12th AIAA Aviation Technology, Integration, and Operations (ATIO) Conference and 14th AIAA/ISSM
17 - 19 September 2012, Indianapolis, Indiana

AIAA 2012-5498

Spot Release Planner: Efficient Solution for Detailed Airport Surface Traffic Optimization

Waqar A. Malik[*] and Gautam Gupta[*]

University of California, Santa Cruz, NASA Ames Research Center, Moffett Field, CA 94035, USA

Yoon C. Jung[†]

NASA Ames Research Center, Moffett Field, CA 94035, USA

The Spot Release Planner (SRP) is an algorithm previously developed by the authors to reduce delay and congestion on the airport surface. The algorithm was developed to provide real time advisories to tower controllers. A Human-in-the-loop (HITL) simulation in April 2010 showed that the SRP reduced the average movement area delay of departure aircraft by 64%. The SRP is a two-stage algorithm that considers runway scheduling in the first stage, and the rest of the ground movement, such as gate pushback and spot release, in the second stage. This decomposition of airport surface scheduling into two stages provides fast computational times and makes the SRP applicable for real-time decision making. However, the two stages also result in the given scheme being a heuristic for solving the complicated airport surface scheduling problem; no guarantees on quality of the obtained solution have been provided. This paper explores the quality of solutions obtained by the SRP and compares them with the optimal solution for airport surface traffic. Simulations conducted for the East side of Dallas/Fort Worth International Airport (DFW) show that the SRP solutions are within $14s$ of the optimal solution for a detailed airport surface planner.

Nomenclature

$\mathbb{A}$	Set of all aircraft
t_{iu}	Time at which aircraft i reaches node u
z_{iju}	Sequencing of aircraft i and j at node u
R_i	Route of aircraft i. $R_i = \{u_{i0}, u_{i1}, ..., u_{in}\}$
E_i	Set of links in route of aircraft i. $E_i = \{(u_{i0}, u_{i1}), (u_{i1}, u_{i2}), ..., (u_{in-1}, u_{in})\}$
α_i	Earliest available time for aircraft i at the first node on its route
δ_{ij}	Temporal separation between aircraft i and j when aircraft j is behind i on the taxiway
δ^r_{ij}	Wake vortex separation between aircraft i and j, when aircraft j is behind aircraft j
$\mathcal{TS}$	Taxi scheduling formulation
$\mathcal{SRP}_1$	SRP Stage 1 formulation
Δ	Separation between SRP and $\mathcal{TS}$
Subscript	
i, j	Individual aircraft
u, v	Nodes on airport
r	Runway node
s	Spot node

[*] Associate Research Scientist, University Affiliated Research Center, MS 210-8, Moffett Field, CA 94035.

[†] AIAA senior member and Aerospace Engineer, NASA Ames Research Center, MS 210-6, Moffett Field, CA 94035.

I. Introduction

Meeting the projected increase in air traffic demand within the National Airspace System (NAS) requires improvements in all areas of air traffic management. Airports, being the origin or destination of the air traffic network, encounter some of the highest traffic density in the NAS. During peak periods at major airports, capacity limitations on the airport surface area create bottlenecks and cause delays to both departures and arrivals. This congestion effect and associated delays persist for a significant part of the peak period, and often restrict an airport's throughput by hampering runway operations. Throughput may be augmented by constructing additional facilities such as runways or taxiways. However, the practical difficulties involved in airport expansion, both geographic and monetary, introduce the need for decision support tools that optimize the use of current airport infrastructure.

Research has been conducted in the United States and Europe in the area of airport surface traffic planning. Various optimization techniques have been used to reduce taxi delays and fuel emissions for surface traffic.[1–16] In the majority of airports in the United States, airlines control the ramp area (non-movement area), while the FAA Air Traffic Control Tower (ATCT) controls traffic on taxiways and runways (movement area). Typically, airlines push-back an aircraft from its gate as soon as the aircraft is ready and during peak periods, the uncoordinated push-backs result in taxiway congestion and large runway queues. Moreover, operational constraints at the runways, such as Miles-InTrail (MIT) restriction over certain departure fixes and Expected Departure Clearance Time (EDCT) for some aircraft, can cause additional delays. Idris et al.[1] observed that a majority of airport surface delay was incurred at the runways. Although optimal runway scheduling[2–6] may alleviate some of the surface congestion, the cause of long queues at the runways is the absence of a gate or spot[a] metering policy. In recent years, a number of human-in-the-loop simulations[7] and field trials[8–10] have assessed the impact of various departure metering concepts. Although these trials have shown significant reductions in taxi delays and fuel emissions, most of the implemented departure metering schemes are based on heuristics with little quantification of the solution quality. Comparing the obtained solution with the optimal throughput or delay would help identify additional benefits which could potentially be obtained.

In the human-in-the-loop simulations,[7] the Spot Release Planner[11] (SRP) was used to advise the Ground Controller (GC) on the spot release times for the aircraft. The purpose of the advisories was to achieve a small queue at the runway, improve throughput and reduce taxi delays. In this paper, we compare the solution obtained by the SRP with the optimal throughput solution for complete airport surface traffic and provide a method to quantify the quality of the solution by measuring the gap (difference) from the optimal solution.

