AN INTEGRATED MODEL FOR FLIGHT SCHEDULING AND FLEET ASSIGNMENT BASED ON THE ANT COLONY META-HEURISTIC

Daniel Jorge Caetano
Universidade de São Paulo, Escola Politécnica, Departamento de Engenharia de Transportes Av. Professor Almeida Prado, Trav. 2, nº 83, Cidade Universitária, São Paulo, SP, Brasil
Phone: 55-11-3091-5731 Fax: 55-11-3091-5716
daniel@caetano.eng.br

Nicolau D. Fares Gualda
Universidade de São Paulo, Escola Politécnica, Departamento de Engenharia de Transportes Av. Professor Almeida Prado, Trav. 2, nº 83, Cidade Universitária, São Paulo, SP, Brasil
Phone: 55-11-3091-5731 Fax: 55-11-3091-5716
ngualda@usp.br

ABSTRACT

Schedule Generation and Fleet Assignment problems are usually solved separately. The integrated solution for both problems, albeit desirable, leads to large-scale models of the NP-Hard class. Some linear constraints on Fleet Assignment may become non-linear in the integrated problem, bringing further complexity to the solution process. This article presents a mathematical formulation of this integrated problem along with a new heuristic approach, called MAGS, based on the ACO metaheuristic. Both implementations include the same set of linear objective functions and constraints, addressing arrival and departure slots at airports and allowing aircraft load factor control. Both the exact solution and the one provided by MAGS are obtained and compared for the case of a Brazilian airline. The results show the applicability of MAGS to real world cases.

KEYWORDS: flight network, fleet assignment, ant colony.

1. INTRODUCTION

Along with the increasing passenger demand over the years, there was an increase in the frequency of flights, and a slight drop in the number of passengers per flight (Swan, 2002). This fact, in addition to the increasing competition among airlines, points toward the need of models to allow the definition of better air transport networks (Klabjan, 2004). Such problems are relatively complex, leading to NP-Hard models, composed of a large number of variables and constraints (Hane *et al*. 1995; Klabjan, 2004). Hence, there is a tendency to split the problem into stages (Rabetanety *et al*. 2006), simplifying the solution but possibly preventing the global optimal solution from being achieved.

Two of the several stages of the operational planning of an airline - the Schedule Generation Problem (SGP) and the Fleet Assignment Problem (FAP), can be solved using integrated linear programming models, usually described as a minimum cost flow over a space-time network (Caetano and Gualda, 2010, 2011; Lohatepanont and Barnhart, 2006). Due to the NP-Hard characteristic of these models, their application to large scale problems is limited, requiring the use of techniques to allow the solution for real sized airline networks, such as node clustering, column generation etc. (Hane et al., 1995).

This paper presents an exact model based on linear programming and a heuristic model that incorporates the Ant Colony Optimization metaheuristic to solve Schedule Generation and Fleet Assignment stages in an integrated way, seeking global optimum solutions and applications to real world problems (Caetano, 2011; Caetano and Gualda, 2010; Rabetanety *et al*. 2006).

Initially, a brief review of the concepts involved in solving SGP and FAP is presented, along with the description of the linear programming model. The mathematical model explanation is followed by the Multiple Ant Colony Group System heuristic, based on the traditional Ant Colony Optimization (Dorigo and Stützle; 2004), detailing the local search methods used. Finally, results obtained through the metaheuristic and the optimal results obtained with the linear programming model are presented, followed by a brief analysis and the conclusions of the study.

2. MATHEMATICAL MODELING

The operational planning of airlines can be divided into three interrelated problems: the definition of which flights to be offered, of which aircraft will be used for each flight, and of which crew will perform these flights. These steps are shown in Figure 1.

The increasing complexity of aircraft, crew and passenger management, and the intensification of the competition between airlines led to the need to create models that include a growing number of routes and restrictions. Such large-scale models pose significant computational challenges, since many of them are of the NP-hard type (Hane et al., 1995). This behaviour may be even worse when models are used to simultaneously solve two or more stages of the problem, seeking a global optimization (Klabjan, 2004; Sherali et al. 2006; Rabetanety et al., 2006).

While the Schedule Generation Problem (SGP) is usually solved observing flight frequency requirements defined by marketing-oriented decisions, the traditional approach to solve Fleet Assignment Models (FAM) is based on a space-time network, in which arrival or departure airports are represented by nodes. There are two basic types of arcs in this representation: flight leg arcs – connecting nodes that represent different airports – or waiting time arcs – connecting nodes that represent different times at the same airport (Berge and Hoperstead, 1993 apud Sherali et al., 2006; Hane et al. 1995).

SGP and FAP may be solved using integer linear programming techniques. However, practical instances, representing the operation of major airlines, remain a challenge, given the computational complexity involved.

Moreover, fleet assignment classical models assume that the flight schedule is previously defined, with every flight being covered. Traditionally, these models do not include operational restrictions at airports. To overcome these limitations, it is necessary to define a more comprehensive model. The model presented herein is based on the concept of space-time modeling, extended to cope with landing and departure slots by the addition of landing arcs that connect an arrival node to the first viable departure node, as shown in Figure 2.

The fleet assignment model can be integrated to schedule generation with the addition of a penalty for non-served demand and relaxing the cover constraint so that not all flights must be assigned.

The following sets, parameters and decision variables are defined to describe the model:

- <u>Sets</u>

  M: set of all markets, indexed by *m*; each market defines a demand and a time window which limits which flights can serve this demand.
  N_f: set of all nodes for aircraft *f*, indexed by *i*, *j*, *o*, *d* or *k*, representing an airport at a specific time.
  Nrd: set of nodes with departure restrictions.
  Nra: set of nodes with landing restrictions.
  F: set of all types of aircraft, indexed by *f*.
  L: set of arcs that represent the movement of aircraft, maintenance, waiting on the ground or wrap, indexed by (*i*, *j*), being *i* the source node and *j* the destination node of the movement.
  Lv: set of arcs that represent flight movements.
  Lvd: set of arcs representing flights assigned to a market.
  L_t: set of arcs the origin time of which is equal to or less than t and destination time is after t. Time t is set to a valid time according to the problem.
  L_m: set of arcs associated to market *m*.

