Abstract—Automation of airport operations can greatly improve ground movement efficiency. In this paper, we study the speed profile design problem for advanced airport ground movement control and guidance. The problem is constrained by the surface four-dimensional trajectory generated in taxi planning. A decomposed approach of two stages is presented to solve this problem efficiently. In the first stage, speeds are allocated at control points, which ensure smooth speed profiles can be found later. In the second stage, detailed speed profiles of each taxi interval are generated according to the allocated control point speeds with the objective of minimizing the overall fuel consumption. We present a swarm intelligence based algorithm for the first-stage problem and a discrete variable driven enumeration method for the second-stage problem, since it only has a small set of discrete variables. Experimental results demonstrate the presented methodology performs well on real world speed profile design problems.

Keywords—Airport ground movement, fuel consumption, particle swarm optimization, smoothness, speed profile design.

I. INTRODUCTION

The air traffic demand is expected to more than double in the near future [1]. Currently airport ground movement has been recognized as the bottleneck of air traffic flow, which is even worse at peak hours and in bad weather or low visibility conditions. One promising way to relieve this predicament relies on the utilization of modern technologies to improve the automation level of airport operations [2], [3]. As a result, the airport ground movement related problems have attracted lots of attention from researchers.

One of the fundamental problems in airport ground movement is taxi planning (TP) [4], which deals with the routing and/or scheduling of aircraft ground movement. To date, researches on TP mainly focus on conflict-free routes and schedules for taxiing aircraft. Due to different airport operational requirements, TP has been modeled with different settings and objectives [5]. For example, [6], [7] studied the scheduling of aircraft ground movement along fixed taxi routes, while [8], [9] presented models for combined optimization of routing and scheduling. On the other hand, precise four-dimensional guidance systems have been planned to guide aircraft ground movement in a highly automated manner in the future [10], [11], where detailed speed profiles should be provided for each aircraft. However, only a few works have been devoted to this topic [12]–[14], which mainly focused on the analysis of the trade-off between taxi time and fuel consumption and aimed at providing meaningful information for TP to produce more fuel-saving routes and schedules. For example, [12] dealt with generating Prato optimal speed profiles with respect to taxi time and fuel consumption for an unimpeded aircraft. Reference [13] extended the work of [12] by considering the interactions between aircraft, but it still mainly focused on the analysis of the trade-off between taxi time and fuel consumption, instead of providing an efficient algorithm to generate feasible speed profiles.

Based on the investigation of the characteristics desirable speed profiles possess, [14] presented a heuristic approach to improve the solution efficiency of the speed profile design problem. However, it again dealt with TP and the speed profile design problem in an integrated approach, which made it difficult to utilize existing results from TP. Therefore, we further study the speed profile design problem on the basis of TP results (i.e., taxi routes and schedules) in this paper. The presented methodology decomposes the speed profile design process into two stages. The first stage properly assigns speeds at control points of the routes so that speed profiles consistent with TP schedules can be generated, while the smoothness of the resulting speed profiles can also be ensured. The second stage finds a fuel-saving speed profile for every interval between consecutive control points, with speeds at the control points constrained by the assigned values in the first stage and taxi time constrained by TP results. Particle swarm optimization (PSO) techniques are used in the first stage to find promising control point speeds, and an enumeration approach is proposed for the second stage problem based on a discretization of the acceleration rate.

II. PROBLEM DESCRIPTION

The TP module provides for each aircraft a taxi route and the related schedules (i.e., the arrival time at every control point along the route). Consequently, the TP results for aircraft $A$ can be described by a sequence of space-time points $t_{A,i} = \left((p_{A,i}, t_{A,i,0}), ..., (p_{A,i}, t_{A,i,n})\right)$. Here, $p_{A,i}$ and $t_{A,i}$ denote the position and arrival time of control point $i$, respectively; $n$ denotes the number of control points. In this paper, we assume $t_{A,i}$ is generated by TP so that feasible speed profiles exist which meet the arrival time requirement at every control point; this requires TP to consider aircraft taxi time in a...
realistic way [15].