The content of the paper is organized as follows: An overview of the two algorithms, the detailed airport surface planner and SRP is provided in section II. The methodology used to compare the SRP solution with the solution to the detailed airport planner is presented in section III. Details of simulation performed for East side operations at the Dallas/Fort Worth International Airport (DFW) are provided in section IV.

II. Background

Existing literature primarily focuses on developing optimal solutions for operations of aircraft on the airport surface, including ramp area, taxiways, and runways. Two algorithms applicable for generating advisories for Air Traffic Control Tower (ATCT) controllers at DFW are the Taxi Scheduler and the Spot Release Planner. They are described in sections II.A and II.B respectively.

II.A. Taxi Scheduling – Detailed Airport Surface Planner

The taxi scheduling problem finds the optimal times for the aircraft to leave the control point and to reach different points (nodes) along its route. It is important that aircraft maintain required separations at all times while moving. There are many variants of the taxi scheduling problem[12–16] – the exact problem description and solution technique depends on the requirements of the airport being modeled. For our comparative analysis between schedulers, the selected taxi scheduler should model the airport surface traffic at DFW[b] accurately. Variants of the taxi scheduling problem have considered pre-determined routes for the

[a] Spots are physical locations at the airport where the control of a departure aircraft transfers from the ramp controller to the ATCT ground controller.

[b] East-side DFW operations with a south flow configuration.

aircraft,[12,13] or have chosen the route from a set of predefined routes.[14–16] DFW has a structured taxiway system and has predefined routes similar to the highlighted routes shown in Figure 1. Moreover, since the taxi scheduler should represent the complete airport surface traffic it should consider the runway as part of an aircraft's route and should incorporate the required safety separations at the runway(s) and taxiways.

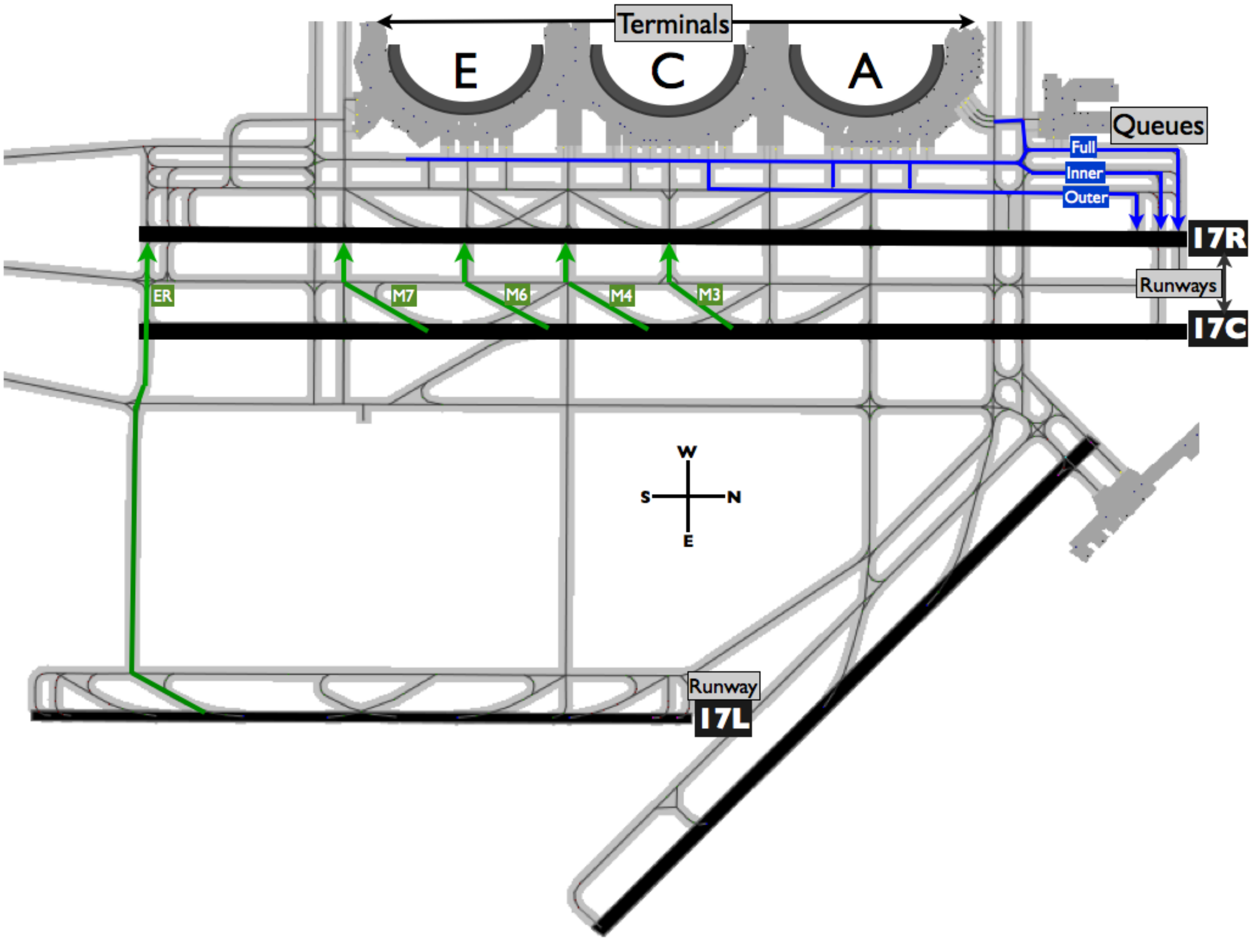

Figure 1. East-side DFW: routes under south flow configuration (arrivals: green, departures: blue).

