Aircraft Ground Routing and Scheduling Optimization

Ludovica Adacher  
Dipartimento di Ingegneria  
Università degli Studi Roma Tre  
Rome, Italy  
Email: adacher@dia.uniroma3.it

Marta Flamini  
Università Telematica Internazionale  
UNINETTUNO  
Rome, Italy  
Email: m.flamini@uninettunouniversity.net

Abstract—In this paper we deal with the ground optimization problem, that is the problem of routing and scheduling airplanes surface maneuvering operations. We consider the specific case study of Malpensa Terminal Maneuvering Area (Italy). Our objective function is the minimization of total tardiness. At first a routing problem is solved to assign a path to each aircraft in the terminal, then the scheduling problem of minimizing the average tardiness is addressed. We model the scheduling problem as a job-shop scheduling problem. We develop heuristic procedures based on the alternative graph formulation of the problem to construct and improve feasible solutions. Experimental results based on real data and analysis are reported.

Keywords—Air traffic control, Aircraft scheduling problem, Job-shop

I. INTRODUCTION

Due to the fast increasing of flights and passengers, the finite airport resources often cannot meet the demand at busy time interval causing possible serious congestion and potential operation safety-risk events. How to effectively guarantee high operation performances in case of airport congestion respecting the safety, has become a relevant issue. Many attentions are addressed to airport ground traffic management systems to improve efficiency and guarantee safety of surface operations, by studying methods for aircraft taxing route optimization problem. Airports management is entrusted to dispatchers in air control, local control and ground control. Air control tasks concern (i) routing decisions, where a route for each aircraft has to be chosen from its current position to its destination, and (ii) scheduling decisions in the vicinity of the airports, where air routes are fixed, and feasible aircrafts sequencing and timing have to be determined such that safety rules are satisfied. Local control is responsible of runways control and consist in sequencing and timing the use of the runway by the aircrafts.

Ground control is responsible for aircraft traffic management in the Terminal Maneuvering Area (TMA), that generally include all taxi ways, inactive runways, yards, aprons or intersections. Currently, controllers perform aircraft traffic control operations supported only by graphical representations of the aircrafts position and speed. Decision support systems would be useful both to optimize management performances and to face unexpected events without neglecting the optimization of ground traffic management even in such cases. Problems arising in aircraft traffic control are usually hard to perform in terms of designing both adequate models and efficient and effective solution techniques, due to the complex real aspects that have to be included in the model representation in order to produce solutions that can be actually implemented.

The importance of the ground movement problem was explained in [8]. Improved ground movement can increase on-time performance at airports, so ground movement simulations and optimizers are extremely useful. Several approaches have been proposed for solving both air and ground scheduling problems ([6],[11], [3],[4],[10]), but most of the early contributions suffer for a substantial lack of information due to the usage of very simplified models. Recent models ([2],[5],[1]) introduce an increased level of realism and incorporate a larger variety of constraints and possibilities, such as no-wait constraints, and earliness/tardiness penalties for each early/tardy aircraft. Most of the research efforts focus on local control [2],[6] where runway scheduling problems are analyzed. Due to the complexity of such problems researchers often propose the developing of heuristic procedures [9],[7],[6],[12], [15] and distributed techniques [13], [14].

In this paper we deal with the ground optimization problem, that is the problem of routing and scheduling airplanes surface maneuvering operations. The case study we refer to is related to the Terminal Maneuvering area of Malpensa Airport (Italy) (from now on TMA). We consider the minimization of the average tardiness. We propose a two-steps solution approach. At first we solve a ground aircraft routing problem, which assigns a ground route to each aircraft; subsequently we apply scheduling algorithms to solve ground aircraft scheduling problem with fixed routes. We test two different approaches to solve the routing problem and we measure their performance on the final solution.

We associate the ground aircraft scheduling problem to a job shop scheduling problem with additional constraints. We model the problem by the alternative graph [?]. The constrains of the problem are those arising by the safety rules regulating the aircraft ground maneuvering.
In Section II a detailed description of the problem is presented. In Section III authors propose a model for the ASP based on the alternative graph formulation [4]. Solution heuristic procedures are described in Section IV and experimental results are reported in Section V. Section VI is dedicated to the conclusions.

II. THE TMA OF MALPENSA AIRPORT

In this paper we minimize the average tardiness, solving a ground optimization problem, by routing and scheduling airplanes surface maneuvering operations, respecting the safety rules regulating ground traffic a Terminal Maneuvering Area.

