k with aircraft fleet e
$L(f)$	Set of aircraft fleets types whose aircraft can cover flight f
$T(e, f)$	Set of time-station nodes for flight f with aircraft fleet e that can accept passengers from other flights
Parameters	
NP_f	Number of passengers originally on flight f
N_e	Number of seats on aircraft in fleet e
a_i^e	Number of aircraft belonging to fleet e that are available at time-station node $i \in I$ at the beginning of the recovery period
c_f	Cost of cancelling flight f per passenger
d_{ij}^{ef}	Delay cost of per passenger in flight f for aircraft belonging to fleet e going from time-station node $i \in I$ to node $j \in I, J$
h_j^e	Number of aircraft belonging to fleet e that should terminate at sink node $j \in J$
e_f^{ek}	Cost of per passenger transfer from flight f to flight k for aircraft in fleet e
Decision variables	
x_{ij}^{ef}	1 if flight f in fleet e traverses the arc from node i to node j, 0 otherwise
z_{ij}^e	Number of aircraft in fleet e flowing from node i to node j at the same station
t_f^{eki}	Number of passengers transferring from flight f to flight k with aircraft fleet e originating at time-station node $i \in I$
r_f	Number of passengers whose tickets are refunded due to cancellation of flight f

The objective is to minimize the total cost associated with recovering flights and aircraft and rerouting passengers to their destinations. The first term in the objective function (1) represents the delay cost of flights, which depends on the number of passengers on those flights and the amount of time they are delayed. The second and the third terms try to balance the number of passengers rerouted and the number of passengers receiving ticket refunds. It is preferable to transfer passengers than to refund their tickets.

$$\text{Minimize} \sum_{e \in E} \sum_{f \in F} NP_f \sum_{i \in M(e,f)} \sum_{j \in G(e,f,i)} d_{ij}^{ef} x_{ij}^{ef} + \sum_{e \in E} \sum_{f \in F} \sum_{k \in P(f)} e_f^{ek} \sum_{i \in Q(e,f,k)} t_f^{eki} + \sum_{f \in F} c_f r_f \tag{1}$$

Constraint in (2) ensures that every flight is either rescheduled or cancelled ($x_{ij}^{ef} = 0$ for all i,j,e,f).

$$\sum_{i \in M(e,f)} \sum_{j \in G(e,f,i)} x_{ij}^{ef} \leqslant 1 \quad \forall e \in E, f \in F(e) \tag{2}$$

Constraints in (3) and (4) ensure aircraft flow balance at time-station nodes and at sink nodes, respectively.

$$\sum_{f \in D(e,i)} \sum_{j \in G(e,f,i)} x_{ij}^{ef} + \sum_{j \in O(e,i)} z_{ij}^e - \sum_{f \in H(e,i)} \sum_{j \in K(e,f,i)} x_{ji}^{ef} - \sum_{j \in S(e,i)} z_{ji}^e = a_i^e \quad \forall e \in E, i \in I \tag{3}$$

$$\sum_{f \in H(e,i)} \sum_{j \in K(e,f,i)} x_{ji}^{ef} + \sum_{j \in S(e,i)} z_{ji}^e = h_i^e \quad \forall e \in E, i \in J \tag{4}$$

Constraints (5) preserve passenger flow balance by considering ticket refunds and rerouting.

$$\sum_{k \in P(f)} \sum_{i \in Q(e,f,k)} t_f^{eki} + r_f = NP_f \Big(1 - \sum_{i \in M(e,f)} \sum_{j \in G(e,f,i)} x_{ij}^{ef}\Big) \quad \forall e \in E, f \in F(e) \tag{5}$$

Constraints in (6) state that the number of passengers transferring to a flight must not exceed the capacity of the flight.

$$\sum_{k\in R(e,f,i)} t_k^{efi} \leqslant (N_e - NP_f) \sum_{j\in G(e,f,i)} x_{ij}^{ef} \quad \forall f\in F, e\in L(f), i\in T(e,f) \tag{6}$$

Constraints in (7)–(11) define the variables as being either integer or binary.

$$x_{ij}^{ef} \in \{0,1\} \quad \forall e\in E, f\in F, i\in M(e,f), j\in G(e,f,i) \tag{7}$$

$$z_{ij}^{e} \in \{0,1,2,\ldots\} \quad \forall e\in E, i\in I, j\in O(e,i) \tag{8}$$

$$t_f^{eki} \in \{0,1,2,\ldots\} \quad \forall e\in E, f\in F, k\in P(f), i\in Q(e,f,k) \tag{9}$$

$$r_f \in \{0,1,2,\ldots\} \quad \forall f\in F \tag{10}$$

Model (1)–(10) is an approximate representation of the real problem that becomes increasingly accurate as the time-band interval shrinks. However, its solution will underestimate delay cost and transiting cost, and the indicated departure and arrival times may be displaced by the length of the time-band. As a consequence, it will generally be necessary to adjust the cost values and schedule obtained to take into account the continuous nature of time.

Complexity analysis. The construction of the original daily flight schedules can be viewed as a multi-depot vehicle scheduling problem (MDVSP), which Bertossi, Carraresi, and Gallo (1987) proved is strongly NP-hard. The aircraft recovery problem (ARP) without considering passengers has two additional components that must be taken into account. The first is the option to cancel flights and the second is the introduction of a departure window as a replacement for the published departure time for each flight.

Proposition 1. *ARP is NP-hard in the strong sense.*

Proof. We will show that the nonhomogeneous, uncapacitated vehicle routing problem with time windows (NU-VRPTW) can be transformed into an instance of the ARP in polynomial time. In our version of the NU-VRPTW, which is essentially an m-TSPTW which strongly NP-hard, we are given m vehicles (aircraft) located at a single depot (airport) and n customers (flights), each of which must be served by a single vehicle within a fixed time window, say, $[a_f,b_f]$ for $f = 1, \ldots, n$. By the end of the planning horizon (recovery period) all vehicles must return to the depot. The objective is to minimize the cost of serving all customers subject to flow balance, routing, and time window constraints. No vehicle capacity limits are assumed. The problem can be represented by a directed graph $G = (V, A)$ with both flight nodes and station nodes similar to the structure depicted in Fig. 1 so it will not be described here.

