

Stern, Helman I.; Gertsbakh, Ilya B.

Article

Using deficit functions for aircraft fleet routing

Operations Research Perspectives

Provided in Cooperation with:

Elsevier

Suggested Citation: Stern, Helman I.; Gertsbakh, Ilya B. (2019) : Using deficit functions for aircraft fleet routing, Operations Research Perspectives, ISSN 2214-7160, Elsevier, Amsterdam, Vol. 6, pp. 1-11,
<https://doi.org/10.1016/j.orp.2019.100104>

This Version is available at:

<https://hdl.handle.net/10419/246390>

Standard-Nutzungsbedingungen:

Die Dokumente auf EconStor dürfen zu eigenen wissenschaftlichen Zwecken und zum Privatgebrauch gespeichert und kopiert werden.

Sie dürfen die Dokumente nicht für öffentliche oder kommerzielle Zwecke vervielfältigen, öffentlich ausstellen, öffentlich zugänglich machen, vertreiben oder anderweitig nutzen.

Sofern die Verfasser die Dokumente unter Open-Content-Lizenzen (insbesondere CC-Lizenzen) zur Verfügung gestellt haben sollten, gelten abweichend von diesen Nutzungsbedingungen die in der dort genannten Lizenz gewährten Nutzungsrechte.

Terms of use:

Documents in EconStor may be saved and copied for your personal and scholarly purposes.

You are not to copy documents for public or commercial purposes, to exhibit the documents publicly, to make them publicly available on the internet, or to distribute or otherwise use the documents in public.

If the documents have been made available under an Open Content Licence (especially Creative Commons Licences), you may exercise further usage rights as specified in the indicated licence.



<https://creativecommons.org/licenses/by-nc-nd/4.0>



Using deficit functions for aircraft fleet routing

Helman I. Stern^{a,*}, Ilya B. Gertsbakh^b

^a Department of Industrial Engineering and Management, Ben Gurion University of the Negev, Israel

^b Department of Mathematics, Ben Gurion University of the Negev, Israel

ARTICLE INFO

Keywords:

Deficit function
Flight schedule
Aircraft fleet
Chain decomposition
Aircraft routing

ABSTRACT

We consider the problem of minimizing the number of airplanes needed to fly a fixed daily repeating schedule of flights. We use deficit functions (DF) to decompose an aviation schedule of aircraft flights into aircraft chains (routes) called a chain decomposition. Each chain visits periodically a set of airports and is served by several cockpit crews circulating along the airports of this set. The initial step in our approach is to find the minimal number of aircraft needed to carry out the flight schedule. This is achieved by using the fleet size theorem based on a DF representation of an aircraft flight schedule. A DF is a step function associated with an aircraft terminal which changes by + 1 and - 1 at flight departure and arrival times, respectively. DF theory was developed in the 1960–70s by Linis and Maksim (1967) and Gertsbakh and Gurevich (1977). Although the initial application of DFs was to the Russian AEROFLOT fleet it has subsequently attracted more attention on bus scheduling than aircraft scheduling. Here we discuss the revival of this method and its crucial use to construct the so-called chain decomposition of the schedule for a single period. We provide a justification for maximizing the number of balanced chains (flight sequences with the same start and end terminals). To do this we propose The Maximal Balanced Chain Problem. These are then converted into a set of infinite periodic flight sequences, each of which can be carried out by a single aircraft. The conversion is carried out by mapping the single period set of chains into an Euler graph. To construct the set of multiperiod chains that are “balanced” (return to the same terminal at the start) we find all edge disjoint cycle covers of the Euler graph using a modified version of Hierholzer's algorithm. These cycles are converted back into a balanced multiperiod chain solution and modified to conform to any maintenance constraints. To insure maintenance check constraints are satisfied for multiperiod chains, it may be necessary to add deadhead flights. Minimizing the cost of deadhead trips and overnight stays provide the basis for selecting an optimal routing solution.

1. Introduction

Because airline scheduling is a very complex problem, it has been tackled by decomposing it into a set of sub problems. The process starts with the design of a flight schedule involving a flight network, based on which markets to serve and their customer demands. Then a set of flights or “flight legs” to meet this demand are determined, each defined by its departure and arrival time. This is followed by the fleet assignment problem which assigns flights to fleets of different aircraft types. Then aircraft routes are determined as the sequence of flight legs flown by individual aircraft. Finally, a crew schedule is determined which consists of a crew pairing followed by crew rostering. In this paper we are interested in the aircraft fleet routing problem.

As the aircraft represents the largest cost associated with the operation of an airline, this prompted us to focus on the problem of finding the least number of aircraft required to meet the demand of a

passenger flight schedule (FS). When reviewing the literature, we found that most problems start with the assumption that the fleet size and its composition are already fixed. For example, in the aircraft assignment problem it is known at the start that the airline has a fixed number of aircraft of each aircraft type. The solution separates this heterogeneous fleet into homogeneous sub fleets of identical aircraft types assigned to different flight collections. There is no consideration whether the given fleet size is optimal or not. An optimal fleet size is the minimal number of aircraft needed to service all flights assigned to it. If this differs from the original fixed number of aircraft, then there is a need to either rent or buy additional aircraft or dispose of surplus aircraft (both most likely at inflated market prices).

Our primary goal then is to find the minimum fleet size for a given fleet type required to service the FS followed by a chain decomposition (CD). A CD is a decomposition of the flights in the FS into chains of sequential flights (or routes) carried out by individual aircraft in the

* Corresponding author.

E-mail addresses: helman@bgu.ac.il (H.I. Stern), elyager@bezeqint.net (I.B. Gertsbakh).

<https://doi.org/10.1016/j.orp.2019.100104>

Received 5 August 2018; Received in revised form 28 January 2019; Accepted 26 February 2019

Available online 06 March 2019

2214-7160/ © 2019 Published by Elsevier Ltd. This is an open access article under the CC BY-NC-ND license (<http://creativecommons.org/licenses/by-nc-nd/4.0/>).

fleet. Each flight leg must be included in exactly one chain. A chain/route must start and end at the same terminal, have feasible flight pair joinings (a flight arriving at a terminal is followed by a flight departing from the same terminal after an acceptable layover time) and satisfy maintenance constraints. This is referred to as the aircraft routing problem. Barnhardt et al. [1] defines a chain (or string) as maintenance feasible, if it satisfies all Federal Aviation Administration and carrier-specified maintenance requirements. The maintenance checks require each aircraft to undergo checks for every 60 h of flying time. However, some airlines may require more severe flying time checks. The maximum time between checks are typically restricted to three to four calendar days. As these checks can be quite long (typically 4–6 h), these checks are performed at night. These checks can be conducted at the aircrafts home base or a designated terminal, where it is assumed the necessary equipment and labor are available. We do not consider the long-term maintenance checks which are typically performed once a year.

The aircraft routing problem has been addressed by many researchers in the past, especially as an integer mathematical programming problem. [2–7]. We adopt a simpler approach based on deficit function (DF) theory developed in the 1960–70s by Linis and Maksim [8] and Gertsbakh and Gurevich [9], primarily because of its transparency, visual appeal, and polynomial complexity. Each DF is associated with a terminal and is a step function which has unit changes at flight departure and arrival times. The DF is a discrete multimodal function with regions of maximal values between which are valleys referred to as “hollows”. This admits to building chains by joining flight arrivals to departures. After a chain is completed, its flights are removed and the DFs are redrawn with the remaining flights. Then other chains are extracted until no flights remain. Each chain then becomes a single aircrafts route.

It is desirable to obtain a CD in which all its chains are fully balanced, that is one in which each aircraft returns to the terminal from whence it started. If the aircraft does not return in time to undergo required maintenance checks, it is necessary to add costly empty (deadheading) flights to bring the aircraft associated with an unbalanced chain and its crew back to the terminal from which it originated. To avoid this situation, we consider a 2-stage process: First a CD is found containing the maximal number of single period chains; and secondly, we offer a method to convert any unbalanced chains into balanced multiperiod chains (MPCs). A MPC is a concatenation of single period unbalanced chains. It should be noted, that although this investigation addresses the airline schedule problem it has applications to other domains as well, such as ground transportation and even job-machine scheduling.

The paper is organized as follows. Section 2 provides some background on DFs and aircraft routing. Section 3 describes the DF approach and the minimum fleet size theorem. Section 4 describes a method for determining a CD. Flight joining rules are introduced with the aid of hollow analysis. In Section 5 the procedure for finding a balanced aircraft routing that insures a *daily repeating schedule of flights*. This proceeds in three stages: Stage 1 - Find the Maximal Single Period Balanced Chain Decomposition, Stage 2 - Converting all single period unbalanced chains into a **Basic Set** of balanced MPCs. This stage employs an Euler graph for converting the unbalanced chains in a CD into a ‘basic set’ of MPCs, Stage 3 - Expanding the basic set of MPCs in order to create a daily repeated aircraft schedule. In 6 the routing solution for a 30 flight - 4 terminal example in terms of a periodic set of balanced MPCs is displayed. Sections 7 and 8 provide a method to ensure that the maintenance constraints are satisfied for each aircraft route, and a means for finding the optimal routing solution among alternatives, respectively. In Section 9 an analysis and comparison of the presented DF model with other approaches such as integer programming formulations is given. The final section provides a conclusion.