We study the specific case of the Terminal (TMA) of Malpensa Airport (Italy). The TMA layout is depicted in Figure II.

The TMA has two parallel runways, namely $35R$ and $35L$, two yards, two aprons, and a taxiway network. The runways are divided into segments, each between two consecutive intersections with the taxiway network. Yards are areas positioned at the beginning of the runways, in which aircrafts wait before departing. Aprons include parking bays in which the aircrafts are parked. Stop bars regulate the access to the intersections.

Runway $35L$ has to be crossed by aircrafts departing/landing on the runway $35R$ and coming from/going to apron A1. Hence, it becomes crucial to develop detailed models able to adequately represents all safety constraints in the TMA in order to produce feasible solutions.

We consider both arriving and departing airplanes, in a given time interval. In the solution of the ground routing problem, each arriving aircraft has associated a ground route from its current position to the parking bay and each departing aircraft has associated a ground route from its current position to the take-off runway. We are given a nominal landing time and an arrival time for the arriving aircrafts. The nominal landing time is the minimum time instant in which the aircraft would be able to land. The nominal landing time is a variable of the problem, since in practice the landing instant could be postponed for optimization scopes. The arrival time represents the time instant in which the aircraft should reach the parking bay. For each departing aircraft we are given a departing time that is the time instant in which the aircraft should take-off. A possible path for a landing aircraft is a sequence of resources from the runway exits to the gate; a possible path for take off aircraft is a sequence of resources from the he parking bay to a given runway takeoff position. In Malpensa airport the are 160 landing routes and 54 take-off routes. Hence, we consider two different lists of possible routes, one for the landing airplanes the other for take-off airplanes.

In the first work [4], we considered a casual assignment of route to the aircraft, i.e. each aircraft had assigned a route based on a uniform distribution association rule.

Now, in order to improve the solution, we propose a new routing approach. We first order the landing/take off aircraft respect to the nominal time. Applying FIFO rule, we assign at each landing/take off aircraft the most promising route, considering two criteria:

- the number of requested common resources in the route
- the running time of route considering the airport empty

Hence, considering our objective function at the first aircraft of the ordered list we assign the shortest route or the route that requests the minimum number of resources. Then we proceed with the following airplanes in the ordered list eliminating the routes already assigned. Also a criteria deriving from the convex combination of the two above is considered. At this point each aircraft has a route associate

Figure 1. Resources of Malpensa TMA

III. GROUND ROUTING PROBLEM

The ground routing problem consists in associating a ground route to each aircraft, either once landed or if parked. We consider all the arriving airplanes and the airplanes moving or parked in the TMA, in a certain time interval.

We are given a nominal landing time and an arrival time for the arriving aircrafts. The nominal landing time is the minimum time instant in which the aircraft would be able to land. The nominal landing time is a variable of the problem, since in practice the landing instant could be postponed for optimization scopes. The arrival time represents the time instant in which the aircraft should reach the parking bay. For each departing aircraft we are given a departing time that is the time instant in which the aircraft should take-off. A possible path for a landing aircraft is a sequence of resources from the runway exits to the gate; a possible path for take off aircraft is a sequence of resources from the he parking bay to a given runway takeoff position. In Malpensa airport the are 160 landing routes and 54 take-off routes. Hence, we consider two different lists of possible routes, one for the landing airplanes the other for take-off airplanes.

In the first work [4], we considered a casual assignment of route to the aircraft, i.e. each aircraft had assigned a route based on a uniform distribution association rule.

Now, in order to improve the solution, we propose a new routing approach. We first order the landing/take off aircraft respect to the nominal time. Applying FIFO rule, we assign at each landing/take off aircraft the most promising route, considering two criteria:

- the number of requested common resources in the route
- the running time of route considering the airport empty

Hence, considering our objective function at the first aircraft of the ordered list we assign the shortest route or the route that requests the minimum number of resources. Then we proceed with the following airplanes in the ordered list eliminating the routes already assigned. Also a criteria deriving from the convex combination of the two above is considered. At this point each aircraft has a route associate

In addition to the minimization of average tardiness, we also focus on some performance indices that give a measure of the performance of terminal control and management. Such indices concern the number on tardy aircraft, absolute average lateness and the pollution.
that from now on we consider fixed. The following phase provides a schedule of the movements of the airplanes in the TMA based on the fixed routing.

