Tail Assignment with Multiple Maintenance Locations Using Network Model

ISBN: 978-81-924713-8-9

Ajyuk J. Raj
Vinay V. Panicker
R. Sridharan
National Institute of Technology Calicut
(ajyuk.jraj@gmail.com)
(vinay@nitc.ac.in)
(sreedhar@nitc.ac.in)

Only a very few practical discrete optimization problems could be solved to optimality in the available time. A pragmatic proposition to these problems is use of heuristic algorithms, which do not guarantee the optimality of the solution. But, it is possible to find nearly optimal solutions within a reasonable amount of computational time for many problems. Tail assignment, which comes in np-hard class of problems, is the allocation of individual aircraft to different flight legs with the consideration of maintenance schedule for each aircraft. The primary aim of tail assignment is making assignments in such way that all maintenance constraints are satisfied. The tail assignment is traditionally solved manually with the manager's intuitive skills and thus the solution will vary from person to person, providing sub-optimal solution. This paper indents to provide a graphical model for solving practical cases of tail assignment with multiple maintenance hub. The model is tested with instances extracted from real-world timetables of an airlines company in India. It is found that the model can be solved using standard type of computers in 3-4 seconds for single fleet airlines with 18 aircraft and 1050 flights.

1. Introduction

All airlines provide air transport services for passengers and freight transfers. Air transport facilitates greater global travel, economic links and trade. It is a key factor in connecting a country to the global economy. For airlines, providing larger capacities implies higher operating costs. On the other hand, aircraft seats are ‘‘perishable’’, that is, unsold seats at the departure of the flight are wasted. Consequently, the ideal strategy is to provide just the ‘‘right number’’ of seats to passengers at the ‘‘right price’’ (Nikolaos Papadakos, 2009). This extols the extensive use of data and information resources from various aspects and analytical methods to solve complexities related to airline operations planning. This includes schedule design, fleet assignment, aircraft maintenance routing (tail assignment), and crew scheduling (Ruther, et al., 2013). The unrestrained magnitude and complexity of the planning process for the airlines has resulted in the decomposition of the overall problem into a set of sub-problems that are solved sequentially (Díaz-Ramírez, et al., 2013). The existing literature on tail assignment optimizes flights and maintenance together, while taking operational costs and constraints into consideration. The paradigm proposed in this paper suggests an alternative way of airline tail assignment which does not use the cost function, but uses the onward flight breakage rule, to take the advantage of real time data from strategic as well as operational aspect of airline operations. The literature classifies heuristic algorithms between two broad spectrums: constructive algorithms and iterative improvement algorithms. A constructive algorithm builds a solution from scratch element by element, by assigning values to one or more decision variables at a time. An improvement algorithm starts with an initial feasible solution and iteratively tries to obtain a better solution. The heuristic techniques are applied based on the problem on hand and there are different principles that govern the design of the same.

1.1 Greedy Heuristics Principles

The most widespread principle used for developing constructive heuristics is the greedy principle. Greedy techniques are the most imperative constructive techniques among single pass heuristics that create a solution in a single sweep through the data. Each successive step is taken to minimize the immediate cost or maximize the immediate gain. Greedy techniques have following the characteristic features.

Incremental Feature

The problem is represented in such a way that a solution can be viewed either as a subset of the set of all elements, or as a sequencing of a set of elements in some order. The approach builds up the solution set, or the solution sequence, one element at a time starting from scratch, and terminates with the first complete solution.

No-Backtracking Feature

In a greedy algorithm, there is never a revision done on the decisions made at any stage in the algorithm. That is, if an element is included in the solution set or is included in the sequence in the current position, it is never taken back or replaced by some other element.

Myopic Feature

At some stage, while selection of an element for inclusion in the solution set or to fill the next position in the sequence, the significant addition to the overall objective of that inclusion at that stage is taken into consideration, not the consequences of that inclusion in later stages.

Greedy Selection Feature

Every element selected to be included in the solution set, or selected to fill the next position in the sequence, is the best among those available for selection at that stage by some criterion, in the sense that it contributes at that stage the least amount to the total cost, or the maximum amount to the total gain, when viewed through that criterion. Thus, only the best possible element is selected at every stage of greedy algorithm.

1.2 Interval Graph

A graph $G = (N,A)$ is defined by a set $N = \{1, . . . , n\}$ of vertices (also known as nodes), and a set A of connecting arcs called edges, where each edge in A joins exactly a pair of nodes in N and has no orientation. If an edge joins nodes i and j, it is denoted by $(i; j)$. A pair of nodes in N are said to be adjacent if there is an edge joining them in A. The degree of a node is the number of edges containing it. An interval graph is an undirected graph formed from a family of intervals
S_i, $i = 0, 1, 2 ... n$

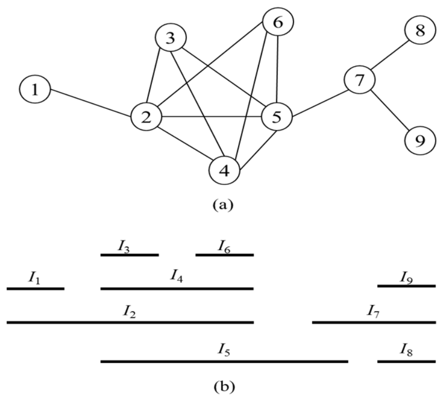

Figure 1.1 *Interval Graph (a) and Timeline (b)*

For each interval S_i, one vertex v_i is created and two vertices v_i and v_j are connected by an edge whenever the corresponding two sets have a nonempty intersection, that is,
$E(G) = \{\{v_i, v_j\} \mid S_i \cap S_j \neq \emptyset\}$.