- <u>Decision Variables</u>
  x^f_{ij}: number of type *f* aircraft flowing through arc (*i*, *j*).
  pa_{ij}: number of passengers flying from node *i* to node *j*.
  d_{ij}: number of potential passengers (demand) associated to the flight from node *i* to node *j*.

- <u>Parameters</u>
  D_m: unrestricted passenger demand on market *m*.
  C^f: number of seats of type *f* aircraft.
  R_{ij}: unitary revenue for a passenger in the flight from node *i* to node *j*. Since (*i*,*j*) represent a specific flight – including day and time – each flight may be associated with a specific unitary revenue.
  A_f: number of type *f* aircraft available.
  μ: Cost coefficient for unoccupied seats.
  ν: Cost coefficient for non-served passengers.

The objective function (expression 1) seeks to minimize the sum of lost revenues. The first term represents the difference between maximum revenue for the assigned aircraft and the revenue received from assigned passengers. The second term is associated to the lost revenue due to lost demand. The μ and ν parameters are used to control minimum aircraft load factor.

$$[Min]\sum_{(i,j)\in Lv}\left\{\mu.\left[\sum_{f\in F}\left(R_{ij}.C^f.x_{ij}^f\right)-R_{ij}.pa_{ij}\right]+\nu.\left[R_{ij}.\left(d_{ij}-pa_{ij}\right)\right]\right\} \quad (1)$$

$$\sum_{f\in F}x_{ij}^f\leq 1 \quad \forall(i,j)\in Lvd \quad (2)$$

$$\sum_{o|(o,k)\in L}x_{ok}^f-\sum_{d|(k,d)\in L}x_{kd}^f=0 \quad \forall k\in N_f,\forall f\in F \quad (3)$$

$$\sum_{(i,j)\in Lt}x_{ij}^f\leq A_f \quad \forall f\in F \quad (4)$$

$$\sum_{f\in F}\sum_{d|(i,d)\in Lv}x_{id}^f\leq 1 \quad \forall i\in Nrd \quad (5)$$

$$\sum_{f\in F}\sum_{o|(o,j)\in Lv}x_{oj}^f\leq 1 \quad \forall j\in Nra \quad (6)$$

$$\sum_{f\in F}C^f.x_{ij}^f-pa_{ij}\geq 0 \quad \forall(i,j)\in Lv \quad (7)$$

$$d_{ij}-pa_{ij}\geq 0 \quad \forall(i,j)\in Lv \quad (8)$$

$$\sum_{(i,j)\in Lm}d_{ij}-D_m=0 \quad \forall m\in M \quad (9)$$

Binaries:

$$x_{ij}^f\in\{0,1\}\ for\ \forall(i,j)\in Lvd \quad (10)$$

Integers:

$$x_{ij}^f\geq 0\ for\ \forall(i,j)\in L\setminus Lvd \quad (11)$$

$$d_{ij}\geq 0\ for\ \forall(i,j)\in Lv \quad (12)$$

$$pa_{ij}\geq 0\ for\ \forall(i,j)\in Lv \quad (13)$$

Expressions 2 to 4 represent the traditional cover, balance and number of aircraft restrictions (Berge and Hoperstead, 1993 apud Sherali et al. 2006; Hane et al. 1995).

Expressions 5 and 6 represent slot constraints, assuring that only one aircraft will depart or land on those nodes, respectively. Expressions 7 to 9 assure that each market demand will be associated to each flight and that the number of passengers of a flight will never be greater than the associated aircraft capacity.

The variables representing demanded flight arcs are binary, and are specified in expression 10. All other arc variables are integers greater than or equal to zero, as stated in expression 11, 12 and 13.

2.2. Minimum Aircraft Load Factor

Using an aircraft designed for *C* passengers on a specific flight, the revenue *R* for each seat may be given by the seat average cost *c* plus a profit *p*. As a consequence, the revenue for *pa* passengers can be obtained using expression 14.

$$R.pa = pa.(c + p) \tag{14}$$

Assuming that there is a specific number of passengers *pa*, smaller than or equal to the aircraft capacity *C* – guaranteed in the model by expression 7 –, which pays the total cost, this number of passengers can be represented by expression 15.

$$pa \geq \frac{C.c}{R} \tag{15}$$

Since this number of passengers varies with aircraft capacity, the minimum load factor is usually given as the ratio *pa*/*C*, dubbed minimum load factor δ. This ratio is used to determine whether a flight is profitable – and therefore should be executed – or not.

Traditional models limit the minimum load factor adopting a constraint forcing passenger per flight ratio above or equal to δ value, as shown in expression 16. The δ value is determined by the "break even" occupation – the exact number of passengers when the revenue is equal to the flight cost.

$$\frac{pa_{ij}}{\sum_{f \in F} C^f . x_{ij}^f} \geq \delta \quad \forall (i, j) \in Lv \tag{16}$$

This approach has drawbacks: it will hinder the creation of solutions in which a flight with load factor below the minimum is required to perform one or more flights that, globally, lead to higher load factors.

Another drawback associated to this approach is the model linearity. When FAP is individually solved, pa_{ij} is a constant value and, therefore, expression 16 represents a linear constraint. However, this is not the case of the proposed integrated model, since pa_{ij} is a decision variable.

One possible solution for both problems is a relaxation of this restriction, obtaining its effect through two objective function parameters: μ and ν.

The adopted objective function (expression 1) focuses on minimizing lost revenues. Assuming that *p* is the number of potential passengers and that $p \leq C$, lost revenue with empty seats (LES) may be calculated by expression 17.

$$LES = \mu.R.(C - p) \tag{17}$$

This expression represents the objective function when the flight is performed and all potential passengers are transported, that is, $p = dij = pa \leq C$ (only the first term of the objective function remains).

