A Mixed Integer Linear Program for Solving a Multiple Route Taxi Scheduling Problem

Justin Montoya *

NASA Ames Research Center, Moffett Field, CA, 94035

Zachary Wood†

Univ. of Calif., Santa Cruz, Moffett Field, CA, 94035

Sivakumar Rathinam‡

Texas A&M University, College Station, TX, 77843

Waqar Malik§

Univ. of Calif., Santa Cruz, Moffett Field, CA, 94035

Aircraft movements on taxiways at busy airports often create bottlenecks. This paper introduces a mixed integer linear program to solve a Multiple Route Aircraft Taxi Scheduling Problem. The outputs of the model are in the form of optimal taxi schedules, which include routing decisions for taxiing aircraft. The model extends an existing single route formulation to include routing decisions. An efficient comparison framework compares the multi-route formulation and the single route formulation. The multi-route model is exercised for east side airport surface traffic at Dallas/Fort Worth International Airport to determine if any arrival taxi time savings can be achieved by allowing arrivals to have two taxi routes: a route that crosses an active departure runway and a perimeter route that avoids the crossing. Results indicate that the multi-route formulation yields reduced arrival taxi times over the single route formulation *only* when a perimeter taxiway is used. In conditions where the departure aircraft are given an optimal and fixed takeoff sequence, accumulative arrival taxi time savings in the multi-route formulation can be as high as 3.6 hours more than the single route formulation. If the departure sequence is not optimal, the multi-route formulation results in less taxi time savings made over the single route formulation, but the average arrival taxi time is significantly decreased.

Nomenclature

(u, v)	An edge from node u to node v.
D	The set of departure aircraft.
A	The set of arrival aircraft.
P	The set of all aircraft.
R_i	The set of all routes for aircraft i.
N_i^p	A set of ordered nodes denoting aircraft i's p^{th} route.
E_i^p	A set of ordered edges denoting aircraft i's p^{th} route.
f_{ik}^p	The kth node in aircraft i's p^{th} route.
o_i	The origin or activation node for aircraft i.
d_i	The destination or final node for aircraft i.

*Aerospace Engineer, NASA Ames Research Center,Moffett Field, CA 94035.

†Software Engineer, University of California, Santa Cruz, Moffett Field, CA 94035. AIAA Member.

‡Assistant Professor,Department of Mechanical Engineering, Texas A & M University, College Station, TX, 77843. AIAA Member.

§Aerospace Engineer, University of California, Santa Cruz, NASA Ames Research Center, Moffett Field, CA 94035.

Υ_i	The set of all nodes aircraft i can visit (e.g., $\Upsilon_i = \bigcup_p N_i^p$).
Λ_i	The set of all edges aircraft i can visit (e.g., $\Lambda_i = \bigcup_p E_i^p$).
Z_{iju}	A binary variable that is equal to 1 if aircraft i arrives at node u before aircraft j and is 0 otherwise.
Γ_{ir}	A binary variable that is equal to 1 if aircraft i uses route r and is 0 otherwise.
ρ_{ij}	A binary variable that is equal to 1 if aircraft i arrives at a runway before aircraft j and is 0 otherwise.
t_{iu}	A continuous variable that represents the time that aircraft i arrives at node u.
Sep	The minimum spatial separation between two aircraft on the taxiway.
l_{uv}	The length of edge (u, v).
ETD_i	The estimated touchdown time for arrival aircraft i.
PBT_i	The earliest push-back time for departure aircraft i.
V_{ij}	The minimum time separation between leading departure aircraft i and trailing departure aircraft j due to wa
e_l	The maximum number of aircraft allowed on runway exit edge l.
T_{ij}^l	The time aircraft i should leave runway exit edge l due to incoming aircraft j.
T_{dep}^i	The average time departure aircraft i requires to depart.
S_{max}^{uv}	The maximum velocity an aircraft can travel on edge (u, v).
S_{min}^{uv}	The minimum velocity an aircraft can travel on edge (u, v).
M	An arbitrarily large positive real number.

I. Introduction

Departure and arrival ground movements on taxiways at busy airports often create bottlenecks [1][2], and surface delays can result in large operating costs [3]. Often, aircraft wait in long queues to depart due to airport capacity limitations or non-optimal surface planning. Currently, ground and local controllers tend to make aircraft sequence decisions on a first-come-first-served basis or with simple heuristics based on local information [5]. While first-come-first-served is a good strategy for providing fair schedules, it guarantees neither maximum throughput nor minimum taxi times. The fundamental problem that this paper addresses is to find optimal taxi schedules for aircraft on the airport surface for a given time horizon.

Various mixed integer linear and integer linear models have been developed to generate optimal taxi schedules. The mixed integer linear programs in [6] and [7] yield good taxi solutions, but neither of them include routing decisions for aircraft. Moreover, [6] has been shown to have a large number of unnecessary decision variables [7]. Integer programs in [8] and [9] have successfully incorporated multiple route decisions for taxiing aircraft, but both have neglected aircraft type. By neglecting aircraft type both models cannot allow for a precise separation between taxiing aircraft resulting in loss of optimality. Also, the computation times of models [9] and [8] are directly proportional to the fidelity of the model.

No optimization model has effectively incorporated alternative route decisions for aircraft, while simultaneously enabling precise aircraft separation. The multi-route formulation described in this paper extends the model presented in [7]. The multi-route model incorporates additional decision variables to effectively model multiple route decisions. Since the model presented in this paper is an adaptation of [7], the authors have included aircraft type along with all other required safety constraints.

At the Dallas/Fort Worth Airport, arrival aircraft often taxi directly across active departure runways to get to their assigned gates while simultaneously preventing departure aircraft from taking-off [9]. The multi-route model can effectively determine when arrivals should use the perimeter taxiway to decrease their taxi-in time. While the concept of using perimeter taxiways [10] has been studied for many airports, the multi-route model has been implemented for an airport modeled after Dallas/Fort Worth Airport.

The paper is organized as follows. In Section II, currently accepted optimization models [9], [8], [7], and [6] are reviewed. The optimization model and notation are introduced in Section III. A comparison framework and results are presented in Section IV.

II. Problem Setup and Background

This paper introduces a mixed integer linear program for solving a Multiple Route Aircraft Taxi Scheduling Problem (MRATSP). The problem is described as follows. Given a *set* of taxi routes for all departing and arriving aircraft, schedule all aircraft along their chosen routes so that the time for all aircraft to finish using the taxiway and runway is minimized subject to the following constraints:

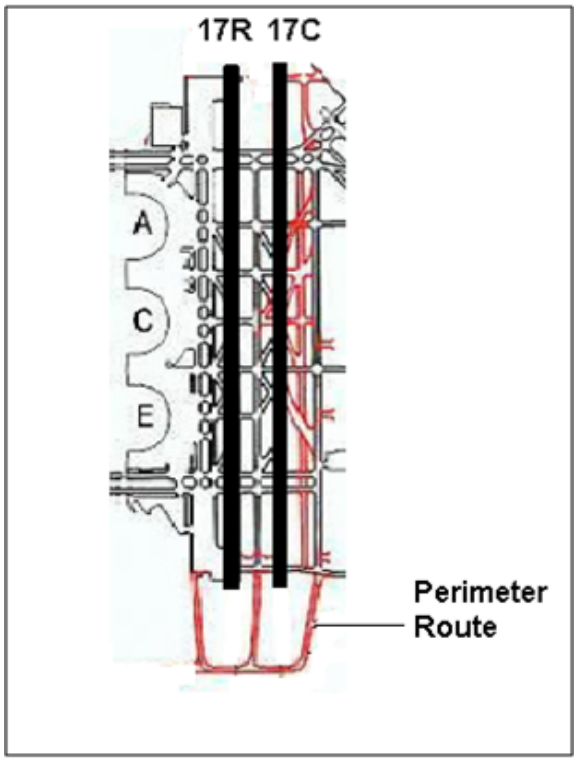


Figure 1: Runway and taxiway layout on the east side of Dallas/Fort Worth International Airport.