Given an instance of NU-VRPTW, we construct an instance of the ARP as follows. At the beginning of the recovery period, without loss of generality, the first g of the m aircraft are grounded at a single airport denoted by the index 0, and at the end of the recovery period all m aircraft must be located at the same airport. Rather than actually removing the g aircraft from the problem they are used to cover flights so if one or more of those aircraft have a route in the solution, the corresponding flights are assumed to be cancelled. In the network, an arc with zero cost is included from each flight node back to the node 0 to allow for the termination of a route after any flight. Now let c_{fk} be the cost on the arc that corresponds to covering flight f with aircraft k, where $c_{fk} = c_f$ if flight f is "covered" by one of the grounded aircraft $k = 1, \ldots, g$, and hence cancelled; otherwise it is equal to the delay cost for $k = g+1, \ldots, m$, which can be calculated from the difference between the actual departure time and the scheduled departure time. The cost of flying the aircraft is not included in the model, only the cost of either cancelling or delaying a flight.

To construct the time window for flight f we set $a_f = std_f$ and $b_f = \min\{maxd + std_f, curf - dur_f\}$. The inclusion of return arcs to the depot after every flight node along with the follow balance requirement allows any of the flights in the route associated with a grounded aircraft to be coved by one of the m–g available aircraft. It also allows a flight assigned to a non-grounded aircraft to be cancelled. The simplest case would be for the grounded aircraft to "fly" their original route if that were feasible.

A solution to this instance to the ARP represents the minimum cost of recovering a disrupted aircraft schedule. If we can find an optimal solution to the NU-VRPTW, then we can find a minimum cost solution to the ARP with g grounded aircraft that meets the time window, flow balance and time window constraints. That fact that an instance of NU-VRPTW can be transformed into an instance of ARP in polynomial time coupled with the fact that a given solution to the latter can be checked in polynomial time completes the proof. □

Remark. For VRPTWs, as the width of the time windows shrink, instances get easier to solve, at least empirically. When the time windows shrink to 0, the vehicles are identical, and there is a single depot, we get the vehicle scheduling problem which is polynomial solvable with a min-cost flow algorithm. For the ARP, the time windows are typically large because the departure times are set as far as possible into the recovery period in order to accommodate as many passengers as possible. In our data sets the average width varies from 3 to 7.5 h, depending on the length of the time-band specified. Also, the average length of a flight is 2.3 h. These statistics taken together imply the existence of feasible cycles in the ARP graph and so confirm its difficulty. If the time windows are sufficiently small however, there will be no cycles in the ARP graph, but the problem remains strongly NP-hard. This can be shown from reduction from a parallel machine scheduling problem with ready times and due dates and the objective of minimizing the maximum lateness.

Corollary 1. *The problem represented by model (1)–(10) is NP-hard in the strong sense.*

Proof. The result follows from the fact that for a given time-band length, the abbreviated model defined by (1)–(4), (7)–(8) corresponds to a modified version of the MDVSP. The remaining constraints and variables which account for passenger transiting do not affect the complexity of the formulation. In this version of the MDVSP, the nodes corresponding to a particular flight are grouped together with the requirement that exactly one node in the group must be visited. This modification gives rise to what is called the *vehicle scheduling problem* represented in Bertossi et al. (1987). As in the proof of Proposition 1, grounded aircraft are used to cover cancelled flights. □

5. Existence of solutions to network model

In this section, the existence of solutions to model (1)–(10), denoted by *ARPR*, is analysed. In accordance with the daily operational requirements, a specified number of aircraft must be positioned at each airport in the network for the schedule to be feasible at the beginning of the next day. If grounded aircraft are taken out of service and cannot be repaired by the end of the recovery period, sufficient resources may not be available at all airports. Based on the assumption of no ferrying, only aircraft assigned to flights can arrive at an airport. In some circumstances, then, aircraft flow balance may be not satisfied before curfew time. To address this issue, we would like to know under what circumstances a solution exists to *ARPR*. The answer is given in the form of a necessary condition stated below with the help of some additional notation.

ARPR	Problem represented by model (1)–(10)
ARPR'	Relaxation of problem *ARPR* such that there is no curfew or maximum delay time limit, i.e., the recovery period can be extend without limit.
sol_{ARPR}	Feasible solution for problem *ARPR*
S_{ARPR}	Set of feasible solutions for problem *ARPR*; $sol_{ARPR} \in S_{ARPR}$
$sol_{ARPR'}$	Set of feasible solutions for problem *ARPR'*
$S_{ARPR'}$	Set of feasible solutions for problem *ARPR'*; $sol_{ARPR}' \in S_{ARPR}'$

We also introduce a new network denoted by *CN* with node set *V* and arc set *A*, where $CN = (s, t, V, A, b)$ and $V = SN \cup FN \cup TN$, where, with some abuse of notation, *s* and *t* are now origin and termination nodes of *CN* and the definition of *b* is given below. The set $A = \{(s,i), i \in SN_a\} \cup \{(i,j), i \in SN_a, j \in FN\} \cup \{(i,j), i,j \in FN\} \cup \{(i,j), i \in FN, j \in TN_a\} \cup \{(i,t), i \in TN_a\}$, which consists of five different types of arc sets whose definitions are as follows.