On the other hand, assuming that the flight is not performed, no potential passenger will be transported. As a result, the lost revenue by not executing the flight (RLN) may be calculated by expression 18.

$$RLN = \nu . R . p \tag{18}$$

This expression represents the objective function when the flight is not performed at all and no passengers are transported, that is, $p = dij$ and $pa = 0$ (only the second term of the objective function remains).

A flight is considered viable by the model if LES ≤ RLN. This relation means that the limit between a flight to occur or not is defined by expression 19, which may be rewritten as expression 20.

$$\mu . R . (C - p) = \nu . R . p \tag{19}$$

$$\frac{\mu}{\nu} = \frac{p}{C - p} \tag{20}$$

This expression can be rewritten dividing every member of the right side by C, resulting in expression 21.

$$\frac{\mu}{\nu} = \frac{\frac{p}{C}}{\frac{C}{C} - \frac{p}{C}} \tag{21}$$

On the other hand, the p/C ratio is exactly the load factor δ of the aircraft, which allows the expression 21 to be rewritten as expression 22.

$$\frac{\mu}{\nu} = \frac{\delta}{1 - \delta} \tag{22}$$

Although there is no direct relationship between the proposed objective function terms and the flight cost, control of minimum load factor δ is possible using the objective function coefficients μ and ν.

Since the objective function is calculated for all flights, μ and ν will, in fact, control the minimum average load factor, weighted by the unitary revenue associated to each flight seat. Thus, the model will allow suboptimal flights to be executed in order to achieve global optimal flight sequences, when low occupation flights are considered as repositioning flights. The use of unitary revenue as weights leads to solutions in which low revenue flights are more easily discarded than high revenue ones.

It is important to notice that there are differences between the terms presented in equation 19 and those in the objective function (expression 1). These differences exist only to accommodate situations in which the number of potential passengers is greater than the capacity of the aircraft selected to perform a specific flight. Expression 1 considers that each potential passengers not transported by insufficient aircraft capacity will cost the same as those not transported when the flight is not performed at all.

3. ANT COLONY MODEL

SGP and FAP are traditionally solved using integer linear programming techniques such as node clustering and constraint relaxation. However, practical instances, representing the operation of major airlines, remain a challenge, given the computational complexity involved. On the other hand, there are many heuristics capable of finding very good solutions to several types of combinatorial problems (Rayward-Smith et al., 1996 apud Abrahão, 2005), suggesting the search for heuristics that can provide appropriate solutions for the problem in lower processing times. The successful application to problems such as the Vehicle Routing Problem (VRP) and Aircraft Rotation Problem (ARP) draws attention to the metaheuristic known as Ant Colony Optimization (ACO), one of the many swarm intelligence metaheuristics (Teodorovic, 2008).

3.1. Ant Colony Optimization

The Ant Colony Optimization meta-heuristic (ACO) is described by the logic shown in Figure 3.

In short, given a graph (N, A) with N nodes and A arcs, the solutions are built in an iterative process, governed by probabilistic decisions. Starting from a node **n** $\in$ N, an arc **a** $\in$ A which departs from **n** is selected accordingly to its probability. This probability is calculated through two fundamental values associated to each arc (i,j): one of them is the heuristic value η_{ij} and shall be related to the problem being solved. The other value is the amount of pheromone τ_{ij} associated to the arc. Assuming that N_i is the set of all arcs leaving node **i**, the probability of choosing an arc can be obtained by expression 23, where α and β are parameters that need to be adjusted to the problem (Dorigo and Stützle, 2004; Bonasser and Gualda, 2006).

$$p_{ij} = \frac{{\tau_{ij}}^{\alpha}.{\eta_{ij}}^{\beta}}{\sum_{l|(i,l)\in N_i} {\tau_{il}}^{\alpha}.{\eta_{il}}^{\beta}} \qquad (23)$$

Applications described by Dorigo and Stützle (2004) reveal that updating the pheromones only when the full path has already been composed by the ant yields better results than the pheromone update upon the selection of each arc, during the construction of a solution. Additionally, to assure that the pheromone level will increase faster in better paths, the pheromone increment to the arcs of a solution should keep an inverse proportion with respect to the total solution length or cost, when coping with minimization problems.

Dorigo and Stützle (2004) defined auxiliary actions, such as the evaporation of pheromones over time, which reduce the attractiveness of bad, seldom selected paths; at the same time, it slows down the convergence towards a solution, broadening the solution space search for global optimum.

Since the pheromone value multiply the numerator of the probability equation, as shown in expression 23, it cannot be initialized as zero. For the TSP, Dorigo and Stützle (2004) proposed an initial deposit of pheromone associated with each of the arcs of the graph, the value of which should be inversely proportional to the length of the solution obtained by the nearest neighborhood method.

This kind of initial pheromone distribution provides similar probability for all arcs when the process starts, causing a broader search in the solution space and providing an initial diversification. Due to processes of pheromones updates and evaporation, the neighborhood search narrows as the heuristic evolves and thus promotes a gradual intensification. This approach, combined with an appropriate choice of α and β values, decreases the chance of a fast convergence towards a local optimum.

The initial solutions for some combinatorial problems may have very poor quality if the problem characteristics are not well represented by the adopted heuristic value in expression 23. The heuristic value has a crucial role for the proper exploration of the specific problem solution space (Dorigo and Stützle, 2003). On the other hand, the definition of a proper heuristic value may be complex for some problems. The heuristic value tends to be myopic due to solution building rules, since the real quality of a given arc to the solution may also depend on the arcs to be selected later in the process. Thus, Dorigo and Stützle (2003) suggest the adoption of local search methods as one of the auxiliary actions in the ACO. It may be simpler to set up local searches that take into account the problem global characteristics. The principle is that the ACO will provide good initial solutions and the local search leads those initial solutions toward the optimum in its neighborhood.

