

Design Characteristics of a Terminal Departure Scheduler

Alan Capps*
Mosaic ATM, Fort Worth, TX, 76155

Matthew S. Kistler†
Mosaic ATM, Leesburg, VA, 20175

Shawn Engelland‡
NASA Ames Research Center, Fort Worth, TX, 76155

A terminal departure scheduler designed to work with varying degree of precision across many airports is required to operate with high levels of uncertainty, multiple disparate departure constraints and substantial volatility. This paper describes fast time simulation modeling of terminal departure traffic to assess performance of a terminal departure scheduler. A prototype terminal departure scheduler is developed and exposed to a range of air traffic constraints, departure time uncertainty and terminal transit time uncertainty. Terminal transit error and surface error are varied to assess the robustness of scheduler design to these variations. Current day manual terminal departure scheduling practices are simulated and compared against performance of a prototype terminal departure scheduler. Simulation is used to assess the tradeoffs of sequence and schedule freeze methodologies in the terminal departure environment. Sequence freeze capability demonstrates lower average delay than schedule freeze capability for expected levels of OFF time compliance in future automation. Dallas/Fort Worth TRACON simulation results indicate the possibility of a delay reduction of 35 percent and increased departure throughput of 17 percent for commonly used terminal departures constraints. The results of this study are used to inform the design of a terminal departure scheduler which will undergo evaluation at NASA's North Texas Research station.

I. Introduction

Recent NASA research¹⁻³ has focused on improving tactical departure scheduling in scenarios where well-equipped airport Towers interact directly with Center Traffic Management Units (TMUs) to implement departure management initiatives such as Call For Release (CFR). The research presented in this paper is part of an effort to extend tactical departure scheduling improvements to lesser-equipped airports and to address constraints that exist in the terminal environment. The FAA's Next Generation Air Transportation System (NextGen) plans,^{4,5} call for the ability to accurately schedule a flight from its departing gate to its arrival gate in advance of its actual gate departure (i.e. gate-to-gate scheduling). Specifically, gate-to-gate scheduling presumes the planning and control of a flight from its departure gate to the runway, to the terminal departure fix, Center departure metering fix, through En Route airspace to the arrival metering fix, runway and finally to the arrival gate. For gate-to-gate scheduling to be effective in the NextGen environment, surface, terminal, Center, and national constraints must all be simultaneously satisfied by the departure scheduling tool. NextGen gate-to-gate scheduling also requires accurate prediction and execution of, trajectory-based operations in the terminal area. Observations at the Dallas/Fort Worth

* National Airspace System Engineer, NASA/FAA North Texas Research Station, AIAA Senior Member.

† Analyst, Mosaic ATM, Inc., 540 Fort Evans Rd NE, Leesburg, VA, AIAA Member.

‡ Aerospace Engineer, NASA/FAA North Texas Research Station, AIAA Senior Member.

Terminal Radar Approach Control facility (D10 TRACON) revealed substantial delay and inefficiency attributable to the workload-intensive process of handling terminal departures during constrained operations.

Considerable research has been performed on *arrival* scheduling in the terminal area,⁶⁻⁸ however, terminal *departure* scheduling research is substantially less mature in comparison. Prior NASA work utilized highly-precise terminal departure trajectories to enable expedited departures,⁹⁻¹¹ however, no departure decision support tool exists within the National Airspace System (NAS) today to implement this capability. More recent investments in terminal departure research have enabled advances in this area,¹²⁻¹⁴ but still lack a terminal departure scheduler capable of providing guidance with direct integration of surface and En Route systems in a multi-airport scheduling environment as well as supporting tactical decisions such as which flight should depart next from a group of airports.

In recent years technological advances have enabled improvements to departure predictions which can serve to reduce departure uncertainty. Examples of these advances are integration of surface decision support systems with En Route decision support tools and airline data interfaces that provide more accurate gates and gate pushback times.¹⁻³ As observed in D10 departure scheduling and a nationwide analysis of TRACON departure operations,¹⁵ a need exists for a terminal departure scheduler that can work effectively amidst substantial uncertainty while also considering equity amongst the larger and smaller airports.

