Scheduling Flight Perturbations with Ant Colony Optimization Approach

F. Mokhatab Rafiei [1*], S.M. Manzari [2], M. Khashei [3]

[1]Department of Industrial Engineering, Isfahan University of Technology, Isfahan, Iran, 84156-83111

[2]Faculty of Engineering, Tarbiat Modares Univer1sity, Tehran, Iran

[3]Department of Industrial Engineering, Isfahan University of Technology, Isfahan, Iran

[1]Farimah@cc.iut.ac.ir; [2]Eemanzari@yahoo.com; [3]Khashei@in.iut.ac.ir

DOI: 10.5963/IJCSAI02002001

***Abstract*— The disturbance incidences are common events in the flight networks in many countries. On the other hand, the survival of busy airlines in this strict and competitive business is crucial and tough. Therefore solving flight perturbation scheduling problem, in order to provide an optimized schedule in low computational time, is very important. This problem and its related literature have not yet been investigated seriously in Iran. One of the smart methods in solving routing and scheduling problems is ant colony optimization method which creates high quality results. In this paper, a model and a number of solution strategies for flight perturbation problem are presented. An approach based on ant colony optimization algorithm is used to solve the model. The computational results with real problem data show that the proposed method is highly efficient and effective in solving complex perturbation problems. The use of the proposed method can increase fleet life time and decrease air traffic and fuel consumption.**

Keywords- Flight Perturbation Scheduling; Smart Optimization; Ant Colony Meta-Heuristic Algorithm

I. INTRODUCTION

Airline Operations Control Center (AOC) in airlines is responsible for operational scheduling of flights. Safety assurance and flight efficiency in timetables are two objectives of these centers. There might be perturbations, such as airplane technical problems, absence of crew members, and bad weather conditions. In order to resolve such perturbations, airlines employ strategies like flight cancellation, delaying flights, and swapping among aircrafts. The most robust plans lose their credibility in such perturbations. In these circumstances, airlines make use of manual planning to minimize the negative effects of perturbations. However, as manual methods result in longer problem solving time, less accurate results, increasing workload on flight control personnel, and reducing reliability, companies tend to use new computer based solutions and technologies. On the other hand, the enumeration of all possible schedules is time consuming.

Flight Perturbation Problem (FPP) has been addressed in the literature and different solution methods have been proposed. Teodorovic and Stojkovic [1] proposed a lexicographical model in which flight cancellation, delaying flights, and airplane swapping are allowed, and developed an algorithm based on dynamic programming for solving FPP. The main objective function of the research was to minimize the number of cancelled flights. The second objective function, which is used in case of finding multiple solutions with equal number of cancelled flights, is to minimize total passenger delays.

Airplane shortage is studied by Jarrah *et al.* [2] in which an airplane is assumed to be out of service for a period of time. In this paper, two models were proposed for an isolated airport, based on Network Flow Model. In the first model, problem was solved by means of delay in flights and airplane swapping, while in the second model, flight cancellation and airplane swapping was employed. In both models, the goal is to minimize the costs associated with replanning the purturbated plan. These costs for the two models are delay and swapping, and cancellation and swapping, respectively. Cao and Kanafani [3] extended the aforementioned paper for multiple airports and proposed a Quadratic Binary Programming method for it. However, the types of perturbations in this research have not been clearly stated.

There are four different models to deal with FPP [4, 5]. The first model permits only flight cancellation, the second one allows both flight cancellation and the use of vacant airplanes, the third one authorizes the use of flight cancellation and flight delay, and finally the last model enables the model to use flight cancellation, flight delay, and vacant airplanes. In all models, the objective function is to minimize replanning costs, the solution methods are Lagrangian relaxation and subgradiant optimization, and the perturbation is the unavailability of one airplane [4, 5].

There is also a multi-fleet version of the model in Yan and Tu's [5] study in which the use of a substitute airplane for any high capacity airplane of other fleets is possible.

Arguello *et al.* [6] proposed a model in which the perturbation is defined as the unavailability of one airplane. A greedy randomized search procedure is proposed in which flight cancellation, flight delay, and flight swapping areused in order to minimize total cost of the airline. Thengvall *et al.* [7] has extended the work of Yan and Yang [4] and introduced a parameter as deviation penalty which measure the deviation from the initial schedule. Although this research considers the unavailability of multiple airplanes, it is claimed that the model is suitable in the presence of other types of perturbation.

The solution proposed by Love *et al.* [8] employs a local search algorithm which tries to minimize the objective function consists of weighted sum of the number of flight cancellations, flight delays, and plane swapping. In this method, manipulation of weights results in different solutions.

FPP is known to be NP-Hard. On the other hand, due to rapid growth in many airlines, the reviewed heuristic methods are time consuming and inefficient and lose their credibility in real cases. Therefore, in this research we employ meta-heuristic method. Due to the success of ant colony algorithm, which is a type of parallel random search method, in delivering good results in a wide range of optimization problems, e.g. scheduling and routing, and its high level of compatibility with real world problems, we choose this meta-heuristic algorithm to solve the FPP. We believe that the use of this method will reduce the time and cost associated with rescheduling and results in longer lifetime of the fleet.