1. *Safety Constraints*: Each taxiing aircraft must be separated by a minimum distance from any other aircraft. Any two aircraft traversing in the same direction on the same taxiway segment can not overtake each other. Similarly, any two aircraft which will traverse in opposite directions on a bidirectional taxiway segment can not simultaneously occupy the taxiway segment (no head-on collisions).

2. *Runway Constraints*: Departing aircraft off the same runway must be separated by the minimum wake vortex separation while departing in sequence. Arrivals which occupy a departure runway prevent departures from using the runway.

3. *Speed Constraints*: All taxiing aircraft are subject to not exceed maximum speed. A minimum taxiing speed can also be considered.

The problem of optimally scheduling aircraft along a given single route subject to the above constraints is known as the Aircraft Taxi Scheduling Problem (ATSP) [6],[7],[8] and [9]. An ATSP is identical to a MRATSP with the exception of multiple route selection.

Although there are no known formulations which use both continuous and discrete variables for solving a MRATSP, there exist formulations for solving an ATSP. These formulations are known as mixed integer linear programs (MILP). By utilizing a finite number of continuous time variables, MILP formulations can be formulated with infinite fidelity for a fixed number of spatial locations. Because of the high fidelity that MILP formulations have, they can account for various aircraft types and include precise aircraft separation on the taxiway and runway. [7].

Integer programs (IP), which only use discrete variables, have been formulated for solving a MRATSP. In the literature, [8] and [9] are IP for solving a MRATSP, where time and space is partitioned into blocks of uniform or non-unform size. A typical construction is to allow a set of time slots for an aircraft at several chosen locations on the airport surface, thereby modeling aircraft movements. The integer program modeled in [8] decomposes the airport into discrete taxiway segments and ensures each taxiway segment has single occupancy.

There are limitations of the IP formulations in [8] and [9]. Increasing fidelity of these formulations can be problematic since the computational complexity of the problem is highly dependent on the number of time slots (e.g., there is a strong tradeoff between optimality and fidelity). Without good fidelity though, considerable taxi space can be wasted. The model in [8] decides aircraft taxi separation based on the largest aircraft size. Another issue is that these IP formulations do not consider the runway scheduling problem as part of their model. Without optimal sequencing between departures and arrivals at runways there can be unnecessary delay to aircraft on the surface.

Table 1 summarizes for each model, the formulation type (IP or MILP), advantages, and limitations. The table is not meant to be exhaustive as there exists other formulations with similar characteristics.

Table 1: Summary of Existing Mixed Integer Linear Programs and Integer Programs

Model	IP or MILP	Problem Scope	Advantages	Limitations
[6]	MILP	Taxi scheduler	Includes aircraft type	Missing runway constraints, necessary safety constraints, and routing decisions.
[7]	MILP	Taxi and runway scheduler	Includes aircraft type and models all safety constraints	No route choice
[8]	IP	Taxi scheduler	Includes routing decisions	Coarse estimate of aircraft separation. Missing runway constraints.
[9]	IP	Taxi scheduler	Includes routing decisions	Coarse estimate of aircraft separation. Missing runway constraints.

To encapsulate multi-route decisions, high temporal fidelity, and runway scheduling, a mixed integer linear program is built on the existing decisions variables presented in [7] with additional architecture to support multi-route decisions. The rest of this paper is dedicated to describing the optimization model as well as showing the benefits of using multiple routes at Dallas/Fort Worth International Airport.

III. Optimization Model

In the following section we provide a mixed integer linear programming for the Multiple Route Taxi Scheduling Problem. Section B illustrates each of the decisions variables and Section C describes how those decision variables are used to build the model. Some important definitions are provided below to aide in understanding the optimization model. All other definitions can be referenced from the nomenclature.

A. Definitions

Given a taxi layout of an airport, each taxi intersection on the layout is represented by a node. Each taxiway joining any two intersections is represented by an edge. Let $G = (V, E)$ be a directed graph with V denoting the set of all nodes and E denoting the set of all edges. Each edge, $(v_1, v_2) \in E$ joining nodes v_1 and v_2, is directed and indicates the direction an aircraft can travel along that edge. When a taxiway is bi-directional (e.g., aircraft can travel in both directions along the taxiway), two directed edges are used. Let D be the set of all departing aircraft, and A be the set of all arriving aircraft. Then all aircraft are elements of the set P (e.g., $A \cup D = P$).

An aircraft is allowed to choose any route from a set of available routes. Let n_i be the total number of routes specified for aircraft i and let these routes be denoted by $R_i = \{N_i^1, N_i^2, ..., N_i^{n_i}\}$ where N_i^p is the pth route for aircraft i. The route N_i^p, is denoted by an ordered set of nodes, $\{o_i, f_{i1}^p, f_{i2}^p, ..., f_{im_{ip}}^p, d_i\}$ where $o_i, f_{i1}^p, f_{i2}^p, ..., f_{im_{ip}}^p, d_i \in V$. Here, the nodes $o_i \in V$ and $d_i \in V$ are the nodes that respectively denote the origin (activation) and destination (termination) of aircraft i. If aircraft i chooses to follow its pth route, then i will first travel along the taxiway joining origin o_i to node f_{i1}^p, and then from f_{i1}^p to f_{i2}^p, and so on until it reaches its destination d_i. Let the set of all nodes present in all the routes of aircraft i be denoted by

Υ_i, i.e., $\Upsilon_i = \bigcup_k N_i^k$ for all $i \in P$. Let the set of all edges present in all the routes of aircraft i be denoted by Λ_i, i.e., $\Lambda_i = \bigcup_k E_i^k$ for all $i \in P$.

B. Decision Variables

Mixed Integer Linear Programs are built by using decision variables to construct logical and meaningful linear constraints. To build the model, we use sequencing variables Z_{iju}, timing variables t_{iu}, runway variables ρ_{ij}, and routing variables Γ_{ir}.

Z_{iju} is adapted from [6]. Z_{iju} is a binary variable which is equal to 1 if aircraft i arrives at node u before aircraft j. If Z_{iju} is equal to 0 aircraft i does not arrive to node u before aircraft j or that either i or j did not use a route with node u. These variables will be used to model most taxiway decisions.

Furthermore, t_{iu} is a continuous timing variable which represents the time that an aircraft i arrives at u (a node in its route). These variables will serve as the primary outputs for this problem.