This paper uses fast-time simulation modeling of terminal departure traffic to analyze design characteristics of a terminal departure scheduler for a multi-airport environment including real-time estimates of uncertainty. Simulation is used to assess the sensitivity of a prototype terminal departure scheduler to varying uncertainty, traffic demand and air traffic constraints.

This paper begins with a description of the terminal departure scheduler developed for this research, followed by information on the simulation capability used in this research. The fast time simulation setup and results for each scenario executed is discussed. The paper ends with a brief discussion and concluding remarks.

II. Terminal Departure Scheduler

A. Unique Considerations of a Terminal Departure Scheduler

A key challenge for the terminal departure scheduler is high departure demand uncertainty and controllability. Departure time error is generally accepted as the largest source of error in the NAS.^{16,17} Unlike other NAS decision support tools, all the terminal demand is comprised of flights that originate from within its airspace. This OFF time uncertainty does not include weather related uncertainty and ascent modeling uncertainty that can also exist in terminal airspace. The cumulative uncertainty from these events creates complications for a terminal scheduler that can result in instability and render the schedule unusable in real-world operations. Thus, the terminal scheduling algorithm must be especially robust to multiple forms of uncertainty.

Another challenge unique to the terminal departure scheduling environment is the opaqueness of the plan. That is, the Towers that control the release of the flights and the pilots that are flying the departing aircraft often do not know the departure time in advance of issuing the clearance. In today's operations during terminal constraints a departure clearance is not communicated until immediately prior to the departure's release. Based upon observations of D10 traffic, it is not uncommon for a departure at a nearby airport to have substantial delay due to a departure at a separate airport given they are competing for resources at the same departure fix. Because of this, the departure controller cannot reliably inform the dependent flight when it is likely to depart given the high uncertainty associated with the departure prediction. Thus, the terminal departure scheduling algorithm should seek to support transparency in the scheduling process.

Another complexity of scheduling flights in the terminal environment is the existence of multiple departure constraints. Examples of this are controlled departure times from an Expect Departure Clearance Time (EDCT) and/or CFR. Based upon field evaluations of tactical departure scheduling technology¹ it is known that approximately 8% of all CFR flights in the NAS are also subject to an EDCT constraint. In addition, observations of terminal departure operations for this research have shown that departures may be subject to terminal and Center constraints at the same time.

Volatility of the terminal restriction is a significant challenge for terminal departures. It is not uncommon for a constraint to change several times within an hour as a weather constraint moves through the region. Thus, the terminal scheduler must be capable of providing a stable and fair solution amidst changing constraints.

Lastly, the terminal departure scheduling solution should serve to reduce the controller workload. The uncertainty associated with weather events, demand/capacity imbalances, required vectoring and increased

communication requirements all add substantial workload to air traffic control. Thus, a terminal solution that increases workload during these busy periods is unlikely to be accepted by operational personnel.

B. Terminal Departure Scheduler

The prototype terminal departure scheduler seeks to resolve many of the unique challenges mentioned in the previous section. The following sections discuss the process employed to sequence and schedule each flight.

1. Sequencing

The terminal departure scheduler gives greater priority to the flights that are ordered earlier. The process the terminal departure scheduler uses to decide what order is used in scheduling is referred to as the sequencing logic. This section briefly describes the sequencing logic which was derived, in large part, from two existing schedulers, the Traffic Management Advisor (TMA) Dynamic Planner (DP)¹⁸ and the Surface Decision Support System (SDSS)¹⁹ surface scheduler. These two schedulers were chosen as a basis for terminal departure scheduling logic because of their relevance to the problem at hand, demonstrated success in operational environments and their ability to handle flights at various stages in the departure process.

The terminal departure scheduler runs on a user defined periodic rescheduling interval, hereafter referred to as the scheduling cycle. Currently a five second scheduling cycle is used for processing given it matches the frequency of position data updates. For each scheduling cycle, flights are re-sequenced and rescheduled. The sequencing order ensures that flights which are higher priority (see Table 1) from an operational readiness standpoint are scheduled

first and that frozen flight times do not change from one iteration to the next.

Table 1 describes the categories that are used to sequence flights. This table is listed in priority order from highest to least. Thus all flights that are in the first category (crossed departure fix) are scheduled prior to the flights in the second category (terminally controlled airborne).