In this research, we propose a mixed integer multi commodity model with flight cancellation, flight delay, inter- and intra-fleet airplane swapping, and the use of vacant airplanes in other airports allowed. Having defined and stated the problem, we explore possible solution methods. Our investigations lead to the selection of ant colony algorithm for which we modelled the problem. We used to proposed algorithm in real cases and also in available datasets in the literature. After comparison among the results, we note improvement tips on our proposed method.

II. Flight Perturbation Problem

Flight scheduling in airlines is based on maximization of profit, efficiency, and flight safety. Due to perturbations, however, these schedules lose their credibility and should be revised. In this context, Flight Perturbation Problem (FPP) is the effort puts in the rescheduling of the flight plans in order to minimize negative effects of perturbations. There have been identified different types of perturbations some of which are as follows:

- Airplane breakdown. Prior to any flight, airplane's systems are examined by flight engineers, and in case of any deviation from standards, the airplane is prohibited from flying and the flight is delayed until the problem is resolved.
- Absence of crew members. By international regulations, there should be a standard number of crew members. This number is based on flight sensitivity and the number of passengers. The absence of crew members, caused by illness, delay in previous flight, etc., results in delay in flight.
- Bad weather conditions. Icy runway, heavy storms against the flight, etc. can result in perturbations of flight schedules and in severe cases which might cause closure of the airport.
- Air Traffic Control (ATC) issues. ATC is responsible for the control of space and the safety assurance of flights. This center is authorized to interfere with flight schedules of all airlines. Postponing departure time, confining an airplane to remain in the sky for a period of time, delaying the arrival of the plane, etc., are of such interferences.
- Perturbation Propagation Effects. In most of airlines, flights occur on a chain basis. In other words, if an airplane flights from place "A" to place "B", it will continue its tour from place "B" to place "C", and so on. Obviously, least among of perturbation in each airport will be propagated to other airlines. This phenomenon intensifies the need for a rapid rescheduling [15].

A. Perturbation effects reduction strategies

Whatever the cause of perturbation is, the AOC's main goal is to minimize its negative effects. Prior to any decision making, responsible employees at AOC negotiate the situation with flight controllers, crew controllers, and maintenance schedulers. Main strategies for reducing the effects of perturbations are [9].

- Delay. Applying delays in departure times, affects not only direct passengers of that flight, but also passengers of consecutive flights with narrow time gaps.
- Airplane swapping. Airplane swapping is defined as the allocating an airplane to a route which in not in its main route list. This solution is effective in passengers view. If the airplane is chosen from another fleet, its capacity should be examined to cover all passengers. This strategy's drawback is in crew planning, as the crew members are usually trained for specific types of planes.
- Flight cancellation. While in passengers' perspective, flight cancellation should be the last resort, it is a suitable low cost option for airlines if they could place passengers in other flights. It should be noted that flight cancellation leads to a serious perturbation in the airplane's flight route, as the plane should by any means arrive at the next airport.
- Airplane positioning. Plane's flight between two airports with no passengers is called airplane positioning. It occurs in some circumstances that an airplane should be in a specific time at a specific airport and no scheduled flight can be carried out in this time. Due to the costs of flight (fuel, personnel, etc.) and unavailability of the plane in positioning time, this strategy is too expensive for airlines. Therefore, such flights usually occur at nights [10].

III. Ant Colony Algorithm

Ant Colony Optimization (ACO) is a meta-heuristic algorithm for finding near optimal solutions to NP-Hard optimization problems in a logical time. This method, which was introduced by Dorigo [11], is inspired by the actual behaviour of ants. Whenever ants find food, in real world, they transfer the food to the nest. In the beginning, each ant travels a different path and on their way back, they discharge a chemical substance known as pheromone. The amount and intensity of the release pheromone depend on the quality and amount of food. In the upcoming travels, this chemical substance acts as a communication medium, due to the fact that closer distances will have more pheromone which makes more ants to follow that path. The main characteristics of ants' system are [12, 13]:

- A positive feedback is used (by means of pheromone model, a parametric probability function on the solution space will be provided which results in fast recognition of a good solution)
- based on distributed computing (In order to alleviate the amount of pheromone, a vaporization method is employed which can prevent rapid convergence)
- Useing a greedy heuristic method
- population based (leads to parallel search in solution space and increasing the convergence rate of the algorithm).

ACO has been employed in many highly complex optimization problems. This algorithm is currently the newest method for solving sequential ordering problems (SOP), project resource-constraint scheduling, and other types of scheduling problems [14].

The algorithm's steps are as follows:

1) To Determine the initial parameters of the algorithm,

2) To Start the algorithm

   a) 'm' ants have travelled from the nest to the resource, therefore we have 'm' feasible solutions

   b) Update the amount of pheromones on all paths

   c) If the stopping condition is met, go to step 3, otherwise go to step 2.1

3) To Report the best result of all the iterations as the global optimal solution.

The algorithm is further described in the proposed algorithm section.

IV. Flight Network Design

In order to solve the FPP problem by means of ant colony, it is necessary to model the problem in the network format. Such a network should consist of nodes and arcs so that mobile agents (ants) can travel the nodes and complete their path. Therefore, all the available flights and airplanes in the period of perturbation are presented as a network depicted in Fig.1. This network includes three types of nodes: aircraft source nodes, flight nodes, and flight sink nodes. Each node belongs to a station (airport). Each aircraft source node represents a specific airplane and this airplane belongs to a specific station from which the plane is planned to start its tour or to which the plane will land at the time of perturbation.