2. Background on deficit functions

Much of the research on aircraft routing assumes a single fleet for a fixed type of aircraft. Two approaches have been taken. The **first** is to solve a fleet assignment problem where the numbers of aircraft of different types, owned by the airline, are given constants. The fleets of each aircraft type have different capabilities to service flights according to the number of passengers, range of travel, costs, etc. The solution to the fleet assignment problem decomposes the set of flights in the original flight schedule into separate subsets to be operated by fleets of a single type. Mancel et al. [10] provide a state of the art for the airline fleet assignment problem. Most authors formulate the problem as an integer linear program. For example, Ozdemir et al. [11] solve a fleet assignment problem using Turkish Airlines data and Markus et al. [12] apply it to Lion Air in Indonesia. The **second** approach is to determine directly the minimum fleet size (of a given type) required to service the FS assigned to the fleet. Once the minimum number of aircraft is determined, the routes, or chain of flights, for each aircraft is found. This in fact partitions the FS into subsets, each being serviced by a single aircraft.

The seminal paper of Dilworth [13] on a decomposition theorem of partially ordered sets was the first to address this problem. Methods such as mathematical programming and network flows, were carried out in the 1950s and 70s by Dantzig and Fulkerson [14] Bartlett [15] Salzbom [16]. Then there is the infinite vehicle chain problem approached from a theoretical point of view of periodic scheduling problems and partial orders of flights Serafini and Ukovich [17] Orlin [18] Gertsbakh and Serafini [19]. In parallel, around the decade of the 70s, we see the start of the DF approach for solving minimum fleet size and CD problems. DFs were first introduced by Linis and Maksim [8] Gertsbakh and Gurevich [9] Gertsbakh and Gurevich [20] for airline scheduling. An English translation of Linis and Maksim’s 1967 paper [8] is provided in Linis and Maksim [21], along with a discussion of its merit in Gertsbakh et al. [22]. This thread has been continued with a few scattered papers in the 80s, and subsequently by the work of Ceder and colleagues with regard to public transit bus and rail scheduling. Liu and Ceder [23] provide an excellent 50-year retrospective of the use of the DF approach related to public transport. Lui and Ceder [24] incorporate the DF approach to help solve an integrated public transit timetable and vehicle scheduling problem.

The insertion of “deadheading trips” to further decrease the fleet size was developed by Ceder and Stern [25], Stern and Ceder [26] in the context of bus transit scheduling. A second method for reducing the fleet size, using possible shifts in departure times within given tolerances, is described in Ceder and Stern [27]. DFs have also been applied to machine job scheduling by Gertsbakh and Stern [28]. In a recent paper, Gertsbakh and Stern [29] describe the use of DFs for crew planning and rostering in aviation.

3. Deficit function and minimum fleet size theorem

3.1. Definitions and notations

Let $I = \{i: i = 1, \dots, n\}$ denote a set of required flight legs. The flights are conducted between a set of terminals (airports) $K = \{k: k = 1, \dots, q\}$. Each flight is to be serviced by a single aircraft and each aircraft is able to service any flight. For a flight i departing from terminal k_a^i and arriving at terminal k_b^i let t_a^i and t_b^i represent its departure and arrival times, respectively. The arrival time includes an extension of the duration of each flight by a minimum turn-time. A flight leg i is represented as a quadruple $(t_a^i, t_b^i, k_a^i, k_b^i)$. A flight schedule FS is a set of all flights $\{(t_a^i, t_b^i, k_a^i, k_b^i): k_a^i, k_b^i \in K, i \in I\}$. Two flights i, j may be serviced sequentially (feasibly joined) by the same aircraft only if the precedence relation R is satisfied.

$$R: i < j \Rightarrow t_a^i \leq t_a^j \text{ and } k_a^i = k_a^j \quad (1)$$

Here the arrival time has been prolonged to include the minimum turn time between any two flights. Denote $[0, T]$ as a daily schedule horizon (say 24 h) where flights are excluded from crossing the 24:00 h line. All flights depart and arrive within this time interval, i.e., $0 \leq t_a^i < t_a^i \leq T$. Such a daily FS is said to be balanced.

A deficit function, $DF(k, t)$, is a step function defined for each terminal k , whose value at time t is equal to the number of flight departures less the number of arrivals over the interval $[0, t]$. The function changes by $+1$ and -1 at flight departure and arrival times, respectively. At these times, the DF is right-continuous

Theorem 1. Minimum Fleet Size Theorem

The minimum number of aircraft required to service a FS equals the sum of the Max values of q DFs. Here m represents the minimum fleet size.

$$m = \sum_{k=1}^q \text{Max}\{DF(k, t) | t \in [0, T]\} \tag{2}$$

Proof. The proof may be found in Linis and Maksim [8].

Corollary 1. The DFs also provide the starting numbers of ac at each terminal k as:

$$\text{Max}\{DF(k, t) | t \in [0, T]\}$$

3.2. DF as a function of plateaus and hollows

A DF is a discrete step function containing multiple regions or intervals of maximal values (plateaus). Regions between plateaus are denoted as hollows. Let the maximal value of $DF(k, t) = DF(k)$. Let $r(k)$ equal the total number of such maximal regions for $DF(k)$ defined by a tuple of adjacent points $M_r^k = [s_r^k, e_r^k]$, $r = 1, \dots, r(k)$. Here, r represents the r th maximal interval ordered from the left. Note, s_r^k and e_r^k represent departure and arrival flights from and to terminal k , respectively (not necessarily the same). The one exception occurs when the DF reaches its maximum value at the end of the horizon in which case $M_{r(k)}^k$ has a departure not followed by an arrival, and $e_{r(k)}^k = T$. Define the set of hollow intervals (regions) for each $DF(k)$ as; $H_0^k = [0, s_1^k]$, $H_1^k = [e_1^k, s_2^k], \dots, H_{r(k)}^k = [e_{r(k)}^k, T]$. H_0^k and $H_{r(k)}^k$ may consist of only one point. It is now possible to describe the partition of the schedule horizon of $DF(k, t)$ into a sequence of alternating hollow and maximal intervals, i.e., $(H_0^k, M_1^k, H_1^k, \dots, M_r^k, H_r^k, \dots, M_{r(k)}^k, H_{r(k)}^k)$. An example of a FS named FS30 is given in the Appendix. The FS and its corresponding set of DFs is displayed in Figs. 1 and 2, respectively.

In Fig. 1, we see that all flights are carried out within a daily 24 h schedule. A homogeneous fleet is assumed allowing any aircraft to carry out any flight. For this example, we see from the DF's max values that the minimum fleet size is 12. From Corollary 1 we see that the starting number of aircraft at terminals A, B, C, D are 1, 4, 4, 3, respectively.

4. Chain decomposition (aircraft routing)

Below we show how to use DFs for a decomposition of the FS into vehicle routes or chains. It should be noted, that aircraft route and chain are used interchangeably.

A Chain (C) is an ordered sequence of flights $[f_1, f_2, \dots, f_i, \dots, f_c]$ such that, $t_a^i \leq t_a^{i+1}$, and $k_a^i = k_a^{i+1}, \forall i = 1, \dots, c-1$. A set of chains is a CD; if all flights in a chain are serviced by a single aircraft, and each flight $i \in I$ is included exactly once in a chain.

It is convenient at this point to introduce the importance of hollows in the construction of aircraft chains.

4.1. Hollow analysis

Hollows are an important region of the DF to look for possible

joinings between arrival and departure flights to form chains. In order for a flight j to be joined to a previous flight i it needs to be a feasible joining satisfying (1). In fact, the 2 hollows in $DF(C)$ allow for a single feasible joining. We refer to such hollows as a V hollow of depth 1. A V-Hollow of depth n is monotone decreasing and increasing and corresponds to a sequence of n arrivals followed by n departures with the number of possible multi-joinings is $n!$. Any arrival in a V-Hollow can be joined with any departure. However, there are more complicated hollows which we refer to as U-Hollows.

A U-Hollow also has n arrivals and n departures, but exhibits at least one arrival event which is preceded by one or more departure events. For U-Hollows, with n arrivals and departures, the number of feasible joining pairs is less than $n!$. This feature implies that the number of CDs is finite and bounded.

4.2. Feasible flight joinings

The problem of finding the number of feasible pairs of joinings can be formulated as an assignment problem with inadmissible cells whereby only a feasible solution is sought. The inadmissible cells are determined by identifying the cases in which an arrival event is preceded by one or more departure events. For example, H_3^B in Fig. 2.

Theorem 2. (Necessary Hollow Joinings) In creating chains, an arriving flight in any hollow must be connected to a departing flight in the same hollow, in order to achieve a chain decomposition equal to the minimal fleet size. Conversely, connecting an arrival flight in a hollow to a departing flight in any other hollow will result in a chain decomposition exceeding the minimal fleet size.

Proof. A proof can be found in Gertsbakh and Gurevich [9].

Corollary 2.1. For any feasible CD, it is necessary that the single arrival and departure of all Hollows of depth 1 must be joined.

Proof. The theorem states for “any hollow”. A Hollow of depth 1 is the simplest hollow.

Corollary 2.2. The two flight legs comprising a hollow of depth 1 may be replaced by a single representative flight.

Let t_a^i and t_d^i be the arrival and departure times of flights i and j defining a hollow of depth 1 in terminal k . According to Corollary 2.1 flights i and j must be joined. Hence, they can be replaced by a dummy representative flight $f_{ij} = (t_a^i t_a^j k_a^i k_a^j)$, where $t_a^i = t_a^j$, $t_a^j = t_a^i$, $k_a^i = k_a^j$, $k_a^j = k_a^i$.