ρ_{ij} is a sequencing variable that is particular to runway events. The logic is as follows: ρ_{ij} is equal to 1 if aircraft i arrives at a shared runway before j, otherwise it is equal to 0. This variable is used to build many constraints around the runway, such as wake vortex separation and runway incursions. Although it is possible to model all runway constraints with Z_{iju}, it is more convenient to use ρ_{ij} without adding complexity to the problem.

Finally, the last decision variable is Γ_{ir}. Γ_{ir} is equal to 1 if aircraft i uses route r and is equal to 0 otherwise. This variable is used to incorporate multiple route decisions for taxiing aircraft.

C. Constraints

In the following section the constraints of the optimization mode are describedl. Frist, the basic components of the formulation such as domain constraints are described,followed by the more complicated constraints such as aircraft separation and runway constraints.

The following constraints constrain the variables to their respective domains:

$$Z_{iju} \in \{0,1\} \qquad \forall i,j \in P, \forall u \in \Upsilon_i \tag{1}$$
$$\Gamma_{ir} \in \{0,1\} \qquad \forall i \in P, \forall r \in R_i \tag{2}$$
$$\rho_{ij} \in \{0,1\} \qquad \forall i,j \in P \tag{3}$$
$$t_{iu} \in \Re^+ \qquad \forall i \in P, \forall u \in \Upsilon_i \tag{4}$$
$$\tag{5}$$

The following constraint ensures that any aircraft is permitted to take only one route.

$$\sum_r \Gamma_{ir} = 1 \qquad \forall i \in P \tag{6}$$

For all $i,j \in P$ such that $i \neq j$, for all $u \in \Upsilon_i \cap \Upsilon_j$,

$$Z_{iju} \leq \sum_{r:u \in N_i^r} \Gamma_{ir} \tag{7}$$
$$Z_{iju} \leq \sum_{r:u \in N_j^r} \Gamma_{jr} \tag{8}$$

Equations (7) and (8) imply that if either aircraft i or j do not use a route with u in it, then no sequencing variable Z_{iju} can be equal to 1. These constraints complete the logic behind decision variable Z_{iju}.

1. *Sequencing Constraints*

For all $i,j \in P$ such that $u \in \Upsilon_i \cap \Upsilon_j$,

$$Z_{iju} + Z_{jiu} \leq 3 - (\sum_{r:u \in N_i^r} \Gamma_{ir} + \sum_{r:u \in N_j^r} \Gamma_{jr}) \tag{9}$$

$$Z_{iju} + Z_{jiu} \geq 2(\sum_{r:u \in N_i^r} \Gamma_{ir} + \sum_{r:u \in N_j^r} \Gamma_{jr}) - 3 \tag{10}$$

Equations (9) and (10) are used jointly to imply sequence consistency. Together, (9) and (10) express that if both aircraft i and j use a route with node u, then only one can come first through that node.

2. *Overtaking and Head-on Constraints*

For all $i, j \in P$ such that $(u, v) \in \Lambda_i \cap \Lambda_j$,

$$Z_{iju} - Z_{ijv} \leq 2 - (\sum_{r:(u,v) \in E_i^r} \Gamma_{ir} + \sum_{r:(u,v) \in E_j^r} \Gamma_{jr}) \tag{11}$$

$$Z_{iju} - Z_{ijv} \geq (\sum_{r:(u,v) \in E_i^r} \Gamma_{ir} + \sum_{r:(u,v) \in E_j^r} \Gamma_{jr}) - 2 \tag{12}$$

Equations (11) and (12) are used to make sure that the no overtaking can occur. Any two aircraft which share the same edge and are traveling in the same direction must not pass each other.

For all $i, j \in P$, for all $(u, v) \in \Lambda_i$, and for all $(v, u) \in \Lambda_j$,

$$Z_{iju} - Z_{ijv} \leq 2 - (\sum_{r:(u,v) \in E_i^r} \Gamma_{ir} + \sum_{r:(v,u) \in E_j^r} \Gamma_{jr}) \tag{13}$$

$$Z_{iju} + Z_{ijv} \geq (\sum_{r:(u,v) \in E_i^r} \Gamma_{ir} + \sum_{r:(v,u) \in E_j^r} \Gamma_{jr}) - 2 \tag{14}$$

Similarly, constraints (13) and (14) are required to keep aircraft from colliding head-on with each other. Pictorially, every edge can only be used in one direction (Fig. 2).

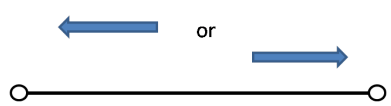


Figure 2: A edge can only be used in one direction, in which overtaking of aircraft is not permitted.

3. *Release Constraints*

Departing aircraft i can not be scheduled any earlier than its earliest available push-back time, PBT_i. Likewise, an arrival aircraft j can not be scheduled any earlier than its estimated touch-down time, ETD_i.

For all $j \in A$,

$$t_{jo_j} \geq ETD_j \tag{15}$$

For all $i \in D$,

$$t_{io_i} \geq PBT_i \tag{16}$$

4. *Speed Constraints*

For all $i \in P$ and for all edges $(u, v) \in \Lambda_i$,

$$\sum_{r:(u,v) \in E_i^r} \Gamma_{ir}(t_{iv} - t_{iu}) \leq \frac{l_{uv}}{S_{max}^{uv}} \tag{17}$$

$$\sum_{r:(u,v) \in E_i^r} \Gamma_{ir}(t_{iv} - t_{iu}) \geq \frac{l_{uv}}{S_{min}^{uv}} \tag{18}$$

Equations (17) and (18) set limits on aircraft taxi speed. These constraints are highly nonlinear and non-convex and are not able to be solved with a guarantee on optimality. A linearization, however, is possible by using a well known technique. Constraints are linearized by choosing a suitably large positive value M. The linearizations of the constraints are expressed as follows.

For all $i \in P$ and for all edges $(u, v) \in \Lambda_i$,

$$(t_{iv} - t_{iu}) \leq \frac{l_{uv}}{S^{uv}_{max}}(M - M \sum_{r:(u,v)\in E^r_i} \Gamma_{ir} + \sum_{r:(u,v)\in E^r_i} \Gamma_{ir}) \tag{19}$$

$$(t_{iv} - t_{iu}) \geq \frac{l_{uv}}{S^{uv}_{min}}(M \sum_{r:(u,v)\in E^r_i} \Gamma_{ir} - M + \sum_{r:(u,v)\in E^r_i} \Gamma_{ir}) \tag{20}$$

5. *Separation Constraints*

Separation between aircraft is provided at intersecting nodes and are formulated below. The separation constraints will also depend on the selection of route (e.g., Γ). Two distinct separation constraints have been identified in [7] and are discussed for clarification.

The first separation situation occurs when aircraft i is leading aircraft j and i just leaves node u. Then j cannot reach node u until i has gone far enough so that the minimum separation distance Sep is maintained. Situation one is shown in Fig. 3a. This separation depends on the speed of i on (u, v) and the speed of j on (w, v).

The second separation situation occurs when the approach speed of aircraft j is considered. Aircraft j must not enter (w, v) at a speed which could possibly violate separation parameter Sep. Situation two is shown in Fig. 3b.