Each flight node and flight sink node is representative of a specific flight. The position of a node in the network shows the planned departure time and destination. To every flight sink node, an aircraft source node is related. In other words, every flight route starts from an aircraft source node and ends in a flight sink node. Considering the type of aircraft source node, to ensure that every route ends in a proper destination, extra constraints should be enforced. Flight sink nodes assign a specific flight to every aircraft which is specified by the aircraft source node. In other words, these nodes represent the assigned flight to aircrafts in the original scheduled, at the end of perturbation time. As soon as the perturbation period being finished, these nodes turn the rescheduled plan back to the original schedule.

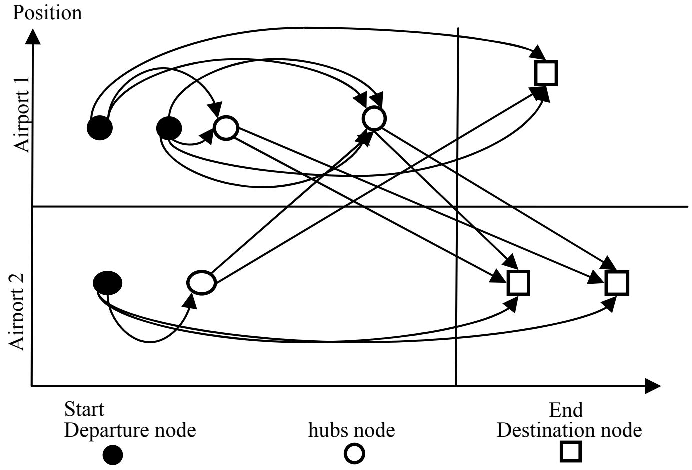

Fig. 1 Flight network modelling in node and act format

The interval between the start and the end of perturbation is called decision time in which any changes in the original schedule is allowed. For domestic flights, decision time is one day or less. However, in case long haul operations are included, this time is extended to more than a day. Arcs in the network define every possible connection between aircraft source nodes and flight sink nodes. Therefore, if there is an arc between aircraft source node and flight sink node, the available airplane in the source node can be assigned to the flight.

V. Mathematical Modelling of Flight Perturbation Problem

A mixed integer multi commodity model is employed in order to model the FPP. In the FPP literature, commodity implies different types of airplane. The sets, parameters, and variables of the model are as follows:

- Sets

  A: set of airplane source nodes

  F: set of flight nodes

  S: set of flight sink nodes

- Parameters

  c_{ij}^{k}: the income earned if the airplane k is assigned to flight j after flight *i*

  c_i: the cost per time period of delay in flight *i* (proportional to the number of passengers of the flight *i*)

  c^k: capacity of the airplane *k*

$$s_i^k = \begin{cases} 1 & i = k \text{ (the airplane is assined to its respective source node)} \\ 0 & i \neq k \text{ (the airplane isn't assined to its respective source node)} \end{cases} \quad (1)$$

$$t_i^k = \begin{cases} 1 & i = k \text{ (the airplane is assined to its respective sink node)} \\ 0 & i \neq k \text{ (the airplane isn't assined to its respective sink node)} \end{cases} \quad (2)$$

AD_i: the airport from which the flight *i* departs

AA_j: the airport to which the flight *j* arrives

TD_j: the departure time of the flight *j*

TA_j: the arrival time of the flight *i*/ the availability time of the airplane *i*

TG_{ij}^{k}: the time between two flights if the airplane *k* is assigned to flight *j* after flight *i*

D_i: the maximum allowed delay for the flight *i*

p_j: the number of passengers for the flight *j*

M: a big positive number

- Variables

$$x_{ij}^k = \begin{cases} 1 & \text{if the airplane } k \text{ flies the flight } j \text{ after the flight } i \\ 0 & \text{otherwise} \end{cases}$$

d_i: the amount of delay in flight *i*

The proposed model will be:
Maximize:

$$\max \sum_{i \in A \cup F} \sum_{j \in F \cup S} \sum_{K} c_{ij}^k x_{ij}^k - \sum_{i \in F} c_i d_i \tag{3}$$

Subject to

$$\sum_{j \in F \cup S} x_{ij}^k = s^k \qquad i \in A, k \in A \tag{4}$$

$$\sum_{j \in F \cup S} x_{ij}^k - \sum_{j \in A \cup F} x_{ji}^k = 0 \qquad i \in F, k \in A \tag{5}$$

$$\sum_{k \in A} \sum_{j \in F \cup S} x_{ij}^k \le 1 \qquad i \in F \tag{6}$$

$$\sum_{j \in A \cup F} x_{ji}^k = t_i^k \qquad i \in S, k \in A \tag{7}$$

$$x_{ij}^k \left(AD_i - AA_j\right) = 0 \qquad i \in A \cup F, j \in F \cup S, k \in A \tag{8}$$

$$TA_i + TG_{ij}^k + d_i - \left(TD_j + d_j\right) + M\left(x_{ij}^k - 1\right) \le 0 \qquad i \in A \cup F, j \in F \cup S, k \in A \tag{9}$$

$$d_i \le D_i \qquad i \in F \tag{10}$$

$$x_{ij}^k \in \{0،1\} \qquad \forall\, i, j, k \tag{11}$$

$$\begin{cases} d_i \ge 0 & i \in F \\ d_i = 0 & i \in A \cup S \end{cases} \tag{12}$$

$$x_{ij}^k P_j \le C^k \qquad i \in A \cup F, j \in F, k \in A \tag{13}$$

In constraint (4), an airplane is assigned to every source node. Constraint (5) establishes the flow equilibrium in every node. In other words, this constraint ensures that the number of incoming arcs is equal with the number of outgoing arcs. Constraint (6), which hardens solving the model, ensures the acceptability of the solutions by assigning at most one airplane to every flight. Constraint (7) is used to make sure that the sink node for every airplane is the same as the original schedule.