4.3. Chain decomposition

Proposition 1. Given a FS and its associated set of q DFs. Let m equal the minimum fleet size. Then m flight chains C_1, C_2, \dots, C_m may be constructed.

Proof. Gertsbakh and Gurevich [9].

Such a construction is said to be a CD, and may be determined by the following algorithm.

Chain Decomposition Algorithm

For a FS of n flight legs, q terminals, and a fleet size of m .

Initialization: Replace all necessarily joined jobs in hollows of depth 1 by a single job according to Corollary 2.2.

- 1 Starting from the maximal valued DF, say $DF(k, t)$. Select any flight i with its starting point $t_a^i \in H_0^k$. Find the arrival time of this flight, t_a^i , at some terminal k_a^i . Let $k = k_a^i$ and join it to a feasible flight departure $t_d^j \in H_r^k$ according to Eq (1). Continuing joining a flight arrival with a departure within each hollow visited, until some flight v arrives at some end hollow $H_{r(k)}^k$ at time t_a^v . Here no departure flight is available to feasibly connect a flight v .
- 2 Remove the chain of flights from the DFs. Recompute the DFs. The value of the total Max DF decreases by one.

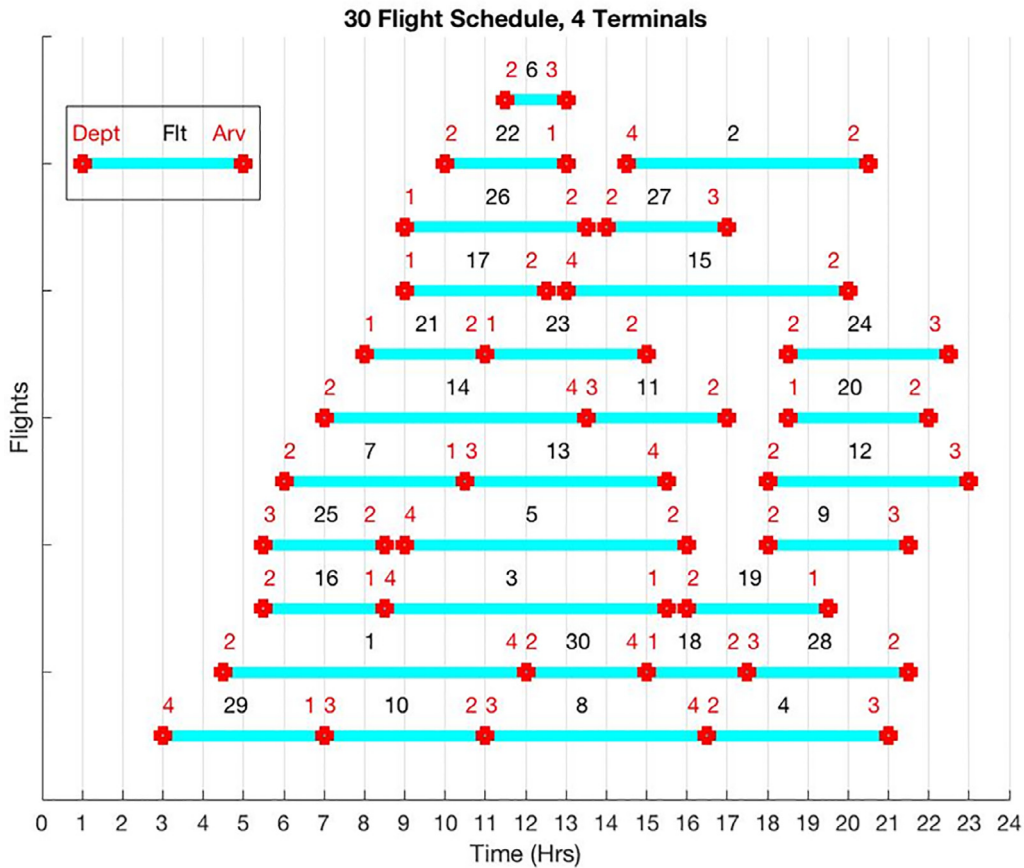


Fig. 1. The flight schedule FS30 ($k = 1, 2, 3, 4 = A, B, C, D$).

- 3 Repeat until all $DF(k)$ s have $MaxDF(k) = 0$
- 4 This results in a feasible CD.

Gertsbakh and Gurevich [9] also prove that the algorithm terminates after a finite number of m steps.

Proposition 2. The Chain Decomposition Algorithm is of polynomial complexity $O(nq)$.

Proof:

Step 1. Find max valued $DF \Rightarrow O(q)$. Find a flight number in the FS array $\Rightarrow O(n)$.

Step 2. Sequentially check list for element $k > arvertime$, and if $k < current\ min \Rightarrow O(2n)$

Step 3. Update FS and n by removing the flights in the chain $C(i)$, by finding the flight number in a list $\Rightarrow O(n)$. Update DFs and remove flights in FS, Precompute plateaus and hollows. All that is needed is a list of max values of each DF (Find all max values of a list and repeat for each $DF \Rightarrow O(nq)$)

Repeat all m times: $m[O(q) + O(n) + O(2n) + O(n) + O(nq)] = (q + 4n + nq)m O(nq)$.

5. Finding balanced aircraft routes

Given a CD comprised of a set of chains C of cardinality m . If for a given chain $C = [f_1, f_2, \dots, f_i, \dots, f_c]$, the start and end terminals are the same, i.e.; $k_d^{f_1} = k_a^{f_c}$ then the chain is said to be a **balanced chain**. Otherwise it is an **unbalanced chain**. These chains cover a single period and are referred to as **IT chains**. A C can be partitioned into β and $\bar{\beta}$ of balanced and unbalanced chains, respectively.

It is rare that a **IT CD** is fully comprised of balanced chains. In order

to obtain a **IT CD** (a set of aircraft routes) that are fully balanced it is necessary to add costly empty (deadheading) flights to bring the aircrafts associated with the unbalanced chains and their crews back to the terminal from which they originated. To avoid this situation a method is proposed to convert the set of unbalanced **IT** chains into a family of balanced chains. Such chains are formed by concatenating several **IT** unbalanced chains into **Balanced MPCs**. Also, longer MPCs increase the number of nights and the cost for crew and aircraft to be away from the home base. Finally, the number and length of MPCs subsequently increases the complexity of finding crew pairing solutions. All of this provides a justification for finding a **IT CD** that has the minimum number of MPCs, or equivalently a CD that has the maximum number of **IT** balanced chains. Thus, we propose a **Maximal IT Balanced Chain Decomposition problem (MBCP)**.

As our final goal is to obtain balanced aircraft routings that insure a daily repeating schedule of flights we propose a three-stage process. In Stage 1 we solve the **MBCP**. In Stage 2 we convert any remaining **IT** unbalanced chains into a **Basic Set of MPCs**. In Stage 3 we expand the **Basic set of MPCs** in order to create a daily aircraft schedule.

5.1. Stage 1: finding the maximal balanced chain decomposition

Here we wish to find the **Maximal IT Balanced Chain Decomposition** among all possible CDs. Let S represent the set of all possible CDs associated with a FS where, each CD is a set C of chains of cardinality m ($m = \min$ fleet size). Let $C_i(k) \in S$ be the i th set of chains partitioned into β and $\bar{\beta}$ of balanced and unbalanced such that the cardinality of β is k . Partition S into subsets $S(k)$; $k = L(k), \dots, U(k)$, where $S(k)$ is the set of all CDs of the form $C_i(k)$ and $S(k) = \emptyset, \forall k \in (L(k) \text{ and } \forall k)U(k)$. Note, that $L(k)$ and $U(k)$ are not known a priori. Then the **MBCD** problem is defined as find the subset:

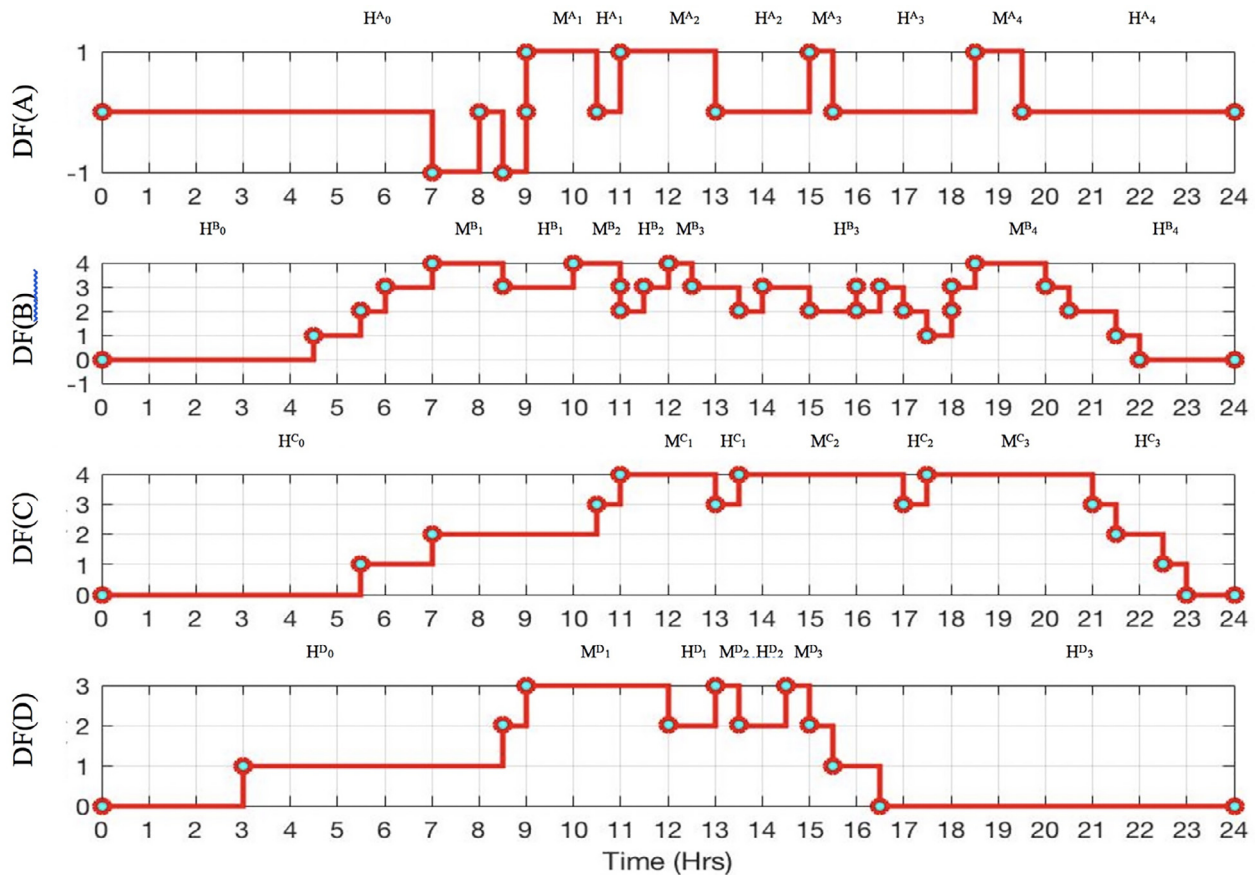


Fig. 2. Deficit functions for flight schedule FS30, $k = 1, 2, 3, 4$ (A, B, C, D).

$$S(k) \ni k = \text{Max}\{k: k = L(k), \dots, U(k)\}, i. e. ; k = U(k) \quad (3)$$

The MBCP can be formulated as a multi commodity network problem which is NP-hard, and therefore we provide a probabilistic heuristic method. For this purpose, we have developed a method for generating random chains and selecting the “best” from this set. To generate a random chain each arrival in a hollow is joined to a randomly chosen (feasible) departure in the same hollow. These arrival - departure pairs are then placed in a “list of flight joinings”. A chain builder procedure then traces through the pairs to extract a chain and the list is updated. The procedure terminates when the list is empty. The formal algorithm follows.

5.1.1. Random Chain generation algorithm

- 1 Denote each pairwise arrival - departure joining as a tuple (i,j) .
- 2 Combine all tuples into a list of joinings J .
- 3 Chain Builder (Create a Chain from J)
 - a Define a **flight chain connection rule (FCCR)**. For any two tuples in J , if $(i - j)$ and $(j - k)$ connect them into a 3-flight chain of the form $(i - j - k)$.
 - b Begin with a Start tuple $(0-i)$ in J
 - c Using the FCCR find the tuple (i,j) and connect it to $(0-i)$ to obtain $(0-i-j)$. Find the tuple (j,k) connect to $(0-i-j)$ to obtain $(0-i-j-k)$. Continue until an end tuple $(\nu-0)$ is found.
 - d The resulting chain is $C = [0-i-j-k-\dots-\nu-0]$
 - e Update J .
- 4 After the chain is found remove the tuples from J that formed it.
- 5 If J is empty **STOP**, otherwise return to Step 3.

5.1.2. Balanced chains and 100 random CDs

For the example FS30 we generated 100 random CDs. Let $RS(k)$

denote the set of randomly generated CDs with k balanced chains. Let the cardinality of $RS(k)$ be $r(k)$. The distribution of $r(k)$ is shown in Fig. 3. This figure helps us understand the underlying structure of the solution space of balanced CDs.

In column 3 of Table 1 ($Term(s,i) - Term(e,i)$) represents the i th chain's start and end terminals.

5.2. Stage 2: converting unbalanced chains into a basic set of balanced MPCs

At this stage we describe a method to convert the set of unbalanced IT chains into a family of balanced chains. We introduce the notion of a chain period equal to the number of unbalanced IT chains combined to form a MPC. For example, consider two unbalanced IT chains, represented by their terminal endpoints (A-B) and (B-A). Then joining (B-A) to (A-B) forms a chain [A-B-A] with a chain period of $2T$. Note, connecting the last flight, say x , in B-A to the first flight, say y , in A-B is a feasible joining because it occurs at the same terminal A and $t_a^x > t_a^y$ is satisfied since y departs the day after x arrives. If the schedule is balanced there always exist a set of balanced MPCs. This conversion can be carried out by constructing an Euler graph, and using a modified version of Hierholzer's algorithm, to extract a set of disjunctive cycles. Each cycle becomes a MPC. As an example, we show, using the CD RS38, how to construct the associated Euler graph, and from it extracting the basic set of balanced MPCs. Also, after how to take a basic MPC and after applying circular shifts generate shifted copies in order to insure all flights in the FS are serviced daily.

Definition. A MPC is a balanced chain comprised of n ordered unbalanced IT chains $[C_1, C_2, \dots, C_i, C_{i+1}, \dots, C_n]$. Each pair of successive unbalanced chains C_i, C_{i+1} has $Term(e, i) = Term(s, i + 1)$. Also, as a multiperiod balanced chain $Term(s,1) = Term(e, n)$.

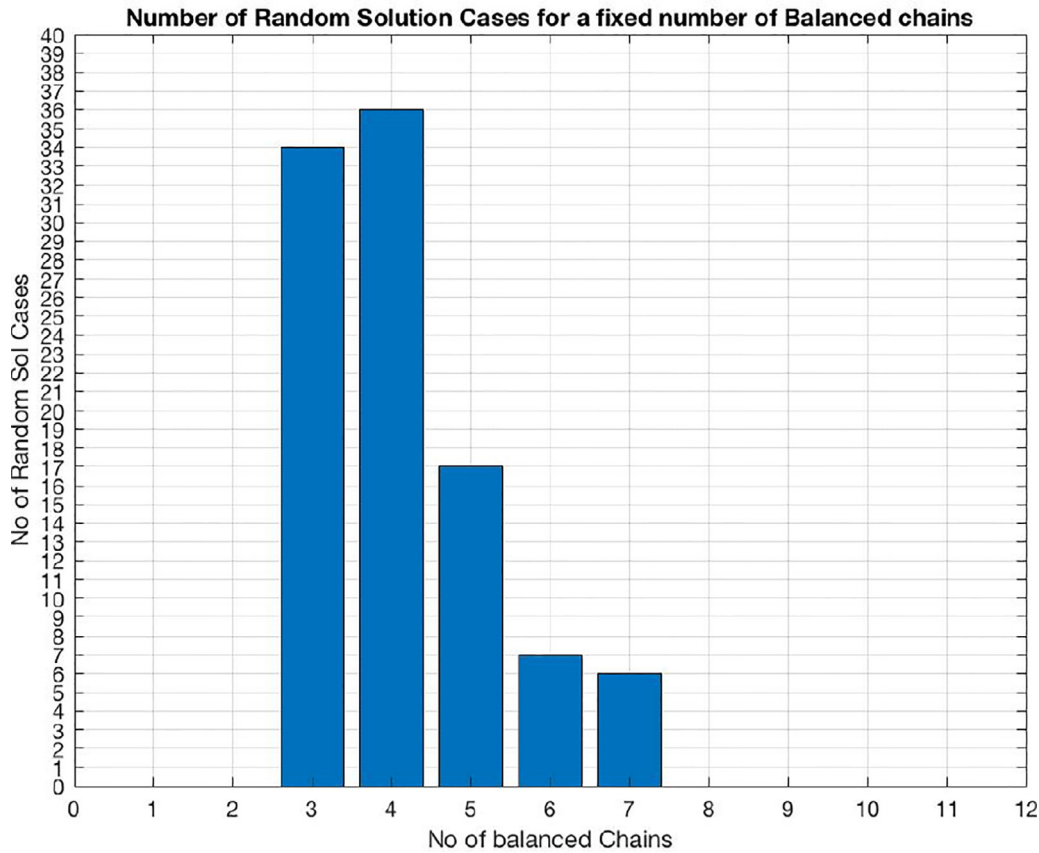


Fig. 3. The distribution of $r(k)$ for example FS30.

Table 1
Chain decomposition for RS38 (* = balanced chain).

Chain	Flights	(Start-End)
1	[17-19]	(A-A)*
2	[1-15]	(B-B)*
3	[16-26-27-28]	(B-B)*
4	[14-2]	(B-B)*
5	[25-22-18-9]	(C-C)*
6	[10-6-11-12]	(C-C)*
7	[29-21-30]	(D-D)*
8	[13]	(C-D)'
9	[5-24]	(D-C)
10	[8]	(C-D)''
11	[3-20]	(D-B)
12	[7-23-4]	(B-C)

A MPC does not have any interior balanced sub chains. It is convenient at times to refer to an MPC with n ordered IT chains as a nT MPC. Such a chain is said to have a Chain Period of nT .