3.2. ACO Applied to the Integrated SGP and FAP

Although it was possible to adapt the basic ACO metaheuristic to solve the integrated flight schedule and fleet assignment problem, the results obtained through such an approach were not satisfactory. Since the basic ACO leads to a single shortest path, it must be executed several times, assigning one aircraft at a time and removing the selected arcs from the list, leading to suboptimal solutions, with objective function values almost three times the optimal ones.

One way to improve the results is the adoption of a representation with multiple colonies in which each colony represents one fleet. In this case, ants from one colony are repelled by other colonies pheromones, as proposed by Vrancx and Nowé (2006), which led to better results, but they remained far behind those obtained by linear programming, with objective function values of almost twice the optimal ones. A possible reason for these poor results is the dynamic characteristic of the demand assigned to each arc, since several arcs may share the demand of a market. Once more than one flight can compete for exactly the same passengers, the decisions made by one ant should be more tightly related to other ants' paths. The inability to cope with this specific characteristic leads to unrealistic solutions and cause convergence problems.

To improve the heuristic behaviour, the decisions made by some ants should produce more changes to the environment than simply adding attractive or repellent pheromones. The demand information should be updated upon the heuristic execution, improving each ant's decision quality. However, the results obtained with the changes enumerated above are not the most satisfying, because probabilistic decision proposed by the ACO metaheuristic assumes that all arcs are directly comparable, i.e.: the heuristic information of different arcs are compatible and comparable, which allows an adequate exploration of the problem solution space.

In the case of the mathematical model presented, there are different types of arcs. Each type of arc has a heuristic value which is not directly comparable to others: flight arcs can be compared by their direct profit, but how could one of them be compared to an arc that represents the aircraft that will wait on the ground? The direct profit of a ground arc is equal to zero or negative, if one considers the opportunity cost or the cost of staying at the airport. On the other hand, solutions using waiting arcs should be explored since sometimes it is better for an aircraft to wait for a flight with lots of passengers than not to wait and fly almost empty. There are particular situations in which waiting on the ground, instead of flying, leads to better overall solutions.

Since the problem has specific characteristics that can be used to improve the overall solution, an alternative heuristic is proposed, called Multiple Ant Group System (MAGS), incorporating elements of Multiple Ant Colony Optimization (MACO)(Vrancx and Nowé, 2006), Multiple Ant Colony System (MACS) and Elitist Ant System (EAS)(Dorigo and Stützle, 2004), as well as new elements not present in other ACO metaheuristic variants.

3.3. Multiple Ant Group System – MAGS Heuristic

MAGS is a multiple ant colony heuristic, such as MACS and MACO. As in MACO, a solution is represented by multiple ants; on the other hand, the number of ants that build a specific solution is previously known: there must be one ant per aircraft. The ants that compose a solution are called an *ant group*. A group may be composed of ants from different colonies and, similarly to MACS, each colony has a different objective function. This means that ants from each colony make decisions based on different criteria. In MACS, however, pheromones are identical for all colonies, which means it is essentially different from MAGS. Each ant group in MAGS has the same role of a single ant in the basic ACO: the group of ants represents the complete objective function, and each ant is associated with a different term of this function.

The proposed solution construction process is substantially different from the classical ACO in order to reduce the number of invalid and unrealistic solutions. During the construction of a solution, the ants of a group will alternately choose graph arcs. The group ant which will take the next step is randomly selected and whenever a flight arc is associated to an ant, this arc will be no longer available to other ants in the same group, ensuring the construction of solutions in which flight arcs are not shared by two or more aircraft.

Additionally, when a flight is selected by an ant, part of the flight market demand is also allocated, reducing the demand available for other flights that share the same market. This strategy avoids the association of ants to flight arcs for which the demand is no longer available in that solution. As the demand for each arc becomes dynamic during the construction of the solution, the problem presents similar characteristics to the dynamic routing in communication networks, as solved by the AntNet heuristic (Dorigo and Stützle, 2004). The exclusion of arcs and the demand allocation during the solution construction have relevant effects on the results, which are complementary to those provided by the repellent pheromones proposed in MACO, which continues to affect the selection probability of each arc available.

Considering the described construction process, each ant group has the same role as a single ant in the basic ACO: the group of ants represents the complete objective function, each ant associated with a different term of it. The MAGS basic logic is presented in Figure 4.

As proposed by Dorigo and Stützle (2003), the nearest neighborhood solution can be adopted as an initial solution. In the problem addressed, the "nearest neighbor" was defined as the arc associated with the minimum revenue loss, avoiding waiting arcs whenever possible. The objective function value for this solution is used to determine the initial pheromone deposit in each arc.

As defined by ACO, the arc selection follows a probabilistic selection, as depicted in Figure 5. On the other hand, differently from the basic ACO, the initial pheromone deposit is not the same for all arcs. Arcs associated with smaller heuristic values must receive substantially more pheromones in the initial distribution than those associated to higher heuristic values. Considering the basic ACO probability equation (expression 23), an increased pheromone deposit value in arcs that have low heuristic value will also increase their likelihood of being chosen, at least in the initial heuristic stage.

The probability equation adopted (expression 24), however, presents some additional parameters. The first one is ϕ_{ij}, which represents the amount of pheromone of other colonies, as in MACO, with their respective coefficient γ. Additionally, parameter ρ_{ij} reduces the probability of selecting a sequence of several unprofitable flights. The ρ_{ij} value is always 1.0 for profitable flight, maintenance, and waiting arcs. For unprofitable flight arcs, its value starts as 1.0 but upon the addition of an unprofitable arc to the solution, the value of ρ_{ij} is reduced by 50% for the next unprofitable flight arc. This value is only reset to 1.0 when a profitable flight is selected to compose the solution.

$$p_{ij} = \frac{\tau_{ij}^{\alpha}.\eta_{ij}^{\beta}.\left(\frac{1}{\phi_{ij}}\right)^{\gamma}.\rho_{ij}}{\sum_{l|(i,l)\in N_i} \tau_{il}^{\alpha}.\eta_{il}^{\beta}.\left(\frac{1}{\phi_{il}}\right)^{\gamma}.\rho_{il}} \qquad (24)$$

The η_{ij} value is proportional to the flight profit, given that there is demand available in the market. The η_{ij} is made equal to τ_{ij} for repositioning flight arcs – which have no markets associated to them –, waiting arcs and maintenance arcs, since the heuristic value based on lost revenue in these arcs would always be non-positive. The objective of this measure is also to reduce the myopic heuristic behavior, adding more emphasis on the historical quality of the solutions containing a specific arc, which is represented by the pheromone deposit value.