In order to model the flight j to be carried out after the flight *i*, constraints (8)-(10) are added to the model. The first one implies that the flight j should start from the airport in which the flight i has ended. In constraint (9), we make sure that the departure time of the flight j is greater than or equal with the arrival time of the flight *i*. Constraint (10) confines the possible delay to be less than maximum allowed delay (Di). *Di* can be set variably by the flight controller's opinion. Finally, constraint (13) allows x_{ij}^k being "1" only if the capacity of the airplane is enough.

VI. Parameter Setting Approaches

There are two methods for parameter setting: using real costs and using weights. When we use real costs, c_{ij}^k will include different costs such as: the cost of using a substitute airplane for the flight j (in case of substitution), the cost of carrying out flight j after flight *i* (transition cost), the cost of delays (c_i, which includes the refund paid by the airline to customers), etc. Exact calculation of these numbers in real world is not trivial and some of them can only be estimated.

When weights are used in parameter setting, model users, which are in fact flight controllers, can define parameters freely. For example, airplane substitution within a fleet might cost 20 units, airplane substitution between the fleets can cost 100 units, the weight of flight cancellation might be 10, and the weight of delay in a flight might be 1. These weights will be converted into c_{ij}^k and c_i.

One of the advantages of using weights in problem modelling is the freedom of decision makers to change the weights in case the results of the model do not satisfy their perceptions. The new model will be solved and new results will be calculated. In other words, by using weights, the model will act as an interactive decision support system [15]. Hence, we use this approach in our modelling.

VII. Implementation of the Proposed Model

Review of related literary of application of TSP, has shown that Ant Colony System (ACS) has achieved better results [16, 17]. Therefore this method is used in this research.

A. Building the Graph

The first step in solving the problem by means of ant colony algorithm is to present the problem as a graph.

Therefore, the network which is described in section 5 is employed. In this network, which is defined as N (A, F, S, L), elements of set A are the sets of airplanes of the airline which are grouped by their type, F is the set of all planned flight in decision time, S is the set of flight nodes at the end of perturbation and the number of its members is equal to the number of the members of A, and L is the set of arcs that connect nodes to each other.

B. Heuristic Information

One of the differences between artificial ants in ACO and real ants, which accelerate the convergence of the algorithm, is the calculation of heuristic information or the evaluation of the desirability of a node and its prioritization.

By means of heuristic information, smart agents (ants) should choose a node with both maximum gain, compared with the neighbour nodes, and minimum deviation from the original schedule as the next node. The first candidate for such heuristic information is the revenue from a flight between the current node and the destination node (Equation 14) in which W_c, W_d, and W_s are weights for cancellation and delay in flights' cost, FP_i is the number of passengers of flight *i*, d_{ij}^k is the amount of delay of flight j if it occurs after flight *i* by airplane *k*, SW_{ij}^k is the flight substitution cost if airplane *k* (which is not the prescheduled airplane) flies the flight j after flight *i*.

$$\eta_{ij}^{k} = W_c * FP_i - W_d * FP_i * d_{ij}^k - SW_{ij}^k \tag{14}$$

Another candidate, who is more efficient in selecting flight nodes, is to use relative flight revenue instead of absolute values. In order to calculate the relative revenue, the above revenue is divided by the number of passengers of the flight j. Therefore, the priciest node might not be included in the selection, as its revenue per passenger value is lower. Relative revenue is calculated as in equation 15.

$$\eta_{ij}^{k} = \frac{W_c * FP_i - W_d * FP_i * d_{ij}^k - SW_{ij}^k}{FP_i} \tag{15}$$

η_{ij} is a heuristic value which is defined based on the objective function, N_i^k is the set of possible neighbourhood of the *k*th ant when it is the city i. α and β are two parameters with which the extent effects of pheromone and heuristic data are defined.

In fact, with probability of q_0, the ant will do the best move based on the pheromone trail and heuristic information, while with probability of $1 - q_0$, it will do a bias exploration. Therefore, tuning q_0 determines whether to focus on the best found solution or to explore new ones.