For a given trajectory \( tr_i \), we study the problem of designing detailed speed profiles which ensure eco-friendly movement and passenger comfort. In addition to the schedules specified by \( tr_i \), this problem is also constrained by several realistic restrictions including the unimpeded taxi speed \( v_{0i} \), the allowed taxi speed range \( \{v_{\min}, v_{\max}\} \), and the maximum acceleration rate \( a_{\max} \), which can vary with aircraft types and airport regulations.

### III. METHODOLOGY

In general, smooth speed profile design is difficult due to varying movement modes and complicated constraints and objectives. In this section, a two-stage methodology is presented to efficiently generate smooth speed profiles for a given taxi trajectory.

In the first stage, we focus on determining proper speed allocations at control points using a heuristic to ensure smoothness. To achieve computational efficiency, we utilize the PSO technique [16] to address this problem.

The second stage takes the result of the first stage as additional constraints and tries to figure out the most fuel-saving speed profiles for each taxi interval. This problem is more similar to the trajectory planning problem described in [12], [13], but with stricter constraints on the travel time in each taxiway part.

#### A. Speed Allocation at Control Points

The first-stage problem is crucial for generating smooth and fuel-saving speed profiles, since its output acts as an additional constraint for the second-stage problem.

1. **Heuristics for Smoothness**
   
   Let \( T_i \) and \( D_i \) be the taxi time and distance between control point \( i-1 \) and \( i (i=1,2,\ldots,n) \) of a given trajectory \( tr_i \). We refer to the corresponding movement process as moving in interval \( i \).
   
   The average speed of interval \( i \) is \( \bar{v}_i = D_i / T_i \). The difference \( \delta = v_{0i} - \bar{v}_i \) reflects the amount of waiting time in interval \( i \) due to interactions between nearby aircraft, where \( v_{0i} \) is the normal unimpeded taxi speed for interval \( i \). If \( \delta \) is close to zero, the movement in interval \( i \) is almost unimpeded, for which the best speed profile in this interval is to taxi with a constant speed \( v_{0i} \) close to \( v_{0i} \). Otherwise (if \( \delta \) is large), it would be reasonable for the speeds at the two ends of this interval to be much more different. Therefore, we can use \( \delta \) to penalize the deviation of control point speeds from the average speed, which leads to the following heuristic:

\[
    h_i = h_0 + \delta (v_{0i} - \bar{v}_i + |v_{0i} - \bar{v}_i|), \quad i=1,2,\ldots,n, \tag{1}
\]

where \( v_{0i} \) and \( v_{0i} \) represent the speed at the start and end position of interval \( i \) respectively; \( h_0 \) and \( h_2 \) are two positive constants.

To ensure movement smoothness, we prefer speed allocation schemes that minimize the sum of \( h_i \) over all the taxi intervals.

![Fig. 1 Upper bound of the traveling distance: (a) \( v_i \leq v_{\max} \), (b) \( v_i > v_{\max} \)](image)

2. **Constraints**

   For taxi interval \( i \), the values of \( v_{ei} \) and \( v_{si} \) should meet certain requirements to be feasible for aircraft to follow, which depend on the aircraft’s physical movement ability. The first class of constraints pertains to the acceleration ability: It should be possible for the aircraft to reach speed \( v_{ei} \) in the specified time window of taxi interval \( i \) starting from speed \( v_{si} \). This condition can be described as

\[
    -a_{\max} \leq (v_{ei} - v_{si}) / T_i \leq a_{\max} \tag{2}
\]

Then, to set up the constraints imposed by TP, we first consider two extreme cases of the movement in interval \( i \).

In the first case, the aircraft moves from the start position with speed \( v_{si} \) and the maximum acceleration at first, and then decelerates with the maximum deceleration rate until the end time of interval \( i \), reaching a speed equal to \( v_{ei} \). This relates to the longest distance the aircraft can travel within the specified time window of taxi interval \( i \). The second case is to the contrary, and relates to the shortest distance within the specified time window.