The following propositions will be useful in designing an algorithm to perform this conversion.

Proposition 3. Given a Balanced FS, and its CD with a set of chains $C = \{C_1, C_2, \dots, C_k, \dots, C_m\}$. Let C_k have a start flight i departing from $Term(s, i)$, and an end flight j arriving at terminal $Term(e, j)$ Let C_s be the set of chains departing from $Term(s, i)$, and C_e the set of chains ending at $Term(e, j)$ where $C_s \subset C$ and $C_e \subset C$ then the total number of chains with $Term(e, j)$ and the total number of chains with $Term(s, i)$ are equal, i.e.; $|C_s| = |C_e|$.

Proof. This is true since IT chains must start at some terminal and end at another terminal within a single time period (since the FS is balanced i.e.; all flights start and end within a single time period)

Proposition 4. Given a Balanced FS, and a CD, for all chains in $\bar{\beta}$ the total

number of chains whose starting terminal is $Term(s, i)$ equals the total number of chains whose ending terminal is $Term(e, j)$, where for a given chain in $\bar{\beta}$ $Term(s, i) \neq Term(e, j)$.

Proof. This follows from Proposition 3 after removing all balanced IT chains from the CD.

For example, in Table 1, the number of unbalanced chains that start with terminal C equals 2 as do those that end with terminal C. This is also true for terminal D. Also, chain 12 starts at B and chain 11 ends at B.

5.2.1. Euler graph and Hierholzer's algorithm

Euler's Theorem, first given by Hierholzer in 1873 states: A directed graph has an Eulerian Cycle if the graph is connected and all vertices have even degree Biggs et al. [30]. To convert IT unbalanced chains to balanced MPCs. Define a directed graph $G(E, V)$, where V is a set of vertices and E is the set of directed edges (i, j) from vertex i to j . Let each vertex in G correspond to a terminal in V . Let each unbalanced chain i in $\bar{\beta}$, with $Dept Term(s, i) \neq Arv Term(e, j)$, be represented as a directed arc (i, j) in E . Balanced IT chains are loops at a single node, and after stripping away looped edges corresponding to balanced chains it becomes a directed Euler graph

(see Fig. 4.). All vertices are of even degree according to Proposition 4. A cycle in an Euler graph comprised of n directed edges $(1, 2), (2, 3), \dots, (n, 1)$, corresponds to a nT MPC of the type $[1-2 \ 2-3 \ \dots \ n-1]$. Hierholzer's algorithm (1873) finds a set of disjunctive cycles as it traverses a Euler graph. We denote this set of disjunctive cycles as the basic set.

5.2.2. Finding the basic MPC set

Theorem 3. Conversion Unbalanced to Balanced

Given a balanced FS whose IT CD chains are partitioned into β and

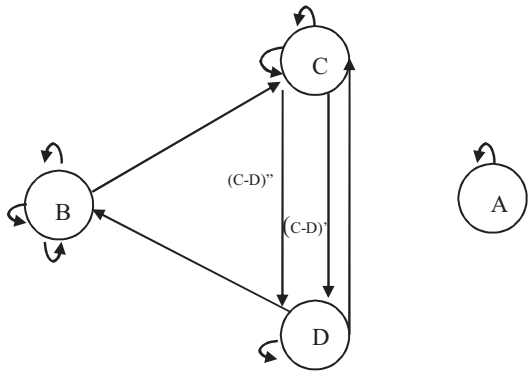


Fig. 4. Euler Graph for Example RS38.

$\tilde{\beta}$. There exists a conversion of $\tilde{\beta}$ into a set of balanced MPCs (Using a modified Hierholzer's algorithm). We denote this set as the basic set of MPCs.

Proof. After constructing an Euler graph from the set of unbalanced single period chains, according to Proposition 4 each terminal type appears the same number of times at the start and end of the chains. This insures that each vertex in G , corresponding to a terminal in V , is of even degree. A cycle in an Euler graph comprised of n directed edges $(1, 2), (2, 3), \dots, (n, 1)$, corresponds to a nT MPC of the type $[1-2-2-3 \dots n-1]$. Each edge represents an unbalanced single period chain between a pair of vertices representing its start and end terminals. Traversing each cycle from any of its vertices and back covers a sequence of edges (unbalanced chains). Starting and ending at the same vertex is tantamount to starting and ending at the same terminal, and hence the cycle represents a balanced multiperiod chain.

Corollary 3.1. The conversion of a set of unbalanced chains from a feasible CD results in a basic set of MPCs such that each $1T$ unbalanced chain appears exactly once in some MPC and the set of MPCs contains all $1T$ unbalanced chains.

Proof. This follows from the fact that a complete cover of the Euler graph provides a set of disjunctive cycles.

The CD chains for example RS38 are shown in Table 1. Note, since chains 8 and 10 have different flight sequences, we denote them by C-D' and C-D'', respectively. Balanced chains are indicated by an asterisk. In the CD, there are 7 balanced $1T$ chains, the remaining are unbalanced $\tilde{\beta} = \{(B-C), (C-D)', (C-D)'', (D-B), (D-C)\}$. However, according to Proposition 4 for each terminal type that appears at the start of an unbalanced chain it also appears end of some other unbalanced chain. The corresponding Euler graph is shown in Fig. 4. where the loops represent the $1T$ balanced chains. After stripping the loops, we are left with a 3 vertex directed connected Euler graph.

Referring to the Euler graph in Fig. 4 start from an arbitrary node B and traverse the unvisited edges (B, C), (C, D)', (D, B). Remove this cycle from the graph, but retain nodes C and D in the reduced graph as they have unvisited arcs attached to them. Select any one of the retained nodes, say C, and traverse the arcs until a second cycle (C, D)'', (D, C) is obtained. We denote this set of disjunctive cycles as a basic set: [(C-D)'', (D-C)], [(B-C) (C-D)' (D-B)]. Here we have one $2T$ MPC and one $3T$ MPC. Note, because of the two arc traversal options from vertices C to D, there exists a second basic set from the set of disjunctive cycles [(C-D)' (D-C)], [(B-C) (C-D)'', (D-B)].

The conversion takes $O(E)$ time, using Hierholzer's algorithm for an Euler graph $G(E, V)$, since each arc is traversed exactly once.

5.3. Stage 3: Expanding the basic set of MPCs

Because the basic set of MPCs has "collected" the unbalanced $1T$ chains to cover a span of several periods, it is no longer the case that all

flight legs are serviced on a daily basis. Thus, it is necessary that all interior $1T$ chains of the MPCs be clockwise circular shifted in order to ensure that all flight legs in the FS appear in a daily schedule. The number of disjunctive cycles and hence the number of basic MPCs is less than the number of unbalanced $1T$ chains. This is true because a cycle is comprised of at least 2 $1T$ unbalanced chains. It follows that the number of basic MPCs plus the number of $1T$ balanced chains is less than m . Because the unbalanced chains have been collapsed into the basic chains, we need to resurrect the total chain count to m . This can be done by expanding each MPC in the basic MPC set into additional MPCs, in order to restore the total number of MPCs (including the $1T$ balanced chains) to m chains. This is necessary because after the concatenation, the $1T$ chains in the basic MPC appear in successive periods. In order expand the basic MPCs, to ensure that all flights in the FS are flown each day, we need to apply several Clockwise Circular Shifts (CSS)s to each basic MPC. This will insure the construction of an aircraft routing that includes a daily repeating schedule of flights.

Corollary 3.2. Clockwise Circular Shifts: Given a MPC of n ordered $1T$ chains (a chain period of nT), where $MPC = [C_1, C_2, \dots, C_i, C_{i+1}, \dots, C_n]$. The chain can be split at any interior point into two parts (say, between C_i and C_{i+1}) and a CCS operation applied. The new $MPC^* = [C_{i+1}, \dots, C_n, C_1, \dots, C_i]$. After $n-1$ CCS operations the MPC provides $n-1$ offspring MPC^* s. When added to the original MPC a set of n different nT MPCs is obtained. The CCS operation preserves the balanced aspect of the MPC.

Proof. After applying the CCS operation to all possible interior points, $n-1$ offspring MPC^* s are obtained. These $n-1$ MPCs each of period nT , plus the original nT MPC results in n different nT MPCs. Also, after the CCS the new MPC is balanced. Let the shift point for the MPC be between C_i and C_{i+1} . Recall for a feasible MPC each pair of successive unbalanced chains C_i, C_{i+1} has $Term(e, i) = Term(s, i + 1)$. After the CCS, the first $1T$ chain becomes C_{i+1} as with starting terminal $Term(s, i + 1)$ and the last $1T$ chain becomes C_i with ending terminal $Term(e, i)$.

For a basic set of MPCs, each nT MPC may be expanded in $O(n)$ operations because the CCS operation finds $n-1$ shift points and for each step it takes one operation to rearrange the two parts.