C. Producing Solutions

Because of the different types of node we observe in FPP, the solution procedure is to start from aircraft source nodes, and stop at flight sink nodes. Therefore, instead of placing ants on all nodes, they will be distributed randomly on aircraft source nodes and consequently further nodes are selected in ACO algorithm until a complete path is formed. In ACO approach, every node is selected by means of state transition rule. In other words, when the *k*th ant is in node i, the next node, j, is selected by the equation (16).

$$j = \begin{cases} \arg\max_{l \epsilon N_i^k} \{\tau_{il}[\eta_{il}]^{\beta}\} & q \le q_0 \\ J & \text{otherwise} \end{cases} \tag{16}$$

In equation (16), q is a uniform random variable in [0, 1], q_0 is a parameter, and J is a random variable which calculated based on distribution function as equation (17).

$$p_{ij}^k = \frac{[\tau_{ij}]^{\alpha} [\eta_{ij}]^{\beta}}{\sum_{l \epsilon N_i^k} [\tau_{ij}]^{\alpha} [\eta_{ij}]^{\beta}} \quad \text{if} \quad j \epsilon N_i^k \tag{17}$$

η_{ij} is a heuristic value which is defined based on the objective function, N_i^k is the set of possible neighbourhood of the *k*th ant when it is the city *i*. α and β are two parameters with which the extent effects of pheromone and heuristic data are defined.

In fact, with probability of q_0, the ant will do the best move based on the pheromone trail and heuristic information, while with probability of $1 - q_0$, it will do a bias exploration. Therefore, tuning q_0 determines whether to focus on the best found solution or to explore new ones.

D. Pheromone Trail and Its Update

In global updating of pheromone trails in ACS, after each iteration only one ant is allowed to release pheromone. The amount of pheromone is calculated by equation (18).

$$\tau_{ij} \leftarrow (1 - \rho)\,\tau_{ij} + \rho \Delta\tau_{ij}^{bs} \quad ,\forall \left(i,j\right) \epsilon T^{bs} \tag{18}$$

τ_{ij} is the amount of pheromone on the arc (i, j) and its initial value is τ_0. $\Delta\tau_{ij}^{bs}$ in FPP is:

$$\Delta\tau_{ij}^{bs} = rrba \times \frac{Q}{irr} \tag{19}$$

$rrba$ is the Root Revenue for Best Ant, and irr is the Initial Root Revenue in the original schedule. Q is a parameter whose optimum value should be defined using experiments. Beside the global update rule for pheromone trails, there is also a local update rule which is activated as soon as an ant passes the (i, j) arc in its close path construction. The local rule is formulated in Equation (20).

$$\tau_{ij} \leftarrow (1 - \xi)\,\tau_{ij} + \xi\,\tau_0 \quad (0 < \xi < 1) \tag{20}$$

τ_0 is the initial value of the pheromone trail.

E. Parameter Setting

In the first step, the values of the parameters are set. Parameter setting is an important factor in the success of meta-heuristic algorithms. These values differ from one problem to another. Common procedure for finding suitable values for parameters is to use many experiments on every specific problem. Repeated experiments have been carried out on four datasets, s1a, s1b, s2a, and s2b, which are presented in the following parts. The results revealed that some parameters are not significantly sensitive (NSS) on the type of problem, and therefore can be kept constant. The parameters of the proposed algorithm and their values are presented in Table 1.

VIII. Local Search Algorithms

The literature on meta-heuristic algorithms reveals that a promising method for finding a high quality solution is the simultaneous use of a local search method and an initial solution generation mechanism. There have been developed different local search methods in the literature, from which K-OPT algorithms are of the most effective ones [14, 18]. In this paper, as a local search method 2.5-OPT is employed to be added to FPP-ACO, in order to improve the solutions in each stage. This method is a bit more time consuming than 2-OPT, while its results are much better than 2-OPT.

TABLE II THE PARAMETERS VALUE.

Parameter	Value	Comments
m	2A	The best value for the number of mobile agents are 2A where A is the number of flight nodes
ρ	NSS	Evaporation rate in global pheromone trail update
α	NSS	The coefficient of pheromone trail's effect in movement transition rule
β	NSS	The coefficient of heuristic information's effect in movement transition rule
τ_o	0.0001	Initial pheromone on paths
ξ	0.1	Evaporation rate in local pheromone trail update
MI		Maximum iteration
q_0	0.5	State parameter in transition rule
Q	1	The coefficient of the effect of pheromone change by the best ant (ant on the path with the most revenue)
NS	*NN*	The number of steps, which is equals with the number of nodes (NN)
NRS	1	Number of random steps is the number of ants' random movements (without following the state transition rule) in the beginning of the algorithm. Due to the fact that in the start of the algorithm in FPP problem, ants are placed on flight source nodes, random moves ensure that ants spread on all the flight nodes and then start to build up routes.
NSO	NSS	The number of save options which is the number of answers presented to the flight controller

A. Limitations of 2.5-OPT in FPP

Considering that there are 3 types of nodes in FPP with different characteristics, and time is a key factor in determining the position of nodes, the arcs in this problem is directed and therefore, there are limitations in reassigning arcs. In other words, in addition to the limitations of 2.5-OPT algorithm, other limitations, such as maximum allowed delay for the next flight, sufficiency of the capacity of the airplane, and the feasibility of departure and arrival of every node, should be taken into account.

IX. CONVERGENCE OF THE PROPOSED ALGORITHM

In order to probe the convergence of the proposed algorithm, we should investigate whether ants converge on one or some limited routes in later times or they continue to move randomly. This hypothesis is tested by means of entropy theory. The total probability entropy of all the routes is calculated using equation 21:

$$E_{ij} = -\sum_{\forall \text{Path}} P_{ij} \tag{21}$$

Where P_{ij} is the probability of choosing node j by the ant, if it was on node *i*. Based on the entropy theory and its application in mathematical algorithms [19, 20] the more an algorithm converges, its entropy will be closer to zero. In other words, if by increasing the number of iterations of the algorithm, the average entropy of the probability of path selection, i.e. $\overline{\mathrm{E}}$, tends to zero, we can infer that the ants' movements are convergence to one or a few paths. Fig. 2 shows the changes in $\overline{\mathrm{E}}$ by the increase in the number of iterations in the proposed problem.

