Planning Aircraft Taxiing Trajectories via a Multi-Objective Immune Optimisation

Jun Chen
School of Engineering
University of Lincoln, Brayford Pool
Lincoln, United Kingdom

Paul Stewart
School of Engineering
University of Lincoln, Brayford Pool
Lincoln, United Kingdom

Abstract—Airport operations include departure sequencing, arrival sequencing, gate/stand allocation and ground movements (taxying). During the past few decades, air traffic at major airports has been significantly increased and is expected to be so in the near future, which imposes a high requirement for more efficient cooperation across all airport operations. A very important element of this is an accurate estimation of the ground movement, which serves as a link to other operations. Previous researches have been concentrated on the estimation of aircraft taxi time. However, such a concept should be stretched more than just predicting time. It should also be able to estimate the associated cost, e.g. fuel burn, for it to achieve such an expected time. Hence, in this paper, an immune inspired multi-objective optimisation method is employed to investigate such trade-offs for different segments along taxiways, which leads to a set of different taxiing trajectories for each segment. Each of these trajectories, on the one hand, provides an estimation of aircraft taxi time, and on the other hand, has great potential to be integrated into the optimal taxiway routing and scheduling process in a bid to find out the optimal taxiing not only in terms of reducing total taxi time but also in terms of lowering fuel consumption.

Keywords-multi-objective immune algorithm; aircraft ground movement; optimal taxiing trajectories

I. INTRODUCTION

Airport ground movements played a major part in the ever increased annual average delays of the flights [1] for a simple reason that they serve as links to other airport operations, such as departure sequencing, arrival sequencing and gate/stand allocation. Any decision based on the wrong hypothesis of the ground movements will be quickly propagated, if not exaggerated, to other parts of operations. Although expanding or changing the existing airport layouts may represent a potential way to mitigate this effect, physical and environmental constraints often overrule such an option as a feasible solution. Hence, the call for a more effective use of existing airport resources becomes more pressing. Responding to such a requirement, taxi planning, which includes the aircraft routing and scheduling on the airport ground, plays a central role and often relies on the accurate estimation of ground movements. Such estimation forms the basis not only for optimal allocation of airport ground resources within a single airport at the tactic level, but also for the optimal air flow management across multiple airports at the strategic level [2]. The main focus of this paper has been placed on the tactic level so that further taxiway routing and scheduling can be built upon it.

Most of previous works on ground movements rely themselves more or less on the estimated taxi time and can be broadly classified into the following categories depending on how taxi time is specified: 1) in [1] and [3], a constant speed is assumed either for each individual link depending on its location on the taxiways [1] or for the whole taxiways [3] so that taxi time can be obtained by dividing the length of the link by its corresponding speed; 2) in [4-5], a maximum taxi speed constraint is specified either for the whole taxiways [4] or for each individual link [5] within an optimisation model with the aim to reduce the total taxi time; in such a case, taxi time for each link is governed by a feasible taxi speed constrained by the defined speed limit; 3) [2] and [6-9] represent a statistical way of predicting taxi time by fitting some types of linear regression to historical data.

The assumption made by all the aforementioned works is that by reducing the total taxi time one can improve the efficiency of airport operations and can reduce the fuel consumption at the same time. This is because, under this assumption, aircraft engines are only turned on with less amount of taxi time. However, such an assumption may not always be true since the detailed relationship between fuel consumption and the corresponding speed profile is not investigated in such a consideration. Hence, it may often be the case that more fuel is burned in exchange for a faster speed so that the total taxi time can be reduced. Hence, an optimal taxi planning in terms of taxi time may not be optimal in terms of fuel consumption. And for this reason, the study of the ground movement should not be restricted to estimating taxi time. The corresponding fuel consumption to achieve such an expected time should also be provided so that both of them can be considered simultaneously in a ground optimisation model.

In the light of the above discussion, in this paper, an immune inspired multi-objective optimisation algorithm [10-11] is employed to work along with a simplified aircraft dynamic model in a bid to find out different optimal taxiing trajectories for each segment of taxiways. Each of these trajectories represents an optimal trade-off between taxi time
and fuel consumption. With such information available, further insight into optimal taxi planning not only in terms of taxi time but also in terms of fuel burn can be gained. Hence, the rest of the paper is organised as follows: Section II gives the detailed problem description and the way to tackle it; the emphasis of this section has been placed on how to effectively reduce the search space so that the problem becomes more tractable; Section III briefly discusses a Population Adaptive based Immune Algorithm (PAIA), which is used as the multi-objective search engine for finding out optimal taxiing trajectories; in Section IV, experimental results for a particular taxiway on Manchester Airport are presented; and finally, conclusions and future research directions are given in Section V; the potential to integrate the obtained trajectories into the existing taxi planning optimisation framework is also highlighted in this section.