For all $i, j \in P$ such that $i \neq j$ and for all $u \in \Upsilon_i \cap \Upsilon_j$ where $(u, v) \in \Lambda_i$,

$$(Z_{iju})(\sum_{r:(u,v)\in E^r_i} \Gamma_{ir})(\sum_{r:u\in N^r_j} \Gamma_{jr})(\ell_1 - Sep) \geq 0 \tag{21}$$

For all $i, j \in P$ such that $i \neq j$ and for all $v \in \Upsilon_i \cap \Upsilon_j$ where $(w, v) \in \Lambda_j$,

$$(Z_{ijv})(\sum_{r:(w,v)\in E^r_j} \Gamma_{jr})(\sum_{r:v\in N^r_j} \Gamma_{ir})(\ell_2 - Sep) \geq 0 \tag{22}$$

These constraints will not work in a linear programming environment since they are nonlinear. Instead, an application of the technique used in (17) and (18) is provided. Now both separation equations read:

For all $i, j \in P$ such that $i \neq j$ and for all $u \in \Upsilon_i \cap \Upsilon_j$ where $(u, v) \in \Lambda_i$,

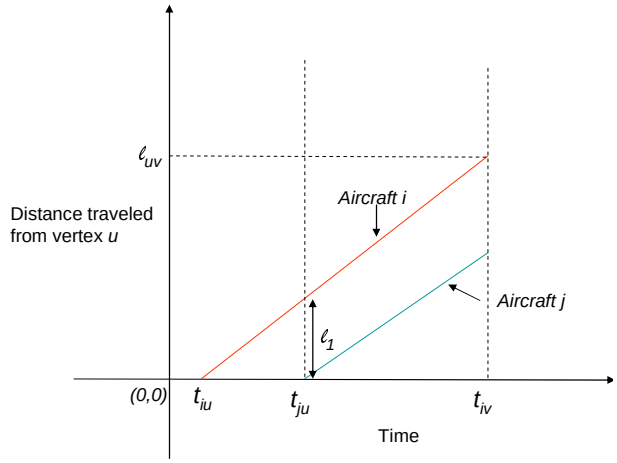
$$t_{ju} - t_{iu} - (t_{iv} - t_{iu})\frac{Sep}{\ell_{uv}} \geq -[3 - (Z_{iju} + \sum_{r:(u,v)\in E^r_i} \Gamma_{ir} + \sum_{r:u\in N^r_j} \Gamma_{jr})]M \tag{23}$$

For all $i, j \in P$ such that $i \neq j$ and for all $v \in \Upsilon_i \cap \Upsilon_j$ where $(w, v) \in \Lambda_j$,

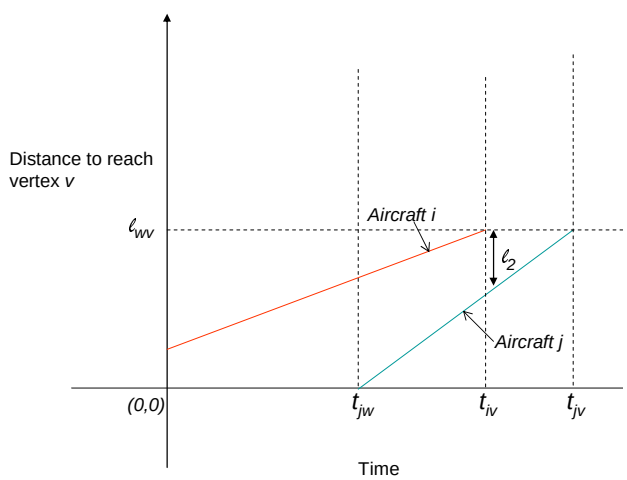
$$t_{iv} - t_{jv} - (t_{jv} - t_{jw})\frac{Sep}{\ell_{wu}} \geq -[3 - (Z_{jiv} + \sum_{r:(w,v)\in E^r_j} \Gamma_{jr} + \sum_{r:v\in N^r_i} \Gamma_{ir})]M \tag{24}$$

$$\ell_1 = \frac{t_{ju} - t_{iu}}{t_{iv} - t_{iu}}\ell_{uv} \tag{25}$$

$$\ell_2 = \frac{t_{jv} - t_{iv}}{t_{jv} - t_{jw}}\ell_{wv} \tag{26}$$



(a) Separation Situation 1



(b) Separation Situation 2

Figure 3: Space-time diagrams to calculate aircraft separation at nodes.

6. *Runway Occupancy Constraints*

Departures are not allowed to takeoff sequentially from the same runway without proper separation. Departure separation time is dependent the departure sequence and the weight class of aircraft departing. Physically, this is due to restrictions on aircraft departing into another aircraft's wake vortex. If aircraft i departs before aircraft j then the proper wake vortex separation, given in units of time, is denoted by V_{ij}. The constraint which prevents a violation in the separation between consecutive departures is given by:

$$\rho_{ij}(t_{jd_j} - t_{id_i} - V_{ij}) \geq 0 \qquad \forall i, j \in D, i \neq j \tag{27}$$

The linearization is as follows:

$$t_{jd_j} - t_{id_i} - V_{ij} \geq -(1-\rho_{ij})M \qquad \forall i,j \in D, i \neq j \tag{28}$$

A runway will usually have multiple entrance and exit points and should be protected from all possible conflict combinations. For example, Dallas/Fort Worth International Airport has arrival crossing points along the edge of 17R (Fig. 4). Equation (2) restricts an arrival aircraft from crossing 17R while a departure is occurring. Similarly, an aircraft wishing to depart must wait for the arrival to finish crossing. This is expressed below in (30). We have denoted T^i_{dep} as the time it takes a departure to take-off (usually taken as 40-60 seconds). Note that b_k and a_k are nodes corresponding to the kth runway exit.

$$\rho_{ij}(t_{jb_k} - t_{id_i} - T^i_{dep}) \geq 0 \qquad \forall j \in A, \forall i \in D \tag{29}$$

$$\rho_{ji}(t_{id_i} - t_{ja_k}) \geq 0 \qquad \forall j \in A, i \in D \tag{30}$$

The corresponding linearization:

$$t_{jb_k} - t_{if^i_{n_i}} - T^i_{dep} \geq -M(1-\rho_{ij}) \qquad \forall j \in A, \forall i \in D \tag{31}$$

$$t_{if^i_{n_i}} - t_{ja_k} \geq -M(1-\rho_{ji}) \qquad \forall j \in A, i \in D \tag{32}$$