After the initial pheromone is distributed, **g** ant groups are generated, but no changes are made in pheromones, as in the basic ACO. As the generation of the ant groups is completed, pheromone evaporation takes place, at a fixed rate, and then all **g** ant groups will update their pheromone trails. As in the EAS, the best solution will reinforce its own pheromone trail, leading to convergence toward that solution.

The pheromone deposit for each ant group is proportional to the objective function value, as in the basic ACO, but each ant of that group shall deposit only part of the group total pheromone: the amount of pheromone each ant of a group deposits is proportional to the ant contribution to the quality of the solution represented by that group. The proposed distribution rule is defined by expression 25, where τ_f is the deposit of each ant, with $\tau_g = 1/C^g$, where C^g is the cost of the solution represented by the group, calculated through the objective function. R_f is the revenue generated by that ant and MR_g is the maximum revenue that could be generated by that group of ants.

$$\tau_f = \tau_g.\{0.5 + [R_f / 2.MR_g]\} \tag{25}$$

This formulation guarantees that each ant will deposit a value not smaller than 50% of the deposit calculated for the group and also ensures that it will increase when the ant has a large contribution to the group total revenue.

Even adopting the techniques proposed previously, very bad solutions are commonly built, limiting the basic heuristic quality. Hence, when each ant group finishes its solution construction, a local search is performed to improve that solution.

3.3.1. MAGS Local Search Methods

This local search is divided into two steps: LS, which is quicker and handles all the solutions, and LS2, which is slower and processes only the solutions that have a value close to the optimum one or are too far from it.

LS is a procedure that removes sequences of two unprofitable flight arcs, as shown in Figures 6a and 6b. This enhances the chance of improvement by LS2, which supplements LS, looking for profitable flights that could replace waiting arcs in each ant path, as shown in Figure 6c.

LS procedure also includes a corrective heuristic, which adjusts the solution so that each ant terminal and initial airports are the same, since this is not guaranteed by the basic heuristic. The procedure considers three ways of correcting the final destination: removing final solution flights, as shown in Figures 7a and 7b, adding profitable flights, as shown in Figure 7c, or even adding non-profitable arcs, since aircraft solutions with different initial and final airports are not considered viable.

4. APPLICATION AND RESULTS

The linear programming model and the MAGS heuristic were applied to instances based on a domestic regional airline case that encompasses 104 weekly flights and uses three ATR-42/300 aircraft (for 50 passengers each). Alternative flight networks were generated, composed by different fleet configurations, involving Embraer 120 (for 30 passengers each) and Embraer 170 (for 70 passengers each). Several instances (3 to 9) also include alternative flights for a new destination, expanding the base network to 164 weekly flights, plus thousands of potential repositioning flights, with no markets associated to them.

Three types of demand distribution are considered in different instances.

- Fixed: the demand associated to each flight is fixed at 50 passengers (Instances 1 to 3).
- Flight: the demand is associated to each flight and is the average demand per flight, based on values provided by the Brazilian Civil Aviation regulatory agency – ANAC (2007) (Instances 4 to 6).
- Period: the demand between two airports associated to a period of day – morning or evening – is the average demand by day period, based on values provided by ANAC (2007) (Instances 7 to 9).

Moreover, every instance was presented in two forms: instances marked as "A" were solved with $\mu = \nu = 1$, which means a minimum average load factor $\delta = 50\%$; instances marked as "B" were solved with $\mu = 3$ and $\nu = 1$, leading to a minimum average load factor $\delta = 75\%$.

The instances were solved by integer linear programming techniques through Gurobi Optimizing software version 4, in an Intel Core2 Quad computer with 2GB of memory, using four processing cores and 200GB of available virtual memory. The MAGS heuristic was implemented using the Java SE version 6, running in the same equipment while using only one of the cores, since MAGS was not implemented using parallel programming.

Table 1 shows the results obtained by both the exact mathematical model and MAGS for some of the instances, given a weekly schedule. The resulting values of the objective function represent the total lost revenue and, thus, the lower the value, the better the solution.

Analyzing these results, it is possible to notice that the MAGS heuristic leads to values very close to those obtained by the exact mathematical model, with small standard deviations. When comparing processing times, MAGS heuristic times are more consistent with the problem complexity in terms of arcs and fleets. On more complex problems, involving more fleets, arcs and with multiple flights competing for the demand of each market (instances 5 to 9), MAGS heuristic processing times are usually lower when compared to processing times of the exact model solved with the Gurobi Optimizer, even without a parallel implementation for MAGS. The average MAGS processing time per arc was of 0.04 seconds.

When considering the more complex problems (instances 5 to 9), it is also possible to notice that "B" instances are usually solved in smaller processing times than "A" instances, for both exact and heuristic models. One possible reason for this behavior is the fact that the number of profitable flight sequences is smaller when higher load factors are required.

The minimum values obtained, shown in Table 2, are even closer to the optimal ones: while the average values are distant by up to 7.9% of the optimum, the minima are no more than 3% greater than the optimum value for each case.

The number of transported passengers and resulting load factors are shown in Table 3, which is organized in a different way to allow easy comparison between occupations on "A" and "B" instances, as well as the results achieved by the exact model, tagged as "E", and heuristic model, tagged as "H".