A graph G is an interval graph if and only if the maximal cliques of G can be ordered $M_1, M_2, ..., M_k$ such that for any $v \in M_i \cap M_k$, where $i < k$, it is also the case that $v \in M_j$ for any M_j, $i \leq j \leq k$, which is known as interval graph isomorphism (Fishburn, Peter, 1985).

For instance, the following interval graph represents the corresponding interval in timeline as given in Figure 1.1 (a) & (b) respectively.

2. Tail Assignment

Tail assignment is the problem of deciding which individual aircraft, which is identified by its tail number, should cover which flight leg taking into consideration, the maintenance schedule for each aircraft. In light of the heavy competition, it is mandatory that airline companies keep their operating costs to a minimum by effective management of their flights, aircraft, and crew. Safety is the most imperative prerequisite for any airline operations and cannot be waived at any cost to meet other goals (Ageeva & Clarke, 2000). Therefore, it is essential to provide each type of aircraft in the fleet with a separate maintenance-routing plan. All routing plans must be coordinated to provide the best overall service. Currently, most airline companies plan and solve the problems in operations in a sequential manner. The schedule design process begins 12 months prior to implementation, and involves designing flight schedules, i.e., the origin, destination, and timing of each flight.

Based on this flight schedule, a flight network is constructed for solving the fleet assignment problem, whereby a fleet (aircraft type) is assigned to each flight segment, which is performed about 12 weeks in advance (Nikolaos Papadakos, 2009). Subsequently, about 2-4 weeks in advance, aircraft routing and crew scheduling decisions are made. Each problem is modelled and solved independently of the remaining ones, even though there is a clear interaction between them: the optimal solution of one problem becomes the input for the subsequent problem as provided in Figure 2.1. Some research papers on maintenance scheduling have considered the human interaction in obtaining an optimal solution (Brio, 1992). Ritcher suggests that if nothing else helps, a good way to find the best partial solution is by making the human planner try his hand (Ritcher, 1989). These approaches were innovative, but more emphasis is placed in human judgment and cannot be utilized in the current scenario due to scalability of the problem.

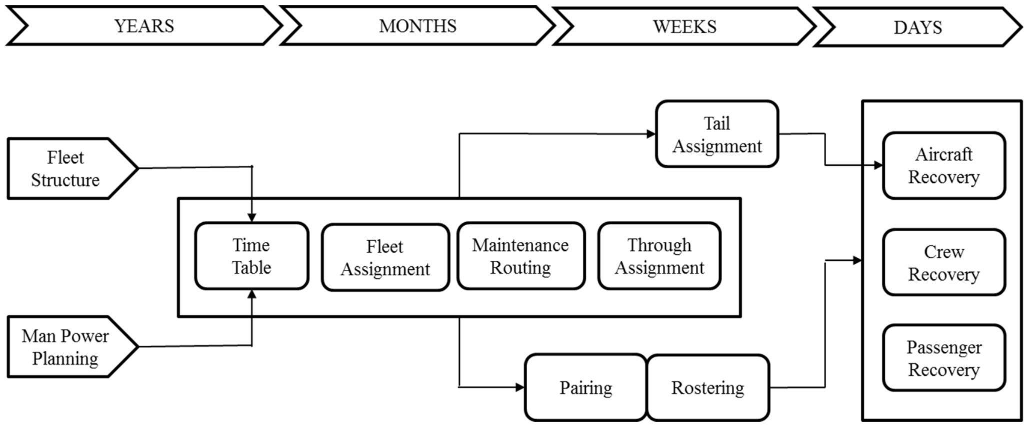

Figure 2.1 *Airline Planning and Scheduling*

1.3 Maintenance Constraints

The main focus of tail assignment is making assignments in such way that all maintenance constraints are considered. Furthermore, aircraft rotations should be cost-efficient and robust on the day of operations. Federal Aviation Administration (FAA) in US has established maintenance regulations which is described in (Talluri, 1998). The FAA mandates that the airlines perform four types of aircraft maintenance, commonly referred to as A-, B-, C- and D-checks. These checks vary in scope, duration, and frequency. The most common maintenance check is the A-check, which involves a visual inspection of major systems. The FAA mandates that airlines perform the A-checks approximately every 60 flight hours. This is equivalent to four–eight operating days depending on aircraft utilization. If an aircraft does not receive the A-check within this period, it is grounded until such maintenance is performed. B-checks involve a thorough visual inspection and lubricating of all moving parts. This type of maintenance is performed every 300 to 600 hours of flight. C- and D-checks involve taking the aircraft out of service, and are performed every one to four years (Gopalan & Talluri, (1998), Qi, et al., (2004)).