While some flights may fall into multiple traffic management initiative (TMI) categories, a flight can only belong to one sequencing category. The highest priority sequencing category a flight qualifies for is assigned to it. Thus, a flight with both a terminally controlled frozen OFF time and a CFR will be assigned the higher priority sequencing category associated with terminally controlled frozen flights.

Table 1. Sequencing categories determine the order a flight is scheduled.

Sequencing Category (in priority order from greatest to least)	Description	Aircraft Location	Sorted By
Crossed Departure Fix	The flight has crossed the departure fix.	Center Airborne	Actual departure fix crossing time
Terminally Controlled Airborne	Flights that have a terminal constraint and are airborne.	Terminal Airborne	Undelayed estimated departure fix time
Uncontrolled Airborne	Flights that have no terminal constraint and are airborne.	Terminal Airborne	Undelayed estimated departure fix time
Terminally Controlled Frozen	Terminally controlled flights have a frozen OFF time. This category is the focus of this research.	Surface Active	Terminally controlled frozen OFF time
Call For Release	Flights that are surface active and have a Call for release TMI.	Surface Active	Call For Release time
Strategic TMI	Flights that are surface active and have an EDCT.	Surface Active	EDCT time
Terminally Controlled Unfrozen Surface Active	Terminally controlled flights that are surface active but not yet frozen.	Surface Active	Estimated undelayed OFF
Surface Active	Flights that are surface active with no TMI constraint.	Surface Active	Estimated undelayed OFF
Terminally Controlled Surface Inactive	Terminally controlled flights that are not yet surface active.	Surface Inactive	Estimated undelayed OFF
Surface Inactive	Flights that have no terminal constraint and are not surface active.	Surface Inactive	Estimated undelayed OFF

Each sequencing category has its own sorting rules. For flights that have already crossed the departure fix, they are sorted by their crossing time. Airborne flights that have yet to cross the departure fix are sorted by their undelayed estimated departure fix crossing time. Surface flights which have a TMI use the controlled OFF time associated with that constraint, while all other surface flights use their undelayed estimated OFF time to determine sequencing order.

2. Scheduling Processing Logic

Rather than accomplishing this with a single monolithic entity, a collection of smaller schedulers are orchestrated by a master scheduler. This design approach was chosen to model industry best practices of loosely

coupled, course grained services²⁰ and to maximize reuse of existing components from prior research. The individual schedulers can be seen at the top of Fig. 1 and are comprised of a pre-scheduler, departure fix scheduler, airport scheduler and a post-scheduler. The terminal departure scheduling process begins by a call to the master scheduler from the main processing logic of the terminal departure system. The frequency of the call to the master scheduler is configured in system properties files. For this research a frequency of five seconds was utilized. Once invoked, the first step the master scheduler performs is initializing a temporary copy of the flight object for the scheduling process. The primary purpose of this activity is to know the starting time for each flight and allow it to be bound by time ranges in later processing if necessary. Given the scheduler must resolve times at multiple locations (airport and fix) a temporary copy of the flight is initialized at both locations. If the flight has a CFR or EDCT time, this time is used to set the latest time the flight can depart. For CFR and EDCT flights, a single minute-level of granularity is used for departure time as opposed to a departure time window.

After initializing the flights, the master scheduler calls the pre-scheduler component. The pre-scheduler's primary role is to address flights that have missed their terminally controlled OFF times and thus need to be rescheduled. The pre-scheduler will evaluate all the flights being scheduled to determine if any have missed their coordinated OFF time by the configured number of seconds. If so, the flight will lose its controlled time which will result in the flight having lower sequencing priority as described in the previous section.

Once flights have been initialized, they are sorted according to the sequencing categories listed in Table 1. Each flight then undergoes scheduling from earliest to latest in each sequencing category.

For each flight, the terminal departure transit time is calculated. For the simulation, this transit time is supplied by a flight time decision tree and terminal departure transit time error described later in this document. For real-time prototype system processing, the terminal departure transit time prediction is provided by the research Traffic Management Advisor

(rTMA) system. The rTMA terminal departure transit time prediction includes the effect of winds at crossing altitude. Once the flight time is calculated, it remains constant for the remainder of the scheduling cycle assuming no changes to departure fix have occurred. If the flight is airborne, the remaining terminal departure transit time will be calculated by subtracting the amount of time already spent in transit.