SN_a	Node set representing available aircraft at airport *a*
$SN = \cup_a SN_a$	Node set representing available aircraft
FN	Node set representing flights as scheduled
$TN = \cup_a TN_a$	Node set representing aircraft that sink at the end of the recovery period
TN_a	Node set representing aircraft that sink at the end of the recovery period at airport *a*
(s, i), $i \in SN_a$	One aircraft is initially available at node $i \in SN_a$
(i, j), $i \in SN_a$, $j \in FN$	Indicates that one aircraft is available at node $i \in SN_a$ for the flight corresponding to flight node *j*
(i, j), $i, j \in FN$	Represents the situation where the arrival airport of flight corresponding to flight node *i* is the same as the departure airport of the flight corresponding to flight node *j*
(i, j), $i \in FN$, $j \in TN_a$	Aircraft sink at airport *a* after implementing the flight corresponding to flight node *i*
(i, t), $i \in TN_a$	Aircraft sink at airport *a* at the end of the recovery period
$f(i, j)$	Flow from node *i* to node *j* for arc $(i, j) \in A$
$b(i, j)$	Flight arc capacity where $b(i, j) = 1$, $\forall\ (i, j) \in A$ (*A* includes all arcs in *CN*), 0 otherwise

Let the maximum number of vertex-disjoint paths from *s* to *t* in *CN* be denoted by *dstn* and let the maximum number of paths from *s* to *t* in *CN* be denoted by *stn*. Fig. 3 provides a partial example of *CN* for the sample network in Bard et al. (2001). From the definition of *CN*, we know that $dstn \leqslant stn \leqslant |SN| = |TN|$. We can easily obtain the value of dstn by the Ford-Fulkerson labeling method (Ford & Fulkerson, 1956), which is based on the max-flow, min-cut theorem.

To provide necessary conditions for the existence of solutions to *ARPR*, we first provide such conditions for *ARPR'*.

Theorem 1. *There exists a* $sol_{ARPR'} \in S_{ARPR'} \iff dstn = |SN|$.

Proof.

(1) There exists a $sol_{ARPR'} \in S_{ARPR'} \Rightarrow dstn = |SN|$

We will prove the converse negative proposition: $dstn < |SN| \Rightarrow \nexists sol_{ARPR'}$

There are two situations for $dstn < |SN|$; one is $dstn \leqslant stn < |SN|$, and the other is $dstn < stn = |SN|$.

If $stn < |SN|$, for instance, then $stn = |SN| - 1$. According to the definition of *CN*, only $|SN| - 1$ aircraft arrive at the disrupted airport, and one aircraft violates constraints (4). Therefore, $\nexists sol_{ARPR'}$.

If $dstn < stn = |SN|$, for instance, $dstn = stn - 1$. Thus, $\exists\ i, j_1, j_2 \in FN$, j_1, $f(i, j_1) = 1$ and $f(i, j_2) = 1$. According to the definition of *CN*, there is one flight implemented by two aircraft, which violates constraint (2). Therefore, $\nexists sol_{ARPR'}$.

(2) There exists a $sol_{ARPR'} \in S_{ARPR'} \Leftarrow dstn = |SN|$

Because $dstn = |SN|$, according to the definitions of *CN* and *ARPR'*, the recovery period of *ARPR'* has no limit. Thus, during the recovery period, there are $|SN|$ available aircraft flows, and they satisfy constraints (2)–(4), (7) and (8). Now, for all $f \in F$, if $\sum_{i \in M(e,f)} \sum_{j \in G(e,f,i)} x_{ij}^{ef} = 1$, we set $r_f = 0$; otherwise, we set $r_f = Np_f$. Also, we set $t_f^{eki} = 0$, $\forall e \in E, f \in F, k \in P(f), i \in Q(e,f,k)$. Accordingly, r_f and t_f^{eki} satisfy constraints (4), (5), (9) and (10). Assigning these values for x_{ij}^{ef}, z_{ij}^{e}, r_f and t_f^{eki} gives us an element in $sol_{ARPR'}$. Therefore, $\exists sol_{ARPR'} \in S_{ARPR'}$. □

Lemma 1 gives a necessary and sufficient condition, that is, $dstn = |SN|$, for solutions to exist for the *ARPR'*. A necessary condition for the existence of solutions to the *ARPR* can be deduced from the lemma as follows.

Corollary 2. *There exists a* $sol_{ARPR} \in S_{ARPR} \Rightarrow dstn = |SN|$

Proof. From Theorem 1, we know that $sol_{ARPR'} \in S_{ARPR'} \Rightarrow dstn = |SN|$.From the definitions of *ARPR* and *ARPR'*, we know $S_{ARPR} \subseteq S_{ARPR'}$, which implies that $\exists sol_{ARPR} \in S_{ARPR} \Rightarrow \exists sol_{ARPR'} \in S_{ARPR'}$. Therefore, $\exists sol_{ARPR} \in S_{ARPR} \Rightarrow dstn = |SN|$. □

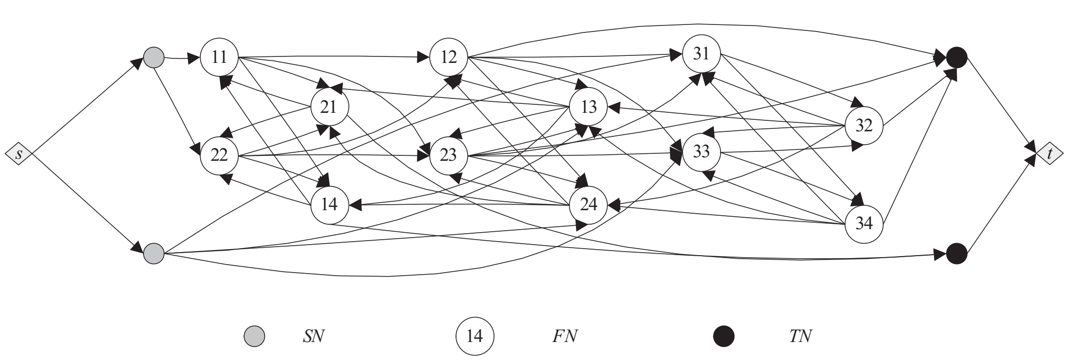

Fig. 3. Example of *CN*.

6. Computational results

The flight and schedule data used to evaluate the effectiveness of our model for the ARPRP was provided in part by a big Airline in China. In this section, we discuss the construction of problem instances, the preprocessing steps, and the experimental results. All computations were performed on a DELL INSPIRON N4110 with CPU i5-2410 and 2.3 gigabits of RAM. The methodology, built around the reduced time-band network and passenger transiting relationship, was coded in C++ programming and all integer programs were solved with CPLEX 12.3.