Although in general the average load factor is above 75% (even for "A" instances), several results in Table 3 show the effect of changing μ and ν values. "B" instances present increased load factors compared to the corresponding "A" instances, for both exact and heuristic models. Also, the improvement in the average load factor was similar for both models. This can also be observed by comparing, in relative terms, the reduction of transported passengers, which is generally smaller than the reduction of executed flights.

It is possible to notice that the minimum load factor usually increases from "A" to "B" instances, but the acceptance of occupation ratios lower than the minimum load factor is common. This means that those flights are necessary to deploy the aircraft at the origin of a sequence of flights with high occupation, to attain lower objective function values. Flights below the minimum required load factor are not a problem when the resulting average load factor is higher than the minimum required load factor. This requirement is not met for instance 22H, which means the result is not viable.

CONCLUSIONS

This study presents and compares results of two types of models to solve the flight schedule and the fleet assignment problems in an integrated way. Both models incorporate the same objective function and constraints, including real world operational restrictions such as slots at airports and control over minimum average aircraft load factor without losing their linear characteristics.

The first model relies on Linear Programming and considers several constraints common to real airport operations such as competing flight markets and departure and arrival slots. Furthermore, the model incorporates a new way to set a minimum average load factor, eliminating the need of non-linear constraints.

The second model, a heuristic approach called MAGS – Multiple Ant Group System – is based on the ACO metaheuristic and incorporates the same constraints present in the exact model. In order to represent and to solve the proposed problem, MAGS presents a distinctive way to determine the initial pheromone level on each arc, as well as new alternative rules for the construction of solutions, each one being represented by multiple ants that compete among themselves for arcs and demands associated to those arcs. In addition, the multiple ant solution representation required a new rule for pheromone updating.

The exact model could reach optimal solutions for relatively simpler instances. When considering more complex instances, MAGS reached very close results to the optimal ones in smaller processing times, addressing the possibility to utilize it to solve larger real world problems.

ACKNOWLEDGEMENTS
The authors acknowledge CAPES (Coordenação de Aperfeiçoamento de Pessoal de Nível Superior) and CNPq (Conselho Nacional de Desenvolvimento Científico e Tecnológico) for financial support and LPT/EPUSP (Laboratório de Planejamento e Operação de Transportes da Escola Politécnica da USP) for technical support.

REFERENCES
Abara, J. (1989) Applying Integer Linear Programming to the Fleet Assignment Problem. *Interfaces* , v.19, n.4, p.21-28.

Abrahão, F.T.M. (2005) *A Meta-Heurística Colônia de Formigas para Solução do Problema de Programação de Manutenção Preventiva de uma Frota de Veículos com Múltiplas Restrições: Aplicação na Força Aérea Brasileira.* Thesis. Universidade de São Paulo. São Paulo, Brazil.

ANAC (2007) *Anuário Estatístico da Agência Nacional de Aviação Civil vol.I* . Available at http://goo.gl/GJsba . Accessed on 27/05/2011.

ANAC (2008) *ANAC cria regra de slots para aumentar concorrência em aeroportos saturados* . Available at: http://goo.gl/qXmsi . Accessed on 27/05/2011.

Barnhart, C., A.M. Cohn, E. Johnson, D. Klabjan, G.L. Nemhauser and P.H. Vance (2003) Airline Crew Scheduling in *Handbook of Transportation Science*. 2nd ed. Kluwer's International Series.

Bonasser, U.O. and N.D.F. Gualda (2006) Hybrid Meta-Heuristics for Solution of the Vehicle Routing Problem with Multiple Depots and Heterogenous, Fixed Fleet. In: *XIV Panamerican Conference of Traffic and Transportation Engineering*, Las Palmas de Gran Canária, Spain. L05_30, 14p.

Caetano, D.J. and N.D.F. Gualda (2010) A Flight Schedule and Fleet Assignment Model. In: *12th World Conference on Transport Research*, Lisboa (Portugal). *Selected Procedures of the 12th WCTR (Paper ID: 02492).* WCTRS, v. 1, p. 1-18.

Caetano, D.J. (2011) *Modelagem Integrada para a Programação de Voos e a Alocação de Frotas: Abordagens Baseadas em Programação Linear Inteira e na Meta-Heurística Colônia de Formigas.* Thesis. Universidade de São Paulo. São Paulo, Brazil.

Dorigo, M. and T. Stützle (2003) The Ant Colony Optimization Metaheuristic: Algorithms, Applications and Advances. In: Glover, F; Kochenberger, G.A. *Handbook of Metaheuristics*. Kluwer Academic Publishers. Chap.9, p.251-285.

Dorigo, M. and T. Stützle (2004) *The Ant Colony Optimization*. Bradford Book, 2004.

Gomes, W.P.; Gualda, N.D.F., 2010. Airline crew scheduling in the Brazilian context. In: *12th World Conference on Transport Research*. Lisboa (Portugal). *Selected Procedures of the 12th WCTR (Paper ID: 02861).* WCTRS, v. 1, p. 1-19.

Hane, C., C. Barnhart, E. Johnson, R. Marsten, G. Nemhauser and G. Sigismondi (1995) *A fleet assignment problem: Solving a large-scale integer program*, Technical report, Georgia Institute of Technology, School of Industrial and System Engineering. Report Series 92-04.

Klabjan, D. (2004) Large-scale models in the airline industry. In G. Desaulniers, J. Desroriers, MM Solomon, editors, *Column Generation*, Kluwer Academic Publishers.

Lohatepanont, M and C. Barnhart (2006) Airline Schedule Planning: Integrated Models and Algorithms for Schedule Design and Fleet Assignment. *Transportation Science*, v.38, n.1, p.19-32.

Oliveira, A. (2009) *Transporte Aéreo – Economia e Políticas Públicas*. Pezco Editora. São Paulo.
Rabetanety, A., J. Calmet and C. Schoen, 2006. *Airline Schedule Planning Integrated Flight Schedule Design and Product Line Design.* Thesis. Universität Karlsruhe. Karlsruhe.