Next, the scheduler resolves the departure fix and airport times for each flight. The following steps are taken until both the departure fix time and runway departure time are fully resolved. Fully resolved means that both locations have a time that has no scheduling conflict with other flights and meets all specified traffic management constraints. To accomplish this objective the master scheduler calls the departure fix scheduler which schedules each flight to the appropriate departure fix based upon the OFF time estimate and the terminal departure transit time. Once a departure fix time is obtained, the scheduler then calls the airport scheduler. To resolve the flight's scheduled OFF time, the airport scheduler uses the later of the initial OFF time calculated in an earlier step or the adjusted OFF time derived from the departure fix time. The reason for this is the derived OFF time can be no earlier than the flight can achieve. The adjusted OFF time is calculated by subtracting the terminal departure transit time from the resolved departure fix crossing time. If a CFR or EDCT time exists for this flight, this constraint is taken into account in the

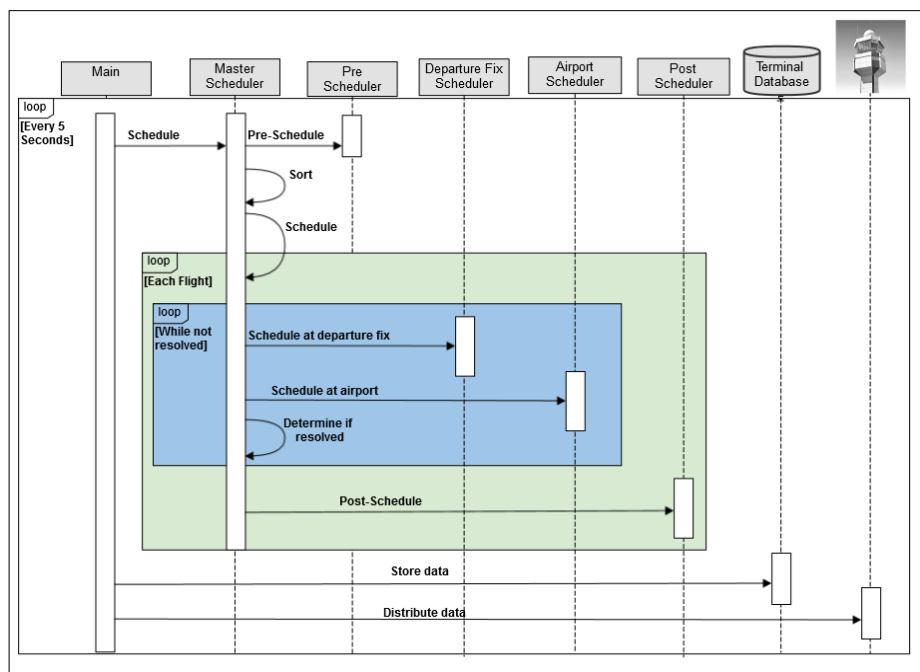


Figure 1. Terminal Departure Scheduler Processing Flow.

airport schedule time. In the current implementation, the SDSS scheduler is used for Dallas Fort Worth (DFW) flights. A simplified model was used for less-equipped airports, which included all D10 airports except DFW. The simplified model for less-equipped airports utilizes fewer site adaptation files than the SDSS scheduler. If the resulting OFF time from the airport schedule time matches the required OFF time to meet the departure fix, scheduling for this flight is complete. If the times do not match, the scheduling process continues again using the updated OFF time as a starting point for scheduling to the departure fix.

Once all flights are scheduled in the manner described above, the master scheduler calls the post-scheduler process. The primary purposes of the post scheduler are to determine if a flight should be frozen, store the data and distribute the data to the appropriate processes. The purpose of freezing a flight is to ensure that the controlled departure time that has been communicated to a surface local controller does not change. A flight is frozen if its time to departure is less than the configured freeze horizon value. The results from the scheduling process for each flight are stored in the terminal database. The purpose of the database is to assist in system processing and post operational analysis. The results from the scheduling process are then distributed to the other system processes. In real-time operations, this makes the scheduling results available to Tower personnel.