Illustrations of the two cases are given in Figs. 1 and 2. The crossover point of the acceleration and deceleration curves is denoted by \( v_{ci} \). The area of shaded region represents the distance traveled in each case. Here we use \( d_{ai} \) and \( d_{ci} \) to denote the upper and lower bound on the traveling distance, corresponding to the two extreme situations respectively. And \( v_{ci} \) and \( v_{ci} \) are used to replace \( v_{ei} \) and \( v_{si} \) for clarity. According to the scenarios in Figs. 1 and 2, we have...
and is a small positive constant used to denote the maximum and minimum constraint on the speed at the control points. Let \( r^i_1 \) and \( r^i_2 \) be two random numbers drawn from a uniform distribution over \((0,1)\); \( w \) is called inertia weight, which is introduced by [17] as a trade-off between the global and local search ability. A larger inertia weight is more appropriate for global search while a smaller value is more appropriate for local search [18].

To prevent premature convergence that often occurs in the canonical PSO algorithm, we utilize the randomization technique introduced in [19], which uses a random position to guide the flying of particles. When the difference between the maximum and minimum fitness value of particles in the swarm is small, there will be a large probability that a randomization operation upon the local or global best position takes place. To enhance the global search ability in the early stage of the optimization process and encourage particles to fly toward the global optima at the end of the search, this randomization happens for the global best position in the first half of iterations and for the local best position in the second half. The probability of randomization is calculated by

\[
P_r = \exp\left(-\left(f_{\text{max}} - f_{\text{min}}\right)/(f_{\text{max}} + \varepsilon)\right)
\]

where \( f_{\text{max}} \) and \( f_{\text{min}} \) denote the maximum and minimum fitness value respectively; \( \varepsilon \) is a small positive constant used to prevent division by zero.

In this paper, the fitness value of the PSO-based algorithm is defined by the multiplication of the value of the objective function (6) with a penalty term related to constraint violations. To make violations of different kinds of constraints comparable, all of them are normalized before adding to the penalty term. Let \( e_i \) denote the penalty of particle \( i \), then its fitness value is defined as

\[
\text{fit}_i = O_{i}\left(1 + e_i\right)
\]

where \( O_{i}\) is the value of (6) with respect to particle \( i \).

### B. Interval Speed Profile Optimization

This section describes the formulation and solution for the second stage problem.

1. Problem Simplification

Here, we consider a typical movement process between two control points consisting of an acceleration phase, a constant speed phase, and finally a deceleration phase to reduce the speed to the specified value of the end control point (see Fig. 3), or vice versa.

Then speed profile design becomes a problem of finding out a speed profile with three phases for every taxi interval of the given trajectory. Notice that given the speed allocation, this
problem always has feasible solutions. And we shall find the optimal one with the least fuel consumption and the highest riding comfort.

2. Formulation

According to [12]–[14], [20], fuel consumption is closely related to the engine thrust level during taxiing, which can be determined by Newton’s second law:

\[ F_i = F_r + ma \]  

(12)

where \( F_i \) is the thrust, \( F_r \) is the rolling resistance, \( a \) is the acceleration, and \( m \) is the weight of the aircraft. The rolling resistance is defined as

\[ F_r = \mu mg \]  

(13)

where \( \mu \) is the rolling resistance coefficient on a concrete surface, and \( g = 9.81 \text{m/s}^2 \) is the acceleration of gravity.

According to the simplification made above, the fuel consumption index for taxi interval \( i \) can be defined as

\[ U_i = \sum_{k=1}^{3} F_{ak} (1 + |a_k|) W_{ki} \]  

(14)

where \( F_{ak} \) and \( a_k \) represent the thrust and acceleration for phase \( k \) of taxi interval \( i \) respectively, and \( W_{ki} \) represents the duration of phase \( k \). Here, the term \( |a_k| \) is used to penalize large acceleration rate for the sake of passenger comfort [12]. Notice that \( a_{3i} \) always equals to zero due to the simplification, so the second stage problem can be formulated as follows:

\[
\text{Min } \sum_{i=1}^{n} U_i \quad (15)
\]

subject to:

\[
\begin{align*}
& v_{i-1} W_i + a_{i} W_{3i}^2 / 2 + (v_{i} + v_{i}) W_{2i}^2 / 2 \\
& + v_i W_{1i} - a_{i} W_{1i}^2 / 2 = D_i \\
& W_{1i} + W_{2i} + W_{3i} = T_i \\
& v_{i} = v_{i}, i = 1, \ldots, n, (16)
\end{align*}
\]