Before presenting the full set of computational results, we look at the solution to the sample problem in Bard et al. (2001) to demonstrate the potential costs reductions that are attainable by explicitly considering passenger disruptions. Since they did not consider passenger disruptions, some additional assumptions are necessary. In particular, we assume that the capacity of each aircraft is 120, the number of passengers on each flight is 100 and the delay cost for each passenger is \$0.2/min. The transiting cost for each passenger is based on \$0.25/min and is determined by multiplying this value by the difference between the actual departure time of the reassigned flight and the scheduled departure time of the cancelled flight.

The optimal objective value obtained from the linear programming relaxation of the integrality requirements (7)–(10) is \$20,912, which is a lower bound. After adjusting the departure and arrival times obtained by solving the reduced time-band network to get the actual aircraft flow, we have the results given in Table 1. For example, the delay time of flight 21 is 15minutes, and the delay cost per passenger is \$0.2/min, so the delay cost of flight 21 is $0.2 \times 15 \times 100 = \$300$. Similarly, the delay cost for flights 22, 23, 24, and 14 are \$300, \$300, \$200, and \$4500, respectively. For cancelled flight 12, 20 passengers are transformed to flight 23 so the transiting cost is $\$0.25 \times 20 \times 220 = \$1100$, and 80 passengers are refunded the cost of their ticket so the cancellation cost $\$102.31 \times 80 = \$8184.8$, where \$102.31 is the cancellation cost for each passenger in flight 12. Similarly, the transiting cost and the cancellation cost for flight 13 are respectively \$1125 and \$5947.2. Summing all the costs give a total of \$21,957, which is less than the solution value in Bard et al. (2001) because some of the passengers on cancelled flights will be reassigned to other flights. Also, note that the gap between the lower bound and the feasible solution is only 4.8%.

For the computational experiments, a total of 20 instances were created. The data required for the ARPRP include flight information (flight ID, flight departure and arrival times, departure and arrival airport stations) and aircraft information (aircraft ID, aircraft fleet type and aircraft capacity). These data along with original schedule for one day were obtained from the Airline, while passenger information (number of passengers on each flight and cancel cost for each passenger) were hypothesized for testing purposes. The number of the aircraft in an instance ranged from 47 to 188, the fleet set varied from 3 to 13, and the flight set ranged from 130 to 628. These values reflect one day of the Airline operations.

Based on the original schedule, the disruption scenarios (instances) were randomly generated as follows. Cost data and passenger statistics were derived from our discussions with the Airline.

- Between 2–10 aircraft are grounded throughout the day. Grounded aircraft remain out of service for the recovery period but are included in the flow balance equations at the end of the period.
- Scenarios do not include delaying aircraft, only grounding them. Grounded aircraft make problem instances much more difficult than delayed aircraft. Therefore, we only considered the former rather than the latter or a mixture.
- A 40-min minimal turnaround time is assumed at all airports.
- Delay cost for each passenger is ¥0.1/min.
- The number of the passengers in each aircraft is 80% of capacity.
- Transiting cost for each passenger is determined according to the difference between the actual departure time of the reassigned flights and the scheduled departure time of the cancelled flight, assuming ¥0.15/min.
- Maximum delay time (*maxd*) varied from 4 h to 16 h.
- The recovery period is one day, that is, from 7 am - 12 pm.

Fleet substitutions and the aircraft fleet capacities are listed in Table 2, where 'y' indicates that the row fleet can substitute for the column fleet to cover a flight as scheduled; otherwise, an 'n' appears. To guarantee the feasibility of each instance, it is assumed that the airport where an aircraft is grounded is the same airport where that aircraft is scheduled to be located at the end of the recovery period.

The objective again is to minimize the total system cost resulting from flight delays, passenger reassignments and cancellations. The output of the optimization solver is used to generate new aircraft routes and passenger transiting itineraries between different flights, and a list of re-quoted and cancelled flights, as well as the number of passengers from cancelled flights who are rerouted.

Tables 3 and 4 contain the results for the 20 scenarios that we investigated. Each table is divided into five sections. The first section presents the size of the recovery problem:

"*flight#*" = number of flights used in the instances
"*fleets#*" = number of fleet types in the instances
"*grdplane#*" = number of grounded aircraft in different disruption scenarios

Table 1
Solution for sample problem.

Aircraft id	Flight id	Origin	Destination	Departure	Arrival	Number of passengers	Delay cost	Transiting cost	Cancellation cost
1	11	BOI	SEA	2:10	15:20	100			
	21	SEA	BOI	16:00	17:15	100	300		
	22	BOI	SEA	17:55	19:05	100	300		
	23	SEA	GEG	19:45	20:45	120	300		
	24	GEG	SEA	21:25	22:25	120	200		
	14	SEA	BOI	23:05	0:20	100	4500		
3	31	GEG	PDX	15:15	16:20	100			
	32	PDX	GEG	17:30	18:30	100			
	33	GEG	PDX	19:10	20:20	100			
	34	PDX	GEG	21:00	21:55	100			
Cancelled	12	SEA	GEG					1100	8184.8
	13	GEG	SEA					1125	5947.2

Table 2
Fleet substitutions and fleet capacity.

Fleet	A319	A320	A321	A332	A343	B733	B736	B737	B738	B744	B752	B767	B772
Capacity	120	150	170	220	220	120	120	120	160	250	200	200	300
A319	–	n	n	n	n	n	n	n	n	n	n	n	n
A320	y	–	n	n	n	n	n	n	n	n	n	n	n
A321	y	y	–	n	n	n	n	n	n	n	n	n	n
A332	y	y	y	–	y	n	n	n	n	n	n	n	n
A343	y	y	y	y	–	n	n	n	n	n	n	n	n
B733	n	n	n	n	n	–	y	y	n	n	n	n	n
B736	n	n	n	n	n	y	–	y	n	n	n	n	n
B737	n	n	n	n	n	y	y	–	n	n	n	n	n
B738	n	n	n	n	n	y	y	y	–	n	n	n	n
B744	n	n	n	n	n	n	n	n	n	–	n	n	n
B752	n	n	n	n	n	n	n	n	n	n	–	y	n
B767	n	n	n	n	n	n	n	n	n	n	y	–	n
B772	n	n	n	n	n	n	n	n	n	n	y	y	–

Table 3
Solution results for disruption cases 1–10.