III. Terminal Departure Scheduling Simulation Capability

The objective of this work is to develop a departure scheduler that can be assessed by air traffic personnel in an operational terminal departure environment. The term *departure scheduler* refers to a software program that is used by terminal personnel to schedule flights from multiple departure airports within their control which possess varying levels of OFF time precision. The departure scheduler receives real-time flight planning data, surface OFF time estimates and terminal transit time estimates as input and produces a controlled wheels off (OFF) time for each flight which meets all required air traffic constraints on the runway, departure fix, downstream Center metering points and strategic traffic management initiatives. The OFF time provided by the scheduler ensures that minimal separation is maintained at both the runway threshold and departure fix.

The OFF times provided to operational personnel are expected to be treated as a controlled OFF time. That is, air traffic personnel will actively control the flight to meet the departure time similar to the process used for EDCT and CFR controlled times.

Fast-time simulation was used to better understand terminal departure scheduler performance when subjected to variances in OFF time error, flight time error and varying traffic constraints. The terminal departure scheduler used in this simulation is also expected to execute as the prototype scheduling software that will be used in terminal departure prototype system processing. The terminal departure prototype is a new decision support tool being developed and evaluated in the D10 terminal environment. To achieve the objective of using the same scheduler for both fast-time simulation and prototype processing, an evaluation harness was developed to allow the terminal departure scheduler to be evaluated in multiple modes including; real-time data mode, playback mode and simulation mode. This paper focuses only on the departure scheduling simulation mode.

The simulation analysis is executed by running the prototype terminal departure scheduler within the fast-time simulation evaluation harness, as illustrated in Fig. 2. Key inputs such as surface taxi time and terminal transit time undergo perturbation by applying stochastic uncertainty. Once the scheduled OFF time for an aircraft is calculated, then the aircraft's actual OFF time is adjusted by a random variable. The application of error to the terminal transit time is called *terminal transit error*, whereas the application of this error to surface events is called *surface error*. This research varies both terminal transit error and surface error to assess the robustness of scheduler design to these variations.

The evaluation harness developed for this work provided required components enabling fast time simulation; namely a component to provide input data to the scheduler, a feedback mechanism to model controller response to scheduler output and an error generation component to inject realistic operational error into the simulation. Figure 2 illustrates the terminal departure scheduler evaluation harness, which is briefly described in the following subsections.

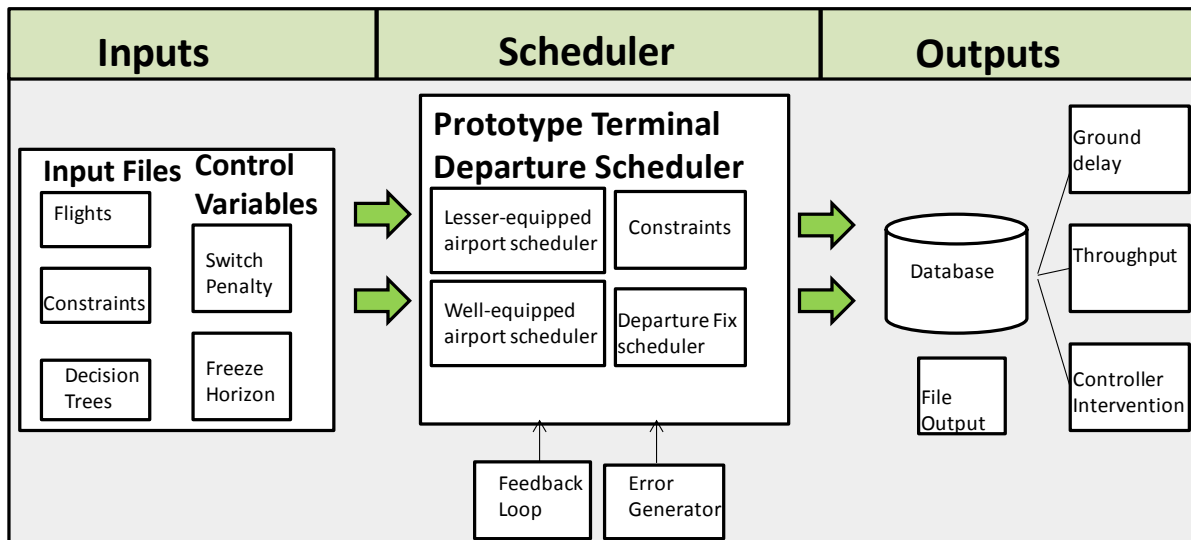


Figure 2. An evaluation harness was developed to assess a prototype terminal departure scheduler.