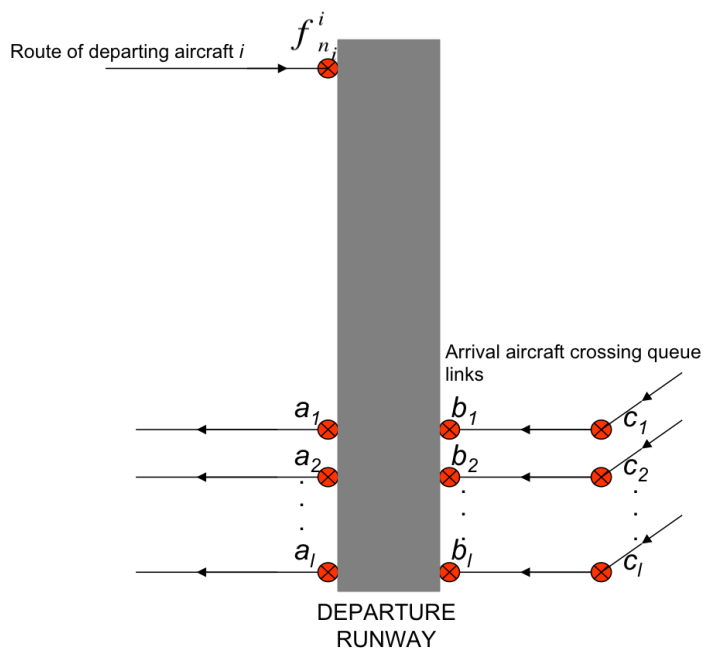


Figure 4: Multiple runway entrance and exit points

7. *Capacity Constraints for Runway Crossing Queue*

At Dallas/Fort Worth International Airport there are four exit edges off the arrival runway 17C. Each of these edges have capacity constraints so that each does not exceed a prescribed capacity, e_l. If c_l denotes a runway crossing node (Fig. 4), then arrival aircraft, which use edge l, must leave that node before the edge capacity e_l is violated. e_l is the capacity, given in units of aircraft, for the lth taxi exit. In order to formulate this mathematically, it is easier to introduce the time T^l_{ij}. T^l_{ij} is equivalent to ETD_j, and also we require that i lands before j, i and j both use the same edge l, and that e_l aircraft arrive between i and j. Furthermore, let L be a set of all exit taxi edges. Then the capacity constraint reads,

For all $i \neq j \in A$, for all $l \in L$

$$t_{ic_l} \leq T_{ij}^l \tag{33}$$

D. Objective Functions

Two objective functions have been identified to solve a MRATSP. The first objective function minimizes the total time that all aircraft spend in the system, denoted by (34).Throughout the rest of this paper the total time spent in the system is equivalent to the total *taxi time* of all aircraft.

$$\min \sum_{i \in P} t_{id_i} \tag{34}$$

The second objective function tries to maximize the throughput of the departure aircraft. This objective function is naturally written in the following way:

$$\min \max t_{id_i} \qquad \forall i \in D \tag{35}$$

And can be rewritten more applicably with some arbitrary real variable S:

$$t_{id_i} \leq S \qquad \forall i \in D \tag{36}$$

$$\min S \tag{37}$$

E. Additional Constraints

Useful departure constraints can be added to the problem to decrease computation time. A difficult aspect of this problem is in deciding which departure aircraft should use the runway first for takeoff. Deciding the departure sequence at the runway is a subproblem in a MRATSP and is known as the departure scheduling problem [13],[14].

For all $i, j \in D$ such that the type of aircraft j is equal to the type of aircraft i and the unimpeded taxi time of i plus PBT_i is less than or equal to the unimpeded taxi time of j plus PBT_j,

$$\rho_{ij} = 1 \tag{38}$$

$$\rho_{ji} = 0 \tag{39}$$

This constraint is only valid when considering departure throughput as an objective and when the departures are released from a nodes where taxiway collisions are not possible. Also, it will not hold if you try to optimize arrivals simultaneously. For example, a departure sequence at the departure runway based on the above constraint may not be optimal because arrivals may cause departures to delay on the taxiway. In section IV.C we discuss how this constraint is used in conjunction with objective functions (35) and, to some degree, (34) to find isolated optimal departure throughput solutions.

IV. Simulation and Results

The following section illustrates how the above optimization model is used to solve a MRATSP problem on the east side of Dallas/Fort Worth International Airport. An arrival taxi time comparison between the model used in [7] and the multi-oute model is made by incorporating additional perimeter routes for arrivals. Section A describes all the model parameters, Section B describes how the inputs are configured for three different scenarios, Section C outlines an efficient method for comparing the single route formulation and the multi-route formulation, and Section D presents the results.

A. Simulation Inputs

This section provides a description of each input and the numerical values which correspond to the simulation. Fig. 5 is the node-edge model used in the simulation. Each node has been accurately placed using Global

Positioning System data. The perimeter taxiway was created using the known distance from the south-end of 17R and represents a decent approximation of the actual perimeter taxiway at Dallas/Fort Worth International Airport. The node-edge model has two active runways: one for departures (17R) and one for arrivals (17C). Arrival aircraft enter the airport just before the crossing nodes off 17C (orange rectangles). All departures are released from queue nodes, which are located in the departure queue just before the runway takeoff point (aqua-teal rectangles). Often at Dallas/Fort Worth International Airport there are large queues that form right before the runway and by releasing departures close to the runway entry we try to mimic this occurrence.

All departures start moving within a few minutes of each other so that there is adequate pressure on 17R from the departures. Arrival aircraft arrive in sixty second intervals and no more than three can occupy the same queue to cross 17R, so that $e_l = 3$ for all runway exit edges. When using the multi-route model, two routes are supplied to arrival aircraft: one shortest path route that crosses the runway and another longer route that goes along the perimeter of 17R (as discussed in section IV.C).

To simulate medium-to-heavy scenarios, we use 22-, 26-, 30-, and 34-sized aircraft problem instances. These scenarios have equally distributed departures and arrivals. Aircraft weight classes are distributed according to the particular scenario, as explained in Section IV.B. The departure separation matrix given in Table 2 was empirically collected using surface surveillance data [11] from Surface Management System records [9].

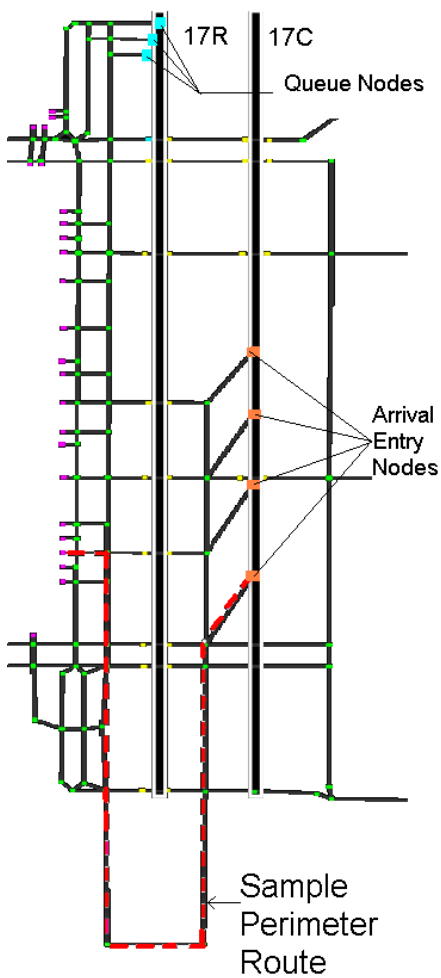


Figure 5: Thee east side of Dallas/Fort Worth International Airport node-edge model used in the simulations. The perimeter taxi route is outlined by a red dotted line.

