AIAA Aviation
13-17 June 2016, Washington, D.C.
16th AIAA Aviation Technology, Integration, and Operations Conference

AIAA 2016-4073

Runway Scheduling for Charlotte Douglas International Airport

Waqar A. Malik* and Hanbong Lee†
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

This paper explores a multiple runway scheduling problem and generates the schedule for arrivals, departures and crossing aircraft planning to use the runways within a given planning horizon. We present a mixed integer linear programming-based model that explicitly considers separation criteria between pairs of aircraft and also incorporates various Traffic Management Initiatives. It also includes constraints that arise due to airport layout. Additionally, we introduce an idea of *selective* Constrained Position Shifting (CPS), which limits the range of position an aircraft can hold in the runway schedule among a *subset* of flights. Constraints are included in the model to limit the relative sequence of the subset of departures under CPS. In 2014, this model was used in the NASA's Spot and Runway Departure Advisor human-in-the-loop simulations for Charlotte Douglas International Airport. In this paper, the presented model was tested in moderate and heavy surface traffic scenarios in a simulated environment, and results indicate an average improvement of 30% in cumulative delay over first-come-first-served.

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. Due to a lack of common situational awareness and limited data sharing among stakeholders, the Air Traffic Control Tower (ATCT) controllers manage departures in a mostly reactive manner. They release aircraft into the movement area based on the order that aircraft arrive at the spots and call for taxi clearance. In this First-Come First-Served (FCFS) paradigm, flights are motivated to pushback as early as possible to get an earlier slot at the runway and also to meet their on-time performance metric. Coupled with the fact that many flights have similar ticketed departure times, this leads to congestion on the airport surface. This congestion effect and the associated delays persist for a significant period, and often restrict an airport's throughput by hampering runway operations. The surface congestion also makes it difficult to accurately predict the take-off times of the departures. In Ref. 1, it is observed that a majority of airport surface delay was incurred at the runways.

To address this lack of efficiency and predictability in current departure operations, NASA, FAA, airport authorities and airlines have developed several surface management concepts and technologies. Over the last few years, NASA has developed the Spot and Runway Departure Advisor (SARDA) concept[2] as a Decision Support Tool (DST) for surface management. The SARDA concept was initially developed as an ATCT tool for the Dallas/Fort Worth International Airport (DFW) tower personnel. It provided runway usage advisories to the Local Controller and spot release advisories to the Ground Controller. This initial concept was demonstrated and evaluated in human-in-the-loop simulations with retired DFW tower controller participants in 2010[3] and 2012.[4-5] Recent SARDA research, in collaboration with American Airlines, has focused on non-movement (i.e., ramp) traffic advisories for the ramp controllers. This concept was evaluated in a human-in-the-loop simulation experiment in 2014[6] with current ramp-tower controllers from Charlotte Douglas International Airport (CLT) as participants. The simulation results showed that the tool helped reduce taxi time by one minute per flight and overall departure flight

*Research Scientist, University Affiliated Research Center, MS 210-8, Moffett Field, CA 94035.
†Associate Research Scientist, University Affiliated Research Center, MS 210-8, Moffett Field, CA 94035.
‡AIAA senior member & Aerospace Engineer, NASA Ames Research Center, MS 210-6, Moffett Field, CA 94035.

fuel consumption by 10-12% without reducing runway throughput. In these simulations, Expect Departure Clearance Time (EDCT) conformance was also improved when advisories were provided to the ramp controllers.

The SARDA concept incorporates three main algorithms: (1) a taxi-time estimator module for predicting when aircraft will arrive at the departure runway, (2) runway scheduling module for computing an efficient take-off schedule for the departures, and (3) spot release and/or gate pushback module for providing clearance advisories to the Ground and/or ramp controller, respectively. The runway scheduling module is the primary component of the SARDA concept since it determines the runway times that drive the computation of spot release or gate pushback times.

The runway scheduling algorithm computes the schedule for each aircraft to use the runway: the wheels-off times for departing aircraft, the wheels-on times for arriving aircraft and the crossing times for aircraft that need to cross an active runway. Moreover, the algorithm has to incorporate numerous physical and operational constraints, such as wake-vortex separation for successive departures, miles-in-trail restrictions over certain departure fixes, and time-window constraints for some aircraft. The algorithm also needs to be tailored to each airport as it depends on the layout of the runway system: the operations on parallel runways (that are close together) or intersecting runways must be coordinated and the actual separation requirements depend on the exact layout and the use of the runways.

The runway scheduling problem is structurally equivalent to a job shop scheduling problem.[7-9] The runways represent the *machines*, and the aircraft represent the *jobs*. The required separation times between pairs of aircraft on the same runway are the (sequence dependent) *processing times*. The earliest possible time an aircraft can use the runway represents the release time of the job and the latest, the due date. A common objective is to minimize the completion time (runway-use time) of the last job, which is equivalent to maximizing throughput. Hence, many of the solution techniques commonly used for solving the job shop scheduling problems have been adapted to the runway scheduling problem: e.g., mixed integer linear programs,[15] branch and bound, branch and cut, dynamic programming,[10-14] heuristics, metaheuristics,[16] and others. The sequence dependent job shop scheduling problem is strongly NP-hard, and consequently, it is not expected to find polynomial time algorithms for the runway scheduling problem.

The majority of the prior papers on runway scheduling have looked at subsets of the general runway scheduling problems: the single runway scheduling for arrivals (arrivals scheduling problem) or departures (departures scheduling problem). Researchers have also made several simplifications to the problem to make it computationally tractable. In Ref. [9], the researchers have relaxed the wake turbulence separation criteria, and scheduling between successive aircraft is based on a constant separation time rather than separation based on aircraft weight class. Researchers have also proposed the idea of constrained position shifting[10] that limits the range of positions an aircraft can occupy in the runway sequence. This reduces the available solution space and leads to computationally tractable solutions.