A taxiway model based on a Mixed Integer Linear Program (MILP) formulation was developed by Smeltink, et al.[12] They model the taxiways at Amsterdam Schiphol Airport as a directed graph (network of links and nodes). The sequencing constraints are modeled using two binary variables, y_{iju} and z_{iju}. Variable y_{iju} equals 1 if aircraft i visits node u *immediately* before aircraft j does and equals 0 otherwise. Variable z_{iju} equals 1 if aircraft i visits node u before aircraft j does and equals 0 otherwise. Variable y_{iju} is only used to support the sequencing model provided by using z_{iju}. The required times for aircraft at the nodes are modeled as continuous variables. This model incorporates various taxiway conflicts, such as overtaking constraints, head-on constraints, merging constraints, and separation constraints between a pair of aircraft on taxiway and runways. Smeltink, et al. analyzed the optimal taxi schedules of different problem instances, and observed that very low taxi speeds rarely occur – most of the time aircraft taxied at their maximum speed or at the maximum speed of a slower aircraft in front of them. Based on this observation, they concluded that minimal speed can be introduced in the model, and if chosen low enough it does not affect the optimal schedule. The minimum speed restriction imposes stricter bounds on variables and thus reduces computation time. Rathinam, et al.[13] simplified Smeltink's taxi scheduling model by removing the variable y_{iju} and adding additional minimum separation requirements.

In this paper, we use a simplified Smeltink model that uses the variable z_{iju} and does not use the variable y_{iju}. The formulation used in this paper is similar to the model given in Rathinam, et al.;[13] however we do not use the additional separation requirements and keep both the lower and upper bounds on travel time on a given link.

The simplified Smeltink model for taxi scheduling is given below. Let $\mathbb{A}$ denote the set of all aircraft, with i, j denoting individual aircraft. u, v represent nodes on the airport and r denotes the runway node. Let $R_i = \{u_{i0}, u_{i1}, ..., u_{in}\}$ be the pre-determined route for the i^{th} aircraft with $u_{i0}, u_{i1}, ...$ being the nodes along its route. If the i^{th} aircraft is a departure, then u_{i0} is a spot node and u_{in} is the runway node. If the i^{th} aircraft is an arrival, then u_{i0} is the runway node and u_{in} is the spot node. Let E_i be the set of links that defines the route for the i^{th} aircraft with $E_i = \{(u_{i0}, u_{i1}), (u_{i1}, u_{i2}), ..., (u_{in-1}, u_{in})\}$. α_i is the earliest time the i^{th} aircraft is available at the first node. The decision variables are the binary sequencing variables z_{iju} and the continuous variable t_{iu}. t_{iu} denotes the time aircraft i reaches node u.

The following is the formulation for the taxi scheduling problem and will henceforth be referred to as $\mathcal{TS}$.

`Formulation` $\mathcal{TS}$:

$$\min \quad \Gamma := \max_{i \in \mathbb{A}} t_{ir} \tag{1}$$

$$z_{iju} + z_{jiu} = 1 \qquad \forall i, j \in \mathbb{A}, \forall u \in R_i \cap R_j \tag{2}$$

$$z_{iju} = z_{ijv} \qquad \forall i, j \in \mathbb{A}, \forall (u, v) \in E_i \cap E_j \tag{3}$$

$$z_{iju} = z_{ijv} \qquad \forall (u, v) \in E_i \text{ and } (v, u) \in E_j \tag{4}$$

$$z_{iju}(t_{ju} - t_{iu}) \geq z_{iju}\delta_{ij} \qquad \forall i, j \in \mathbb{A}, \forall u \in R_i \cap R_j \tag{5}$$

$$z_{ijr}(t_{jr} - t_{ir}) \geq z_{ijr}\delta^r_{ij} \qquad \forall i, j \in \mathbb{A} \tag{6}$$

$$t_{iu} + \frac{l_{uv}}{V^{max}_{iuv}} \leq t_{iv} \leq t_{iu} + \frac{l_{uv}}{V^{min}_{iuv}} \qquad \forall i \in \mathbb{A}, \forall (u, v) \in E_i \tag{7}$$

$$t_{iu_0} \geq \alpha_i \qquad \forall i \in \mathbb{A}, u_{i0} = R_i.front() \tag{8}$$

$$z_{iju} \in \{0, 1\} \qquad \forall i, j \in \mathbb{A}, i \neq j, \forall u \in R_i \cap R_j \tag{9}$$

$$t_{iu} \in \mathbb{R}^+ \qquad \forall i \in \mathbb{A}, u \in R_i \tag{10}$$

- Eq. (1) specifies the objective function for maximizing the throughput, which is equivalent to minimizing the time when the last aircraft uses the runway. The runway usage time t_{ir} is the time when an aircraft starts take-off roll (if i is a departure), or starts crossing the departure runway (if i is an arrival).