Aircraft separation at each node is variable depending on the aircraft sizes (e.g., Table 3) and aircraft speed (e.g., Equations 21 and 22). There is a buffer separation of ten meters at each node. Aircraft sizes

Table 2: Inter-departure aircraft separation in seconds. For proper separation column aircraft lead row aircraft.

	Small	Large	Heavy	B757
Small	59	88	109	110
Large	59	61	109	91
Heavy	59	61	90	91
B757	59	61	109	91

have been correlated to the weight class of the aircraft for simplification and are 30m, 40m, 60m, and 70m for small, large, heavy, and B757, respectively.

Table 3: Aircraft separation at all nodes given in meters.

	Small	Large	Heavy	B757
Small	40	45	55	60
Large	45	50	60	65
Heavy	55	60	70	75
B757	60	65	75	80

For all problems, it is assumed that departures take an average of fifty-five seconds to depart (e.g., $T^i_{dep} = 55$ sec). M is equal to the sum of the unimpeded taxi times for all aircraft. This is a conservative estimate on all linearizing constants and faster convergence can be obtained with tighter constants. The speed bounds are uniform for all aircraft and are 11.6-17.5 knots. Speed bounds will not apply to departure aircraft because their activation nodes are the same as their destination nodes (e.g., $o_i = d_i$ for all departures i). Also, the arrival aircraft are allowed to wait (zero speed) at runway exit nodes c_l subject to the capacity as explained in Section III.C.

B. Scenario Description

Two scenarios have been developed to illuminate various features of the optimization model solutions. This subsection describes discrepancies between two scenarios. Except for the following conditions, all other inputs are identical according to the above description.

Scenario 1

1. Aircraft weight classes are distributed uniformly.
2. Departures are given priority with an optimal departure throughput schedule.

Scenario 2

1. Aircraft weight classes are distributed based on actual data. This distribution has been collected with surface traffic data at Dallas/Fort Worth International Airport and is approximately 85% large, 5% heavy, 5% small, and 5% B757 classes.
2. Departure aircraft are given successively less priority by allowing a maximum increase to the makespan of 15, 30, 45, and 60 seconds.

For the the results section below, it will be useful to distinguish each sub-scenario where the optimal makespan is successively increased. For each sub-scenario that successively gives departures less priority we adopt the notation 2.x. The appended number x indicates the maximum increase to the optimal departure

makespan. For example, 2.15 indicates we are using Scenario 2 conditions with fifteen seconds of maximum increase to the optimal departure makespan. Likewise, 2.0 indicates we are using scenario 2 conditions with zero seconds of maximum increase to the optimal departure makespan. In this way, there is no confusion between any condition lumped in Scenario 2.

With these scenarios, the effects of weight class distribution and departure throughput on the arrival taxi times by virtue of increased or decreased perimeter route usage can be illustrated. The results using the above scenarios are discussed in SectionIV.D.

For each scenario, the authors use traffic level sizes of 22, 26, 30, and 34 aircraft. For all scenarios 100 problem instances are solved for and among those 100, there are 25 attributed to each traffic level. For example, 25 instances of scenario 1 use 22 aircraft, 25 use 26 aircraft, 25 use 30 aircraft, and 25 use 34 aircraft. From instance to instance within the same Scenario, departure aircraft types, arrival aircraft taxi exits, arrival aircraft routes, and arrival aircraft types can vary.

C. Optimization Program Setup

Using optimization suite CPLEX [12] in conjunction with its C++ application programming interface, the authors have developed a four-step optimization technique to find optimal taxi times for arrivals given an optimal departure sequence. The solving scheme is summarized as follows:

1. Use the above optimization model under objective function (35) to solve the departure sequencing problem with no arrivals using cuts described in Section III.E.

2. Setup a new problem considering the arrivals and fix all t_{iu} for all departures based on the solution found in step one. The departure times t_{iu} are not hard constraints for Scenario 2 since they successively suffer an increase in makespan, but the departure time of the last aircraft to depart will be no farther than 15, 30, 45, or 60 seconds from its optimal time.[a]

3. In the new problem, provide two routes for arrival aircraft: a shorter route which crosses the departure runway and a longer perimeter route which will not interfere with departures.

4. Finally, solve this problem with the above optimization model under objective function (34).

The method is formulated for efficient comparison of a single route formulation and the multi-route formulation. In step one, an optimal departure sequence is calculated without considering any arrival aircraft on the surface. Step one is the most computationally expensive process in this methodology, and this is one reason a problem instance is not solved for at once. Using the above methodology, however, a guaranteed solution on the Pareto surface between optimal throughput for departures and optimal taxi times for arrivals is achieved.

The optimal departure sequence for the departure aircraft can be found trivially by noting that the optimal solution is always {Small,Small,...,Large,Large,...,B757,B767...,Heavy,Heavy,...}. To avoid pre-processing, constraints from section Section III.E can be used. These constraints may be used because its assumed all aircraft can arrive at the departure runway at the same time. This assumption is acceptable because during times of peak traffic at Dallas/Fort Worth International Airport departure queues near the runway are full and any departure sequence can be obtained from the spots. This assumption also implies that all departure delay is due to departure queues.

D. Preliminary Results

In this section results from using Scenarios 1 and 2 are compared and contrasted with respect to basic scheduling performance parameters. In order to determine if arrival taxi times are effected by using the perimeter taxiway, the total arrival taxi time savings has been calculated. The total or accumulated arrival

[a]The authors understand that there may be more than one optimal departure sequence when solving Scenario 1 and Scenario 2, which may determine various arrival taxi times. This discrepancy, however, is inconsequential for the model comparison made in this paper. In the future it will be desirable to determine the departure sequence that minimizes the arrival taxi times.

taxi time savings is calculated by taking the difference of the total arrival taxi times found from the single route formulation and the total arrival taxi times found from the multi-route formulation. This value is then summed across all instances for a particular traffic load. While total arrival taxi time savings is an indicator of performance across models, it does not indicate the decency or quality of the taxi schedules. Therefore to capture individual schedule quality, average arrival taxi times are presented. Additional calculations are done to determine the number of problem instances (out of twenty-five) to use the perimeter taxiway.

For Scenarios 1 and 2 the results indicate that the multi-route formulation yields reduced taxi times over the single route formulation for arrivals *only* when a perimeter taxiway is used. The optimal solution of the multi-route formulation and the single route formulation are identical when there is no perimeter taxiway use.

Both graphs in Fig. 6 show how arrival perimeter taxiway use is directly correlated with arrival taxi savings. The top graph is the total arrival taxi savings versus problem instance, and the bottom graph is the associated perimeter use. On the bottom graph the two red bars indicate problem instances where the single route formulation could not find a solution. There are no arrival taxi time savings for problem instances when the single route formulation is infeasible, because there is no baseline for comparison.

An interesting result, which is highlighted by the red bars, from Fig. 6 is that the single route formulation is infeasible only when a perimeter route is used. The number of infeasible values for the single route formulation also show up in Tables 4-9. A complete analysis has not been accomplished to indicate why there are infeasible solutions for the single route formulation; however, an intuitive reason for the infeasible solutions is provided. The capacity constraints on the high-speed taxi edges are violated and cannot be mitigated. The multi-route formulation deals with this by moving an arrival aircraft to the perimeter taxiway.