This paper describes the runway scheduler that was used in the 2014 SARDA human-in-the-loop simulations[6] for CLT. The algorithm considers multiple runways and computes the optimal runway times for departures and arrivals. The paper also provides results for the proposed scheduler from standalone simulations and evaluates the effect of maximum position shift parameter on solution quality and computation times.

II. Problem Setup

In this paper, we consider the problem of scheduling aircraft on multiple runways at CLT. The modeling of the runway scheduling procedure depends on various factors, such as the taxiway layout, availability of holding areas, number of spots and runway configuration. In this section, we provide an overview of CLT operations and describe various separation and time-window restrictions imposed on the aircraft.

A. Characteristics of Surface Traffic at CLT

Situated between the Washington metroplex (300 nm away) and Hartsfield-Jackson Atlanta International Airport (ATL) (~200 nm away), CLT underlies one of the busiest air traffic corridors on the east coast. CLT is located in the northeast corner of Atlanta Center's (ZTL) airspace, ~18 miles from the Center's boundary with Washington Center (ZDC), which significantly influences operations at CLT and imposes several traffic flow constraints. CLT Tower controls around 1,600 operations per day, and provides nonstop service to over 150 destinations. Based on the total passenger count CLT ranks as 8th in the US and 23rd in the world. The distribution of CLT traffic operations by carriers shows that American Airlines (AA) and its affiliated regional air carriers operate nearly 90% of the flights at CLT. Besides the main terminal for commercial and regional airlines, CLT also has the Wilson Air Center for fixed base operators (corporate and private flights), the North Carolina Army Guard and the North Carolina Air National Guard. These general aviation and military flights comprise about 5% of CLT traffic.

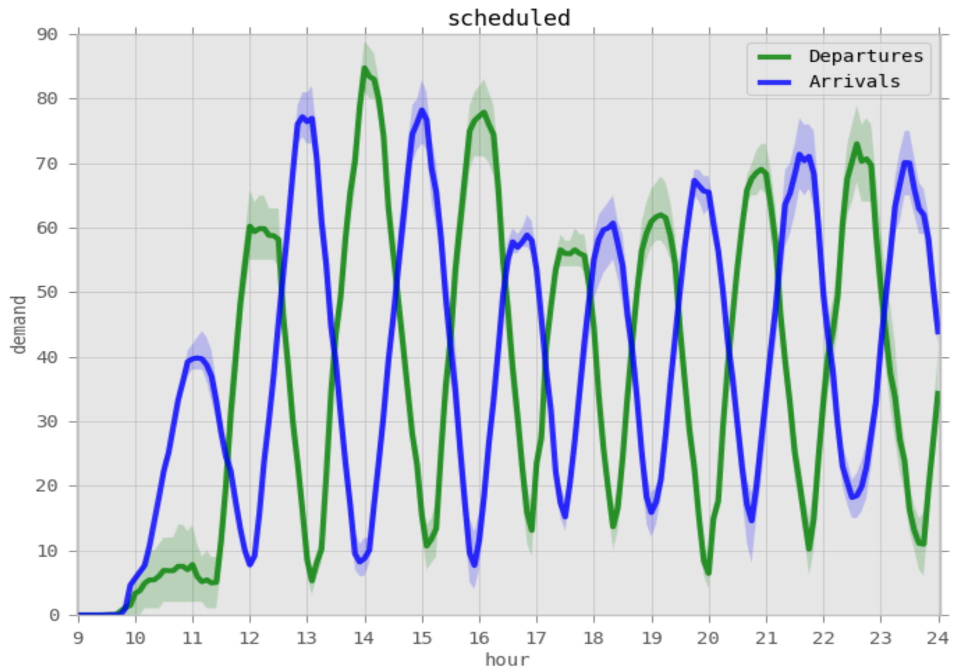

Figure 1. Scheduled-Out and Scheduled-In demand for departures and arrivals.

Figure 1 shows the scheduled gate-in demand for arrivals (blue) and the scheduled gate-out demand for departures (green). The lines show the average values for March 2014 and the shaded areas around the demand lines depict the standard deviations over the whole month. The figure illustrates the definite peaks and valleys that characterize the traffic at CLT. This peak schedule frequently leads to congestion on the airport surface and large queues at the runways. During these departure peaks, the CLT ramp tower employs a departure sequencing procedure wherein they hold departure aircraft at the gate in order to limit the number of departures taxiing to each runway and hence reduce congestion on the airport surface, improve operational efficiency and reduce fuel consumption.

B. CLT Airport Layout

As shown in Figure 2, the airport has three north/south parallel runways (18L/36R, 18C/36C, and 18R/36L) that support simultaneous independent instrument approaches, and a fourth diagonal runway (5/23) that intersects runway 18L/36R.

Runway 18R/36L was commissioned in 2010 and is primarily used for arrivals. Runway 5/23 is used for arrivals in South Flow Configuration and is used as a taxiway during North Flow Configuration. However, due to noise abatement procedures, it is the only runway in use between 2300 and 0700 hours. Runway 18C/36C is a mixed-use runway whereas runway 18L/36R is used primarily for departures. The current operations capacity rate range in visual conditions is 176-182 operations per hour. Annually, for approximately 80% of the time the weather at CLT allows for visual approaches.