- Eq. (2) provides the linear ordering constraints, i.e., given any two aircraft, one always leads the other at a common node u on their route. Eq. (3) prevents aircraft from overtaking each other on a common segment along their routes.

- Eq. (4) represents the head-on or crossing constraints. This constraint ensures that two aircraft, while moving in opposite directions, do not cross each other on a common link.

- Eqs. (5) and (6) enforce the required separation between a pair of aircraft on the taxiway and runway, respectively. δ_{ij} is the minimum time-based separation between aircraft i and aircraft j when aircraft j is behind aircraft i on the taxiway. Similarly, if aircraft j uses the runway after aircraft i, δ^r_{ij} is the required wake vortex separation between them (converted to a time-based separation). These quadratic separation constraints are linearized using big-M formulation.[17]

- Eq. (7) provides the timing constraints which arise due to aircraft i having a speed $V \in [V^{min}_{iuv}, V^{max}_{iuv}]$ on link (u, v) on its route. The constraint allows for different speeds based on aircraft type, and link type.

- Eq. (8) provides the constraints on the time the aircraft can leave the first node. A departure can be released from the spot only after the spot arrival time, and an arrival can cross an active runway only after it arrives at the runway.

- Eqs. (9) and (10) define the domain of the decision variables.

Since the formulation $\mathcal{TS}$ provides a 4-D surface trajectory (required times at all nodes along the route including the runway for all aircraft), we refer to it as a detailed airport surface planner, and it provides the optimal solution for airport surface traffic. As the number of aircraft increases, the computation time of $\mathcal{TS}$ increases significantly, making it unsuitable for use in real-time decision support tools.

II.B. Spot Release Planner (SRP)

The SRP algorithm provides advisories to the ATCT controllers. The main idea is to provide spot release advisories to the ground controller (GC) in order to achieve a small queue at the runway resulting in overall reduction in movement area taxi times. The GC releases the aircraft from the spot at the advised time and is responsible for maintaining required separation on the taxiway.

The calculation of the optimal gate/spot release involves a two stage algorithm. In the first stage, an optimal runway schedule for the set of aircraft (take-off times for departures and crossing times for arrivals) is generated. For each aircraft, its weight class, and earliest available time at the runway are the main inputs to the stage 1 algorithm. The earliest available times at the runway are calculated by assuming that aircraft move with their maximum allowable speed on the taxiway. The optimization problem of this first stage can be formulated either as a mixed integer linear program or a dynamic program. To aid in our analysis, we formulate the first stage of SRP as a MILP to match the formulation of $\mathcal{TS}$. We will henceforth call this formulation $\mathcal{SRP}_1$.

`Formulation` $\mathcal{SRP}_1$:

$$min \quad \Gamma := \max_{i \in \mathbb{A}} t_{ir} \tag{11}$$

$$z_{ijr} + z_{jir} = 1 \qquad \forall i, j \in \mathbb{A} \tag{12}$$

$$z_{ijr}(t_{jr} - t_{ir}) \geq z_{ijr}\delta^r_{ij} \qquad \forall i, j \in \mathbb{A} \tag{13}$$

$$t_{ir} \geq a_i \qquad \forall i \in \mathbb{A} \tag{14}$$

$$z_{ijr} \in \{0, 1\} \qquad \forall i, j \in \mathbb{A}, i \neq j \tag{15}$$

$$t_{ir} \in \mathbb{R}^+ \qquad \forall i \in \mathbb{A} \tag{16}$$

where a_i is the earliest available time at the runway and is given by:

$$a_i = \alpha_i \qquad \forall i \in \text{arrivals} \tag{17}$$

$$= \alpha_i + \sum_{(u,v) \in R_i} \frac{l_{uv}}{V^{max}_{iuv}} \qquad \forall i \in \text{departures} \tag{18}$$

The second stage of the SRP determines optimal times to release aircraft from assigned spots to meet departure schedules calculated by $\mathcal{SRP}_1$. For purpose of this paper, we will calculate the spot time using Eq. (19),

$$t_{is} = t_{ir} - \tau_i, \qquad \forall i \in \text{departures} \tag{19}$$

where τ_i is the unimpeded taxi time for the i^{th} aircraft.

Similarly, the spot times for the arrivals are calculated using Eq. (20),

$$t_{is} = t_{ir} + \tau_i, \qquad \forall i \in \text{arrivals} \tag{20}$$

The calculation of spot times, t_{is}, in stage 2 does not consider the other aircraft on the airport surface. Consequently the release of the i^{th} departure at exactly t_{is} could lead to loss of separation on the airport surface. We assume that local conflicts for departures scheduled to leave the spot within the next few seconds can be resolved by the GC.