IV. GROUND SCHEDULING PROBLEM

The ground scheduling problem we deal with can be modeled as a job-shop scheduling problem with additional constraints. Jobs are represented by aircrafts and resources by parts of the TMA. An operation is the occupation of a resource by an aircraft, therefore the processing time of an operation corresponds to the time necessary to run/occupy a certain resource. Constraints implied by safety rules, typically regulating spatial or temporal security distances, hold for each pair of aircrafts moving in the TMA. We model the TMA as a set of resources of limited capacity. The resources we consider are runway segments, taxiway segments, parking bays, stop bars, yards. All such resources, except for the yards, have unitary capacity. Yards are assumed to have unlimited capacity.

Each route associated to an aircraft corresponds to a list of resources. Standard job-shop constraints hold, namely: (i) precedence constraints: resources composing a route have to be run in a precise order; (ii) capacity constraints: resources with unary capacity can be occupied at most by an aircraft at a time; (iii) preemption constraints: operations can not be interrupted. Additional constraints, that is no-wait, blocking and incompatibility constraints, are introduced to model real aspects of the problem related to the safety rules. No-wait constraints state that a given set of consecutive resources has to be occupied without interruption, this is the case, for instance, of segments composing a runway. Blocking constraints model the absence of buffer between two resources, therefore the occupation of a resource of unary capacity by an aircraft makes such resource unavailable for other aircrafts for the entire occupation time interval. Incompatibility constraints hold when two different resources can not be occupied at the same time. In this case the two resources are said incompatible. For instance, all the segments of a runway can be occupied only by an aircraft at a time. This make each pair of segments of the same runway incompatible.

A conflict arises when two aircrafts have to run the same resource or two incompatible resources at the same time. A conflict is solved by fixing a precedence between the two aircrafts.

Therefore, a feasible solution for ground scheduling problem consists in solving all the conflicts, avoiding deadlocks.

The model we use to represent the problem is the Alternative Graph model [4] briefly described in Section IV-A. Relying on that model, we develop fast heuristic algorithms, to solve the ground scheduling problem.

A. The Alternative Graph Model

An alternative graph is an oriented graph \( G(N,F,a) \), in which \( N \) denotes the set of nodes, \( F \) is the set of fixed arcs and \( a \) is the set of alternative arcs. We denote by \( R \) the set of problem resources, reported in Figure II, and it is possible schematized as:

- runways intersections \( \{35R0, 35R1, \ldots, 35R5\} \) for runway \( 35R \) and \( \{35L1, \ldots, 35L6\} \) for runway \( 35L \);
- taxiway cross points \( \{X0, X1, C0, \ldots, C10, W0, \ldots, W7, K0, \ldots, K7\} \);
- parking bays \( \{P0, P1, \ldots, P7\} \);
- stop bars \( \{sb1, sb2, \ldots, sb10\} \).

For example, if we consider one landing aircraft \( a \) and one departing aircraft \( b \). Routes are

\[
R_a = \{35R5, 35R4, 35R3, sb7, C1, sb2, 35L2, W0, K1, P1\}
\]

and

\[
R_b = \{P2, W1, W0, sb9, 35L3, C1, C2, C3, C4, C5, C8, 35R5 – R0\}
\]

for \( a \) and \( b \) respectively. The two routes are represented as chains including fixed arcs.

Let \( \nu_i^j \in N \) be the node associated to the occupancy of resource \( i \in R \) by aircraft \( j \). For sake of simplicity we use the notation \( \nu_i^j \), both to denote a node of the alternative graph and the associated operation. Set \( N \) includes two dummy nodes, \( s \) and \( t \) associated to time zero and to the completion of the last operation, respectively. A fixed arc \((\nu_i^j, \nu_k^l) \in F\), between two consecutive nodes in an aircraft route represents the precedence constraint. The arc weight \( p_i^j \) indicates the processing time of operation \( \nu_i^j \), that is equal to the minimum occupation time of the resource \( i \). The minimum occupation time for a runway intersection is equal to the running time of the runway segment succeeding the intersection. The minimum occupation time for a taxiway cross point is equal to the running time of the taxiway succeeding the cross point. The minimum occupation time for stop bars and parking bays is fixed to zero.