As an example, let a basic MPC with a chain period of $3T$ be [8] [3-20] [7-23-4]. Here we have 3 $1T$ unbalanced chains. The flights in periods 1, 2 and 3 are 8, 3,20 and 7,23,4, respectively. After providing 2 CCSs we obtain the following 3 MPCs: [8] [3-20] [7-23-4], [3-20] [7-23-4] [8][7-23-4], [8] [3-20]. It can now be seen that in each period all flights 8, 3, 20, 7, 23, and 4 are serviced daily. Continuing with example RS38 we create the set of all m balanced MPCs. Here the elements of each balanced MPC in the basic set are CCSed to obtain additional balanced MPCs. For the $2T$ basic MPC we obtain [(C-D)'', (D-C)] and [(D-C) (C-D)'']. For the $3T$ basic MPC we obtain [(B-C) (C-D)' (D-B)] and [(C-D)' (D-B) (B-C)] and [(D-B) (B-C) (C-D)']. This results in five MPCs (2 of length $2T$, and 3 of length $3T$) as shown in Table 2.

6. Aircraft routes for a daily repeating schedule of flights

Each $1T$ chain is repeated periodically and represents the trajectory of a single aircraft. All $2T$ chains in the basic set are repeated twice after being clockwise circular shifted. This represents the trajectory of two aircraft doing in fact the same sequence of flights with a shift by one T . In general, each nT MPC in the basic set is CCSed n times to represent the trajectories of n aircraft repeating periodically the same sequence of flights with shifts of $1T$ to $(n-1)T$. As an example, the complete set of aircraft trajectories for RS38 is shown in Table 2.

The terminals visited (shown in the 5th column) are those for the start-end terminals of the aircraft flight sequences. The last column shows the aircraft home base, and the MPC lengths in column 3 indicate the number of days to return home after leaving the home base. At each 24 h time period all of the 12 original chains whose flights constitute the entire number of flights in the FS are serviced by the 12 aircraft fleet as shown in Fig. 5. The figure shows three 24 h time periods, because

Table 2
Multi-period Aircraft Trajectories for RS38.

Aircraft Trajectory	Original Chain Sequence	MPC Length	Aircraft Flight Sequence	Start-End Terminals	Home Base
1	1	1T	[17-19]	A-A	A
2	2	1T	[1-15]	B-B	B
3	3	1T	[16-26-27-28]	B-B	B
4	4	1T	[14-2]	B-B	B
5	5	1T	[25-22-18-9]	C-C	C
6	6	1T	[10-6-11-12]	C-C	C
7	7	1T	[29-21-30]	D-D	D
8	8,9	2T	[13] [5-24]	C-D-C	C
9	9,8	2T	[5-24] [13]	D-C-D	D
10	10,11,12	3T	[8] [3-20] [7-23-4]	C-D-B-C	C
11	11,12,10	3T	[3-20] [7-23-4] [8]	D-B-C-D	D
12	12,10,11	3T	[7-23-4] [8] [3-20]	B-C-D-B	B

the longest MPC is of length 3T. This is repeated every 3 days in order to obtain a periodic set of aircraft routings.

7. Checking the maintenance constraints

The schedule must now be checked to ensure that the maintenance constraints are satisfied. If it is necessary to return to the aircrafts home base after 2 days in order to perform maintenance, then it is necessary to introduce deadheading trips [B-C], [C-D] and [D-B] at the end of the second period in the 3T aircraft trajectories of chains 10,11 and 12, respectively. All 1T and 2T trajectories automatically satisfy this requirement. After the deadheading trips are added feasible flight joinings are preserved by switching the last period of the 3T MPCs. To see this more clearly, consider chain 10 in Fig. 5 which starts at terminal C. After inserting a deadheading trip [B-C] at the end of period 2, the aircraft has returned to its base at terminal C for overnight

maintenance. However, now at the start of period 3 the aircraft it must continue from terminal B. It appears that to resolve this situation a second reverse deadheading trip [C-B] must be added. However, this is easily avoided by continuing with the flights in period 3 of MPC 11. Then MPC 11 continues with the 3rd period of MPC 12 and MPC 12 continues with MPC 10's 3rd period. The same principle applies if a maintenance constraint on the total air time of an aircraft is exceeded. In our case even a more severe check of every 40 h of total air flight time is satisfied, as it occurs after the 2-day return constraint. If the maintenance constraint requires each aircraft to return to its home base after every 3 days, then the current schedule need not be changed.

8. Finding the optimal routing solution

Many consider the aircraft routing problem one of finding a feasible solution. However, we have shown that there are many alternative

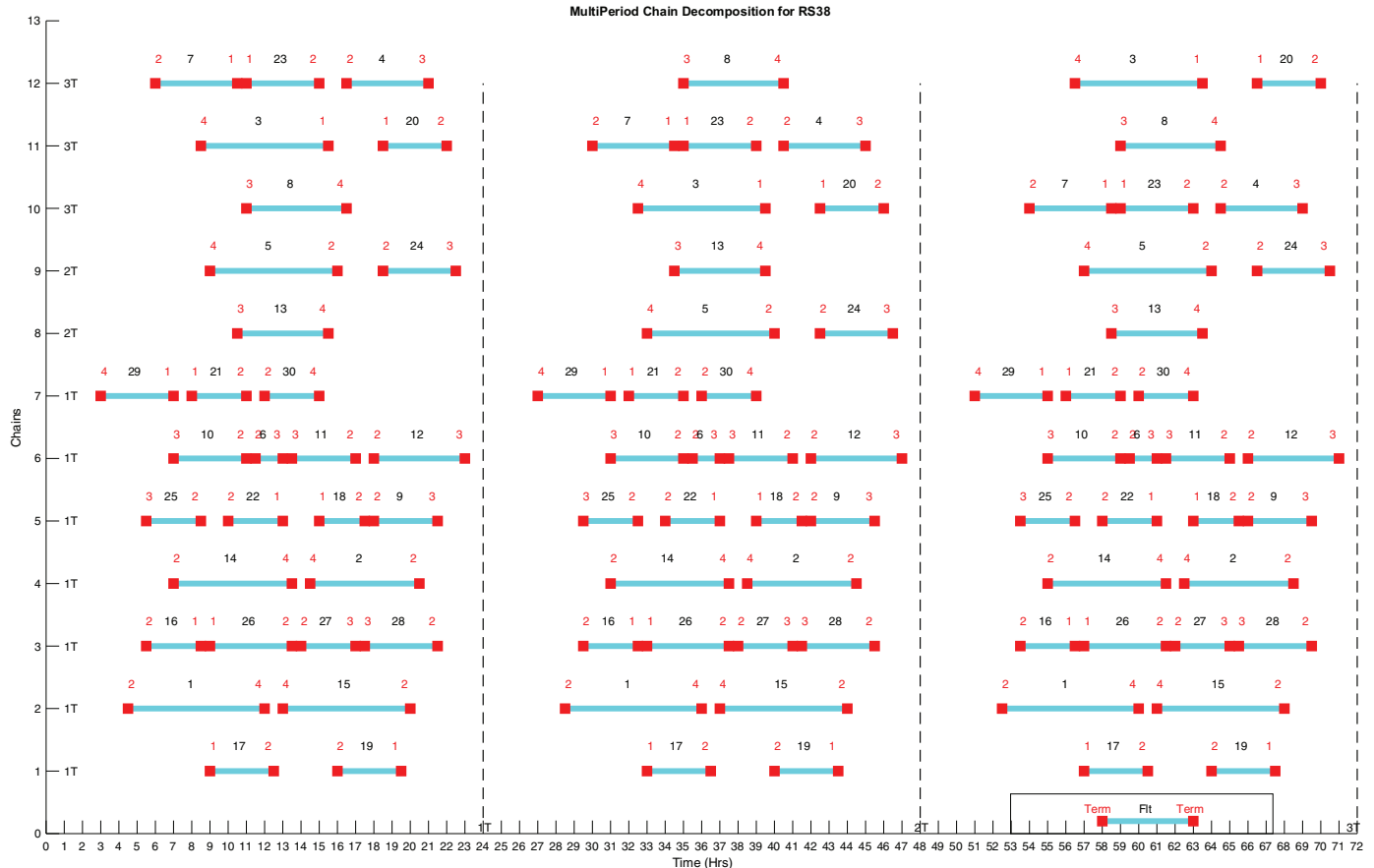


Fig. 5. Aircraft Trajectories for RS38 over a Chain Period of 3T.

feasible aircraft routing solutions in terms of the number of single period CDs. Moreover, because of the requirement that all chains be balanced it is necessary to introduce long multiperiod chains. These chains have increased anomalies that are costly. The key to our approach is to try to reduce the number of solutions that contain these long chains. We show that many of these long chain solutions can be paired out by solving a MBCP and recording all ties. Also, for each tied maximal balanced chain CD, there may be many alternative basic MPCs as a result of tied sets of disjunctive cycles that cover an Euler graph. Such sets will have the same structure in terms of the number and type of nT MPCs. This is noted at the end of section 5.5.2. where a second basic set for example CD RS38 was found. In terms of the MPCs in Fig. 5 it is possible to visualize the second set of balanced MPCs by switching flights 8 and 13. In order to compare alternative feasible MPC solutions (aircraft routings) we can consider the costs of inserting deadheading flights in addition to the cost of overnights when the aircraft and crew are away from their home base. Let the set of alternative routing solutions be R with cardinality r . Denote the i^{th} solution as R_i . Let the deadheading and overnight costs be r_{1i} and r_{2i} respectively. Let the number $RS(k^0)$ be the set of CDs such that k^0 is the maximal number of 1T balanced chains. For each solution ν in $RS(k^0)$, let the number of tied Euler disjoint cycle covers for ν be $D(\nu)$. Then the optimal solution R_{i^*} is found by solving (4) and (5).

$$Z = \text{Min} \{(r_{1i} + r_{2i})\}_{i=1, \dots, r} \quad (4)$$

where,

$$r = \sum_{\nu=1}^{|RS(k^0)|} \nu \cdot D(\nu) \quad (5)$$

9. Analysis and comparison of DF routing model with integer programming approaches

In this section we discuss the position of our work in the context of other approaches in the literature for solving the aircraft routing problem. In particular, those that incorporate integer programming mathematical solution methods. The most popular approaches are based on integer programming models whereby a feasible routing generator is used to provide a large collection of routings from which a desired number is selected based on a minimum cost objective. The main available techniques for addressing aircraft routing almost exclusively relied on up to now is column generation, whereby each column in the set covering integer LP represents a feasible route. Methods for the generation of such feasible routings are heuristic, often based on tree searches. Our random chain generation algorithm of Section 5.1.1 fits in nicely within this approach.