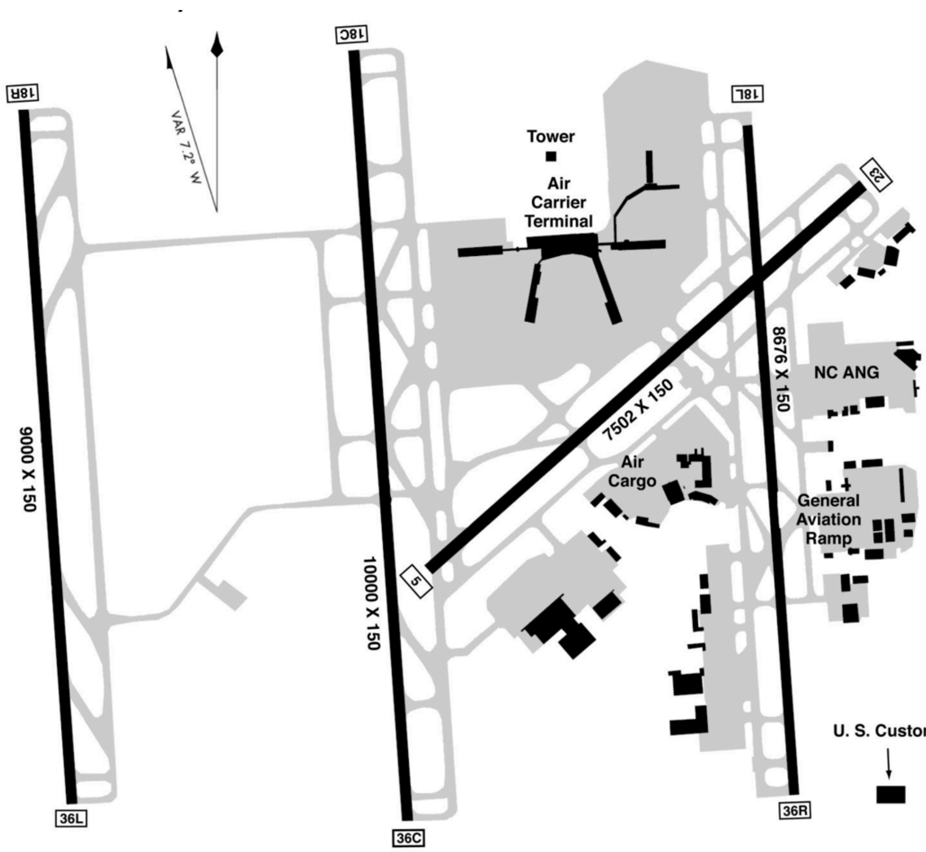

Figure 2. CLT Airport Diagram.

Since March 2015, CLT has implemented the Wake Turbulence Re-categorization[18] (Wake RECAT) that allows for reduction in wake separation standards between certain aircraft. This reduced separation between departures can possibly help by providing increased capacity at the runway, but it imposes new challenges since it also increases the arrival capacity. Since CLT has a very complex layout with limited ramp space, handling a procedure, such as Wake RECAT, that puts more aircraft on the taxiways (through increased arrival capacity) can be a challenge.

CLT operates in either North or South Flow Configuration depending on the primary traffic flow direction. When operating in South Flow Configuration, converging runway operations are normally used. For these converging runway operations, Tower Local controllers are required to adhere to the Arrival/Departure Window (ADW) procedure that restricts departures on runway 18C when there is an arrival on final segment and within a distance of 1.8 nm from the runway 23 threshold. Since runway 23 also intersects with runway18L, there is an indirect coupling of runways 18C and 18L. Moreover, 18L may sometimes be used for departures going to a fix normally associated with runway18C (and vice versa), and this requires direct coordination between the operations on runways 18C and 18L.

C. CLT Surface Constraints and Traffic Management Initiatives (TMIs)

As mentioned in the previous subsection, CLT has implemented the Wake RECAT procedure since March 2015. Wake RECAT is a new FAA procedure that separates aircraft based on their wake profile instead of weight alone. Under Wake RECAT there are six categories (A, B, C, D, E and F) of aircraft for wake turbulence separation purposes. The categories separate the current heavy and large weight classes into four wake categories (two each for heavy (B, C) and large (D, E)). The super heavy aircraft like A388 and A225 are put into a distinct wake category (A), and the current weight class of small remains as its own wake category (F). The wake turbulence separation for departures on the same runway is given in Table 1. Since, Wake RECAT separations was implemented in CLT in March 2015, we used the previous wake-vortex separations (based on four weight-classes) for the 2014 SARDA HITL[6] and the results in this paper. The algorithm proposed in this paper uses the wake-vortex separations as input and future simulations should use the separations provided in Table 1.

Table 1. Wake turbulence separation for departures on the same runway

		Follower					
		A	B	C	D	E	F
Leader	A	.	5 NM	6 NM	7 NM	7 NM	8 NM
	B	.	3 NM	4 NM	5 NM	5 NM	5 NM
	C	.	.	.	3.5 NM	3.5 NM	5 NM
	D	.	.	.	.	.	5 NM
	E	.	.	.	.	.	.
	F	.	.	.	.	.	.

When the "converging runway operations" is in effect, the ADW procedure defines a No Departure Zone (NDZ) from Runway 23 threshold to 1.8 nm out (see Figure 3). Departures on 18C have to start takeoff roll prior to an arrival entering NDZ. This minimizes the risk of separation loss with the departing aircraft in the event the arrival executes a missed approach. The ADW constraint can be expressed as a temporal separation requirement between departures on 18C and arrivals on 23.