Scenarios	1	2	3	4	5	6	7	8	9	10
flights#	215	184	130	184	184	184	184	215	567	628
fleets#	5	3	1	3	3	3	3	5	9	13
grdplane#	2	2	2	4	6	8	10	10	10	10
avaplane#	64	45	28	43	41	39	37	56	153	178
tblength	5	5	5	5	5	5	5	5	5	5
Maxd (hrs)	4	4	4	4	4	4	4	4	4	4
Farc	24,080	12,334	3,346	12,334	12,334	12,334	12,334	24,080	50,607	51,874
Node	12,355	6735	1690	6735	6735	6735	6735	12,355	59,373	88,166
Tran	25,093	11,586	2,370	11,586	11,586	11,586	11,586	25,093	58,823	66,747
trantime (sec)	10.05	3.06	0.89	3.19	3.23	3.14	3.34	10.49	70	72.09
MIPtime (sec)	15.34	3.56	0.23	1.65	2.06	5.23	59	161.32	26.83	32.82
soltime (sec)	0.67	0.25	0.09	0.22	0.15	0.12	0.05	0.1	0.25	0.74
btrantime (sec)	19.84	6.09	1.04	5.91	6.12	6.05	5.73	19.84	150.43	183.63
bMIPtime (sec)	15.69	4.33	0.72	3.56	4.62	12.19	225.37	23.546	54.68	88.84
canceled f#	0	0	0	0	1	1	5	6	6	7
canceled p#	0	0	0	0	96	96	480	384	456	448
trans p#	0	0	0	0	0	0	0	192	256	256
delay cost	13,688	4896	5280	8696	19,062	28,112	47,336	38,072	18,416	23,705
trans cost	0	0	0	0	0	0	0	960	21,088	20,992
canceled cost	0	0	0	0	234,240	235,360	467,236	389,760	314,240	395,840
total cost	13,688	4896	5280	8696	253,302	263,472	514,572	428,792	353,744	440,537
LP value	12,521	4468	4992	8600	252,536	260,836	489,461	423,389	351,089	437,906
Gap	8.53%	8.74%	0.00%	1.10%	0.30%	1.00%	4.88%	1.26%	0.75%	0.60%
cf#	10	12	12	18	26	34	44	44	40	44
cp#	1056	1152	1152	2048	2976	3744	4224	4224	3920	4224
Ccost	1,284,720	1,007,040	1,007,040	2,480,240	3,838,800	4,698,000	4,891,200	4,891,200	4,621,200	4,807,680
Actualcost	474,505	6877	6877	300,386	633,394	637,956	646,638	646,638	771,986	774,461

"*avaplane#*" = number of available aircraft in the scenarios

The second section presents the attributes of the networks associated with *RTBN* and *PTR*:

"*tblength*" = length of time interval in *RTBN*
"*farc*" and "*node*" = number of flight arcs and number of nodes in *RTBN*
"*tran*" = number of feasible transiting connections included in the network between different flights with the same origin and destination in *PTR*.

The third section presents the CPU time to perform the computations. To assess the relative efficiency of our approach, we compared our results with those obtained with the network used by Bard et al. (2001). Their model is similar to model (1)–(10) except that they use a different version of constraints (3) (see constraints (1c) in their paper). In addition, it was necessary to add a passenger transiting component to their approach.

"*trantime*" = CPU time for constructing *RTBN* and *PTR*
"*MIPtime*" = time required to solve model (1)–(10)
"*soltime*" = time required to construct a schedule corresponding to the integer solution obtained from model (1)–(10)
"*btrantime*" = CPU time for constructing time-band network in Bard et al. (2001) and for constructing the *PTR*
"*bMIPtime*" = time required to solve the network model in by Bard et al. (2001) and for solving the *PTR*

From the third section, we see that the CPU time is no more than 1 min for a 30-min time-band length while the largest CPU time is approximately 3 min for a 5-min time-bands. The time required to construct and solve the time-band network in Bard et al. (2001) when the passenger transiting computations are included is approximately twice the time required to solve *RTBN* and *PTR*. Both methods produced almost identical results.

The fourth section highlights the solution and the corresponding costs obtained from model (1)–(10):

"*cancelf#*" = number of cancelled flights
"*cancelp#*" = number of passengers who receive a refund due to flight cancellation
"*transp#*" = number of passengers who are reassigned to other flights

Table 4
Solution results for disruption cases 11–20.

Scenarios	11	12	13	14	15	16	17	18	19	20
flights#	628	628	628	628	628	567	215	628	628	628
fleets#	13	13	13	13	13	9	5	13	13	13
grdplane#	2	4	6	8	10	10	10	10	10	10
avaplane#	186	184	182	180	178	153	56	178	178	178
tblength	30	30	30	30	30	30	30	30	30	30
Maxd (hrs)	4	4	4	4	4	4	4	8	12	16
farc	12,013	12,013	12,013	12,013	12,013	11,446	5522	21,298	26,925	28,486
node	19,344	19,344	19,344	19,344	19,344	12,861	2850	24,622	26,533	26,962
tran	13,461	13,461	13,461	13,461	13,461	11,400	4917	25,449	34,676	37,434
trantime (sec)	3.52	3.15	3.48	3.35	3.34	2.87	0.98	8.36	16.38	18.82
MIPtime (sec)	0.59	0.87	1.47	2.14	4.91	6.58	48.63	9.58	10.36	8.8
soltime (sec)	0.02	0.02	0.02	0.02	0.02	0.02	0.02	0.1	0.3	0.3
btrantime (sec)	9.2	9.28	9.17	9.24	9.28	6.86	1.35	21.13	32.84	38.46
bMIPtime (sec)	4.22	3.58	4.8	10.63	8.32	5.49	2.98	15.04	13.118	15.06
canceled f#	0	0	0	2	0	0	5	2	2	2
canceled p#	0	0	0	0	0	0	288	0	0	0
trans p#	0	0	0	192	0	0	192	192	192	192
delay cost	4120	5064	11,096	17,850	26,448	26,836	32,364	24,360	24,360	24,360
trans cost	0	0	0	1,056	0	0	3664	1056	1056	1056
canceled cost	0	0	0	0	0	0	315,840	0	0	0
total cost	4120	5064	11,096	18,906	26,448	26,836	351,868	25,416	25,416	25,416
LP value	1040	2896	5760	12,922	18,217	21,016	344,425	17,947	17,947	17,947
gap	74.76%	42.81%	48.09%	31.65%	31.12%	21.69%	2.12%	29.39%	29.39%	29.39%
cf#	8	16	28	36	44	44	44	44	42	40
cp#	896	1792	2944	3584	4480	4352	4224	4608	4416	4384
ccost	893,120	1,770,880	2,777,920	3,622,400	4,500,160	4,590,080	4,891,200	4,767,360	5,095,680	4,997,680
actualcost	4581	249,171	257,165	23,163	270,724	278,580	646,638	319,122	334,184	336,779