Maintenance Opportunity (MO) is a deadhead at a maintenance station that lasts at least the minimum maintenance time (Cordeau, et al., 2001). The maintenance activity actually being performed during a MO depends on the operational schedule (Talluri, 1998). The specific aircraft maintenance constraint is considered that establishes that any aircraft route must provide a maintenance opportunity at least every certain number of days. A simplification of this constraint has been considered by the assumption that all overnight deadheads satisfy this maintenance duration requirement, and there is only one maintenance station, which is known as the base (Díaz-Ramírez, et al., 2013). In many research work, these maintenance operations are taken during nights, where none or very few flight legs are scheduled, and the number of maintenance stations per fleet type is small. These assumptions are not strong, and they are commonly used (Talluri (1998), Gopalan & Talluri (1998)). These considerations allow rephrasing the aircraft maintenance constraint considered in the model as: Any aircraft route must spend at least one night at the base every nightsout +1 consecutive day. But such assumption is not considered in this work. Instead, maintenance is considered to occur any point of time, which is provided by the operations department of the airlines. This give rise to a whole new dimension for the tail assignment problem, where the maintenance is not structured. The schedule is, thus, dynamic based on the input provided by the airline.

3. Formulation

The Tail Assignment deals with mainly two entities namely,

- Aircrafts
- Flights

An Aircraft entity will have the following attributes related to it

- Aircraft Registration – Unique Identifier for Aircraft
- Aircraft Type – Specifies the aircraft family
- Seating capacity – With in the same type different aircrafts can have different capacity depending on how the airline configures it

A flight entity will have the following attributes related to it

- Flight Identifier – An Alpha numeric code used to identify a flight
- Flight Date – Date on which the flight is planned to be operated
- Departure Station – The airport at which the aircraft operating the flight is scheduled to depart. The above 3 items together represents an individual flight.
- Scheduled Time of Departure – The departure time of the flight
- Arrival Station – The airport at which the flight is scheduled to arrive
- Schedules Time of Arrival – The arrival time of the flight
- Aircraft Type – The type of the aircraft on which the flight should operate.

- Physical Seating capacity – The seating capacity which is required on the aircraft
- Minimum Ground Time – The minimum time the aircraft should be available at the departure station before the flight's departure.
- Onward Flight Information – The next flight which should be operated by aircraft as per original schedule from planning department.

The tail assignment is traditionally done manually with the manager's intuitive skills and thus the solution will vary from person to person, providing sub-optimal solution. There is a high need for automation of this process as it is tedious and takes a considerable amount of time and energy of the managers. The intendment of this paper is provision of a heuristic solution for maintenance routing of aircraft which would yield in a considerably low computation time. The algorithm developed is based on graph theoretical results which is motivated by the necessity of combining a given flight schedule included in the line of flight with the strict maintenance constraints of the aircraft provided by the operations.

1.4 Assignment Objectives

The final aircraft schedule generated should adhere to the following objectives

1. Maximize Assignment – An ideal solution is one where all flights are assigned. The schedule generated should assign all flights.
2. Minimize Soft Constraint Violation – The solution should try to minimize violations of soft constraints.

1.5 Constraints

The constraints related to tail assignment would vary from airline to airline. The below specified are a list of common constraints

1. The flights should be assigned to aircrafts matching the aircraft type and seating configuration. Example. If Flight 1 requires an aircraft type 320 with seating capacity 135, then it should be assigned to an aircraft of type 320 with matching configuration
2. The Flight assigned against an aircraft should be geographical continuous Example. If flight 1 and 2 are assigned against aircraft AC1 where 2 follows 1, then the flight 1 should be arriving at station A and Flight 2 should depart from station A.
3. The flights should be assigned respecting minimum ground time required. Example. If Flight 1 and 2 are assigned to aircraft AC1 where 2 follows 1, then the Scheduled Time of Departure of Flight 2 – Scheduled Arrival Time of the Flight 1 should be greater than or equal to Minimum ground time required for flight 2.
4. There should be no overlaps between activities scheduled against the aircraft. Example. If an aircraft AC1 has maintenance activities scheduled from 1000 to 1400, then no flight can be scheduled during that period.

The above listed constraints are mainly hard constraints which should not be broken. There are another set of constraints which, if required, can be broken, however, the overall assignment should try to minimize such violations. Such constraints are called soft constraints. An example of this type of constraints is Aircraft Preferences. For example, preference for operating sector A-B might be AC1, however, if required it can be operated using another aircraft.

4. Test Data

The data required for the formulation of the model has been obtained from one of the leading Indian company that specializes in navigational information, operations management and optimization solutions, crew and fleet management solutions and services. The Commercial Department of the airlines prepares the fleet assignment without taking into consideration, the operational constraints, mainly maintenance. The maintenance information is obtained from the Operations team. Both these information is used as the input for the system built.