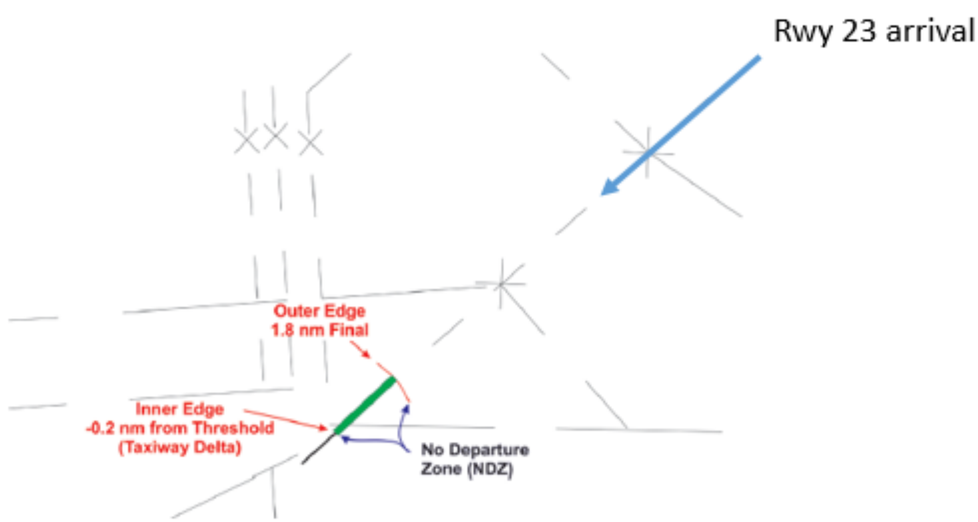

Figure 3. Arrival Departure Window (ADW) for Runway 23.

In addition to the required wake turbulence separation constraints, aircraft may also be restricted by various Traffic Management Initiatives (TMIs). In CLT, there are three main kinds of TMIs that are frequently used. These TMIs can be both strategic and tactical in nature and are normally used to resolve capacity constraints in downstream enroute airspace or at destination airports.

- Expected Departure Clearance Times (EDCTs) are strategic TMIs issued by the Air Traffic Control System Command Center and are used to resolve imbalances in the NAS that are remote from the departure airport. These are provided in the form of controlled take-off times with a +/- 5-minute compliance window.
- Call for Release procedures (CFRs), also called Approval Request or APREQ, are tactical TMIs issued by the Air Route Traffic Control Center (ARTCC or Center) Traffic Management Coordinator (TMC) and are used to manage imbalances locally. They are designed to coordinate the departures' release times from the airport to facilitate overhead stream insertion or merging of traffic at downstream schedule points. They have a take-off time compliance window of -2/+1 minutes.
- Miles-In-Trail (MIT) restrictions are used to reduce the volume of incoming traffic, usually at Center boundaries. They are specific to route, meter points and destination. They are normally issued by the Center TMC and specify a spacing requirement (in miles) between aircraft departing an airport, over a fix, at an altitude, through a sector, or on a specific route.

III. Multiple Runways Scheduling Algorithm

The interdependency of runway operations at CLT requires a runway scheduling algorithm that incorporates the constraints between aircraft using the same runway, as well as considers the effect of the aircraft operations on other runways. In order to obtain an optimal solution for the airport runway operations it is necessary to formulate an algorithm that includes operations at all the runways. Whereas previous studies have only formulated and solved the problem for operations on a single runway, the algorithm described in this paper finds schedules for multiple runways simultaneously while incorporating additional constraints imposed due to the interdependency between the runways.

A. Scheduler Inputs

The scheduler takes as input the current snapshot of the airport traffic, aircraft specific parameters, separation constraints, scheduled pushback times and scheduled arrival times for the aircraft in the next 15 minutes. Based on availability of surveillance and ramp controller input, the aircraft are assigned into different groups. The departures are divided into five groups: (a) *scheduled_out* (aircraft whose scheduled pushback time is within 15 minutes, but the pilot has not called for pushback), (b) *pushback_hold* (the pilot has called for pushback, but the ramp controller has put the aircraft on hold), (c) *pushback_approved* (the ramp controller has provided pushback clearance, but aircraft has no surveillance hits), (d) *taxi_out* (the ramp controller has issued taxi clearance and/or the aircraft is under surveillance), and (e) *unknown*. The arrivals are divided into four groups (a) *scheduled_on* (beyond surveillance range), (b) *airborne* (on final approach segment with surveillance), (c) *taxi_in* (on airport surface and has not reached gate), and (d) *unknown*.

The scheduler also receives a nominal/unimpeded runway entry time and/or runway crossing queue entry time for each aircraft. The scheduler computes the required separation between each pair of aircraft. This consists of a pair of values depending on the runway use sequence and takes into account the most restrictive separation as the required value. The separation criteria considers the wake turbulence separation, separation between aircraft going to the same departure fix, miles-in-trail separation, Arrival Departure Window (ADW) separation, converging runway separation, mixed (takeoff/landing) runway separation, crossing-takeoff separation, crossing-landing separation, and parallel runway separation. The EDCT and CFR/APREQ times for relevant aircraft are also provided as constraints.

B. Runway Scheduler

The runway scheduler for the simulation is implemented as a Mixed Integer Linear Program (MILP) and solves for the optimal system delay using the commercial solver Gurobi.[17] The *scheduled_out* departures are not considered in the MILP planner. Once the MILP provides an optimal runway sequence, empty slots are found in the sequence and *scheduled_out* aircraft are assigned to these slots as long as this assignment does not cause a change in the time for the other aircraft following the inserted aircraft.

Uncertainties in aircraft movement pose a challenge to generating controller advisories. To mitigate the effect of uncertainties, the scheduler gets an updated airport condition snapshot every 10 seconds, which is then used to recalculate the schedule. To improve the stability of the advisories and prevent frequent changes in advisories, a two-minute "freeze window" is implemented.

The core component of the scheduler is a Mixed Integer Linear Program that solves for the entire airport runway system. In our algorithm, we consider the arrival landing time to be a fixed value and plan departures around the arrivals. Arrivals can interact with the departure aircraft in three ways: (a) as part of mixed runway operations, arrivals can land on the same runway from where departures take off (runway 18C in CLT), (b) they could land on a completely separated runway and then cross an active departure runway (arrivals landing on 18R and crossing 18C), or (c) land on a converging runway that effects the departure operations on other runways (arrivals landing on runway 23 impose restrictions on departures on both 18C and 18L).