II. PROBLEM DESCRIPTION

A. Airport Surface Representation

Generally speaking, Airport surface can be viewed as an directed graph $G = (V, E)$ [4] where vertices $V$ are the located points of the airport (they can be taxiway intersections, gates/stands, runway access points and the waiting points within two consecutive taxiway intersections), and edges $E$ represent links between vertices to runways. In this paper, only the departure process is considered. In such a case, $G$ is reduced to a direction graph starting from gates/stands and ending at runway takeoff points. However, it is worth mentioning that the proposed method also works for the taxi-in process for arrivals.

For departures, a set of consecutive vertices $(v_i, v_j) \in E$, for $i \neq j$, defines a path which routes an aircraft from the gate/stand $v_i$ to the runway takeoff point $v_j$. The distance of consecutive vertices $(v_i, v_j)$ is denoted as $d(v_i, v_j)$.

B. Aircraft Model

In order to investigate the speed profile (taxiing trajectory) of a certain taxiing aircraft and its associated fuel consumption, an aircraft model is needed. Since taxi speed is quite slow compared to the speed in flight it is reasonable to assume that there is no aerodynamics associated with the taxiing aircraft in this work. Under such an assumption, the dynamic model of an aircraft is mainly determined by three important factors, namely the maximum takeoff weight, the engine size and the rolling resistance. In the following, Airbus A320 is used as an example to demonstrate how an aircraft model is built based on the aforementioned three factors. A320 is powered by two IAE V2500 engines each of which gives a maximum thrust $TR_m$ around 113kN. When taxiing out, A320 has a maximum takeoff weight $m$ around 78000kg and one of the two engines is used for taxing. The rolling resistance $f_r$ of each tire can be calculated using (1):

$$f_r = C_{rr} \cdot N_f.$$  \hspace{1cm} (1)

Where, $C_{rr}$ is the rolling resistance coefficient ranging from 0.010–0.015 on a concrete surface, and $N_f = m \cdot g$ ($g = 9.81m/s^2$) is the normal force. In this work, $C_{rr}$ is set to 0.015 which leads to $f_r = 11.478kN$. Hence, the total rolling resistance $F_r$ for the whole A320 with 3 tires is $3 \times f_r = 34.433kN$. With all the information, and if A320 is considered as a point mass, then given a speed profile one can work out the needed thrust using Newton’s laws of motion. It is worth noting that fuel consumption is directly related to the use of thrust and hence can be easily obtained. With the aid of A320 model, a multi-objective optimisation problem is formulated, as will be discussed in Section C, in order to search for the optimal taxiing trajectories for each edge $(v_i, v_j) \in E$.

C. Problem and Constraints

The problem is to find out, for each edge, a set of optimal speed profiles, i.e. taxiing trajectories, such that the following two objectives in (2) are minimised simultaneously:

$$\left\{ \begin{align*}
\int_0^T \left[ TR(t) \cdot \left(1 + a(t)\right) \right] \cdot dt. 
\end{align*} \right. \hspace{1cm} (2)
$$

where, $T$ is the taxi time for the edge under consideration when a certain speed profile is given; $TR(t)$ is the thrust required for the speed defined in the speed profile at time instant $t$; and $a(t)$ is the associated acceleration/deceleration rate at time instant $t$. The second objective is directly related to fuel consumption by considering the total use of thrust and penalising the large acceleration rate. The second objective serves as an index rather than a measure of actual fuel burn. The speed profile is subject to the following constraints as described in (3), (4) and (5):

$$Sp(t) \leq Sp_{max} = \begin{cases} 30 \text{ knots} & \text{straight taxiway} \\ 10 \text{ knots} & \text{turning} \end{cases} \hspace{1cm} (3)$$

$$Sp(t = 0) = Sp_{i}; Sp(t = T) = Sp_{j}; . \hspace{1cm} (4)$$

$$a(t) \leq a_{max}. \hspace{1cm} (5)$$

Where, $Sp(t)$ is the speed at time instant $t$ and $Sp_{max}$ is the maximum speed constraint with different values depending on whether it is an edge associated with a straight or turning taxiway segment; $Sp_i$ and $Sp_j$ are the start speed and end speed at two ends of the edge which are specified according to the characteristics of the vertices. For passenger comfort, the acceleration/deceleration rate should not exceed $a_{max}$ which is set to 0.1g in this work.