"*delaycost*", "*trancost*" and "*cancelcost*" = flight delay cost, passenger transiting cost and refund cost
"*totalcost*" = total cost of the solution

The fifth section gives comparative statistics for the relaxed and final solutions:

"*LP value*" = value of linear programming relaxation
"*gap*" = gap between "*totalcost* " and "*LPvalue*"; equals 100%×(*totalcost - LPvalue)/totalcost*
"*cf#*" = number of cancelled fights if we simply cancel the grounded flights
"*cp#*" = number of passengers who have to be refunded their ticket price when their flights are cancelled
"*ccost*" = cost when all grounded flights are cancelled
"*actualcost*" = cost that reflects what the Airline would have incurred if their rules were followed for cancelling and rerouting passengers when faced with a disruption

When the Airline experiences disruptions in their schedules, the dispatchers follow a series of rules to aid in the recovery. In order of priority, flights with very important persons (VIP) are rescheduled first, then international flights that are not carrying any VIPs, and finally, the remaining flights ordered by the affected number of passengers. To calculate the "actual costs" we wrote a heuristic that implements this scheme. For the scenarios tested, no VIPs were assumed to be on any of the flights, and international flights were determined by the actual situation. In the heuristic, when an aircraft associated with an international flight is grounded its remaining legs are assigned to other aircraft whenever possible. Grounded domestic flights are prioritized in decreasing order of the number of passengers aboard and are assigned to available aircraft one by one. Finally, if there are still flights that cannot be accommodated, the passengers on those flights are either transferred to other flights whenever possible or their tickets are refunded.

Cases 1–10 in Table 3 were solved by the network model using a 5-min time band, while cases 11–20 were solved using a 30-min time band since they are much larger in size. This value represents a compromise between solution quality and computational speed, and was arrived at after extensive testing. In the solution of cases 1–17, the maximum flight delay time parameter, *maxd*, was set to 4 h, while in cases 18, 19 and 20, it was set to 8 h, 12 h and 16 h, respectively.

In cases 1–3 and cases 7–10, as the number of fleet types increase, construction of *RTBN* and *PTR* require increasingly more CPU time. Since more options are available to reassign passengers to different flights, this has the effect of decreasing cost, which is more apparent in case 7 to 9. Similarly, as the number of grounded aircraft increases, the CPU time increases substantially, as do the number of cancelled flights, the delay cost, the cancellation cost and the total cost.

The impact of *maxd* on solution quality is demonstrated in cases 15 and 18–20, which have the same aircraft fleet type and the same number of grounded aircraft based on the same disruption scenario. According to airline regulations, when the flight delay time is over 4 h, passengers are due some form of compensation from the Airline. For purposes of comparison we considered four different values for this parameter: 4 h, 8 h, 12 h and 16 h. Figs. 4 and 5 illustrate the relationships between the value of *maxd* and the CPU time, and between the value of *maxd* and the solution cost for one instance with 13 aircraft fleets and 10 aircraft grounded, respectively. The CPU time for constructing the network and running the model increases at a decreasing rate with *maxd*. But the solution value, "total cost," is relatively constant. In practice, airlines must balance solution quality with algorithmic efficiency. To some extent, this tradeoff is reflected in the choice of *maxd*. However, as *maxd* increases, the maximum delay time in a solution converges quickly to 7 h. The results show in Fig. 5 that total cost is relatively insensitive to values of *maxd* above 8 h.

In the fifth section of Tables 3 and 4, "LPvalue" and "totalcost" give lower and upper bounds on the true value of the optimal solution to ARPRP. For a time-band length of 5 min, the gap between the total cost and the lower bound is within 8%. The optimality gap averaged just under 3%. In contrast, when the time-band length is 30 min, the average gap is around 30% of the relaxed

Table 5
Comparison of average results.

Solution approach	*cancelf#*	*transp#*	*cancelp#*	*delaycost*	*transcost*	*cancelcost*
RTBN	2	83	112	20,206	2829	117,626
Heuristic	2.5	23	268	30,023	19,451	364,198

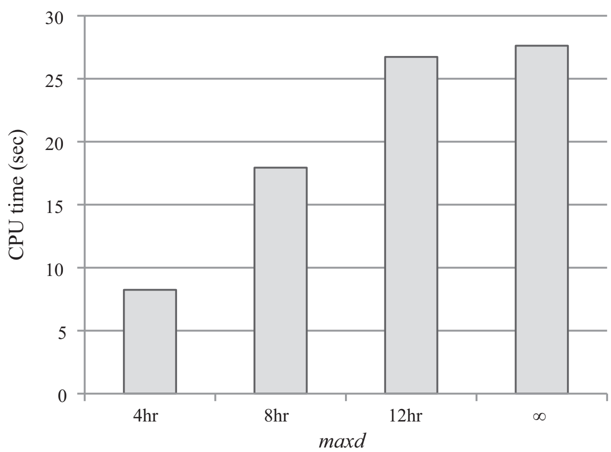

Fig. 4. Comparison of CPU time for different values of *maxd*.