As mentioned above, conflict occurs each time two aircrafts request the same resource or two incompatible resources. Two resources are defined incompatible if they can not be occupied at the same time due to safety rules. Each conflict is modeled by a pair of alternative arcs which represent the two possible precedences between aircrafts involved in the conflict. Conflict arcs have a weight equal to \( \varepsilon \rightarrow 0 \). A conflict is solved when a precedence is established, hence when one of the two arcs of the associated alternative pair is selected and the other canceled. A feasible solution for the ground scheduling problem consists in solving all possible conflicts among aircrafts, avoiding deadlocks (i.e. the associated alternative graph is cycle free). For more details see [4].

B. Scheduling Heuristics

Due to the real nature of the problem we first compute different fast initial solutions, then we try to improve the quality of the solution by applying local search heuristics.
List of real criteria $LP$

- **criterion 0**: when we consider to different landing (departing) airplanes we select in AG the arc of the alternative pair on the bases on FIFO order.
- **criterion 1**: in runway, when we consider a landing and a departing aircraft we always select the arc that gives the precedence at the landing airplane.
- **criterion 2**: in taxiway, stop bar, parking bay, when we consider a landing aircraft and a departing aircraft we always select the arc that gives the precedence at the departing airplane.

![Figure 2. List of precedence $LP$](image)

based on the alternative graph representation. Any potential conflict on resources between landing and departing aircrafts must be detected and solved. We consider a list of real dispatching criteria to assign the precedences (see Fig. IV-B) to find a feasible solution.

The Arc Greedy Heuristic ($AGH$)(see Fig IV-B), based on an iterative arc selection procedure, at each step, $AGH$ selects all the alternative arcs involving a chosen aircraft, so that the sequencing of the operations of this aircraft with respect to the others is fixed. Note that at each step the relative order of remaining jobs with previously selected jobs is already fixed, so it only remains to decide the relative order of unselected jobs. Ordered all the aircrafts according to the $FIFO$ rule respect to nominal landing/departing time, the precedence between two consecutive landing (departing) airplanes is given on $FIFO$ rule, if a conflict occurs in the runway a landing aircraft always precedes a departing aircraft and if occurs in taxiway, stop bars, parking bays, a departing aircraft precedes a landing one. We also use the $EDD$ rule, instead $FIFO$, for the arc selection. The $EDD$ rule solves aircraft conflict situations by assigning each conflicting resource to the aircraft with minimum due date. We applied the $EDD$ rule to optimize the objective function, the other two types of criteria represent safety constrains.

The $AGH$ based on $FIFO$ rule represents a good estimation of the solution currently produced by controllers in terms of both safety and quality. Note that the $AGH$ based on $EDD$ is implemented to reduce the number of tardy aircrafts, the precedence is given on the more pressing due date.

Once a feasible solution $x$ has been constructed, it is modeled by the alternative graph and local search procedures are applied. The alternative graph, is a useful instrument to rapidly estimate the effect produced by modifying the structure of a solution. The Local Greedy Heuristic ($LGH$) is a family of heuristic algorithms (see Fig IV-B). The family is based on the idea of repeatedly choose an alternative arc from the set $a$ and by selecting the other arc of the alternative pair until an improvement is found. The exchange is not accepted if a cycle is detected or the solution does not improve. At each step the arc is selected in an ordered list $S$ constructed on a local rule ($LR$) every time an improvement is obtained the list $S$ and the solution $x$ are updated.

The basic idea is to try and iteratively invert precedences between two aircrafts. An inversion is accepted only if the new solution is feasible and improves. Given a feasible solution from $AGH$, we calculate the lateness for each aircraft, comparing the due date with the heads of nodes representing (i) the arrival to the parking bay for the arriving aircrafts, and (ii) the arrival to the first segment of runway for departing aircrafts. Naturally, if the lateness is negative we have an early aircraft, if positive a tardy aircraft and if equal to zero a on time aircraft. Note that it is impossible to meet exactly the due date, for this reason we have introduced a time interval for each aircraft, instead of time instant, in which it is on time. On the other hand, this really small interval is trivial respect to the whole flight. Obviously, the $LGH$ inverts precedences between aircrafts if and only if the early aircraft precedes the tardy aircraft, otherwise it is not possible to improve the solution since we can improve the first objective function only if we delay a early aircraft.
or if anticipate a tardy aircraft. We develop two heuristic procedures based on two different LR: 

**mML** heuristic inverts precedence between the aircraft with the minimum absolute value of lateness preceding the aircraft with the maximum absolute value of lateness, to reduce the number of tardy aircrafts; 

**ET** heuristic inverts precedence between the aircraft with minimum earliness preceding the aircraft with maximum tardiness, to improve the number of on time aircrafts. 