The optimization problem presented above represents a complex configuration due to the fact that the speed profile is continuous in time and thus contains an infinite number of decision variables. In order to keep the problem tractable, such a speed profile has to be discretized. Instead of direct discretization of the speed profile, in this work, each edge $(v_i, v_j)$ has been divided into four segments with different distances. These four distances correspond to four different aircraft taxiing phases, each of which represents a...
typical taxiing behavior as shown in Fig. 1. The first phase is the acceleration phase in which a constant acceleration rate \( a_1 \) is assigned so that \( Sp \) will be increased from \( Sp_e \) to a speed \( Sp_1 \) depending on the length of the first phase \( d_1 \). For the second phase, aircraft will traverse at the speed \( Sp_1 \) until the length of the second phase \( d_2 \) is reached. After the second phase, aircraft will decelerate across the third and the fourth phases from the speed \( Sp_1 \) to \( Sp_e \). The difference of the last two phases lies in the fact that in the fourth phase a maximum deceleration rate \( a_4 = a_{max} \) is applied so that the speed can be quickly reduced from \( Sp_3 \) to \( Sp_e \). While for the third phase, the deceleration rate \( a_3 \) will be uniquely determined by \( a_4 \) and \( d_4 \) since \( Sp_3 \) can be derived backwards given \( a_4 \), \( Sp_2 \) and \( d_4 \). And the length of the third phase \( d_3 = d(v_i, v_f) - d_1 - d_2 - d_4 \). Hence, there are four free variables, namely \([a_1, d_1, d_2, d_4]\), for each edge \((v_i, v_f)\), and by searching different values for these variables different speed profiles and their corresponding objective values (refer to (2)) will be uniquely defined over this edge.

![Figure 1](image)

**Figure 1.** Discretization of edge \((v_i, v_f)\) via four variables \([a_i, d_1, d_2, d_4]\).

In order to satisfy constraints introduced in (3) ~ (5), each variable in \([a_1, d_1, d_2, d_4]\) has its own bounds which can only be worked out sequentially in the order as follows:

1) The upper \( a_{1,u} \) and lower \( a_{1,l} \) bounds for \( a_1 \): \( a_{1,u} \) equals to \( a_{max} \), the smallest value that \( a_1 \) can be is when the whole edge \((v_i, v_f)\) is reduced into a single segment. In such a case, aircraft constantly accelerates with \( a_1 \) so that \( Sp_e \) can be reached at the end of the edge. Hence, \( a_{1,l} \) can be calculated using (6) as follows:

\[
a_{1,l} = \frac{Sp_e^2 - Sp_1^2}{2d(v_i, v_f)}. \tag{6}
\]

2) The upper \( d_{1,u} \) and lower \( d_{1,l} \) bounds for \( d_1 \): once \( a_1 \) has been fixed with a value within \([a_1, a_{1,u}, a_{1,l}]\), \( d_{1,l} \) will have to satisfy (7), otherwise the end speed of \((v_i, v_f)\) will never reach \( Sp_e \):

\[
d_{1,l} = \frac{Sp_e^2 - Sp_1^2}{2a_1}. \tag{7}
\]

For \( d_{1,u} \), it represents one of the extreme situations where aircraft can only accelerate within such a distance, beyond which the taxi speed \( Sp \) may violate the maximum allowable speed \( Sp_{max} \), or \((d(v_i, v_f) - d_{1,u})\) may not be long enough for aircraft to reduce their speeds from \( Sp_1 \) to \( Sp_e \) even with the maximum deceleration rate.