The various parameters involved are categorised as Flight Parameters, Aircraft Parameters and Maintenance Parameters. The variables included in Flight Parameters are Flight ID, Flight Date, Departure Station, Arrival Station, Standard Time Departure (STD), Standard Time Arrival (STA), Minimum Ground Time, Line of Flight (LOF) Holder and Onward Flight. The various Aircraft Parameters are Aircraft Type, Total Seats, Aircraft Registration, Aircraft Available Station and Availability Time. The various Maintenance Parameters includes Activity ID, Maintenance Start Time, Maintenance End Time and Location. The generic classifications of constraints are Geographical Continuity, Ground Time, No Overlap, Aircraft Type Match and Onward Flight Rule. The Geographical Continuity constraint states that the sequence of flight assigned against an aircraft should always be continuous, i.e., if two flights F1 and F2 are assigned against an aircraft, F2 following F1 then the arrival airport of F1 should be same as departure airport of F2.

The ground time constraint states that there should be minimum ground time available between any two flights. No two activities assigned against an aircraft should overlap. Aircraft Type Match constraint states that the aircraft type and seating configuration against the flight should match the details against the aircraft. The Onward Flight Rule constraint ensures that as many as onward flight rule is respected during allocation of flights to various aircrafts. The Onward Flight Rule constraint is a soft constraint while the remaining are hard constraints. The number of flights that need to be allotted to 18 aircrafts with 19 maintenance slots over a planning horizon of 20 days are 1050. A sample data from the original data set is provided in Table 4.1.

Table 4.1 *Sample Flight Data*

FLIGHT ID	FLIGHT_DATE	DEPARTURE	STD	ARRIVAL	STA	MIN GND TIME	LOF HOLDER	ONWARD FLIGHT	AIRCRAFT TYPE	TOTAL SEATS
XX3319	26-NOV-13	BOM	26/11/2013 04:10	BLR	26/11/2013 09:45	00:45	XX31B001	XX3142/26-NOV-13	31B	141
XX3142	26-NOV-13	BLR	26/11/2013 15:15	CCU	26/11/2013 20:10	00:50	XX31B001	XX3143/27-NOV-13	31B	141
XX3143	27-NOV-13	CCU	27/11/2013 03:55	BLR	27/11/2013 09:15	00:45	XX31B001	XX3324/27-NOV-13	31B	141
XX3324	27-NOV-13	BLR	27/11/2013 15:50	DEL	27/11/2013 21:50	00:50	XX31B001	XX3325/27-NOV-13	31B	141
XX3325	27-NOV-13	DEL	27/11/2013 23:01	BLR	28/11/2013 05:15	00:45	XX31B001	XX3464/28-NOV-13	31B	141
XX3464	28-NOV-13	BLR	28/11/2013 07:55	MAA	28/11/2013 10:15	00:50	XX31B001	XX3477/28-NOV-13	31B	141
XX3477	28-NOV-13	MAA	28/11/2013 10:55	BLR	28/11/2013 13:20	0:40	XX31B001	XX3336/28-NOV-13	31B	141
XX3336	28-NOV-13	BLR	28/11/2013 15:00	TRV	28/11/2013 20:20	00:50	XX31B001	XX3337/28-NOV-13	31B	141
XX3337	28-NOV-13	TRV	28/11/2013 23:20	BLR	29/11/2013 04:40	00:45	XX31B001	XX3140/29-NOV-13	31B	141
XX3140	29-NOV-13	BLR	29/11/2013 09:30	CCU	29/11/2013 14:30	00:50	XX31B001	XX3141/29-NOV-13	31B	141
XX3141	29-NOV-13	CCU	29/11/2013 15:20	BLR	29/11/2013 20:35	00:45	XX31B001	XX3318/29-NOV-13	31B	141
XX3318	29-NOV-13	BLR	29/11/2013 22:00	BOM	30/11/2013 02:50	00:50	XX31B001	XX3319/30-NOV-13	31B	141

5. Solution Methodology

The literature review for each of these planning operations combined with the existing models of integrated airline operations, maintenance routing and airline crew scheduling processes is considered. The solution of the model identifies the route that assigns all the flights in the given horizon with geographical continuity in the network. The solution begins with identifying the large number of possible flights in the planning horizon and related parameters such as time and location of the flight. This information is assumed to be given, known, and fixed for a flight with a specific departure time. This method starts with a base solution from the data provided. The solution is found out in an incremental manner in the sense that the final schedule is obtained by the routes found one at the time.

The tail assignment problem is usually solved after the fleet assignment has been completed. The first phase of this work is modelling of the tail assignment module, in which candidate flight segments are linked to specific aircraft tail numbers within a given sub-fleet of the airline based on maintenance constraints. The data obtained in the form of excel sheets are initial flight assignment to specific Line of Flight (LOF) Holder, which is prepared by the Commercial Department. Also, the maintenance data is obtained from the Operations Department. This data is retrieved and stored in the Database. The solution first starts by assignment of the various LOF holder to aircraft based on the initial location and available time of the aircraft as shown in Figure 5.1