where the variables are \( a_{i}, a_{i}, W_{1i}, W_{2i}, \) and \( W_{3i} \), and we have \( v_{i} = v_{i-1} + a_{i} W_{3i}, v_{i} = v_{i} - a_{i} W_{1i} \). In the last constraint, \( K \) is a finite set containing all the predefined feasible acceleration rates for the aircraft. Such a discretization of the acceleration can facilitate aircraft ground movement control in real-world situations.

3. Solution Approach

The above formulation can be separated by taxi intervals. And only two of the five variables are independent variables, which we choose as \( a_{i} \) and \( a_{j} \). In the following, we drop the subscript \( i \) in \( a_{i}, a_{j}, W_{1i}, W_{2i}, \) and \( W_{3i} \) for clarity.

Since it is required that \( a_{i}, a_{j} \in K \), where \( K \) is a finite set with only a few elements, it is reasonable to use an enumeration approach to find out the best combination of \( a_{i} \) and \( a_{j} \) relative to (15) and (16). We can examine four different cases of movement within taxi interval \( i \) to get the best solution:

Case 1. \( a_{i}, a_{j} \neq 0 \). In this case, \( W_{1i}, W_{2i}, \) and \( W_{3i} \) can be calculated with \( a_{i}, a_{j} \) and \( v_{i} \) as

\[
W_{1i} = (v_{i} - v_{i-1}) / a_{i} \\
W_{2i} = (v_{i} - v_{i}) / a_{j} \\
W_{3i} = T_i - (v_{i} - v_{i}) / a_{i} - (v_{i} - v_{i}) / a_{j}
\]

(17)

(18)

(19)

then, according to the first constraint of (16) we have

\[
k_{2} v_{i}^{2} + k_{1} v_{i} + k_{0} = 0
\]

(20)

where

\[
k_{2} = a_{i} - a_{j} \\
k_{1} = 2(a_{i} a_{j} T_{i} + a_{i} v_{i} - a_{j} v_{j}) \\
k_{0} = a_{i} v_{i}^{2} - a_{i} v_{i}^{2} = 2a_{i} a_{j} D_{i}
\]

(21)

(22)

(23)

To determine whether a combination of \( a_{i} \) and \( a_{j} \) can produce a feasible solution, we first check if there is a root of (20) satisfying

\[
v_{\min} \leq v_{i} \leq v_{\max}
\]

(24)

If (24) is true, we further examine whether \( W_{1i}, W_{2i}, \) and \( W_{3i} \) meet

\[
0 \leq W_{1i}, W_{2i}, W_{3i} \leq T_{i}
\]

(25)

Case 2. \( a_{i} = 0, a_{j} \neq 0 \). In this case, aircraft will move without
the first phase, so \( W_1 = 0 \). Then it’s clear that \( a_3, W_3 \) and \( W_2 \) can be determined by
\[
a_3 = (v_i - v_{i-1})^2 / (2(D_i - v_{i-1}T_i))
\]
(26)
\[
W_3 = (v_i - v_{i-1}) / a_3
\]
(27)
\[
W_2 = T_i - W_3
\]
(28)

Case 3. \( a_i \neq 0, a_3 = 0 \). In this case, aircraft will move without the third phase, so \( W_3 = 0 \). This is similar to the above case. We have
\[
a_i = (v_i - v_{i-1})^2 / (2v_iT_i - D_i)
\]
(29)
\[
W_i = (v_i - v_{i-1}) / a_i
\]
(30)
\[
W_2 = T_i - W_i
\]
(31)

Case 4. \( a_i = a_3 = 0 \). In this case, aircraft will move at a constant speed during the entire taxi interval. The related solution would be feasible if
\[
T_i v_{i-1} = D_i
\]
(32)
\[
v_{i-1} = v_i
\]
(33)
and the values of the remaining variables are
\[
W_1 = W_2 = 0 \quad , \quad W_3 = T_i
\]
(34)

For each taxi interval \( i \), we find all the feasible solutions in the four cases, and choose the speed profiles with the smallest \( U_i \).