For the remainder of this section the authors will refer to Tables 4, 5, 6, 7, 8, and 9. These tables give the total arrival taxi time savings, perimeter route use, feasibility of the single route model, the average arrival taxi times, and computational times. These results are displayed as a function of the number of aircraft being scheduled for each table. Scenario 2 conditions are represented in Tables 5-9 with differences only in the maximum makespan for departures. We use the notation described in Section IV.B, where 2.x indicates we are using Scenario 2 conditions with x seconds of maximum increase to the optimal departure makespan.

The total arrival taxi time savings are significantly larger for Scenario 2.0 conditions. Scenario 2.0 provides departure aircraft with an optimal throughput schedule and uses actual aircraft weight class distributions. Since the majority of aircraft are large (85%) there is simply not enough time between consecutive departures for arrivals to cross. Hence, many perimeter routes are used to accommodate long arrival crossing waiting and feasibility.

While the total arrival taxi savings are the largest for Scenario 2.0, the average arrival taxi time is the greatest. The large taxi times are due to a large number of perimeter routing decisions. From Table 5, one can see that as the traffic load rises from 22 to 34 aircraft the perimeter use goes from 92 (23/25) to 100 (25/25) percent. The consequence of high perimeter use is larger average arrival taxi times. The average arrival taxi times eventually converge to fifteen minutes, which is approximately the time it takes on average for an aircraft to take the perimeter route from 17C.

Table 4: Summary of results for Scenario 1

Number of Aircraft	Total Arrival Taxi Savings (sec)	Average Arrival Taxi Time (sec)	Perimeter Count	Infeasibility Count	Computation Times (sec)		
					Average	Max	Min
22	35.67	537.38	2	1	3	9	2.4
26	302.52	594	3	1	7.01	11.37	3.12
30	884.11	676.75	7	0	15.34	41.87	5.89
34	678.4	734.84	6	0	27.9	52.68	13.49

To illustrate the effect that departure makespan has on the arrival taxi times, consider Scenarios 2.0, 2.15, 2.30, 2.45, and 2.60. From Tables 5-9, we can draw on an apparent phenomenon. The perimeter taxiway use goes down significantly from almost 100% (2.0) use to almost 0% (2.60) use, and the average arrival taxi time decreases largely from about 900 seconds (2.0) to 690 seconds (2.60). For an exchange of

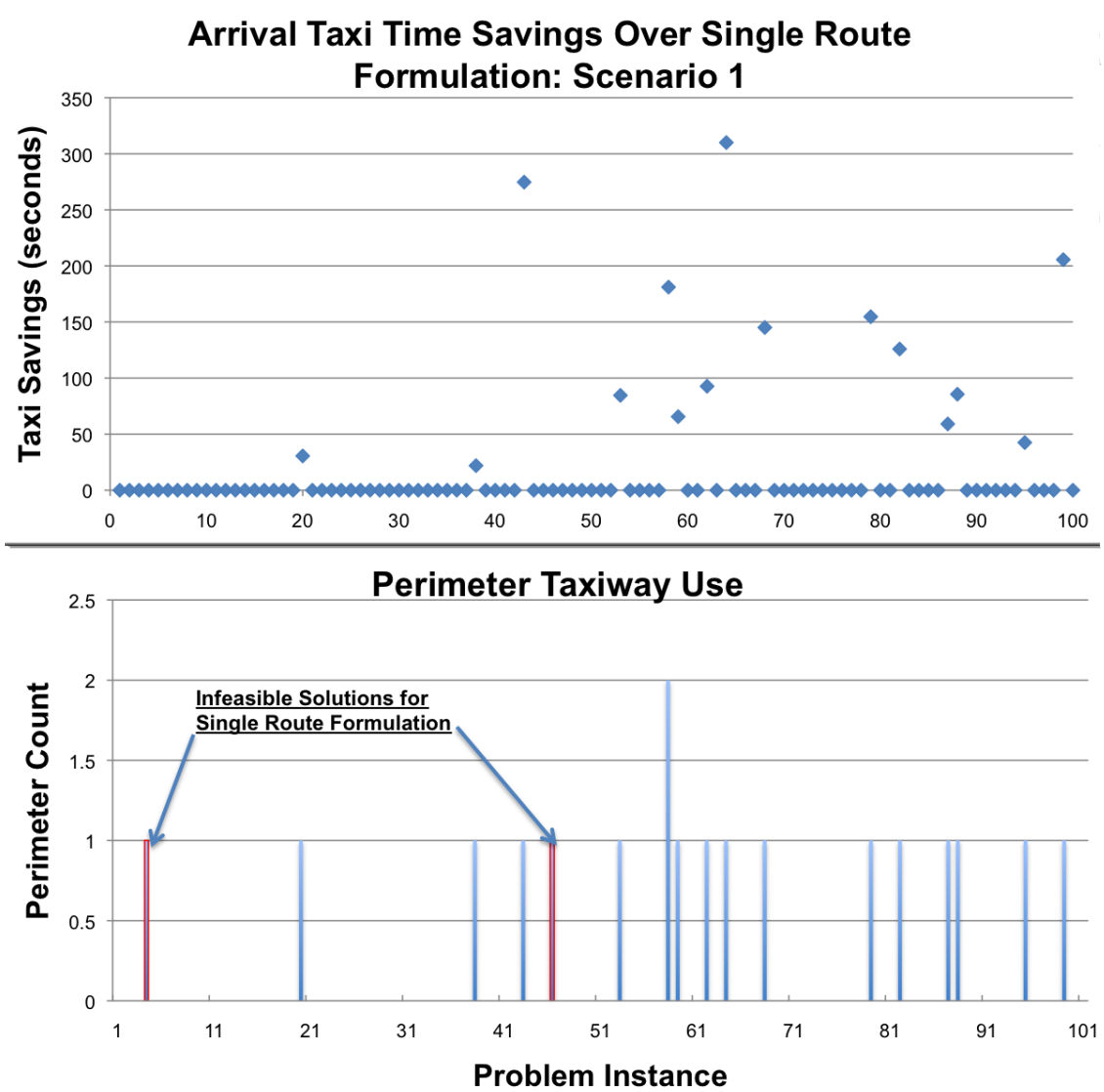


Figure 6: Arrival perimeter taxiway use (bottom) and total arrival taxi savings (top) per instance for Scenario 1.

Table 5: Summary of results for Scenario 2.0

Number of Aircraft	Total Arrival Taxi Savings (sec)	Average Arrival Taxi Time (sec)	Perimeter Count	Infeasibility Count	Computation Times (sec)		
					Average	Max	Min
22	3262.18	669.13	23	4	4.4	8.62	2.32
26	3842.13	751.02	24	12	9.7	23.92	3.14
30	12955	819.5	25	9	19.8	34.86	9
34	7071.9	898.86	25	18	40.5	195.5	14.9

about sixty seconds of makespan in the departures, we can save 210 seconds of taxi time per arrival aircraft on average. For scenarios with thirty-four aircraft, we save about one hour of taxi time per problem instance by increasing the departure makespan by sixty seconds.