III. Analytical Results

The two stage SRP algorithm provides spot and runway-use times for the surface aircraft. The first stage of SRP (the runway scheduler) can be solved using the MILP formulation $\mathcal{SRP}_1$. It can also be efficiently solved using a dynamic program[4, 13] or heuristics.[2] This provides an advantage over the taxi scheduler $\mathcal{TS}$ that has significantly higher computation times. However, the calculation of spot times t_{is} in stage 2 does not consider the other aircraft on the airport surface, so the release of the aircraft at exactly t_{is} could cause conflict on the airport surface. By contrast, $\mathcal{TS}$ provides the time along all nodes for the aircraft and ensures that the times do not lead to separation loss. In the human-in-the-loop simulations[7] that used SRP, GC managed to release the aircraft safely without separation loss when they were provided with a 30-second window within which they had to release the aircraft. Motivated by this observation, we develop a method for comparing the two solutions and provide a metric to measure the difference between solutions of SRP and $\mathcal{TS}$. The details of the analysis are provided next.

Given a surface traffic instance, let $\Gamma^*_{\mathcal{TS}}$ be the optimal throughput value of the $\mathcal{TS}$ and $\Gamma^*_{\mathcal{SRP}_1}$ be the optimal throughput of the corresponding $\mathcal{SRP}_1$. We first make the following proposition.

Proposition 1. *$\Gamma^*_{\mathcal{SRP}_1}$ is a lower bound for $\Gamma^*_{\mathcal{TS}}$.*

Proof. Consider any feasible solution of $\mathcal{TS}$ for the given traffic instance. For each aircraft i a feasible solution is a sequence of times $t_{iu_0}, t_{iu_1}, ..., t_{iu_n}$ for the aircraft to be at successive nodes along its route. Given any such feasible solution for $\mathcal{TS}$, we can construct a feasible solution for $\mathcal{SRP}_1$ as follows: if i is a departure, then $t_{ir} = t_{iu_n}$, and if i is an arrival, then $t_{ir} = t_{iu_0}$. Constraint (13) of $\mathcal{SRP}_1$ is satisfied by this constructed solution since this constraint is the same as constraint (6) of $\mathcal{TS}$. Constraint (14) of $\mathcal{SRP}_1$ is similarly satisfied due to constraint (7) of $\mathcal{TS}$. Additionally, since $\mathcal{TS}$ and $\mathcal{SRP}_1$ have the same objective function, $\Gamma^*_{\mathcal{SRP}_1}$ will not exceed $\Gamma^*_{\mathcal{TS}}$. □

Let ($t_{ir}^{SRP^*}$, $z_{ijr}^{SRP^*}$) be an optimal solution for stage 1 of the SRP, and let $t_{is}^{SRP^*}$ be the corresponding solution from stage 2 of SRP. If we fix the corresponding variables in $\mathcal{TS}$ at the values obtained from SRP, $\mathcal{TS}$ may be infeasible because the solution may violate constraints (3), (4) and (5). However it may be possible to obtain a feasible solution for $\mathcal{TS}$ by perturbing $t_{is}^{SRP^*}$.

In our analysis we use the solution obtained at the runway from stage 1 of the SRP ($t_{ir}^{SRP^*}$, $z_{ijr}^{SRP^*}$) and fix the corresponding variables in $\mathcal{TS}$. We then constrain the spot time in $\mathcal{TS}$ using

$$t_{is}^{SRP^*} - \Delta \leq t_{iu_0} \leq t_{is}^{SRP^*} + \Delta \qquad \forall i \in \text{departures}$$

Similar constraints are put on arrivals for the spot entry times. We refer to formulation $\mathcal{TS}$ with these additional constraints as $\mathcal{TS}^{constrained}$. A feasible solution of $\mathcal{TS}^{constrained}$ is also be a feasible solution of $\mathcal{TS}$. The optimal value of $\mathcal{TS}^{constrained}$ is also an upper bound on the optimal value of $\Gamma^*_{\mathcal{TS}}$.

Definition 2. For a given Δ, if $\mathcal{TS}^{constrained}$ is feasible, then the solution obtained by the SRP is said to be *within* Δ of the optimal detailed airport surface solution ($\mathcal{TS}$).

Proposition 3. *If $TS^{constrained}$ is feasible, then $\Gamma^*_{\mathcal{SRP}_1}$ is the optimal value of $\mathcal{TS}$.*

Proof. From Proposition 1, $\Gamma^*_{\mathcal{SRP}_1} \leq \Gamma^*_{\mathcal{TS}}$. If $\mathcal{TS}^{constrained}$ is feasible, then the value of its objective function will be $\max_{i \in \mathbb{A}} t_{ir}^{SRP^*} = \Gamma^*_{\mathcal{SRP}_1}$. Since any feasible solution of $\mathcal{TS}^{constrained}$ is a feasible solution of $\mathcal{TS}$, it follows that $\Gamma^*_{\mathcal{SRP}_1} = \Gamma^*_{\mathcal{TS}}$. □