Sherali, H.D., E.K. Bish and X. Zhu (2006). Airline fleet assignment concepts, models, and algorithms, *European Journal of Operational Research* , n.172, p.1-30.

Swan, W.M. (2002) Airline route developments: a review of history. Journal of Air Transport Management . n.8. p.349-353.Moore, R., Lopes, J., 1999. Paper templates. In *TEMPLATE'06, 1st International Conference on Template Production*. SciTePress.

Teodorovic, D. (2008) Swarm Intelligence Systems for Transport Engineering: Principles and Applications, *Transportation Research Part C: Emerging Technologies*, v.16, n.6, p.651-667.

Vrancx, P and A. Nowé (2006) Using Pheromone Repulsion to Find Disjoint Paths. In: Dorigo, M; Gambardella, L.M; Birattari, M; Martinoli, A; POLI, R; Stützle, T. (editors) *Ant Colony*

Optimization and Swarm Intelligence 5th International Workshop. ANTS 2006, Brussels, Belgium – Proceedings. Belgium p.522-523.

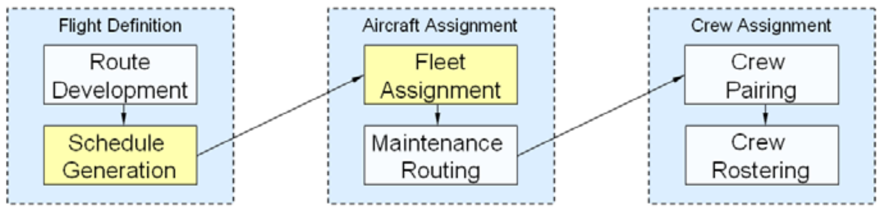

Figure 1: Airline Operational Planning Stages

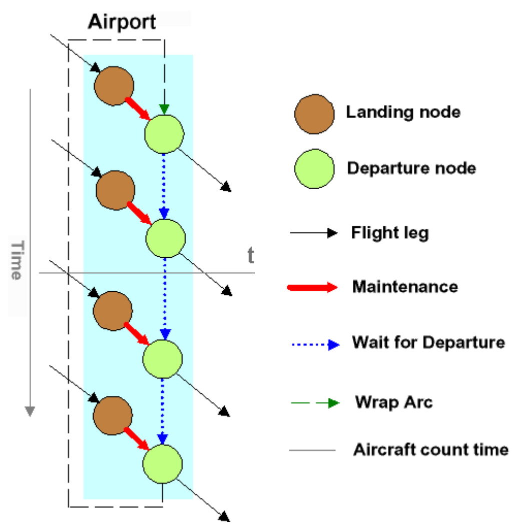

Figure 2: Space-Time Network.

```
procedure ACO
   Scheduled Activities
      Build Ant Solutions
      Update Pheromone Trails
      Auxiliary Actions   %optional
   end Scheduled Activities
end procedure
```

Figure 3: Basic ACO procedure

```
procedure MAGS
  Setup Parameters
  best=Nearest Neighborhood Solution
  Setup Pheromones
  for i Iterations
    for g Ant Groups
      Create Ant Group
      sol=Build Ant Group Solution
      sol=LocalSearch(sol)
      if( sol < 1,05*best OR
          sol > 2*best)
        sol=LocalSearch2(sol)
      if (sol < best) best=sol
    end for g Ant Groups
    Evaporate Pheromones
    Update Pheromome trails
    Update best AntGroup trail
  end for i Iterations
end Procedure
```

Figure 4: MAGS basic logic.

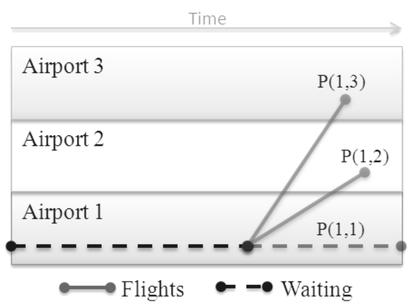

Figure 5: When waiting on airport 1, an ant may choose different arcs to continue, with different probabilities.

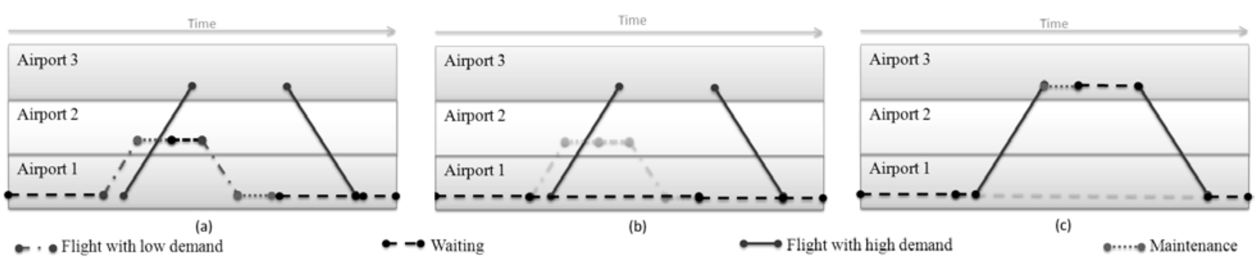

Figure 6: LS and LS2 basic procedures: original solution (a), low demand flight removal by LS (b) and high demand flight inclusion (c).

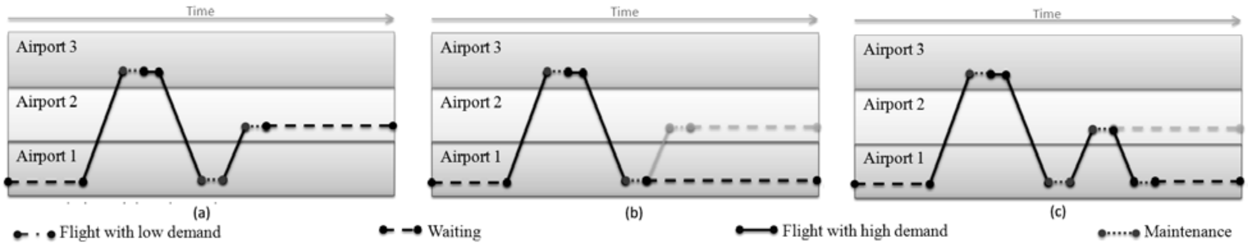

Figure 7: LS correction procedure: original solution (a), flight removal correction (b) and extra flight correction (c).