3) The upper \( d_{2,u} \) and lower \( d_{2,l} \) bounds for \( d_2 \): once \( d_1 \) has been fixed within the feasible bounds, \( Sp_1 \) is fixed and \( d_{2,u} \) and \( d_{2,l} \) can thus be deduced using (8) and (9):

\[
d_{2,u} = d(v_i, v_f) - d_1 - \frac{Sp_1^2 - Sp_e^2}{2a_{max}}. \tag{8}
\]

\[
d_{2,l} = d(v_i, v_f) - d_1 - \frac{Sp_1^2 - Sp_e^2}{2a_{min.d}}. \tag{9}
\]

Where \( a_{min.d} \) is defined in (10) and represents the situation where there is only one deceleration phase with a small deceleration rate \( a_{min.d} \) and aircraft has to decelerate earlier.

\[
a_{min.d} = \frac{Sp_1^2 - Sp_e^2}{2(d(v_i, v_f) - d_2)}. \tag{10}
\]

The upper bound of \( d_2 \) represents the situation where \( d_3 \) does not exist and aircraft has to decelerate with \( a_{max} \).

4) The upper \( d_{4,u} \) and lower \( d_{4,l} \) bounds for \( d_4 \): once \( d_2 \) has been fixed as a value within its feasible bounds \( d_{4,u} \) is calculated using (11), in which case \( d_3 \) does not exist:

\[
d_{4,u} = \frac{Sp_1^2 - Sp_e^2}{2a_{max}}. \tag{11}
\]

The lower bound \( d_{4,l} \) is zero.

As one can see from this discretization procedure, the complexity of the problem has been significantly reduced to only considering four variables for each edge. Once feasible values have been defined, a speed profile, such as the one shown in Fig. 1, will be uniquely determined. Then, using the aircraft model discussed in Section B the associated thrust and brake force can be obtained, and hence the fuel index in (2). The remaining task is to search for the optimal values of \([a_1, d_1, d_2, d_4]\) so that both objectives in (2) can be minimised simultaneously. Since there are more than one objectives to be dealt with at the same time and the bounds of the decision variables are interrelated, natural computation provides an ideal way to tackle it. In the next section, an immune inspired multi-objective optimisation algorithm [10-11] is first described and then utilised to find out the optimal taxiing trajectories.

III. A POPULATION ADAPTIVE IMMUNE ALGORITHM (PAIA)

PAIA is an immune inspired multi-objective optimization algorithm. Fig. 2 summarizes the pseudo-code involved in PAIA and demonstrates how it is adapted for planning the aircraft taxiing trajectories.

As one can see from Fig. 2, first, a random initial population pool is generated with different instantiations of \([a_1, d_1, d_2, d_4]\). The generation procedure is conducted in
exactly the same order as already discussed in Section D so that all values are randomly generated within the feasible regions. Then, one of the good solutions \( x_{\text{identified}} \) from the non-dominated set will be randomly selected in order to activate the rest of the solutions. The corresponding affinity \((\text{fitness})\) for each solution is calculated by calculating its distance to \((\text{fitness})\) for each solution is calculated by calculating its distance to \( (\text{fitness}) \) for each solution is calculated by calculating its distance to another solution.

The computation complexity of PAIA for the block of non-domination() is \( O(N^2) \), where \( N \) is the size of \( \text{pop} \) at each iteration step. For evaluating the objective functions, this complexity is governed by \( O(N_{\text{max}}) \) at each iteration step. It has been shown in [10-11] that \( N_{\text{max}} \) is not problem dependent and 95 is an empirically good number. Increase \( N_{\text{max}} \) does not lead to any improvement of the optimisation. Since \( \text{pop} \) and the clone size for each \( \text{pop}_s \) are adaptive with respect to the search process and only relatively good solutions are selected and maintained, PAIA can largely reduce the number of evaluation times comparing to traditional EAs and is not sensitive to the initial population size.

For the sake of reducing computation time, the fastest taxiing trajectory over the edge \((v_i, v_j)\) is first analytically derived, which corresponds to the situation where \( d_1 = d_4 = d_{\text{max}} \), \( d_2 = d_3 = 0 \). Such fastest taxiing trajectory represents one optimal solution in the Pareto sense and locates at one end of the Pareto front. When constructing the initial population, we intentionally seed such a fastest solution as one of the initial candidates so that the whole population is biased to the Pareto front from the very beginning, which is believed to speed up the search process.

In this section, the proposed optimisation methodology has been tested using a particular taxiing path extracted from Manchester Airport. Table 1 includes some basic information of the different edges along this particular taxiing path.