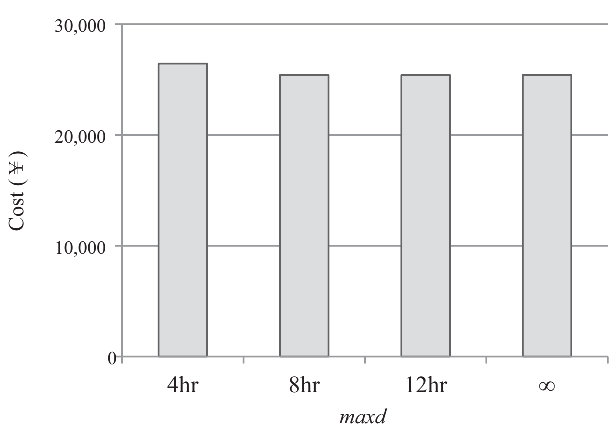

Fig. 5. Comparison of solution cost for different values of *maxd*.

solution. Nevertheless, the solution obtained when using 30 min time-bands is still overwhelmingly better than the solution found with the Airline heuristic, as it is when compared to the cost of cancelling all flights associated with the grounded aircraft. In general, shorter time-band lengths produce smaller gaps between the bounds but require much more CPU time to generate the network and solve the model. Testing indicated that the relationship is approximately exponential.

Fig. 6 displays the average solution costs obtained from the two methods used in the computations. The results from model (1)–(10) are much lower than the values obtained by simply cancelling the grounded aircraft or applying the Airline's heuristic. Table 5 gives a comparison of the average results obtained from the solution from model (1)–(10) in the first row and the solution from the Airline's heuristic in the second row. The two methods have about the same average number of cancelled flights; however, the remaining statistics describe sharp differences. The average number of transiting passengers in the *RTBN* solution is much greater than the number provided by the heuristic. In contrast,

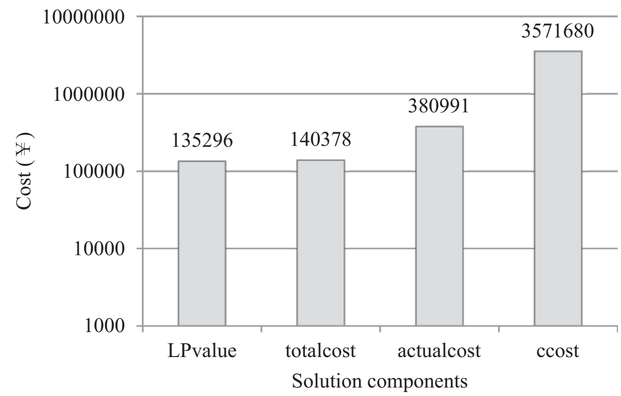

Fig. 6. Comparison of average solution costs.

the average number of refunds determined by the *RTBN* is far fewer than determined by the heuristic. With respect to the distribution of delay, transiting and cancellation costs, the *RTBN* values dominate in all categorizes with a reduction of approximately 85% for transiting costs.

Of course, reading too much into the comparisons would be a mistake since the Airline heuristic lacks any of the intelligence that one would find in such approaches as tabu search or simulated annealing. Nevertheless, the results demonstrate the degree to which the use of optimization technology can improve on practice. One of the reasons why the heuristic did so poorly is that it ranks flights mostly by the number of passengers on board the grounded aircraft. It falls short of the system-wide view embodied in the *RTBN* that implicitly trades off the three cost components to arrive at the best solution within the constraints of the Airline.

7. Summary

In this paper, a new method for solving the combined aircraft and passenger recovery problem is introduced. In a typical scenario, several aircraft are grounded and flights have to be rescheduled and passengers reassigned in a way that minimizes the disruption and corresponding costs. Based on a reduced time-band network and passenger transiting relationship, an integer programming model is presented and solved to optimality with CPLEX in negligible time. Realistic instances with up to 10 grounded aircraft and 13 fleet types required less than 30 s for all the computations using a 30-min time-band length. The comparative analysis indicated optimality gaps between 0 and 8%, with an average of less than 3%, when a 5-min time-band length was used. At the same time, we found that applying the Airline's heuristic to construct recovery schedules the costs were often an order of magnitude greater than those obtained with our model.

A parallel component of the research involved determining the conditions under which an instance will have a solution that does not violate curfew, aircraft availability, and maximum delay restrictions. It was shown that a necessary condition for a solution to exist to *ARPR*, model (1)–(10), is that $dstn \geqslant |SN|$; that is, in the network constructed by airport station connectivity but not the time connectivity, if the maximal number of disjoint paths from the source to the terminal nodes at the end of the recovery period is less than the number of available aircraft, then there will be no feasible solution. In addition, we showed that the same condition is applicable to the *ARPR′*, that is, the relaxed version of the problem where there is no curfew or flight delay limit.

In the future, it would be worthwhile to examine several related issues. First, aircraft maintenance, which is a problem faced by every airline during daily operations, should also be considered when recovering the schedule. Second, if it were possible to develop an expanded model that integrated aircraft, passengers and crew, improved results would likely follow. Finally, other solution approaches, such as metaheuristics or robust optimization might prove advantageous when a multitude of recovery options are called for. Although we were able to find optimal solutions for all the instances examined, stable solutions are more often preferred by practitioners.

Acknowledgements

This work was partially supported by the Institute of Policy and Management, Chinese Academy of Sciences and by the Research center of Overall Planning and Safety Management.