A. Inputs

The inputs to the terminal departure simulation consist of flight data, constraints and decision trees for multiple airports in the simulation. The following subsections will briefly discuss the inputs required for this research.

1. Flight Data Input Files

The terminal departure simulation capability includes the ability to generate input files that match the demand and operational criteria specified. The result of this input generation process is an output file of flights that match the given criteria. Some of the choices available when creating simulation input files are the amount of desired departure fix demand per hour, the percentage of departure fix demand from each airport by departure runway, the percentage of flights subject to other traffic constraints (e.g. EDCT or CFRs) and other variables.

This research modeled D10 airspace to evaluate terminal constraints, as depicted in Fig. 3. This diagram includes

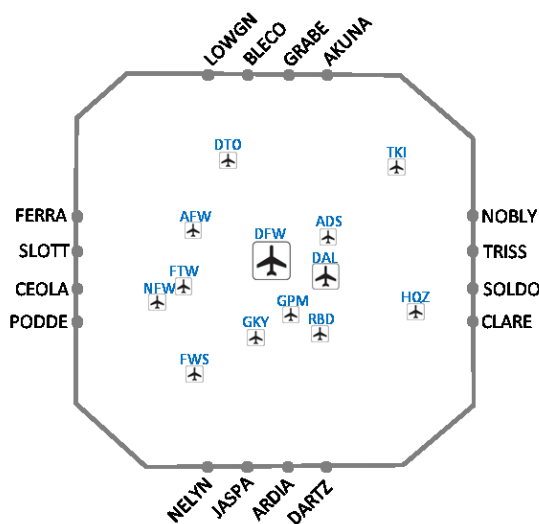


Figure 3. D10 departure airspace was modeled to evaluate terminal constraints.

airports contained within the boundaries and the departure fixes located on the borders. The D10 TRACON is centered on DFW and extends outward approximately forty miles in all directions. It contains two major scheduled passenger service airports, DFW and Dallas Love Field (DAL), which are separated by approximately ten miles. Several busy general aviation airports, a regional cargo hub, and a Naval Air Station Joint Reserve Base contribute to the complexity of this TRACON environment. The sixteen departure fixes are arranged in groups of four called departure gates (not to be confused with airport parking gates), which depict their general location relative to the TRACON boundaries. For example, the north gate includes departure fixes LOWGN, BLECO, GRABE, and AKUNA. It is common for restrictions to be imposed on entire gates, without mention of the fixes, so it is important to understand which fixes belong to which gates. A year of operational data from 2013 was analyzed to determine average flight times and variation to each departure fix from each departure airport.

While simultaneous departure fix constraints are often applied in terminal departure operations, this research found it useful to focus primarily on the effect to system performance with a single departure fix constraint. The scenario used most frequently in this paper was a 10 miles in trail (MIT) constraint over departure fix SOLD0 on April 10, 2013. This day was selected primarily because of availability of firsthand observations of operations from D10 TRACON and detailed output data to further analyze the traffic scenario.

2. Constraints

Terminal departure constraint inputs possess substantial flexibility. This flexibility is demonstrated in the use of routing constraints, flow control constraint (e.g. MIT) and creative combinations of both. This section discusses terminal departure constraints, as listed in Table 2, and their handling by the simulation framework.

The terminal departure constraint is distinct from Center tactical departure metering constraints (i.e. CFRs). However, the terminal constraint and CFR share many properties. For example, they are both local tactical constraints which require a precise departure time and approval prior to releasing the departure. A key distinction

Table 2. Common terminal departure constraints.