The time duration Δ can thus be used as a metric to measure the difference between solutions of $\mathcal{SRP}$ and $\mathcal{TS}$. By using Δ as the objective for $\mathcal{TS}^{constrained}$ we can calculate the minimum separation between solutions of $\mathcal{SRP}$ and $\mathcal{TS}$. The formulation $\mathcal{TS}^{constrained}$, thus modified, is given below:

$$\begin{aligned}
&\min \quad \Delta && && (21)\\
&z_{iju} + z_{jiu} = 1 && \forall i, j \in \mathbb{A}, \forall u \in R_i \cap R_j && (22)\\
&z_{iju} = z_{ijv} && \forall i, j \in \mathbb{A}, \forall (u, v) \in R_i \cap R_j && (23)\\
&z_{iju} = z_{ijv} && \forall (u, v) \in R_i \text{ and } (v, u) \in R_j && (24)\\
&z_{iju}(t_{ju} - t_{iu}) \geq z_{iju}\delta_{ij} && \forall i, j \in \mathbb{A}, \forall u \in R_i \cap R_j && (25)\\
&z_{ijr}(t_{jr} - t_{ir}) \geq z_{ijr}\delta^r_{ij} && \forall i, j \in \mathbb{A} && (26)\\
&t_{iu} + \frac{l_{uv}}{V^{max}_{iuv}} \leq t_{iv} \leq t_{iu} + \frac{l_{uv}}{V^{min}_{iuv}} && \forall i \in \mathbb{A}, \forall (u, v) \in R_i && (27)\\
&t_{iu_0} \geq \alpha_i && \forall i \in \mathbb{A}, u_{i0} = R_i.front() && (28)\\
&t_{ir} = t_{ir}^{SRP^*} && \forall i \in \mathbb{A} && (29)\\
&t_{is}^{SRP^*} - \Delta \leq t_{iu_n} \leq t_{is}^{SRP^*} + \Delta && \forall i \in \text{arrivals} && (30)\\
&t_{is}^{SRP^*} - \Delta \leq t_{iu_0} \leq t_{is}^{SRP^*} + \Delta && \forall i \in \text{departures} && (31)\\
&z_{iju} \in \{0, 1\} && \forall i, j \in \mathbb{A}, i \neq j, \forall u \in R_i \cap R_j && (32)\\
&t_{iu} \in \mathbb{R}^+ && \forall i \in \mathbb{A}, u \in R_i && (33)
\end{aligned}$$

IV. Simulation Results

In the previous section, we defined Δ as a measure of how far the SRP solution is from the optimal throughput solution. We also provided a formulation (Eqs. (21)-(33)) to calculate the minimum Δ for a given airport geometry and traffic scenario. Although the value of Δ would vary with airport geometry and traffic level, for a particular airport it may be possible to empirically calculate a maximum value of Δ over varying traffic levels. The rest of the section provides details of the Monte-Carlo simulation conducted to study the effect of traffic density on the value of Δ, and calculate the maximum separation between $\mathcal{SRP}$ and $\mathcal{TS}$.

In our simulation, we considered only the East side of the DFW airport operating in South Flow configuration (see Fig. 1). Of the 29 spots on East side of DFW we used only 24 spots, 12 for arrivals and 12 for departures. The airport surface is modeled as a directed graph. Departures originated at the spots, and depending on the spot and fix, they were assigned one of the three standard routes (Inner, Outer and Full Length) to the departure runway (17R). Runways 17C and 17L were used for arrivals. Arrivals on 17C took one of the four exits (M3, M4, M6, M7), depending on the spot they were going to. Similarly arrivals on 17L crossed 17R at ER. After crossing 17R, the arrivals taxied to the respective spots using pre-determined routes.

Simulations were carried out for traffic levels ranging from 20-44 aircraft, scheduled within 15 minutes. The traffic scenario levels were increased by increments of 2 aircraft for a total of 13 different levels. In each traffic level there were an equal number of arrivals and departures. Eighty percent of the arrivals were assigned to land on 17C, and 20% on 17L. The earliest available time to the crossings at 17R was one of the random parameters, and they were uniformly distributed between 0-900 seconds. Similarly, the spot arrival time for departures (α_i) were uniformly distributed in 0-900 seconds. The aircraft were randomly assigned a spot. Eighty percent of the aircraft were of weight-class $Large$, 10% were of class $Heavy$ and 10% were $B75x$. The wake-vortex separation between two departures are given in Table 1. Arrivals can cross 17R 40s after a departure and take 21 seconds to clear the runway. If two arrivals cross the runway consecutively, the temporal separation between them is 5s if they are at different crossings, or 20s if they are at the same crossing.

	Large	Heavy	B-75x
Large	61	109	91
Heavy	61	90	91
B-75x	61	109	91

Table 1. Wake vortex separation (in seconds) for departure aircraft.

For each traffic level, 1000 random scenarios were generated. The SRP algorithm was applied to each scenario and then the modified $\mathcal{TS}^{constrained}$ was solved for minimum Δ between $\mathcal{SRP}$ and $\mathcal{TS}$. Figure 2 provides the spread of Δ for each traffic level: the x-axis shows the different traffic level and the y-axis shows the value of six different statistical properties of Δ as follows:

- The bottom and top of the box are the 25^{th} and 75^{th} percentile.
- The band within the box is the median.
- The end of the whiskers depict the 10^{th} and 90^{th} percentile.
- The mean is shown using the marker inside the box.