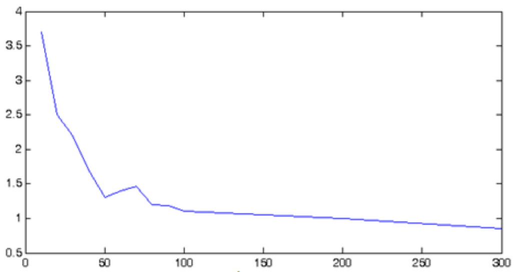

Fig.2 Convergence of the FPP-ACO algorithm.

X. COMPUTATIONAL RESULTS

The purpose of providing computational results in this section is to prove the efficiency of the model and its respective proposed algorithm. Moreover, we show how flight controller can use this method as a decision support tool for assessing different strategies in the time of perturbation.

A. Datasets

The sample datasets with which the efficiency of the proposed algorithm and the algorithms available in the literature are compared belongs to Sweden's domestic flights data. These datasets are provided under the headings of S1 and S2 and their properties are presented in Table 2. Note that domestic flights are categorized in short range flights with 15 to 125 minutes flight time.

TABLE III MAIN PROPERTIES OF SAMPLE DATASETS

	Datasets	
	S1	S2
Number of airplanes	13	30
Types of airplanes	2	5
Number of flights	98	215
Number of airports	19	32

These datasets have faced perturbation in two different types: 'a' and 'b'. In perturbation type 'a', an airplane is broken down while in perturbation type 'b' there is a delay on a group of flights. Combining perturbation types and two datasets, we will have four sets: s1a, s1b, s2a, and s2b. In s1a, one airplane is not available for 5 hours, and in s1b two flights will face a delay for 25 and 30 minutes respectively. The unavailability of an airplane in s2a is 6 hours and the delay in s2b occurs for 4 flights with 15 to 40 minutes time. The maximum allowed delay for every flight (D_i) is 60 minutes and every feasible path has at most 16 flight nodes. The time for every airplane, irrespective of its type, is 10 minutes. This number is changeable and is considered as a fixed number in our tests for comparison purposes only. Every dataset is solved for different weights and their respective costs are calculated.

B. FPP-ACO Implementation

The proposed algorithm has been implemented in MATLAB environment and a laptop with 1.66 GHZ CPU and 1 GB of RAM is used for the tests. For each and every problem, the algorithm is run several times until one of several stopping conditions is satisfied. The stopping conditions are:

1) All mobile agents complete their paths.

2) The number of steps for each ant is greater than the maximum number of moves.

The relationships between the value of the objective function and the run time with the number of iterations are depicted in Fig. 3. Obviously, by increasing the iterations, results will be closer to the optimal value while the run time increases.

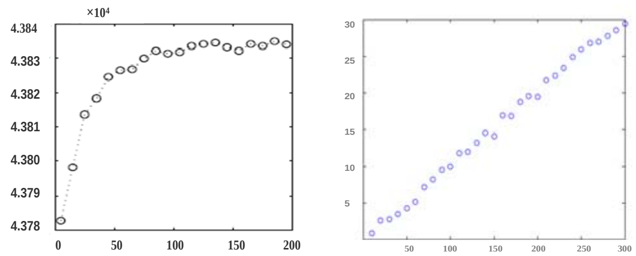

Fig.3 The effect of the number of iterations on the quality and the run time in s1a

The computational results are presented in Table 3. Column 1 shows different runs for different weight settings. The next four columns show user's weights for flight cancellation (can), between fleet swapping (sw), within fleet swapping (sw-i), and delay (del). BKS column shows the best results obtained by means of exact solution method, which is based on the combination of Dantzig–Wolfe decomposition and Lagrangian Heuristic [21] Generate and Solve (LHGS). This method is one of the straightforward strategies in solving FP-MIMF problems. By means of Dantzig–Wolfe decomposition, the main FP-MIMF problem is divided into two easy-to-solve problems and these two problems are solved by LHGS [22]. The BKS column is added to the table for comparison purposes. REV column provides the best generated revenue from paths in the proposed algorithm. The following columns are: 'c' for the number of passengers whose flight has been cancelled and 'D*Pass' for the amount of delay (passenger minute). Finally, the column Δ% is the difference between the generated income in our methods solution and the revenue of the exact solution.

A variety of weight combinations have been adopted for other data sets whose values may be different, regarding the flight controller opinion. As you see in column Δ%, the difference between optimum values calculated by this method and by the precise method increase in larger problems (S2b, S2a) which is unflavoured. The increase suggests that we use local search in the proposed method in order to improve the results' quality. Then we will apply local search algorithm of 2.5-OPT in the proposed method and rerun the data set using the improved version of the proposed method. The results have been presented in Table 4. As you see in the Table 4 the adjusted algorithm is able to improve the optimum results considerably compared to the situation of without local search procedure.