Table 1: Exact model and MAGS average results.

Instance	Demand Type	Fleets / Aircraft	δ	Arcs	Flight Arcs / Repositioning Arcs (Potential)	Exact Model		MAGS (10 Runs Average)		
						O.F. Value	Time(s)	O.F. Value***	Std.Dev.	Avg. Time (s)
1A	Fixed	3 / 3	50%	35,367	312 / 15.636	0	3	0	0	0
1B	Fixed	3 / 3	75%	35,367	312 / 15.636	0	5	0	0	3
2A	Fixed	3 / 3	50%	35,367	312 / 15.636	92,200	3	92,200	0	548
2B	Fixed	3 / 3	75%	35,367	312 / 15.636	230,500	3	238,590	1,061	973
3A	Fixed	3 / 3	50%	44,907	492 / 20.316	175,000	7	177,500	0	1,240
3B	Fixed	3 / 3	75%	44,907	492 / 20.316	175,000	6	177,500	0	1,181
4A	Flight	3 / 3	50%	44,907	492 / 20.316	897,345	14	918,174	2,463	1299
4B	Flight	3 / 3	75%	44,907	492 / 20.316	915,785	18	961,331	14,946	1,096
5A	Flight	5 / 5	50%	74,845	820 / 33.860	809,365	172,800*	819,779	1,351	1,478
5B	Flight	5 / 5	75%	74,845	820 / 33.860	819,815	1,844	882,253	14,007	1,173
6A	Flight	3 / 5	50%	44,907	492 / 20.316	810,345	14,249	838,928	7,882	1,562
6B	Flight	3 / 5	75%	44,907	492 / 20.316	819,815	919	884,689	8,695	970
7A	Period	1 / 3	50%	14,969	164 / 6.772	868,650	68	873,812	4,057	1,844
7B	Period	1 / 3	75%	14,969	164 / 6.772	868,650	43	912,530	4,334	749
8A	Period	2 / 3	50%	29,938	328 / 13.544	812,650	78,133**	827,188	7,152	2,501
8B	Period	2 / 3	75%	29,938	328 / 13.544	812,650	46	854,254	9,510	955
9A	Period	3 / 5	50%	29,938	328 / 13.544	788,550	172,800*	794,782	5,998	6,195
9B	Period	3 / 5	75%	29,938	328 / 13.544	793,950	172,800*	839,344	12,659	1,797

(*) Processing was interrupted after the 2-day time limit (172.800 seconds).
(**) Processing was interrupted due to insufficient memory (2GiB RAM + 200GiB hard disk virtual memory).
(***) Average values do not include the constructive heuristic results.

Table 2: MAGS minimum results.

Instance	MAGS (10 Runs Minimum)		
	Value*	MAGS / Optimum	Average Time (s)
1A	0	100.0%	0
2A	92,200	100.0%	548
3A	177,500	101.4%	1,240
4A	916,475	102.1%	1,299
5A	818,825	101.2%	1,478
6A	832,965	102.8%	1,562
7A	871,170	100.3%	1,844
8A	819,830	100.9%	2,501
9A	790,210	100.2%	6,195

Instance	MAGS (10 Runs Minimum)		
	Value*	MAGS / Optimum	Average Time (s)
1B	0	100%	0
2B	208,000	90.2%**	973
3B	177,500	101.4%	1,181
4B	923,225	100.8%	1,096
5B	839,435	102.4%	1,173
6B	837,065	102.1%	970
7B	878,450	101.1%	749
8B	816,290	100.4%	955
9B	794,150	100,0%	1,797

(*) Minimum values do not include the constructive heuristic results.
(**) Minimum average load factor was not accomplished.

Table 3: Minimum and Average Load Factor.

	A - δ = 50%					B - δ = 75%				
Inst.*	Passengers	Flights	Load Factor (%) Average	Load Factor (%) Minimum	Revenue Lost	Passengers	Flights	Load Factor (%) Average	Load Factor (%) Minimum	Revenue Lost
1 E	5,200	104	100	100	0	5200	104	100	100	0
1 H	5,200	104	100	100	0	5200	104	100	100	0
2 E	5,200	104	71.4	71.4	92,200	0	0	-	-	230,500
2 H	5,200	104	71.4	71.4	92,200	750	15	71.4	71.4	208,000
3 E	4,700	94	100	100	175,000	4,700	94	100	100	175,000
3 H	5,200	104	100	100	177,500	5,200	104	100	100	177,500
4 E	3,555	80	92.3	44	897,345	3,503	76	96.8	66	915,785
4 H	3,934	102	83.1	28	916,475	3,663	82	94.1	28	923,225
5 E	4,993	127	88.7	44	809,365	4,264	101	95.5	66	819,815
5 H	4,912	124	86.9	28	818,825	4,294	107	92.7	44	839,435
6 E	4,931	126	89.2	44	810,345	4,264	101	95.5	66	819,815
6 H	4,827	126	85.8	0	832,965	4,279	107	91.9	28	837,065
7 E	3,930	84	95.6	88	868,650	3,930	84	95.6	88	868,650
7 H	4,090	92	92.8	30	871,170	3,881	84	95	34	878,450
8 E	4,330	84	95.6	88	812,650	4,330	84	95.6	88	812,650
8 H	4,409	86	94.4	70	819,830	4,325	85	94.9	34	816,290
9 E	5,215	143	88.9	46.7	788,550	4,745	109	93.6	56.7	793,950
9 H	5,206	145	88.9	0	790,210	4,850	107	92.3	30	794,150

(*) *E* refers to the exact model and *H* to the heuristic model.