Under low traffic density (20-22 aircraft in 15 minutes), the SRP solutions for 25% of the scenario (1^{st} quartile) are the optimal solution for $\mathcal{TS}$. With increasing traffic levels, SRP solutions for lesser percentage of scenarios are feasible solutions for $\mathcal{TS}$. As the traffic density increases, the spread of Δ gets smaller and all six statistical measures appear to increase and then level-off with increasing traffic levels. Even in very high traffic scenarios, the SRP solutions for 90% of the scenarios are within $7s$ of the optimal solution for $\mathcal{TS}$. Figure 3 plots the maximum recorded value of Δ for each traffic level. The maximum value for all traffic levels is observed to be below $14s$. For no case was $\mathcal{TS}^{constrained}$ infeasible.

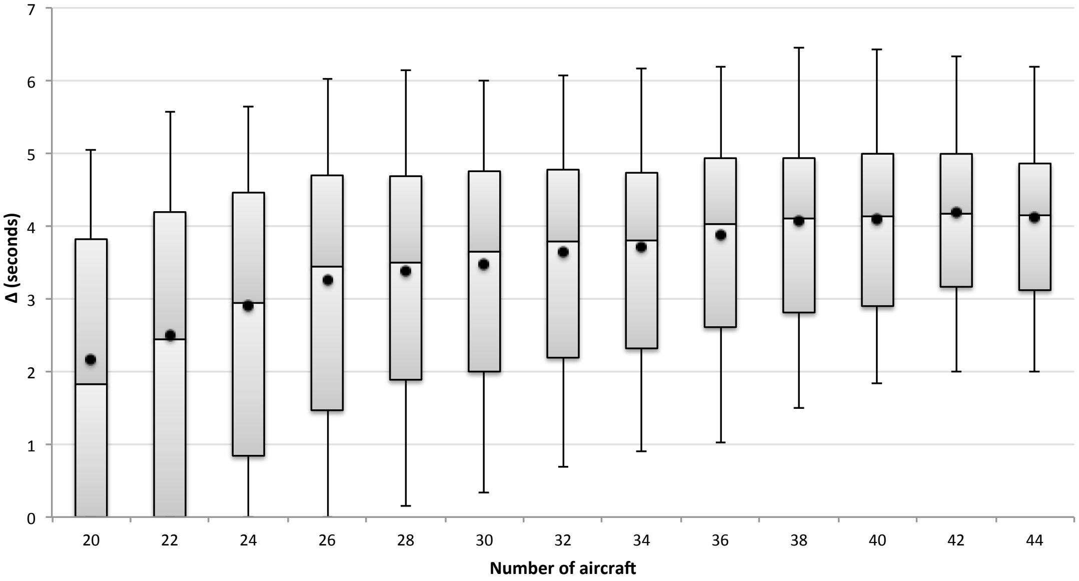

Figure 2. Statistical summary of Δ for varying traffic levels. The $10^{th}, 25^{th}, 50^{th}, 75^{th}, 90^{th}$ percentiles and mean for each traffic level are shown.

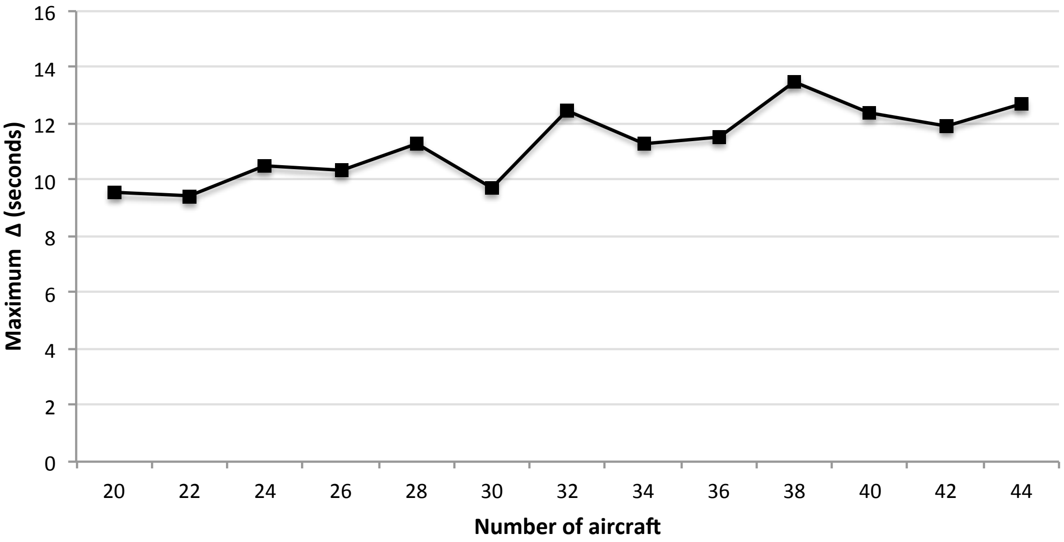

Figure 3. East-side DFW: routes under south flow configuration.

V. Conclusion

This paper is a comparative study of two previously developed algorithms for airport surface traffic, one being the detailed airport surface planner (Formulation $\mathcal{TS}$) and the other being a two stage heuristic SRP. A method to compare the solutions of the two formulations is developed, and a metric to measure the difference between solutions of $\mathcal{SRP}$ and $\mathcal{TS}$ is defined.