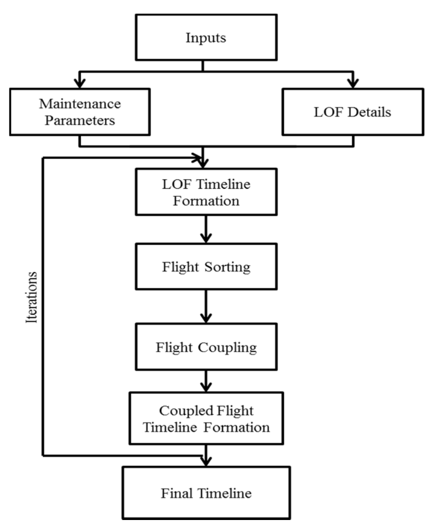

Figure 5.1 *Methodology*

The Airline Tail Assignment Algorithm (ATAA) is proposed in this paper, which is based on interval graph. It is a greedy heuristic that identifies the flight schedule for a single fleet and simultaneously allocates the aircraft routes considering both maintenance constraints and fleet size for a given planning horizon. ATAA, as an assignment problem with side constraints, is solved by the technique mentioned in Figure 5.1. The solution constitutes the aircraft assigned to different flights satisfying the maintenance constraints. An iterative process is used for the identification of routes for each aircraft. Once assigned, all the variables related with the legs covered in the routing solution are set to one and the maintenance conditions are checked one by one.

There are primarily three stages involved in this algorithm: Free Swap, Primary Swap and Secondary Swap. The free swap is the allocation of the flights that are overlapped with the maintenance schedule of the given aircraft to other aircraft, in case any of the aircraft is free, as mentioned in figure 5.2. If there are overlapping flights for the flights to be re-allocated, then the algorithm does a mutual swap between the flights that are feasible. This is based on the fact that if the flights are connected in the interval graph, as shown in figure 5.3, then there is overlap between the flights. In case there are no connections between the flight assigned to an aircraft and any other flights of aircraft *x*, then it means there are no flights that overlap during that flight duration under consideration and a free swap occurs. If the flights of all other aircraft are connected to the flight at maintenance, then there is a mutual swap between the flight during maintenance and the flight such that there is no overlap of the new flight with the maintenance slot and this process is called the primary swap. If there are no flights feasible for primary swap, then the algorithm steps into the secondary swap, wherein the flights of the swapped aircraft is checked with all the remaining aircraft, other than the maintenance aircraft for free and primary swap.

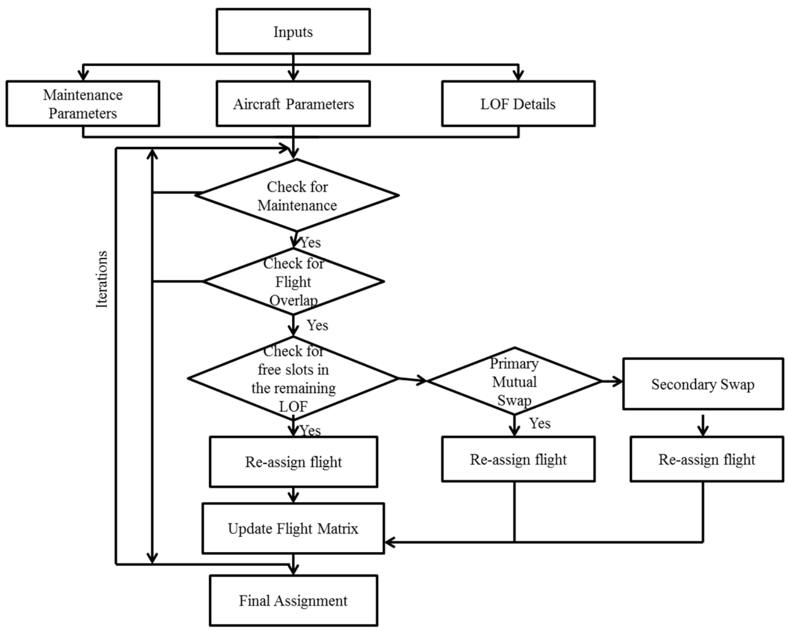

Figure 5.2 *Flight Swapping Algorithm*

This process is repeated for all the maintenance checks stops when all the flights are assigned the aircraft. This solution algorithm is a heuristic one, in the sense that it does not guarantee an optimal solution with minimum breakages to the Line of Flight while covering all flight legs in the network. Thus, the final flight schedule to be covered by the fleet will be in fact defined by the routing solutions satisfying maintenance constraints and fleet size capacity.

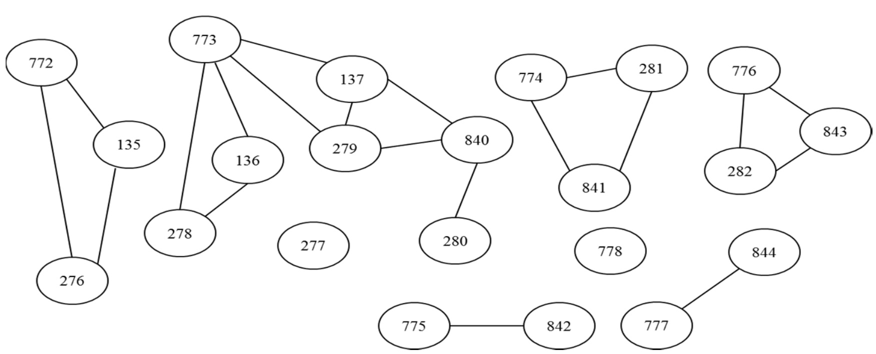

Figure 5.3 *Sample Interval Graph of Flights without Location Constraints*

The definition of flight legs, departure times and maintenance schedule is given by the routes provided. There is no need of a specific or additional formulation for the flight scheduling problem.