Aircraft routing solutions can be found in a number of combined approaches. For example, Barnhart et al. [1] imbed the routing problem and the aircraft assignment together in one mathematical formulation using sequences (called "strings") of maintenance feasible flight legs. Desaulnier et al. [3] proposed a similar approach. The fleet assignment problem determines which fleet (i.e. aircraft type) is going to operate each flight in the timetable. Of course, such a heterogeneous fleet assignment problem can be reduced to a pure routing problem if the initial fleet is set to one. Aircraft routing solutions can be found in a second type of combined approach. For example, the integrated maintenance aircraft routing and crew pairing problem of Ben Ahmed et al. [31]. Imbedded in Ben Ahmed's model is the routing model of Haouari et al. [32]. Haouari described an aircraft routing graph with flight legs as nodes and feasible connections as arcs. Cordeau et al. [33] also presents a simultaneous aircraft routing and crew scheduling problem, and discusses its reduction to an aircraft routing problem. The basis of the reduced formulation is a path-node matrix where paths represent routes and nodes flights. We, in contrast, consider not individual routes; but a collection of individual routes as a complete routing solution for

all aircraft in the fleet which we term a chain decomposition. Of course, our individual routes can be used as the paths in the formulation of Cordeau et al. The routing problems imbedded in these integrated models include a plane count constraint, where by the number of aircraft in the fleet is given a priori. This is in contrast to our deficit function approach, where we make no a priori assumption on the number of aircraft available; but instead find the minimum number of aircraft needed to service the flight schedule.

Finally, there is the pure aircraft routing model itself, where assignment of fleet types has been completed resulting in a set of homogeneous fleets (i.e. a set of airplanes having similar technical characteristics). Among the works we found following this approach are Haouari et al. [32] who solve a routing problem with the objective of deriving cost-effective maintenance feasible routes. Like our approach, Haouari also assumes a daily routing problem where a weekly schedule is derived by solving a sequence of daily problems. In a recent paper, Khaled et al. [34] formulate the problem as a binary linear programming with flight leg connection restrictions and flow conservation type constraints. Unlike our approach the authors do not require that the aircraft routes are cyclic. Eltoukhy et al. [35] use a multi-commodity network flow-based integer LP formulation to solve the maintenance routing problem. The model formulations of all these authors are based on the connection network. This is unlike the column-generation approach, whereby feasible routings are generated outside of the LP and inserted as columns into a set covering /partition integer LP. In a recent paper, Yan and Kung [36] investigate the impact of aircraft routing on delay propagation with an objective of minimizing the propagated delay over all flights. Here the problem is formulated as a set covering problem with flight leg cover and fleet count constraints.

Clark et al. [37] consider a cyclic routing problem which they call the aircraft rotation problem. Their aircraft rotation problem maximizes the *through value* of connecting flights. The *through value* is the potential revenue of passengers remaining on the same aircraft while making a connection at an airport. Their solution methodology is to find an Eulerian tour that maximizes value and satisfies maintenance constraints. The use of an Euler graph to find a rotation is close to our approach. The difference is that we do not use through values to select the best rotation, but instead we perform our maintenance check after the cyclical routes are determined. In particular, Lan et al. [5] state that most aircraft maintenance routing problems have an objective of maximizing *through revenue*, but in practice, this additional revenue is very difficult to determine accurately and the financial impact is relatively small. The aircraft maintenance routing problem can thus be cast as a feasibility problem. Lan's observation on feasibility has been echoed by other authors as well. Liang et al. [38] consider the airline routing problem to include a short connect penalty cost and a *through revenue* between connecting flights, but because these costs are relatively small compared other costs the aircraft routing problem can be considered as a pure feasibility problem. Weide et al. [39], like others, use binary decision variables to indicate if a route is in the solution or not. They also mention that *through-values* are low compared to operational costs of an aircraft, and since the operational cost of all aircraft are identical, the aircraft routing problem reduces to a feasibility problem. Cordeau et al. [33] (2001) state "Since *through values* are not considered in the aircraft routing problem, the only costs that remain in the problem, once fleet size is determined and fixed, are the operational costs that are associated with the flight legs. However, because all legs must be covered exactly once and all aircraft of a given type are assumed to have equal operating costs, this is in fact a fixed cost. The aircraft routing problem thus becomes a feasibility problem to ensure that each aircraft is maintained appropriately." Lacasse-Guay et al. [40] also claim that the aircraft routing problem can be treated as a feasibility problem since the operational costs of aircraft of the same type are identical. Thus, they say other objectives can be used, such as minimizing the number of aircraft, maximizing robustness by avoiding short connections, and maximizing *through values*. This distinct chorus

enforces the point of view that minimizing the number of aircraft is a reasonable objective for the aircraft routing problem.

We found Cardeau et al. [33] to be the closest work to ours. At the start they reduce an aircraft routing and crew scheduling problem to a pure routing problem in order to determine the required minimum fleet size and the initial positions of the aircraft. Starting with a plane count constraint for very large fixed fleet size they introduce a procedure to reduce it to obtain the minimum possible fleet. Using this now as a constraint, they find a set of feasible origin destination paths and construct a path-node coefficient matrix to be solved for the minimum number of paths in a set covering problem. This is, albeit, a much more cumbersome method to find the minimize fleet size than we provide. In a unique application of the min fleet size, Khaled et al. [34] employ a density term to characterize instances of a fleet routing problem. The density of an instance is the ratio of the minimum number of airplanes necessary to cover a flight schedule divided by the number of airplanes in the fleet (assumed greater or equal to the minimum number). Thus, a 100% dense instance implies that the current fleet size is equal to the minimum number required. For lower densities they find that the larger fleet size allows more flexibility and reduced computation times. Although this is true, it is at the high cost of the use of additional aircraft. Khaled's approach requires finding the solution to the min fleet size problem and proceeds to solve it using a complex integer LP, unaware of the simple solution we offer by our construction of DFs according to the Minimum Fleet Size Theorem and it's corollary.

All the aforementioned comparisons allow us to bring forth a number of observations. The first and foremost is that our approach finds and works with the minimum fleet size, whereas others find routings for a fixed fleet size owned by the airline. It was surprising that no literature was found that analyzed the optimization of one of the main capital resources of an airline i.e., that of the decision of the number of planes in the fleet. All scheduling analysis assumed the current number of planes as a given. The advantages of finding the minimum fleet size are that once it is found airlines can reduce or increase their existing fleet accordingly. Placing our work in the context of others, the salient difference is that we do not generate individual feasible single aircraft routings (as in the column generation approach), which are then combined to form a feasible solution by satisfying flight leg cover and plane count constraints. Instead, we find complete feasible collections of routes which we designate as CDs. So that each CD would in effect be a feasible solution to the set covering formulation of the problem. It is, however, possible to extract individual chains (aircraft routings) from our chain construction algorithm, and to use these in the column generating model. Moreover, we found no work that attempts to directly generate short cyclic routes. We, instead, introduced a procedure to find the maximal number of daily (single period) balanced chains. Using a CD with the maximal number of single period balanced (cyclic) chains, induces shorter multiperiod cyclic

routes. These exhibit several good properties of the final solution such as, improved conditions for repair/maintenance activities and crew layovers.

10. Conclusion

This paper considers a DF approach for determining the minimal FS required to service a fleet schedule, followed by a decomposition of the FS into vehicle routes (called a chain decomposition) for each aircraft in the minimal fleet. Our goal here is not to provide another large column generated - integer programming approach to solve these problems. Our goal is to extend DF theory, and use it to provide some new insights on how to solve the aircraft routing problem in a less complex way. After introducing DFs and the minimum fleet size theorem, we attack to problem of aircraft routing through CDs. We noted an important characteristic of a CD in terms of the number of balanced and unbalanced chains. Balanced chains are flight sequences with the same start and end terminals. To convert any unbalanced into balanced chains we concatenate them into multiperiod chains MPC. Because the number and length of MPCs subsequently increases the complexity of finding crew pairing solutions, and extends the time of aircraft away from home base we define a Maximal Balanced Chain Problem (MBCP) to search for CDs with a maximal number of balanced chains The MBCP can be formulated as a multi commodity network problem which is NP-hard, and therefore we provide a probabilistic heuristic method. For this purpose, we develop a method for generating random chains. After finding the maximal number of balanced chains the remaining unbalanced chains are converted into balance MPCs. We do this by mapping the unbalanced chains into an Euler graph and find all edge disjoint cycle covers of the Euler graph using a modified version of Hierholzer's algorithm. Finally, we construct an aircraft routing that insures a daily repeating schedule of flights. All our algorithms have polynomial complexity. To insure maintenance check constraints are satisfied for nT MPCs, it is necessary to add deadhead flights. For multiple routing solutions, minimizing the cost of deadhead trips and overnight stays provides the basis for selecting an optimal routing solution. On the practicality of our method and its results, airlines may look at this work to be only of theoretical interest because in reality they have already invested in an existing fleet of aircraft. To this we argue that the existing fleet is not economical and suggest to sell of excess aircraft or buy/lease additional aircraft to meet the required number needed to service the required flight legs. Also, the minimum number of required aircraft can be used as a bench mark upon which to judge the efficiency of their existing fleet size. In cases of a new airline (perhaps a low-cost contender) starting with a *tabula rasa* the ideas here for determining the minimum fleet size, and its chain decomposition should be appealing.