IV. EXPERIMENTAL RESULTS

In this section, we use experiments to investigate the performance of the presented methodology. TP results for speed profile design are drawn from a taxi-planning module developed for Nanjing Lukou International Airport, including taxi routes and schedules for 50 aircraft. The unimpeded taxi speed is set to 10m/s, while the maximum and minimum taxi speeds are set to 15m/s and zero respectively. The maximum acceleration rate is simply set to 0.1g for experimental purpose, regardless of the type of the aircraft. The parameters for the PSO-based speed allocation algorithm are set as \( c_1 = c_2 = 2.05 \). The algorithms are implemented using MATLAB on a PC with 2.0GHz CPU and 3GB RAM.

A. Comparison of Different Heuristics

In the proposed method, speeds allocated at the control points act as constraints for speed profile optimization, which eventually determines the quality of the generated speed profiles along the entire taxi route. Therefore, it is required that the smoothness heuristic function is also promising for reducing the fuel consumption. To validate the utilized heuristic function, i.e., (1), we compare it with the following competitive alternatives:

\[
h_i^{(2)} = |v_{i+1} - v_i| + |v_{i+1} - v_i|, \quad i=1,2,\ldots,n
\]
(35)
\[
h_i^{(3)} = h_i / (h_i + \delta), \quad h^{(3)}_i \text{ modifies the way to quantify the deviation from the average speed, and } h^{(4)}_i \text{ combines the changes of } h^{(2)}_i \text{ and } h^{(3)}_i \text{ together.}
\]
\[
h_i^{(3)} = \left(\frac{v_{i} + v_{i+1}}{2} - \bar{v}\right), \quad i=1,2,\ldots,n
\]
(36)
\[
h_i^{(4)} = \left(\frac{v_{i+1} - v_i}{2 - \bar{v}}\right), \quad i=1,2,\ldots,n
\]
(37)

where \( h_i^{(2)} \) drops the factor \( h_i / (h_i + \delta) \), \( h^{(3)}_i \) modifies the way to quantify the deviation from the average speed, and \( h^{(4)}_i \) combines the changes of \( h^{(2)}_i \) and \( h^{(3)}_i \) together.

Results of fuel consumption index using the four heuristic functions are illustrated in Fig. 4, where Heur1 corresponds to (1), and Heur2 ~ Heur4 are related to (35)~(37) respectively. According to Fig. 4, it’s clear that Heur3 and Heur4 perform worse than the other two heuristics. The algorithm even failed to find a feasible solution for Aircraft 15 in the case of Heur4. On the other hand, results for Heur1 and Heur2 are close; however, the former results in less fuel consumption in general, especially for Aircraft 33, where the advantage of Heur1 is remarkable.

B. Computational Performance

We test the computational performance of the presented methodology by investigating the average solution time for each aircraft. The problem is solved five times for each aircraft, and the average solution time is calculated using only the feasible solution related values. The results are demonstrated in Fig. 5. As we can see, our two-stage approach runs very fast, needing only about 1.6s to solve a problem instance, and the solution times for different aircraft scatter in a very small range.

Fig. 6 shows the scalability of the two-stage approach, where the taxi steps for each aircraft are scaled up by 2~5 times. The horizontal axis represents the scale of the input data, while the vertical axis represents the average solution time for an aircraft computed over all the 50 instances. The results indicate that the solution time increases almost linearly with the number of taxi steps within the investigated range.

We also investigate the effectiveness of the presented approach. Now each problem is solved ten times, and the resulting feasible solution counts are plotted in Fig. 7. It can be observed that in spite of the random nature of the PSO-based speed allocation algorithm, the presented method can find out feasible solutions with only a few trials.
advanced airport ground movement guidance and control.

ACKNOWLEDGMENT

This research has been supported by the Technology Project of Civil Aviation Administration of China (No. MHRD201124), National Natural Science Foundation of China (No. 61203170), China Postdoctoral Science Foundation funded project (No. 2013T60539) and Innovation Found of Jiangsu Province (No. KYLX_0291).

REFERENCES