The value of Δ, metric for the quality of solution of $\mathcal{SRP}$, is expected to be dependent on airport geometry and traffic scenario. In this paper a Monte-Carlo simulation was conducted for the East side of Dallas/Fort Worth International Airport (DFW) with varying traffic levels, and it was shown that SRP solutions are *always* within $14s$ of the optimal solution for the detailed airport surface planner. For 90% of the scenarios under all traffic levels, allowing up to $7s$ variation in SRP calculated spot times, results in an optimal solution for the $\mathcal{TS}$. Moreover, in all 13000 scenarios tested, the objective value (throughput) obtained by SRP is the optimal throughput for the $\mathcal{TS}$.

The method developed in this paper will next be applied to other airports and the maximum value of Δ for different airports will be empirically calculated. Given the fast computation times of SRP,[11] if the value of Δ is less than some pre-specified value then SRP is an efficient heuristic for detailed airport surface traffic planning. The pre-specified limit for Δ could be based on numerous factors, including, but not limited to, human interaction aspects of the system and uncertainties in surface operations.

References

[1] Idris, H.R., Delcaire, B., Anagnostakis, I., Hall, W.D., Pujet, N., Feron, E., Hansman, R.J., Clarke, J.P. and Odoni, A. , "Identification of flow constraint and control points in departure operations at airport systems," *AIAA Guidance, Navigation and Control Conference*, Boston, USA, 1998.

[2] Dear, R. G., and Sherif, Y. S., "Dynamic scheduling of aircraft in high density terminal areas," *Microelectronics Reliability*, pp. 743-749, Vol. 29, No. 5, 1989.

[3] Psaraftis, H. N., "A Dynamic Programming Approach for Sequencing Groups of Identical Jobs," *Operations Research*, pp. 1347-1359, Vol. 28, No. 6, 1980.

[4] Balakrishnan, H., and Chandran, B., "Efficient and Equitable Departure Scheduling in Real-time: New Approaches to Old Problems," *7th USA - Europe Air Traffic Management Research and Development Seminar*, Barcelona, Spain, 2007.

[5] Rathinam, S., Wood, Z., Sridhar, B., and Jung, Y. C., "A Generalized Dynamic Programming Approach for a Departure Scheduling Problem," *AIAA Guidance, Navigation, and Control Conference*, Chicago, Illinois, 2009.

[6] Gupta, G., Malik, W., and Jung, Y. C. "A Mixed Integer Linear Program for Airport Departure Scheduling," *9th AIAA Aviation Technology, Integration, and Operations Conference (ATIO)*, AIAA, Hilton Head, South Carolina, 2009.

[7] Jung, Y., Hoang, T., Montoya, J., Gupta, G., Malik, W., Tobias, L. and Wang, H., "Performance evaluation of a surface traffic management tool for Dallas/Fort Worth International Airport," *Ninth USA/Europe Air Traffic Management Research and Development Seminar*, Berlin, 2011.

[8] Simaiakis, I., Balakrishnan, H., Khadilkar, H., Reynolds, T.G., Hansman, R.J., Reilly, B. and Urlass, S., "Demonstration of reduced airport congestion through pushback rate control," *Ninth USA/Europe Air Traffic Management Research and Development Seminar*, Berlin, 2011.

[9] Brinton, C., Provan, C., Lent, S., Prevost, T. and Passmore, S., "Collaborative departure queue management: an example of airport collaborative decision making in the United States," *Ninth USA/Europe Air Traffic Management Research and Development Seminar*, Berlin, 2011.

[10] Nakahara, A., Reynolds, T. G., White, T., Maccarone, C., and Dunsky, R., "Analysis of a surface congestion management technique at New York JFK Airport," *11th AIAA Aviation Technology, Integration, and Operations (ATIO) Conference*, Virginia Beach, VA, 2011.

[11] Malik, W. A., Gupta, G., and Jung, Y. "Managing departure aircraft release for efficient airport surface operations," *AIAA Guidance, Navigation, and Control Conference. American Institute of Aeronautics and Astronautics*, Toronto, Canada, 2010.

[12] Smeltink, J. W., Soomer, M. J., De Waal, P. R., and Van Der Mei, R. D., "An optimisation model for airport taxi scheduling," *Proceedings of the INFORMS Annual Meeting*, Denver, Colorado, USA, 2004.

[13] Rathinam, S., Montoya, J., and Jung, Y. C. , "An optimization model for reducing aircraft taxi times at the Dallas Fort Worth International Airport," *26th International Congress of the Aeronautical Sciences*, Anchorage, Alaska, USA, 2008.

[14] Balakrishnan, H., and Yoon, J., "A framework for coordinated surface operations planning at dallas-fort worth international airport," *AIAA Guidance, Navigation, and Control Conference*, Hilton Head, SC, USA, 2007, pp. 2382-2400.

[15] Visser, H. G., and Roling, P. C., "Optimal airport surface traffic planning using mixed integer linear programming," *AIAA Aviation Technology, Integration and Operations (ATIO)*, Denver, Colorado, 2003.

[16] Clare, G.L. and Richards, A.G., "Optimization of taxiway routing and runway scheduling," *IEEE Transactions on Intelligent Transportation Systems*, pp. 1000-1013, Vol. 12, No. 44, 2011.

[17] G. L. Nemhauser and L. A. Wolsey, "Integer and Combinatorial Optimization," *John Wiley & Sons*, 1988.