Let R_1 and R_2 be the two primary runways for which a schedule is generated. Let A_{R_1}, D_{R_1} and C_{R_1} be the set of arrival, departure and crossing aircraft planning to use R_1. Runway R_2 is a departure only runway and let D_{R_2} be the set of departures planning to use R_2. Let R_3 be a diagonal arrival runway that has converging operations with departures on both R_1 and R_2. Let A_{R_3} be the set of arrivals planning to use runway R_3. Let F be the set of all flights in the planning horizon, $F = A_{R_1} \cup D_{R_1} \cup C_{R_1} \cup D_{R_2} \cup A_{R_3}$. For each flight, the earliest available time at the runway is known. Let this time be α_i for all aircraft $i \in F$. For arrivals this is the landing time, for crossing aircraft it is the time the aircraft can start the crossing operation, and for departure it is the earliest take-off time. Furthermore, for a subset of departure flights, a TMI in the form of EDCT or CFR may exist which restricts the departure to depart within a time window $[TMI_L, TMI_H]$. Let the set of TMI aircraft be denoted by TW. For each departure aircraft in the set F, its weight class and departure fix are known and govern the separation between leading and trailing aircraft.

For the majority of departures, the take-off times are independent of operations on the other runway. A departure aircraft on R_1 influences departures on R_2 only if it is going to a departure fix that is normally associated with departures on R_2, and vice versa. Let these aircraft belong to the set $D_{R_{1\to 2}}$ and $D_{R_{2\to 1}}$, respectively. Let $\mathbb{F}$ describe the set of ordered pairs of aircraft that interact with each other. These are the pairs of aircraft for which safety separations need to be defined. $\mathbb{F}$ is defined as,

$$\mathbb{F} = (A_{R_1} \times D_{R_1}) \cup (D_{R_1} \times A_{R_1}) \cup (A_{R_1} \times C_{R_1}) \cup (C_{R_1} \times A_{R_1}) \cup (D_{R_1} \times C_{R_1}) \cup (C_{R_1} \times D_{R_1}) \cup (C_{R_1} \times C_{R_1}) \cup (A_{R_3} \times D_{R_1}) \cup (D_{R_1} \times A_{R_3}) \cup (D_{R_2} \times D_{R_2}) \cup (D_{R_2} \times A_{R_3}) \cup (A_{R_3} \times D_{R_2}) \cup (D_{R_{2\to 1}} \times D_{R_1}) \cup (D_{R_1} \times D_{R_{2\to 1}}) \cup (D_{R_{1\to 2}} \times D_{R_2}) \cup (D_{R_2} \times D_{R_{1\to 2}})$$

There are various separation requirements between any two pairs of aircraft. At any time, only one aircraft can occupy the runway. Between two departures we have the required wake-vortex separation, a reduced separation in case of divergent heading, or an increased separation due to MIT constraints. There are separation requirements between all combinations of arrivals, departures and crossings. Since we consider the arrival aircraft times to be hard constraints, we do not have to impose inter-arrival separations. They are implicitly dealt with as inputs. Let $\Delta_{i,j}$ be the safety separation time for any aircraft pair $(i,j) \in \mathbb{F}$.

Decision Variables:

- For each aircraft $i \in F$, let t_i denote the calculated time at which the aircraft uses the runway (take-off, land or cross).
- Let $Z_{i,j}$ $\forall (i,j) \in \mathbb{F}, i \neq j$ be a binary variable that specifies the relative order of runway use.
  - $Z_{i,j} = \begin{cases} 1 & \text{if aircraft } i \text{ uses the runway before aircraft } j \\ 0 & \text{otherwise.} \end{cases}$

Cost Function:

- Reference [15] shows that optimizing for total delay results in small deviations from the optimal throughput, whereas optimizing for throughput results in large deviations in total delay. For this reason, total delay was chosen as the objective for the scheduler.

$$\text{minimize} \sum_{i \in F} (t_i - \alpha_i)$$

Constraints:

1. Linear ordering constraints, i.e. given any two aircraft, at least one leads the other in runway use.
$$Z_{i,j} + Z_{j,i} = 1, \quad \forall\, (i,j) \in \mathbb{F}, i \neq j$$

2. Aircraft can use the runways only at or after the corresponding earliest available time.
$$t_i \geq \alpha_i, \qquad \forall\, i \in F$$

3. The arrival landing time cannot be changed. A small value of δ is used for numerical reasons (feasibility), and to account for variations in touchdown prediction times.
$$t_i \leq \alpha_i + \delta, \qquad \forall\, i \in A_{R_1} \cup A_{R_3}$$

4. For TMI aircraft, constrain the departure time of the aircraft within the specified time window.
$$TMI_L \leq t_i \leq TMI_H, \qquad \forall\, i \in TW$$

5. Ensure separation when aircraft i uses the runway before aircraft j.
$$Z_{i,j}\left(t_j - t_i - \Delta_{i,j}\right) \geq 0, \qquad \forall\, (i,j) \in \mathbb{F}$$

6. Two crossing aircraft use the runway in the order of their First Come First Served (FCFS) sequence.
$$Z_{i,j} = 1, \text{if } \alpha_i < \alpha_j, \qquad \forall\, (i,j) \in (C_{R_1} \times C_{R_1})$$

7. Two MIT aircraft are sequenced in FCFS order.
$$Z_{i,j} = 1, \text{if } \alpha_i < \alpha_j, \qquad \forall\, i,j \text{ under MIT constraints.}$$

8. Let $(a_{R_i}^1, a_{R_i}^2, \dots, a_{R_i}^{m_i})$ be a sequence of departures in D_{R_i} excluding the TMI and MIT aircraft. Let this sequence be ordered by the earliest available time. We can impose a Constrained Position Shifting (CPS) constraint on these aircraft that limits the number of positions the aircraft can occupy in this sequence. Let MPS be a parameter specifying the maximum number of position shifts. For all positions $p = 1, \dots, m_i$, the CPS imposes a relative order given by,

$$Z_{a_{R_i}^p, a_{R_i}^{p+k}} = 1, \forall\, k \geq MPS$$
$$Z_{a_{R_i}^p, a_{R_i}^{p-k}} = 1, \forall\, k \geq MPS$$

The constraints above define the *core* MILP used to solve the airport optimization problem. The following are additional constraints for operational procedures at CLT airport.