The pseudo-code for the program is as follows:

1. Begin
2. Read the Flight ID, Flight Date, Departure Station, Arrival Station, Standard Time Departure, Standard Time Arrival, Minimum Ground Time, Line of Flight Holder, and Onward Flight for a set of flights assigned to a specific fleet (identified by fleet routing).
3. Read the Aircraft Type, Total Seats, Aircraft Registration, Aircraft Available Station, and Availability Time from the aircraft data.
4. Read the Activity ID, Maintenance Start Time, Maintenance End Time and Location based on the operations planning data.
5. Create all possible valid routings for aircraft incorporating turn-around times – store as cell.
6. Attach two adjoining flights with the maintenance hub as common station for arrival and departure - Do this for all aircraft.
7. Examine each element of this cell According to the following criteria:
    a. There is geographic continuity.
    b. Each day, flights start at the city where the aircraft ended the day before.
    c. The minimum ground time is satisfied before each flight.
8. Check for maintenance conditions that includes the Aircraft, Time and Location.
9. Allocate the flights in the maintenance slots to free slots in other aircraft, if available.
10. Allocate the flight to the overlapped slot, provided the overlap is minimum and outside maintenance slot.
11. Primary Swap involves a mutual swap between the aircraft under maintenance and the aircraft considered for swap.
12. Secondary Swap involves the swap of the flights between the swapped aircraft and the remaining aircrafts, in case there are flights remaining to be swapped
13. Assign the overlapped flight to the aircraft with maintenance.
14. Continue the cycle till no overlap.
15. Go to step 7 till all the maintenance conditions are checked.
16. Display the re-assigned aircraft-flight pair.
17. End

The output is the re-assigned flights to aircrafts based on the maintenance constraints so that the onward flight rule is not broken beyond a limit. The final flight schedule is identified together with the aircraft routes while satisfying maintenance requirements as in figure 5.4. The methodology used is advantageous while dealing with some of the problems that arise in the flight scheduling process such as feasibility: the inclusion of any flight leg in a route establishes its feasibility.