References

Andersson, T., & Varbrand, P. (2004). The flight perturbation problem. *Transportation Planning and Technology, 27*, 91–117.
Argüello, M. F., Bard, J. F., & Yu, G. (1997). A GRASP for aircraft routing in response to groundings and delay. *Journal of Combinatorial Optimization, 1*(3), 211–228.
Arıkan, U., Sinan Gürel, S., & Aktürk, M. S. (2013). Integrated aircraft and passenger recovery with cruise time controllability. *Annals of Operations Research*, 11.
Artigues, C., Bourreau, E., Afsar, H. M., Briant, O., & Boudia, M. (2012). Disruption management for commercial airlines: Methods and results for the ROADEF 2009 Challenge. *European Journal of Industrial Engineering, 6*, 669–689.
Babić, O., Kalić, M., Pavković, G., Dožić, S., & Čangalović, M. (2010). Heuristic approach to the airline schedule disturbances problem. *Transportation Planning and Technology, 33*(3), 257–280.
Ball, M., Barnhart, C., Nemhauser, G., & Odoni, A. (2007). Air transportation: Irregular operations and control. In J. Jan Lenstra, J.G. Dai (Eds.), Handbook in operations research and management science (pp. 1–68), Atlanta.
Bard, J. F., Yu, G., & Argüello, M. F. (2001). Optimizing aircraft routings in response to groundings and delays. *IIE Transactions on Operations Engineering, 33*(10), 931–947.
Bertossi, A. A., Carraresi, P., & Gallo, G. (1987). On some matching problems arising in vehicle scheduling models. *Networks, 17*(3), 271–281.
Bisaillon, S., Cordeau, J.-F., Laporte, G., & Pasin, F. (2011). A large neighborhood search heuristic for the aircraft and passenger recovery problem. 4OR. *A Quarterly Journal of Operations Research, 9*(2), 139–157.
Bratu, S., & Barnhart, C. (2006). Flight operations recovery: New approaches considering passenger recovery. *Journal of Scheduling, 9*(3), 279–298.
Castro, A. J. M., & Oliveira, E. (2011). Airline operations control: A new concept for operations recovery. In C.R. Walsh, (Ed.), *Airline industry: Strategies, operations and safety* (pp. 61–97), New York.
Chan, F. T. S., Chung, S. H., Chow, J. C. L., & Wong, C. S. (2013). An optimization approach to integrated aircraft and passenger recovery. In Y.-K. Lin, Y.-H. Tsao, S.-W. Lin, (Eds.), *Proceedings of the Institute of Industrial Engineers Asian Conference* (pp. 729–737), Taipei.
Clause, J., Larsen, A., Larsen, J., & Rezanova, N. J. (2010). Disruption management in the airline industry-concepts, models and methods. *Computers & Operations Research, 37*(5), 809–821.
Eggenberg, N., Bierlaire, M., & Salani, M. (2010). Constraint-specific recovery network for solving airline recovery problems. *Computers & Operations Research, 37*(6), 1014–1026.
Etschmaier, M. M., & Mathaisel, D. F. X. (1985). Airline scheduling: An overview. *Transportation Science, 19*(2), 127–138.
Filar, J. A., Manyem, P., & White, K. (2001). How airlines and airports recover from schedule perturbations: A survey. *Annals of Operations Research, 108*(1–4), 315–333.
Ford, L. R., & Fulkerson, D. R. (1956). Maximal flow through a network. *Canadian Journal of Mathematics, 8*(1), 399–404.
Gershkoff, I. (1987). Aircraft shortage evaluator. ORSA/TIMS Joint National Meeting, MO, St. Louis.
Hu, Y., Xu, B., Gao, M., & Chi, H. (2011). The study on the aircraft recovery with consideration of passenger transiting. In *Proceedings of the Fifth International Conference on Management Science and Engineering Management*, Macau.
Jafari, N., & Zegordi, S. H. (2010). Simultaneous recovery model for aircraft and passengers. *Journal of the Franklin Institute, 348*(7), 1638–1655.
Jarrah, A., & Yu, G. (1993). A decision support framework for airline flight cancellations and delays. *Transportation Science, 27*(3), 266–280.
Kohl, N., Larsen, A., Larsen, J., Ross, A., & Tiourine, S. (2007). Airline disruption management – Perspectives, experiences and outlook. *Journal of Air Transport Management, 13*(3), 149–162.
Le, M., Wu, C., Zhan, C., & Sun, L. (2011). Airline recovery optimization research: 30 years' march of mathematical programming-a classification and literature review. International Conference on Transportation, Mechanical, and Electrical Engineering (TMEE), Institute of Electrical and Electronics Engineers (IEEE) (pp. 113–117), Changchun.
Lettovsky, L. (1997). Airline operations recovery: An optimization approach. Ph. D thesis, Georgia Institute of Technology, Atlanta, GA.
Liu, T.-K., Chen, C.-H., & Chou, J.-H. (2010). Optimization of short-haul aircraft schedule recovery problems using a hybrid multiobjective genetic algorithm. *Expert Systems with Applications, 37*(3), 2307–2315.
Løve, M., & Sørensen, K. (2001). Disruption management in the airline industry. Master's thesis, Informatics and Mathematical Modelling (IMM), Technical University of Denmark (DTU), Kongens Lyngby.
Løve, M., Sørensen, K., Larsen, J., & Clausen, J. (2005). Using heuristics to solve the dedicated aircraft recovery problem. *Central European Journal of Operations Research, 13*(2), 189–207.
Ryerson, M. S., & Churchill, A. M. (2013). Aircraft Rerouting due to abrupt facility outages: Case study of the 2011 great Tōhoku earthquake, Japan. *Transportation Research Record: Journal of the Transportation Research Board, 2336*, 27–35.
Sinclair, K., Cordeau, J., & Laporte, G. (2014). Improvements to a large neighborhood search heuristic for an integrated aircraft and passenger recovery problem. *European Journal of Operational Research, 233*(1), 234–245.
Teodorovic, D., & Guberinic, S. (1984). Optimal dispatching strategy on an airline network after a schedule perturbation. *European Journal of Operational Research, 15*(2), 178–182.
Thengvall, B. G., Bard, J. F., & Yu, G. (2000). Balancing user preferences for aircraft schedule recovery during irregular operations. *IIE Transactions, 32*(3), 181–193.
Thengvall, B. G., Yu, G., & Bard, J. F. (2001). Multiple fleet aircraft schedule recovery following hub closures. *Transportation Research Part A: Policy and Practice, 35*(4), 289–308.
Yan, S., & Lin, C. G. (1997). Airline scheduling for the temporary closure of airports. *Transportation Science, 31*(1), 72–82.
Yan, S., & Tu, Y. P. (1997). Multifleet routing and multistop flight scheduling for schedule perturbation. *European Journal of Operational Research, 103*(1), 155–169.
Yan, S., & Yang, D. H. (1996). A decision support framework for handling schedule perturbation. *Transportation Research Part B: Methodological, 30*(6), 405–419.
Zhang, Y., & Hansen, M. (2008). Real-time intermodal substitution: Strategy for airline recovery from schedule perturbation and for mitigation of airport congestion. *Transportation Research Record: Journal of the Transportation Research Board, 2052*, 90–99.