Appendix: Data. for FS30 (terminals A,B,C,D = 1,2,3,4)

FS30					
Flt No	DepTime	ArvTime	DepTerm	ArvTerm	
1	4.5	12.0	2	4	
2	14.5	20.5	4	2	
3	8.5	15.5	4	1	
4	16.5	21.0	2	3	
5	9.0	16.0	4	2	
6	11.5	13.0	2	3	
7	6.0	10.5	2	1	
8	11.0	16.5	3	4	
9	18.0	21.5	2	3	
10	7.0	11.0	3	2	
11	13.5	17.0	3	2	
12	18.0	23.0	2	3	
13	10.5	15.5	3	4	
14	7.0	13.5	2	4	

15	13.0	20.0	4	2
16	5.5	8.5	2	1
17	9.0	12.5	1	2
18	15.0	17.5	1	2
19	16.0	19.5	2	1
20	18.5	22.0	1	2
21	8.0	11.0	1	2
22	10.0	13.0	2	1
23	11.0	15.0	1	2
24	18.5	22.5	2	3
25	5.5	8.5	3	2
26	9.0	13.5	1	2
27	14.0	17.0	2	3
28	17.5	21.5	3	2
29	3.0	7.0	4	1
30	12.0	15.0	2	4

References

- [1] Barnhart C, Boland NL, Clarke LW, Johnson EL, Nemhauser GL, Shenoi RG. Flight string models for aircraft fleet and routing. *Trans Sci* 1998;32(3):208–20.
- [2] Daskin MS, Panayotopoulos ND. A Lagrangian relaxation approach to assigning aircraft to routes in hub and spoke networks. *Trans Sci* 1989;23(2):91–9.
- [3] Desaulniers GJ, Desrosiers J, Dumas Y, Solomon MM, Soumis F. Daily aircraft routing and scheduling. *Magmt Sci* 1997;43(6):841–54.
- [4] Kabbani NM, Patty BW. Aircraft routing at american airlines. Proceedings of the thirty-second annual symposium of AGIFORS. 1992.
- [5] Lan S, Clarke JP, Barnhart C. Planning for robust airline operations: optimizing aircraft routings and flight departure times to minimize passenger disruptions. *Trans Sci* 2006;40(1):15–28.
- [6] Mercer A, Cordeau JF, Soumis F. A computational study of Benders decomposition for the integrated aircraft routing and crew scheduling problem. *Comput Oper Res* 2005;32(6):1451–76.
- [7] Stojkovic G, Soumis F, Desrosiers J, Solomon MM. An optimization model for real-time flight scheduling problem. *Trans Res A* 2002;36:779–88.
- [8] Linis VK, Maksim MS. On problem of constructing routes (in Russian). *Proc Civil Aviat Inst Riga* 1967;102:36–45.
- [9] Gertsbakh I, Gurevich Y. Constructing an optimal fleet for a transportation schedule. *Trans Sci* 1977;11(1):20–36.
- [10] Mancel C, Antonio F, -Camino Mora. Airline fleet assignment: a state of the art. Proceedings of the tenth air transportation research society conference, ATRS. 2006.
- [11] Ozdemir Y, Basligil H, Sarsenov B. A large-scale integer linear programming to the daily fleet assignment problem: a case study in turkey. *Proc Soc Behav Sci* 2012;62:849–53.
- [12] Markus F, Blegur A, Bakhtiar T, Aman A. Scenarios for fleet assignment: a case study at lion air. *OSR J Math (IOSR-JM)* 2014;10(5):64–8. Ver. I.
- [13] Dilworth RP. A decomposition theorem for partially ordered sets. *Ann Math* 1950;51(1):161–6. Second Series.
- [14] Dantzig GB, Fulkerson DR. Minimizing the number of tankers to meet a fixed schedule. *Naval Res Log Quart* 1954;1:217–22.
- [15] Bartlett TE. An algorithm for the minimum number of transport units to maintain a fixed schedule. *Naval Res Log Quart* 1957;4(2):139–49.
- [16] Salzbom FJM. Minimum fleet size models for transportation systems. In: Buckley DJ, editor. Proceedings of the sixth international symposium on transportation and traffic theory (ISTTT6). 1974. p. 607–24.
- [17] Serafini P, Ukovich W. A mathematical model for periodic scheduling problems. *SIAM J Discret Math* 1989;2:550–81.
- [18] Orlin JB. Minimizing the number of vehicles to meet a fixed periodic schedule: an application of periodic posets. *Ops Res* 1982;30:760–76.
- [19] Gertsbakh I, Serafini P. Periodic transportation schedules with flexible departure times: an interactive approach based on the periodic event scheduling problem and the DF approach. *Eur J of Oper Res* 1991;50(3):298–309.
- [20] Gertsbakh I, Gurevich Y. Homogeneous optimal fleet. *Trans Res Part B Methodol* 1982;16(6):459–70.
- [21] Linis VK, Maksim MS. On the problem of constructing routes, Part II: methodology and numerical example, translated and edited by Ilya Gertsbakh and Tao Liu. *Transp Telecommun* 2017;18(3):234–9.
- [22] Gertsbakh I, Liu T, Ceder A. On the problem of constructing routes, Part I: preface. *Transp Telecommun J* 2017;18(3):231–3.
- [23] Liu T, Ceder A. Deficit function related to public transport: 50 year retrospective, new developments, and prospects. *Trans Res Part B* 2017;100:1–19.
- [24] Liu T, Ceder A. Integrated public transport timetable synchronization and vehicle scheduling with demand assignment: a Bi-objective Bi-level model using deficit function approach. *Trans Res Part B Methodol* 2017. <https://doi.org/10.1016/j.trb.2017.08.024>. online.
- [25] Ceder A, Stern HI. DF bus scheduling with deadheading trip insertions for fleet size reduction. *Trans Sci* 1981;15(4):338–63.
- [26] Stern HI, Ceder A. A DF approach for bus scheduling. In: Wren A, editor. Computer scheduling of public Transport. North-Holland Publishing Company; 1981. p. 85–96.
- [27] Ceder A, Stern HI. The variable trip procedure used in the AUTOBUS vehicle scheduler. In: Rousseau JM, editor. Computer scheduling of public transport 2. Amsterdam, Netherlands: North Holland Publishing; 1985. p. 371–90.
- [28] Gertsbakh I, Stern HI. Minimal resources for fixed and variable job schedules. *Ops Res* 1978;26:68–85.
- [29] Gertsbakh I, Stern HI. Using DFs for crew planning in aviation. *Transp Telecommun* 2017;18(4):289–96.
- [30] Biggs NL, Lloyd EK, Wilson RJ. Graph theory 1736-1936. Oxford University Press; 1999. p. 240.
- [31] Ben Ahmed M, Mansour FZ, Haouari M. Robust integrated maintenance aircraft routing and crew pairing. *J Air Transp Manag* 2018;73:15–31.
- [32] Haouari M, Shao S, Sherali HD. A lifted compact formulation for the daily aircraft maintenance routing problem. *Trans Sci* 2013;47(4):508–25.
- [33] Cordeau JF, Stojković G, Soumis F, Desrosiers J. Benders decomposition for simultaneous aircraft routing and crew scheduling. *Trans Sci* 2001;35:375–88.
- [34] Khaled O, Minoux M, Mousseau V, Michel S, Ceugniet X. A compact optimization model for the tail assignment problem. *Eur J of Oper Res* 2018;264(2):548–57.
- [35] Eltoukhy AEE, Chan FTS, Chung SH, Qu T. Optimization model and solution method for operational aircraft maintenance routing problem,. Proceedings of the world congress on engineering. 2017. II 2017.
- [36] Yan C, Kung J. Robust aircraft routing. *Trans Sci* 2018;52(1):118–33.
- [37] Clarke L, Johnson E, Nemhauser G, Zhu Z. The aircraft rotation problem. *Ann Oper Res* 1997;69:33–46.
- [38] Liang Z, Feng Y, Zhang X, Wu T, Chaovalitwongse WA. Robust weekly aircraft maintenance routing problem and the extension to the tail assignment problem. *Trans Res Part B* 2015;78:238–59.
- [39] Weide O, Ryan D, Ehrgott M. An iterative approach to robust and integrated aircraft routing and crew scheduling. *Comput Oper Res* 2010;37(5):844–83.
- [40] Lacasse-Guay E, Desaulniers G, Soumis F. Aircraft routing under different business processes. *J Air Transp Manag* 2010;16(5):258–63.