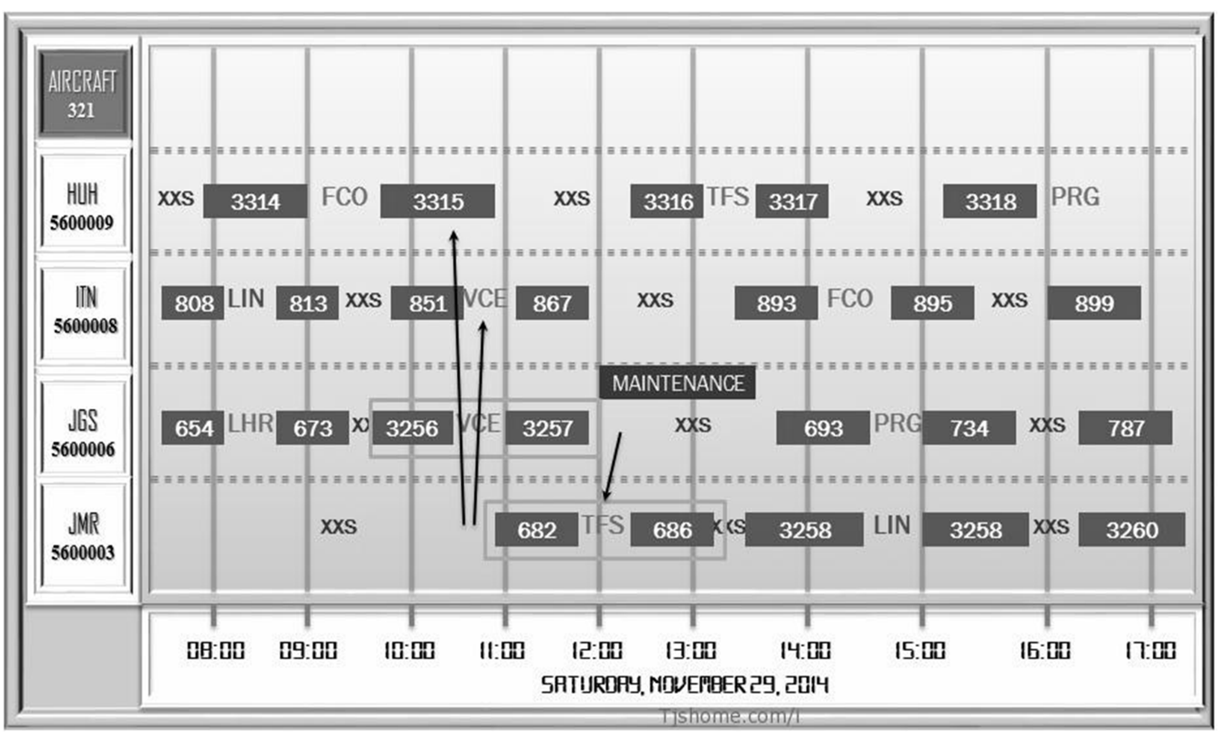

Figure 5.4 *Flight Pair Swaps*

6. Results

The Airline Tail Assignment Algorithm presented in the previous section has been implemented using MATLAB, with Java as the Data Convertor. The solution is run on a HP Pavilion M6 Notebook PC with Core i5-3210M 2.5GHz Intel processor and 6GB RAM running Windows 7 Professional. As the data is obtained directly from the company as Excel Sheet, the significant data has to be converted to MATLAB friendly format. Variability in run times is experienced even when running the exact same algorithm. Therefore, the solution was run five times and the average of these results is used when evaluating the performance of runtime. The model is tested on real world instances of a major airline company in India. The output is obtained as the flights that are initially assigned to each aircraft as well as the re-allocated flights based on the maintenance schedule of the aircraft. All the 1050 flights are assigned to the aircrafts with geographical continuity constraint satisfied.

Table 6.1 *Sample Output File*

FLIGHT ID	FLIGHT_DATE	DEPARTURE	STD	ARRIVAL	STA	INITIAL ONWARD FLIGHT	ACTUAL ONWARD FLIGHT	A/C REG
XX3141	13-DEC-13	CCU	13/12/2013 15:20	BLR	13/12/2013 20:35	XX3318/13-DEC-13	XX3318/13-DEC-13	KMD
XX3318	13-DEC-13	BLR	13/12/2013 22:00	BOM	14/12/2013 02:50	XX3319/14-DEC-13	XX3319/14-DEC-13	KMD
XX3319	14-DEC-13	BOM	14/12/2013 04:10	BLR	14/12/2013 09:45	XX3318/14-DEC-13	XX3318/14-DEC-13	KMD
XX3318	14-DEC-13	BLR	14/12/2013 22:00	BOM	15/12/2013 02:50			KMD
XX3339	26-NOV-13		26/11/2013 08:35	BLR	26/11/2013 13:55	**XX480/26-NOV-13**	**XX476/26-NOV-13**	KOY
XX476	26-NOV-13	BLR	26/11/2013 14:45	AGR	26/11/2013 16:00	XX487/26-NOV-13	XX487/26-NOV-13	KOY
XX487	26-NOV-13	AGR	26/11/2013 16:40	BLR	26/11/2013 17:55	XX474/26-NOV-13	XX474/26-NOV-13	KOY
XX474	26-NOV-13	BLR	26/11/2013 19:00	AGR	26/11/2013 20:10	XX483/26-NOV-13	XX483/26-NOV-13	KOY
XX483	26-NOV-13	AGR	26/11/2013 20:50	BLR	26/11/2013 22:05	**XX432/27-NOV-13**	**XX478/27-NOV-13**	KOY
XX478	27-NOV-13	BLR	27/11/2013 06:25	AGR	27/11/2013 07:35	XX479/27-NOV-13	XX479/27-NOV-13	KOY
XX479	27-NOV-13	AGR	27/11/2013 08:15	BLR	27/11/2013 09:30	XX426/27-NOV-13	XX426/27-NOV-13	KOY
XX426	27-NOV-13	BLR	27/11/2013 10:35	AMD	27/11/2013 11:40	XX447/27-NOV-13	XX447/27-NOV-13	KOY
XX447	27-NOV-13	AMD	27/11/2013 12:20	BLR	27/11/2013 13:35	**XX476/27-NOV-13**	**XX3474/27-NOV-13**	KOY
XX3474	27-NOV-13	BLR	27/11/2013 15:00	MAA	27/11/2013 17:20	XX3471/27-NOV-13	XX3471/27-NOV-13	KOY
XX3471	27-NOV-13	MAA	27/11/2013 18:00	BLR	27/11/2013 20:20	**XX472/28-NOV-13**	**XX3464/29-NOV-13**	KOY
XX3464	29-NOV-13	BLR	29/11/2013 07:55	MAA	29/11/2013 10:15	XX3477/29-NOV-13	XX3477/29-NOV-13	KOY
XX3477	29-NOV-13	MAA	29/11/2013 10:55	BLR	29/11/2013 13:20	**XX3338/29-NOV-13**	**XX3336/29-NOV-13**	KOY
XX3336	29-NOV-13	BLR	29/11/2013 15:00	TRV	29/11/2013 20:20	XX3337/29-NOV-13	XX3337/29-NOV-13	KOY

The schedules provided are domestic schedules which includes 1050 flights, 20 stations and 18 aircrafts, with multiple maintenance hubs and planning horizon of 20 days. The solution is obtained at an average runtime of 3.34 seconds. The sample of output file is shown in Table 6.1. All the hard constraints are satisfied while the total number of onward flight rule breakages are 13 with the letters bolded. The algorithm gives an output satisfying all the maintenance schedules and swaps the necessary flights between aircraft thus completing all line of flights with some breakages as shown in figure 6.1. In the sample provided, the flight number 772, 773, 774, 775, 776, 777 and 778 are initially assigned to aircraft number 9. After the maintenance constraints are considered, it changes to 772, 136, 137, 775, 776, 777 and 778. Similarly, the flight sequence of aircraft 12 changes from 276 – 277 – 278 – 279 – 280 - 281 - 282 to 276 – 277 – 840 – 841 – 282 and aircraft 18 changes from 840 – 841 – 842 – 843 – 844 to 278 – 279 – 280 – 281 – 842 – 843 – 844.