TABLE IV THE RESULTS OF THE PROPOSED MODEL IN SAMPLE DATASETS WITHOUT LOCAL SEARCH

Run Labels	Weights				Results						
	Can	Sw	Sw-i	del	BKS	REV	Δ%	C	Sw	Sw-i	D*pass
S1a1	20	100	10	1	43,792	42,920	1.99	90	0	0	0
S1a2	100	100	10	1	218,980	216,170	1.28	74	0	1	20
S1a3	20	100	10	1	43,792	42,920	1.99	90	0	0	0
S1b1	20	100	10	1	44,539	42,500	4.57	103	0	16	0
S1b2	20	100	100	1	43,786	41,880	4.35	112	0	6	0
S1b3	20	100	400	1	43,140	41,610	3.54	143	0	0	250
S2a1	20	100	10	1	68,480	63,580	7.15	307	0	3	0
S2a2	20	1,000	100	1	68,000	64,220	5.55	253	0	0	20
S2a3	100	1,000	10	1	341,865	318,570	6.81	184	8	19	1340
S2b1	20	100	10	1	69,080	64,435	6.72	230	2	4	25
S2b2	20	100	10	10	68,720	63,770	7.20	263	2	7	0
S2b3	20	100	50	1	68,900	64,460	6.44	201	3	2	420

TABLE V THE RESULTS OF THE PROPOSED MODEL IN SAMPLE DATASETS WITHOUT LOCAL SEARCH

Run Labels	Weights				Results						
	Can	Sw	Sw-i	del	BKS	REV	Δ%	C	Sw	Sw-i	D*pass
S1a1	20	100	10	1	43792	43650	0.32	42	0	0	230
S1a2	20	1000	10	1	43760	43600	0.36	56	0	0	0
S1a3	100	1000	10	1	218980	218970	0.004	44	0	2	210
S1b1	20	100	10	1	44539	43890	1.45	40	0	3	0
S1b2	20	100	100	1	43786	43450	0.76	62	0	3	0
S1b3	20	100	400	1	43140	43140	0.00	0	0	0	1580
S2a1	20	100	10	1	68480	67260	1.78	89	2	6	0
S2a2	20	1000	100	1	68000	66324	2.46	133	0	2	116
S2a3	100	1000	10	1	341865	333975	2.30	98	2	7	655
S2b1	20	100	10	1	69080	67655	2.06	45	1	43	216
S2b2	20	100	10	10	68720	67190	2.22	84	0	4	0

S2b3	20	100	50	1	68900	67565	1.93	30	2	16	135

C. *Software output*

As we mentioned earlier, the output of the software designed to solve FPP problem is a set of ranked answers, based on the income of the routes. The user may choose each route as the Flight Company's strategy to encounter the problems according to experimental considerations. This leads to higher flexibility in decisions of the flight controllers (Table 5).

TABLE VI SOME OF THE ANSWERS OBTAINED WITH ANT COLONY OPTIMIZATION APPROACH TO SOLVING S2A1

No	REV	CAN	SW	SW-i	D*Pass
1	67260	89	2	6	0
2	**67055**	**93**	**3**	**6**	**25**
3	66745	120	0	13	25
4	63670	258	1	3	340
5	63080	317	2	4	10
6	63030	316	3	1	0
7	62960	355	0	0	0
8	62710	311	3	7	0
9	61408	377	2	4	112
10	61382	354	6	14	98

D. *Calculation Time*

Considering the effectiveness level (Quality) of the answers achieved by the proposed method we would analyze its efficiency (Convergence speed). In this regard we would compare the proposed algorithm's calculation time with the precise method based on integration of Lagrangian innovative algorithm and Frank-Wolfe analysis method [23] (LHGS). Both strategies LHGS and FPP-ACO have been applied in two small (S1b, S1a) and large (S2b, S2a) data sets Using various weights in order to calculate the expenditures and the results have been presented in Table 6. The difference in changes in answers achieved by the proposed method for the optimum answers is presented in fig. 4.

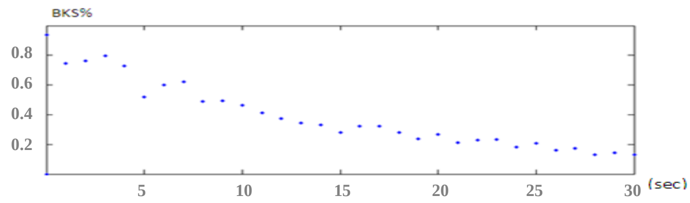

Fig.4 Diagram for the difference changes in answers achieved by the proposed method for the optimum answers

TABLE VII COMPARISON OF FPP-ACO AND LHGS APPROACHES

Run Labels	Weights				Results		Calculated time (sec)	
	Can	Sw	Sw-i	del	LHGS	FPP-ACO	LHGS	FPP-ACO
S1a1	20	100	10	1	43792	43650	4	3
S1a2	20	1000	10	1	43760	43600	4	2
S1a3	100	1000	10	1	218980	218970	6	2
S1b1	20	100	10	1	44600	43890	5	2
S1b2	20	100	100	1	44070	43620	2	2
S1b3	20	100	400	1	43140	43670	1	1
S2a1	20	100	10	1	68480	67260	645	17
S2a2	20	1000	10	1	68000	66324	526	15
S2a3	100	1000	10	1	3418565	333975	1139	23
S2b1	20	100	10	1	69080	67655	1118	14
S2b2	20	100	10	10	68720	67190	1150	15
S2b3	20	100	50	1	68900	67565	583	11

XI. CONCLUSION

In this research we could design and run an efficient and effective instrument for solving the flight tribulation problem, through developing ACO intelligent method and the local search approach of 2.5-OPT.