In practice, 

**MmL** (maximum lateness-minimum lateness) detects the aircraft with maximum absolute value of lateness and the aircraft with minimum absolute value of lateness if there exists the arc from maximum and minimum then **MmL** selects the other arc of the alternative pair. If this exchange does not improve the solution the **MmL** continues with the second minimum and the maximum and so on. Notice that for the **MmL** a change is done if and only if it delays an early aircraft and anticipates a tardy aircraft. 

**Heuristic ET** (earliness-tardiness) detects the aircraft with minimum earliness and the aircraft with maximum tardiness. If there exists the arc from early to tardy aircraft then **ET** selects the other arc of the alternative pair; if this exchange does not improve the solution, or a positive length cycle is detected, the **ET** continues with the second minimum early and the maximum tardy aircrafts, then with the third minimum early and so on. If a improvement solution is not found the **ET** tries with the aircraft with the second maximum tardiness and first minimum early aircraft then with the second minimum early aircraft and so on. 

These two types of exchange are done on the **AGH** feasible solution, when a change improvements the solution the **LGH** rules are applied on this new solution, and the scheme is repeated until a new improvement is found or a feasible solution does not exist. 

V. Computational Experiments 

Experiments are carried out considering as a reference test case one busy hour of arrivals and departures at the Malpensa international airport. From the study a we have tested different instances (denoted as n,m,x,t) based on four parameters (n,m,x,t): 

- **n**: number of departures (n ∈ {4,5,6,8,10,15})
- **m**: number of arrivals (m ∈ {4,5,6,8,10,15})
- **x**: number consecutive arrivals/departures (x ∈ {2,3,4,5,6})
- **t**: simulation constant (t ∈ {4,5,6,8,10,12,20})

The simulation constant t and the TMA average throughput time X defines the simulation time interval T = t * X. The routing problem is solved off-line in a preliminary step and the we apply on real-time scheduling decisions with fixed routes (we have tested only real routing). In particular we deal with the problem of scheduling airplanes moving on ground, with the objective of minimizing the average tardiness. In **AGH** the **EDD** rule gives better results in comparison to **FIFO**: if we consider the number of tardy aircrafts the **EDD** improvement is around 25%, while the average tardiness improves of 60%. Instead if we consider on time aircrafts the improvement is obviously inferior and it is attested around 5%, and respect to the average lateness is 30%. The main results are shown in the following tables, where only the **EDD** rule is considered. The behavior of these two types of local search **ET** and **MmL** is more or less the same. Also the different criteria of choosing route give more or less the same results, the improvement given by the less common rescouses is limited at 0.5%, often the shortest route is also the route with less resources. In the following, the routing fixed by casual assignment based on uniformed distribution of real routes is called stochastic routing and the routing fixed by proposed criteria is called static routing. 

### Table I 

**Average Tardiness: Comparison Between Static and Stochastic Routing. The Results are Given in Second** 

<table>
<thead>
<tr>
<th>Class</th>
<th>stochastic routing</th>
<th>static routing</th>
</tr>
</thead>
<tbody>
<tr>
<td>4, 4, 4, 4</td>
<td>54 41</td>
<td>45</td>
</tr>
<tr>
<td>4, 4, 4, 8</td>
<td>91 74</td>
<td>92</td>
</tr>
<tr>
<td>5, 5, 5, 5</td>
<td>79 76</td>
<td>73</td>
</tr>
<tr>
<td>5, 5, 5, 10</td>
<td>161 140</td>
<td>146</td>
</tr>
<tr>
<td>6, 4, 4, 6</td>
<td>128 115</td>
<td>110</td>
</tr>
<tr>
<td>6, 4, 4, 6</td>
<td>191 178</td>
<td>185</td>
</tr>
<tr>
<td>6, 6, 6, 6</td>
<td>236 207</td>
<td>225</td>
</tr>
<tr>
<td>6, 6, 6, 12</td>
<td>372 370</td>
<td>380</td>
</tr>
<tr>
<td>8, 4, 4, 8</td>
<td>91 83</td>
<td>82</td>
</tr>
<tr>
<td>4, 8, 4, 8</td>
<td>96 82</td>
<td>88</td>
</tr>
<tr>
<td>10, 5, 5, 8</td>
<td>128 108</td>
<td>108</td>
</tr>
<tr>
<td>5, 10, 5, 8</td>
<td>139 130</td>
<td>124</td>
</tr>
<tr>
<td>10, 5, 5, 8</td>
<td>145 130</td>
<td>132</td>
</tr>
<tr>
<td>5, 10, 5, 8</td>
<td>156 139</td>
<td>148</td>
</tr>
<tr>
<td>10, 10, 5, 10</td>
<td>149 133</td>
<td>131</td>
</tr>
<tr>
<td>10, 10, 5, 20</td>
<td>180 165</td>
<td>174</td>
</tr>
</tbody>
</table>