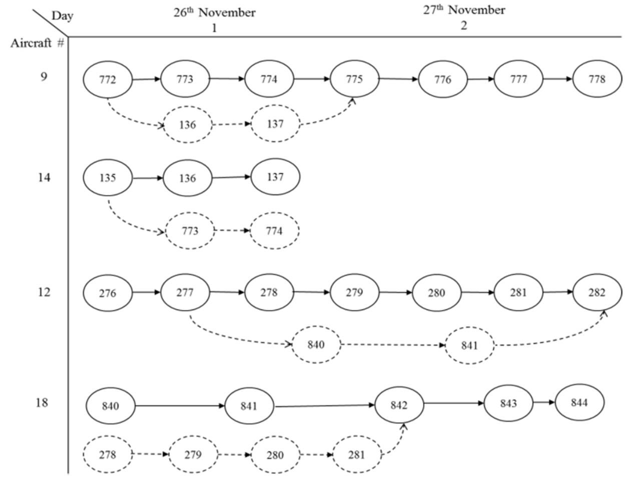

Figure 6.1 *Airline Tail Assignment Algorithm (ATAA) Output*

7. Conclusions

In this paper, a new paradigm for tail assignment, where routes and pairings are generated based on the onward flight rule rather than the cost function is introduced. Routes are generated specifically for each aircraft in accordance with the operational and maintenance constraints, thereby eliminating the need for manager to solve it manually. These routes consider the location, maintenance and flying history of the individual aircraft. Unlike standard aircraft routing, all maintenance requirements that the aircraft has during the planning horizon is considered. A formulation called Airline Tail Assignment Algorithm (ATAA), based on a greedy heuristic approach is presented that result in significant reduction in the computational time, so that swift operational decisions could be taken by the manager.

8. References

1. Cordeau, J. F., Stojkovic, G., Soumis, F., Desrosiers, J., (2001), 'Benders Decomposition for Simultaneous Aircraft Routing and Crew Scheduling', Transportation Science, Vol. 35(4), 375-388.
2. Dániel M., (2004), 'Graph Coloring Problems and Their Applications in Scheduling', Periodica Polytechnica Ser. El. Eng., Vol. 48, No. 1, Pp. 11–16.
3. Fishburn, Peter C., (1985), 'Interval orders and interval graphs: A study of partially ordered sets', Wiley-Interscience Series in Discrete Mathematics, New York: John Wiley & Sons.
4. Gopalan, R., Talluri, K. T., (1998), 'Mathematical Models in Airline Schedule Planning: A Survey', Annals of Operations Research, 76(0), 155-185.
5. Gopalan, R., Talluri, K. T., (1998), 'The Aircraft Maintenance Routing Problem', Operations Research, 46(2), 260-271.
6. Jenny Díaz-Ramírez, José Ignacio Huertas, Federico Trigos, (2013), 'Simultaneous Schedule Design & Routing With Maintenance Constraints for Single Fleet Airlines', International Journal of Engineering and Applied Sciences, ISSN 2305-8269.
7. Karine Sinclair, Jean-François Cordeau, Gilbert Laporte, (2014), 'Improvements to a large neighborhood search heuristic for an integrated aircraft and passenger recovery problem', European Journal of Operational Research, 233 (2014) 234–245.
8. Lloyd Clarke, Ellis Johnson, George Nemhauser, Zhongxi Zhu, (1997), 'The Aircraft Rotation Problem', Annals of Operations Research, 69(1997) 33 – 46.
9. Mattias Grònkvist, (2005), 'The Tail Assignment Problem', Department of Computer Science and Engineering, Chalmers University of Technology.
10. Nikolaos Papadakos, (2009), 'Integrated airline scheduling', Computers & Operations Research, 36 (2009) 176 – 195.
11. Qi, X., Yang, J., Yu, G., (2004), 'Scheduling Problems in the Airline Industry', In: J. Y. Leung, ed. Handbook of Scheduling. Algorithms, Models and Performance Analysis, Boca Raton (FL): Chapman & Hall/CRC.
12. Sebastian Ruther, Natashia Boland, Faramroze Engineer, (2013), 'Integrated aircraft routing, crew pairing, and tail assignment: branch-and-price with many pricing problems', School of Mathematical and Physical Science, University of Newcastle.
13. Yanina V. Ageeva, John-Paul Clarke, (2000), 'Approaches to Incorporating Robustness into Airline Scheduling', MIT International Centre for Air Transportation.