- For two departures i, j in the Active Movement Area (AMA) and planning to use runway 18L via queue C12, the relative departure sequence is fixed if the aircraft have crossed C11 threshold. Once the aircraft have crossed taxiway C11, the taxiway layout does not allow the aircraft sequence to change. In certain extreme situations, involving TMIs/MIT, the controllers may move the aircraft across the runway via taxiway D to D8 and then back-taxi on the runway 18L for take-off (See Figure 4).
- For two departures to runway 18L, the departure at Spot 12 cannot cut in front of an immediate aircraft on taxiway C ready to turn onto C12. The aircraft on taxiway C is already committed to move onto the runway.
- For two departures (non-TMI) in the ramp planning to use runway 18L via Spot 12, the relative sequence is fixed dependent on the distance from Spot 12. The scheduler does not fix the relative sequence for TMI aircraft as the controllers would normally hold the TMI aircraft on the taxiway D/D8 to the East of 18L until required by the TMI constraint.
- For two non-TMI departures to 18C in the AMA, the relative sequence is fixed depending on the distance from the runway. The controller/scheduler should be able to re-sequence aircraft around TMI aircraft.
- For two departures on runway 18C, the departure sequence is fixed once the aircraft crosses E10.

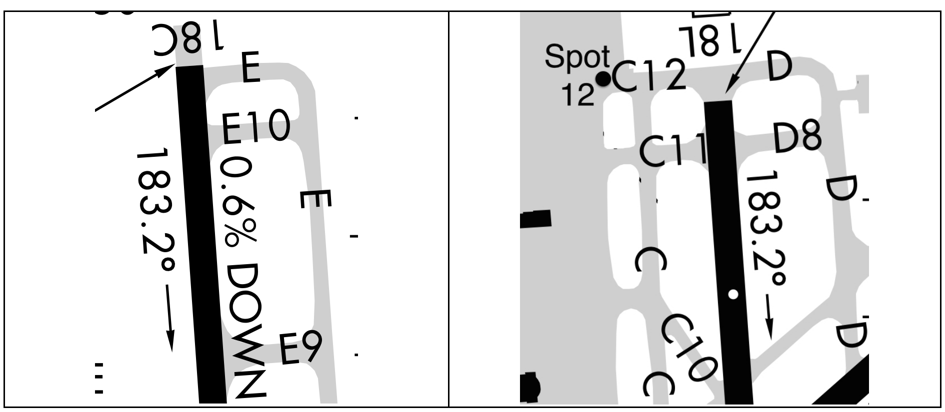

Figure 4. Taxiway constraints near the runway threshold at runways 18C (left) and 18L (right).

In the SARDA simulation[6], this MILP was started with a modified FCFS solution, and was constrained to provide a solution within 7 seconds, though in most cases the computations were completed much faster. The faster computation times were mostly due to the additional constraints that impose precedence constraints on the aircraft, thereby reducing the search space.

The MILP scheduler calculates the runway times for the given group of aircraft. Since *scheduled_out* departures were not considered in the MILP formulation, empty slots are found in the calculated schedule and *scheduled_out* aircraft are assigned to these slots as long as the assignment does not cause a change in the time for the other aircraft following the inserted aircraft. Aircraft already scheduled in the MILP, but held at the gate (i.e., aircraft in group *pushback_hold*) are also considered for open slots. If such a move is available, these aircraft might get an earlier slot at the runway.

IV. Results

The multiple runways scheduling algorithm described in this paper has been used in the SARDA human-in-the-loop simulation[6] and demonstrated that the pushback advisories reduced the departure taxi delay by one minute per flight, and decreased fuel consumption in departure flights by 10-12%. The tool successfully reduced the runway queue length, improved EDCT conformance and reduced tower controllers' self-reported workload.

In this paper, the results from the runway scheduling using the *core* MILP formulation are compared with a FCFS solution to examine the benefits of the proposed algorithm. This work was conducted in standalone simulations. The addition of landing aircraft and/or departures with time-window restrictions makes the computation of the FCFS sequence tricky, since sorting by the earliest available times may give rise to an infeasible sequence for arrivals and departures with time-window restrictions. In this case, a FCFS solution is generated by first considering the time-window constrained aircraft and assigning them a runway use time. The other aircraft are then sorted and sequentially inserted, in ascending order, into the solution to use the first available slots while ensuring that it does not cause any conflicts with the aircraft previously considered in the solution. The MILP was solved using Gurobi,[17] a commercially available optimization software package.

In our simulations, we considered a mixed operation runway with arrivals, departures and crossing traffic. Another stream of arrivals was modeled to simulate converging runway operations. The planning window was set to 15 minutes. The earliest available times (α_i) were randomly chosen from a uniform distributed in 0-900 seconds. The departure aircraft types were randomly assigned to provide a mix with 80% of weight-class Large, 10% of Heavy and 10% of B757 class. The departures were also randomly assigned a departure fix from six discrete choices.

Table 2. Wake vortex separation (in seconds) for departure aircraft.

	Heavy	Large	B757
Heavy	75	90	90
Large	38	38	38
B757	75	75	75

The wake vortex separation matrix, given in Table 2, was used for the simulations. The columns represent the weight class for following aircraft whereas the rows represent the leading aircraft. For example, if a large aircraft follows a heavy aircraft, then they must be separated by 90 seconds. Besides the wake vortex separations between departures provided in Table 2, there are additional separations between different operation types. Aircraft can cross the runway 42 seconds after a departure and take 15 seconds to clear the runway. If two arrivals cross the runway consecutively, the temporal separation between them is 5 seconds. A departure cannot be scheduled on the runway if the arrival on the converging runway is within 50 seconds of its landing time. Similarly, a departure cannot be scheduled on the runway if the arrival on the same runway is within 40 seconds of its landing time. After an arrival has landed it takes 10 seconds to clear the runway. Moreover, two departures going to the same fix have to separated by at least 60 seconds. These separation values were tuned for the human-in-the-loop simulation environment[6] conducted in 2014.