Table 6: Summary of results for Scenario 2.15

Number of Aircraft	Total Arrival Taxi Savings (sec)	Average Arrival Taxi Time (sec)	Perimeter Count	Infeasibility Count	Computation Times (sec)		
					Average	Max	Min
22	260.31	538.18	2	0	48.05	437.3	2.52
26	380.06	607.22	7	0	950	14885	5.71
30	1935.32	684.53	12	0	4078	18808	10.3
34	2201.3	760.45	16	0	12866	20178	184.67

Table 7: Summary of results for Scenario 2.30

Number of Aircraft	Total Arrival Taxi Savings (sec)	Average Arrival Taxi Time (sec)	Perimeter Count	Infeasibility Count	Computation Times (sec)		
					Average	Max	Min
22 Aircraft	0	511.38	0	0	5.17	10.73	2.66
26 Aircraft	0	574.9	0	0	18.33	177.47	5.89
30 Aircraft	156.16	644.53	1	0	205.19	2260.63	9.95
34 Aircraft	52.8	756.64	1	0	1014.57	3849.9	38.49

Computationally, the multi-route formulation is relatively efficient for problem instances where the departure makespan is fixed. However, when the departure makespan is progressively changed with 0 to 60 seconds of maximum deviation from the optimal makespan, the computation times become significant. The average computation time for Scenario 2.0 (fixed makespan) for thirty-four aircraft is 40.5 seconds and for Scenario 2.15 the average computation time is 3.6 hours.

Table 8: Summary of results for Scenario 2.45

Number of Aircraft	Total Arrival Taxi Savings (sec)	Avgerage Arrival Taxi Time (sec)	Perimeter Count	Infeasibility Count	Computation Times (sec)		
					Average	Max	Min
22	0	504.98	0	0	5.0	11.35	2.55
26	0	565.37	0	0	11.67	27.36	4.16
30	0	632.34	0	0	41.9	319.63	9.23
34	38.3	694.82	1	0	295.38	4318.13	27.3

Interestingly, the average computation times reduce for even larger potential changes in the optimal departure makespan. In Scenarios 2.30, 2.45, and 2.60 the average computation times for thirty-four aircraft are 1014.57 seconds, 295.38 seconds, and 27.9 seconds, respectively.

V. Conclusion

A mixed integer linear program to solve a Multiple Route Aircraft Taxi Scheduling Problem (MRATSP) was introduced. The model is a generalization of the model presented in [7] to include routing choices. MRATSP models the airport as a node-edge graph to incorporate fundamental safety considerations such as head-on collisions, safety separation between aircraft, and proper departure constraints. New and necessary constraints are included to model capacity constraints on high speed arrival runway exits.

Table 9: Summary of results for Scenario 2.60

Number of Aircraft	Total Arrival Taxi Savings (sec)	Avgerage Arrival Taxi Time (sec)	Perimeter Count	Infeasibility Count	Computation Times (sec)		
					Average	Max	Min
22	0	499.25	0	0	5.36	10.78	2.5
26	0	559	0	0	.01	27.58	4.78
30	0	626.5	0	0	15.34	67.71	17.72
34	0	688.6	0	0	27.9	339.28	38.02

A comparison is made between a similar single route formulation in [7] and the multi-route formulation to understand if arrival taxi time savings can be realized by using additional perimeter taxiways under optimal and suboptimal departure throughput. Results indicate that the multi-route formulation can yield better arrival taxi schedules under conditions when the departures have a fixed makespan or when an actual aircraft weight class distribution is used. Additional comparisons suggest that by increasing the departure makespan successively from 0 to 60 seconds, approximately 210 seconds per arrival aircraft of taxi time can be avoided resulting in less frequent perimeter taxiway use.

Perimeter taxiway decisions are just one application of the multi-route formulation.In the future it will be useful to explore alternative applications of the multi-route formulation. For example, another application can consider multiple routes for departing aircraft to multiple runways. In this way, optimal runway use can be better understood. This problem is usually referred to as a runway balancing problem. To extend on the current application it will be useful to build a framework where departure/arrival priority can be adjusted by preference. For instance, scalarization or weighting techniques may be applied to individual aircraft time variables.

References

[1] S. Zelinkski and T. Romer, "An Airspace Concept Evaluation System Characterization of National Airspace System Delay," AIAA 4th Aviation Technology, Integration and Operations (ATIO), September 2004.

[2] Mueller E.R. and Chatterji G.B., "Analysis of Arrival and Departure Delay Characteristics," AIAA Aircraft Technology, Integration, and Operations (ATIO) Forum, Los Angeles, CA, October 1-3, 2002.

[3] "Costs of Air Transport Delay in Europe," 2000 Eurocontrol Final Report." [http://www.eurocontrol.int/prc/gallery/content/public/Docs/stu2.pdf]

[4] S. Aktins, Y. Jung, C. Brintons, L. Stell, T. Carniol, and R. Ragowski, "Surface Managment System Field Trials Results," AIAA 4th Aviation Techonology, Innovation and Operations (ATIO) Forum, Chicago Illonoise, September 2004.

[5] Z.Wood, M. Kistler, S. Rathinam, and Y. Jung, "A Simulator for Modelling Aircraft Surface Operations at Airports," AIAA Modeling and Simulation Technologies Conference, Chicago, Illonoise, August 2009.

[6] J.W. Smeltink, M.J Soomer, P.R. De Waal, and R.D. Van Der Mei, "An Optimisation Model for Airport Taxi Scheduling," Thirtieth Conference on the Mathematics of Operations Research, Lunteren, The Netherlands, January 2005.

[7] S. Rathinam, J. Montoya, and Y. Jung, "An Optimization Model for Reducing Taxi Delays at DFW," The 28th International Council of the Aeronautical Sciences, Anchorage, Alaska, 2008.

[8] H. G. Visser and P. C. Roling, "Optimal Airport Surface Traffic Planning Using Mixed Integer Linear Programming," AIAA Aviation Technology, Integration and Operations (ATIO) Conference, Denver, CO, 2003.

[9] H. Balakrishnan, and Y. Jung, "A Framework for Coordinated Surface Operations Planning at Dallas-Fort Worth International Airport,"AIAA Guidance, Navigation, and Control Conference, Hilton Head, SC, August 20-23, 2007.

[10] S. Satyamurti, "Runway Incursion Mitigation, Capacity Enhancement, And Safety Improvements With Perimeter Taxiway Operations," Ph.D. Thesis, The University of Texas at Arlington, 2007.

[11] Airport Surface Detection Equipment Model X, Sensis Corporation. [http://www.sensis.com/docs/128]

[12] IBM ILOG CPLEX, Version 12.0, 2008 Release.

[13] G. Gupta, W. Malik, and Y. Jung, "A Mixed Integer Linear Program for the Airport Departure Scheduling Problem," 9th AIAA Aiviation Technoology, Integration, and Operations Conference (ATIO), Hilton Head, SC, September 2009.

[14] S. Rathinam, Z.Wood, B. Sridhar, and Y. Jung, "A Generalized Dynamic Programming Approach for a Departure Scheduling Problem," AIAA Conference on Guidance, Navigataion, and Control, Chicago, Illonoise, August 2009.