### Table I. Facts of Different Taxiways Along One Particular Taxiing Path of Manchester Airport

<table>
<thead>
<tr>
<th>Edge No.</th>
<th>Edge Description</th>
<th>( S_{p_a} ) (knot)</th>
<th>( S_{p_e} ) (knot)</th>
<th>( d(v_i, v_j) ) (m)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Straight T.W.(^a)</td>
<td>0</td>
<td>10</td>
<td>153.8</td>
</tr>
<tr>
<td>2</td>
<td>Turning T.W.</td>
<td>10</td>
<td>10</td>
<td>61.5</td>
</tr>
<tr>
<td>3</td>
<td>Straight T.W.</td>
<td>10</td>
<td>10</td>
<td>92.3</td>
</tr>
<tr>
<td>4</td>
<td>Turning T.W.</td>
<td>10</td>
<td>10</td>
<td>61.5</td>
</tr>
<tr>
<td>5</td>
<td>Straight T.W.</td>
<td>10</td>
<td>10</td>
<td>246.2</td>
</tr>
<tr>
<td>6</td>
<td>Turning T.W.</td>
<td>10</td>
<td>10</td>
<td>153.8</td>
</tr>
<tr>
<td>7</td>
<td>Straight T.W.(^b)</td>
<td>10</td>
<td>0</td>
<td>215.4</td>
</tr>
<tr>
<td>8</td>
<td>Straight T.W.</td>
<td>0</td>
<td>10</td>
<td>138.5</td>
</tr>
<tr>
<td>9</td>
<td>Turning T.W.</td>
<td>10</td>
<td>10</td>
<td>92.3</td>
</tr>
<tr>
<td>10</td>
<td>Runway Acc. H.(^c)</td>
<td>10</td>
<td>0</td>
<td>230.8</td>
</tr>
</tbody>
</table>

\( a. \) Taxiway; \( b. \) Taxiway waiting point; \( c. \) Runway access holding point.

For this particular taxiing route, an aircraft will taxi out with initial speed 0 on a straight taxiway ‘Edge #1’ and will enter ‘Edge #2’ with a speed of 10 knot. Since ‘Edge #2’ is a taxiway with turning, aircraft will keep its speed at 10 knot until entering ‘Edge #3’, which is a straight taxiway and a variety of taxiing trajectories are available. For ‘Edge #7’, it is a straight taxiway with waiting point at the end. Hence, the
speed will be reduced to 0. Similar situation happens for runway access point at the end of ‘Edge #10’.

PAIA is run with an initial population of 7. The number of generations is set to 100. Other user specified parameters are set according to [10-11]. Fig. 3 shows the obtained Pareto front which corresponds to the planned taxiing trajectories for 10 consecutive edges. The obtained trajectories are optimal in Pareto sense which is an important property since for each taxi time the corresponding fuel consumption is the least that one has to burn, and vice versa for a given fuel consumption the corresponding speed profile is the fastest that one can achieve.

As can be clearly seen from Fig. 3, a shorter total taxi time may not necessarily mean that more fuel consumption can be reduced. On the contrary, such efficiency often comes with the price of excessive use of acceleration/deceleration. Hence, detailed investigation about the relationship between fuel and taxi time, as the one shown in Fig. 3, is important before any conclusions can be reached for taxi planning. Two red circles at two ends of the Pareto front represent the fastest and the one with minimum fuel consumption. Their corresponding speed profiles and the thrust-brake force profiles are shown in Figs. 4 and 5.

V. CONCLUSIONS AND FUTURE RESEARCH DIRECTIONS
In this paper, an investigation into planning taxiing trajectories has been carried out using an immune inspired multi-objective optimization algorithm. A discretization procedure has been first carried out so that the complexity of a previously intractable problem has been significantly reduced. It is argued that the resulted optimal taxiing trajectories will be useful when they are integrated into the taxi routing and scheduling problem since, this time, not only speed but also corresponding fuel consumption are available to support the decision making process. The proposed method can be used off-line for such purpose so that a look-up table can be built which contains different trade-off trajectories for each edge on the airport.

ACKNOWLEDGMENT
The authors wish to express their gratitude to EPSRC for their financial support under Grant EP/H004424/1, and also invaluable discussions with Stefan Ravizza, Geert De Maere and Jason Atkin from the Dept. of Computer Science at the University of Nottingham.