A 15-minute planning horizon was considered for each scenario and the number of aircraft in the scenarios was varied from 10 to 35 in increments of 5, which represent moderate to congested traffic conditions. For each case (number of aircraft), 100 different random instances were generated. Each aircraft in the scenario were randomly assigned an earliest available time, weight class and departure fix (for departures). Sixty percent of the traffic was chosen to be departures, 20% arrivals and 20% crossing aircraft.

The MPS parameter in Constraint 8 was varied from 0 to 3. Figure 5 shows the average improvement in total delay over FCFS for 100 runs for each traffic level. A MPS of 0 implies that the relative departure sequences are fixed. A 5-30% improvement in this case is achieved only through proper sequencing of crossing aircraft by enabling multiple crossings to occur simultaneously. For each MPS value, Fig. 5 shows an increase in average improvement in total delay over FCFS solution with increasing traffic levels. These increases can be attributed to the larger number of aircraft, especially crossing aircraft, providing a better opportunity to sequence the aircraft. The figure also shows that as the value of MPS increases, the average improvement in total delay also increases. The average improvement when the value of MPS is increased from 2 to 3 is only marginal. Note that the CPS scheme is applied to departures only, and the arrival landings and crossings are not constrained by the MPS parameter.

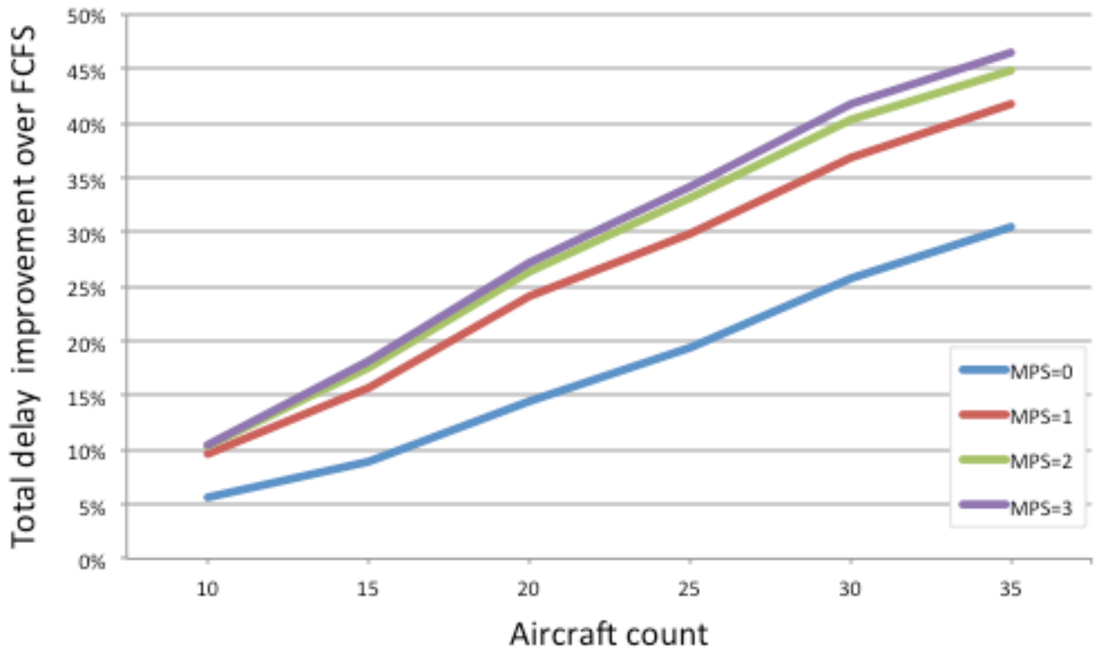

Figure 5. Total delay improvements over FCFS for 10 to 35 aircraft.

Figure 6 shows the average computation time for varying traffic level and different values of the MPS parameter. An increase in computation times is observed with increasing traffic level. Moreover, as the MPS parameter increases, the computation time also increases. These increases can be attributed to a larger search space with increase of these parameters. When the value of MPS parameter is increased from 2 to 3, there is a substantial increase in computation time, whereas the average improvement in total delay is only marginal. Hence, a MPS value of 2 indicates a good trade-off between solution quality and computation times.