In Table 1 is reported the value of the objective function in second. It is important to notice that when we apply the stochastic routing if the local search **ET** gives always an improvement of the the found feasible solution by the **AGH**, and when we apply the static routing the local search does not work. In practice, the static routing gives good solution and an improvement is not possible: fixed the more promising route, the local search is not able to find possible change of disjunctive arcs. Note that in these tables we have only considered the **AGH** solution based on **EDD** rule that gives an improvement of 25% compared to **AGH** solution. 

In Table 2 the improvement of **MmL** compared to the **AGH** solution is reported. The first column reports the percentage of on time aircrafts for **AGH** and the second for **MmL**. The last column represents the absolute lateness improvement due the local search. 

The comparison puts in evidence that the local search applied to the static routing gives solutions very similar to
the static routing based on average tardiness, number of on time aircraft and absolute lateness. Completely different is the case of pollution. A preliminary choice of the route is fundamental. It is possible limit the use of the common resources, reducing the waiting time with the engine on. The analysis of the pollution is reported in Table 3. It is represented the total waiting time with the engine on. The improvement given by a good choice of the route is fundamental, it is around 50% respect to the stochastic one. Moreover the \( GH \) applied to the static routing gives always an admissible solution whereas the \( GH \) applied to the stochastic routing find an admissible solution only in the 70% of cases.

**TABLE II**

\( M_{on}L \) IMPROVEMENT RELATED TO THE MAXIMIZATION OF THE NUMBER OF ON TIME AIRCRAFTS. THE RESULTS ARE GIVEN IN PERCENTAGE

<table>
<thead>
<tr>
<th>Class</th>
<th># On time aircrafts</th>
<th>Improvement Average Lateness [%]</th>
</tr>
</thead>
<tbody>
<tr>
<td>4_4_4_4</td>
<td>28</td>
<td>1</td>
</tr>
<tr>
<td>4_4_4_8</td>
<td>28</td>
<td>7</td>
</tr>
<tr>
<td>5_5_5_5</td>
<td>16</td>
<td>5</td>
</tr>
<tr>
<td>5_5_5_10</td>
<td>12</td>
<td>4</td>
</tr>
<tr>
<td>6_4_6_6</td>
<td>10</td>
<td>5</td>
</tr>
<tr>
<td>6_6_6_12</td>
<td>11</td>
<td>4</td>
</tr>
<tr>
<td>8_4_4_8</td>
<td>9</td>
<td>5</td>
</tr>
<tr>
<td>4_4_4_8</td>
<td>30</td>
<td>3</td>
</tr>
<tr>
<td>10_5_5_8</td>
<td>7</td>
<td>4</td>
</tr>
<tr>
<td>5_10_5_8</td>
<td>8</td>
<td>2</td>
</tr>
<tr>
<td>10_5_5_8</td>
<td>12</td>
<td>5</td>
</tr>
<tr>
<td>5_10_3_8</td>
<td>9</td>
<td>9</td>
</tr>
<tr>
<td>10_10_5_10</td>
<td>12</td>
<td>8</td>
</tr>
<tr>
<td>10_10_5_20</td>
<td>3</td>
<td>4</td>
</tr>
</tbody>
</table>

**VI. CONCLUSION**

In this paper we introduced the alternative graph model for ASP in the TMA of Malpensa airport. We show that even a simple greedy heuristic is able to reduce when compared to commonly adopted policies. We observe that an advanced real-time scheduling system may be useful to optimize the traffic conditions, improving the safety and the pollution on ground. We also evidence the importance of good choice of the routing also to find a good scheduling. We are investigating if changing the static route with more sophisticated method can give better solution. Other local search based on different policies will be considered to reduce the time with the engine on, for instance by transferring the waiting time from the taxy ways to the parking bays.

**REFERENCES**