In other words, we provide a common field between ACO and FPP in an efficient way. This instrument will support the decisions of the flight controllers in the operation centers for the flight companies, When the flight network faces a problem the flight controllers achieve a set of ranked answers using this instrument.

The answers vary in structure. The flight controller can choose one of them as his or her strategy to solve the problem; otherwise he or she may reset the weights according to his or her experience and opinion and solve the problem. Applying this method makes the flight networks tribulations time schedule more economic in real-world and makes the flight navigation's life prolonged.

ACKNOWLEDGMENT

This project was supported by the transportation research center of the Route and Transportation Ministry. We also kindly appreciate the honest members of the center's deputy for developing scientific cooperation.

REFERENCES

[1] D. Teodorovic and G. Stojkovic, Model for operational daily airline scheduling, Transportation Planning and Technology 14, pp. 273–285, 1990.

[2] A.I. Jarrah, G. Yu, N. Krishnamurthy and A. Rakhit, A decision support framework for airline flight cancellations and delays, Transportation Science 27, pp. 266–280, 1993.

[3] J. Cao and A. Kanafani, Real-time decision support for integration of airline flight cancellations and delays part 1: mathematical formulation, Transportation Planning and Technology 20, pp. 183–199, 1997.

[4] S. Yan and D.H. Yang, A decision support framework for handling schedule perturbation, Transportations Research 30(6), pp. 405–419, 1996.

[5] S. Yan and Y. Tu, Multifleet routing and multistop flight scheduling for schedule perturbation, European Journal of Operations Research 103, pp. 155–169, 1997.

[6] M. Arguello, J. Bard and G. Yu, A GRASP for Aircraft Routing in Response to Groundings and Delays, Journal of Combinatorial Optimization 1(3), pp. 211–228, 1997.

[7] B. Thengvall, G. Yu and J. Bard, Multiple fleet aircraft schedule recovery following hub closures. Transportation Research, A35, pp. 289–308, 2001.

[8] M. Love, K. Sorensen, J. Larsen and J. Clausen, Using heuristics to solve the dedicated aircraft recovery problem, Informatics and Mathematical Modeling, pp. 1-19, 2001.

[9] M. Arguello, J. Bard and G. Yu, A GRASP for aircraft routing in response to groundings and delays, Journal of Combinatorial Optimization 1(3), pp. 211–228, 1997.

[10] J. Andrew and G. Schaefer, Improving airline operational performance through schedule perturbation, Annals of Operation Research, 144, pp.3-16, 2006.

[11] M. Dorigo, Optimization, Learning and Natural Algorithms, PhD thesis, Politecnico di Milano, Italie, 1992.

[12] M. Birattaeri, P. Balaprakash and M. Dorigo, ACO/F-Race: Ant colony optimization and racing techniques for combinatorial optimization under uncertainty, 6th Metaheuristics International Conference, 2005.

[13] D. F. Boris Bontonou, Ant colony optimization for the traveling purchaser problem, Computers & Operations Research 35, pp. 628 – 637, 2008.

[14] M. Dorigo and L.M. Gambardella, Ant colony system: a cooperative learning approach to the traveling salesman problem, IEEE Transaction on Evolutionary Computation 1, pp. 53-66, 1997.

[15] T. Anderson, Solving the flight perturbation problem with Meta heuristics, Heuristics, 12, pp. 37–53, 2006.

[16] L. Lessing, I. Dumitrescu and T. Stutzle, A comparison between ACO algorithms for the set covering problem, Lecture Notes in Computer Science. 3172, pp. 1-12, 2006.

[17] B. Crawford, R. Soto, E. Monfroy, F. Paredes and W. Palma, A hybrid Ant algorithm for the set covering problem, International Journal of the Physical Sciences, 6(19), pp. 4667-4673, 2011.

[18] M. Dorigo, and T. Stutzle, Ant Colony Optimization, Cambridge, U.S.A., MIT press, 2005.

[19] D. S. Johnson and L. A. McGeoch, The Traveling Salesman Problem: A Case Study, Local Search in Combinatorial Optimization, pp. 215–310. Wiley and Sons, New York, 1997.

[20] P.Y. Yin, J. Y. Wang, Ant colony optimization for the nonlinear resource allocation problem, Applied Mathematics and Computation 174, pp. 1438-1453, 2006.

[21] G. B. Dantzig, P. Wolfe, Decomposition Principle for Linear Programs, Operations Research 8, pp. 101–111, 1960.

[22] Y. Guo, A. Lim, B. Rodrigues and J. Tang, Using a Lagrangian heuristic for a combinatorial auction problem, Proceedings of the 17th IEEE International Conference on Tools with Artificial Intelligence, pp.-103-116, 2005.

[23] B. Thengvall, J. Bard and G. Yu, Balancing user preferences for aircraft schedule recovery during irregular operations, IIE Transactions 32(3), pp. 181–193, 2000.