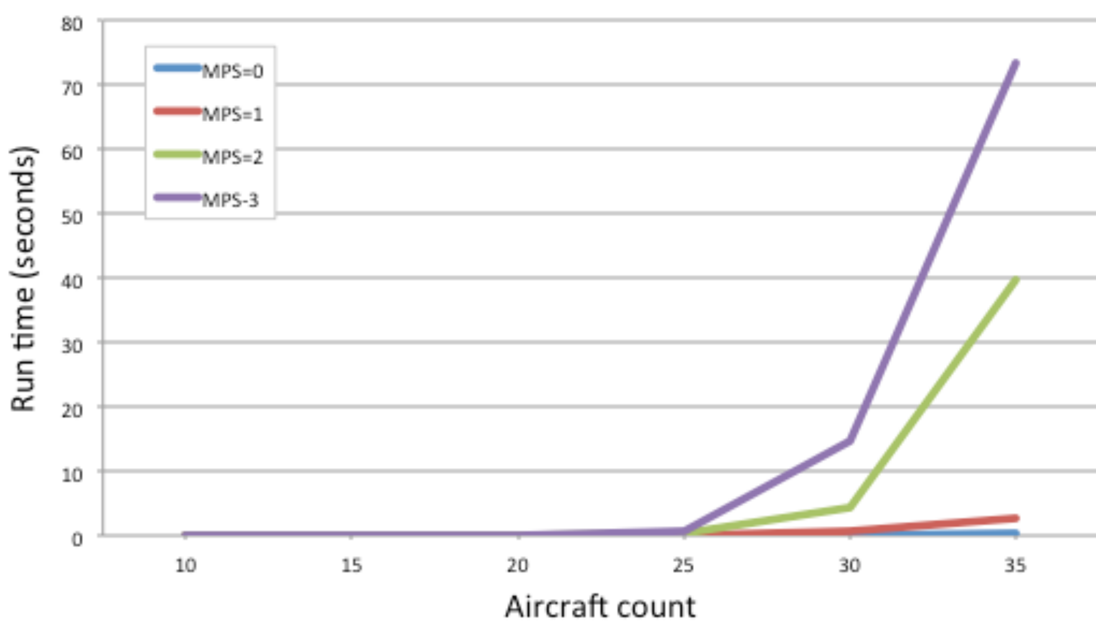

Figure 6. Average computation times (in seconds) for varying traffic levels and different MPS values.

V. Conclusion

This paper provides an overview of surface operations at CLT and presents a mixed integer linear program (MILP) for multiple runway scheduling. The multiple runway scheduling algorithm described in this paper has previously been used in the SARDA human-in-the-loop simulation.[6] The MILP considers various separation criteria along with additional operational constraints such as Arrival Departure Window (ADW) constraint for converging runway operations. It also considers various Traffic Management Initiatives, such as Call For Release (CFRs), Expected Departure Clearance Times (EDCTs) and Miles-in Trail (MIT) constraints. Simulation results indicate substantial benefits over a first-come-first-serve solution. The algorithm selectively applies Constrained Position Shifting (CPS) constraints to only a subset of aircraft, and the results indicate a maximum position shift (MPS) parameter value of 2 as a good trade-off between solution quality and computation times.

References

[1]Idris, H., Delcaire, B., Anagnostakis, I., Hall, W. D., Pujet, N., Feron, E., et al., "Identification of Flow Constraint and Control Points in Departure Operations at Airport Systems," *AIAA Guidance, Navigation, and Control Conference and Exhibit*, Boston, MA, August 10-12, 1998.

[2] Malik, W., Gupta, G., and Jung, Y., "Managing Departure Aircraft Release for Efficient Airport Surface Operations," AIAA Guidance, Navigation, and Control (GNC) Conference, Toronto, Canada, August 2010.

[3]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," the Ninth USA/Europe Air Traffic Management Research and Development Seminars, Berlin, Germany, June 2011.

[4]Gupta, G., Malik, W., Tobias, L., Jung, Y., Hoang, T., and Hayashi, M., "Performance Evaluation of Individual Aircraft Based Advisory Concept for Surface Management," the Tenth USA/Europe Air Traffic Management Research and Development Seminars, Chicago, IL, June 2013.

[5]Hayashi, M., Hoang, T., Jung, Y., Gupta, G., Malik, W., and Dulchinos, V., "Usability Evaluation of the Spot and Runway Departure Advisor (SARDA) Concept in a Dallas/Fort Worth Airport Tower Simulation," the Tenth USA/Europe Air Traffic Management Research and Development Seminars, Chicago, IL, June 2013.

[6]Hayashi, M., Hoang, T., Jung, Y., Malik, W., Lee, H., and Dulchinos, V., "Evaluation of Pushback Decision Support Tool Concept for Charlotte Douglas International Airport Ramp Operations," the Eleventh USA/Europe Air Traffic Management Research and Development Seminars, Lisbon, Portugal, June 2015.

[7]Bianco, L., Rinaldi, G., Sassano, A., delle Ricerche, C.N. and Manzoni, V., "A Combinatorial Optimization Approach to Aircraft Sequencing Problem," *Flow Control of Congested Networks*, edited by A. R. Odoni, L. Bianco, and G. Szego¨, NATO ASI Series, Series F: Computer and Systems Science 38, Springer-Verlag, Berlin, (1987), pp. 323–339.

[8]Bianco, L., Ricciardelli, S., Rinaldi, G. and Sassano, A., "Scheduling Tasks with Sequence-Dependent Processing Times," *Naval Research Logistics*, Volume 35, Number 2, 1988, pp.177–184.

[9]Bianco, L. and Bielli, M., "System Aspects and Optimization Models in ATC Planning, "*Large Scale Computation and Information Processing in Air Traffic Control*, edited by L. Bianco and A. R. Odoni, Springer-Verlag, Berlin, 1993, pp. 47–99.

[10]Dear, R. G., and Sherif, Y. S., "Dynamic Scheduling of Aircraft in High Density Terminal Areas," Microelectronics Reliability, Vol. 29, No. 5, 1989, pp. 743-749.

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

[12]Balakrishnan, H., and Chandran, B., "Scheduling Aircraft Landings Under Constrained Position Shifting," *AIAA Guidance, Navigation and Control Conference and Exhibit*, Keystone, CO, August 2006.

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

[14]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, IL, 2009.

[15]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, SC, 2009.

[16]Atkin, J.A.D., Burke, E.K., Greenwood. J.S., and Reeson, D., "A Metaheuristic Approach to Aircraft Departure Scheduling at London Heathrow airport," 9th International Conference on Computer-Aided Scheduling of Public Transport, San Diego, CA, 2004.

[17]Gurobi Optimization, Inc. "Gurobi Optimizer Reference Manual," http://www.gurobi.com, 2014.

[18]Federal Aviation Administration (FAA), JO 7110.659A, "Wake Turbulence Recategorization," 2014. http://www.faa.gov/documentlibrary/media/order/7110.659a_wake_recat_order_final_format.pdf