Constraint Name	Constraint Type	Description
Complete Departure Fix Combine	Route	All traffic that was assigned to the original fix is moved to one or more alternate departure fixes.
Limited Departure Fix Combine	Route	A select set of flights bound to a departure fix is moved to one or more alternate departure fixes.
Stream based Departure fix Combine	Route	A departure fix becomes the only location terminal flights bound to the destination in question may depart the terminal area.
Gate Swap	Route	Changes the gate a departure fix traffic is bound to one or more alternate gate(s). This adds the requirement for departing flights to file a new flight plan.
Miles in Trail	Flow	Requires MIT separation enforced at the departure fix. This flow constraint is often enforced with a routing constraint.
Speed Constraint	Flow	Requires departures to meet a speed restriction, typically until reaching the departure fix.

between terminal constraints and CFRs is the domain that implements the restriction. While an underlying reason for a terminal constraint may originate from the Center environment, the entire process is implemented in terminal airspace by terminal personnel. Another distinction between terminal constraints and CFRs is the process used to regulate the departure. Unlike CFRs, terminal constraints typically do not come with a specific

departure time window but rather only the expected sequence of departing flights. For this reason, the process is often called *departure sequencing* by terminal personnel. The departure sequencing process is used instead of specific departure times due to extremely high levels of uncertainty that are present in the terminal departure environment.

Table 2 lists commonly used terminal departure constraints. The restrictions modeled in this research are complete departure fix combine, MIT and a speed constraint. The gates referred to in these restrictions are groups of departure fixes. For example using the illustration in Fig. 3, departure fixes NOBLY, TRISS, SOLD0 and CLARE all belong to the East departure gate.

The terminal departure simulation input files control the type of constraint the simulation provides to the scheduler, the time at which it is injected into the system and the duration of the constraint.

3. Decision Trees

Simulation input files are used to control the size and distribution of surface and terminal transit error. These input files are called decision trees because they provide branch like options that allow methodical selection of a property based upon one or more decision variables. Surface taxi times and terminal departure transit times are supplied by decision trees. The primary purpose of surface taxi times in simulation is to allow analysis of delay distribution on the airport surface. Terminal transit times are used to simulate realistic flight times from each departure airport to each departure fix. These decision tree distributions are specified by a mean value, to which a Gaussian error distribution is applied. The distributions used in simulation were determined by analysis of operational data, information learned in first-hand observations of operational events and prior research.¹

The OFF time and flight time error associated with the terminal departures in the simulation is also governed by decision trees. Error is expressed as a stochastic value with the specific mean and standard deviation. Error values used in this research were obtained through a combination of operational data analysis, direct operational

observations and results from prior research.¹ Using the decision trees, error can be applied to flights at several locations in the departure process, including the pushback time, surface taxi, departure queue and terminal transit. For this research error was applied to surface taxi, departure queue (as controlled OFF time error) and terminal transit. The distribution of error is assumed to be Gaussian, which is consistent with prior research on tactical departures.¹

4. *Simulation Variables*

The terminal departure simulation framework uses an input file to control simulation variables. This section discusses frequently modified parameters.

The simulation requires a scenario start and stop date and time. While the simulation will be the duration the user specifies, it is typically best to ensure that all flights in the input file have sufficient time to cross the departure fix. While it may be desirable to evaluate a fixed traffic demand period, terminal departure pushes tend to have the highest error near the end. Thus, if comparing two scenarios to one another, the average departure delay values may be misleading if the entire flight demand has not been resolved.

Error can be applied on the surface in several areas rather than all in the departure queue. The purpose of this capability is to provide a more realistic model of where delay would occur on the airport surface when terminal delays are encountered. Booleans exist in simulation input to allow error to be applied at pushback, taxi and to the controlled OFF time. If these Boolean values are set to true, the decision tree associated with the surface event is used for the distribution. Error can also be applied to the airborne flight time. If this Boolean is set to true then the flight time error decision tree is used to apply the specified error distribution to departure transit time of flights.

A simulation input variable specifies the freeze type. The options are either sequence freeze or schedule freeze. A variable exists to specify the freeze horizon, which is the number of seconds prior to OFF that a flight is frozen by the scheduler. Sequence freeze ensures that this departure scheduler maintains the order of departing flights once the flight reaches its freeze horizon. Schedule freeze requires a flight to meet its departure time within the specified parameters in addition to sequence freeze requirements of maintaining departure order. For more information on sequence and schedule freeze capability, see the freeze section of the results. For schedule freeze a variable must be set that specifies the number of seconds past the controlled OFF time a flight is automatically rescheduled.

Lastly, an airport switch penalty variable is used to model the current day behavior when departure control alternates from one airport to another (i.e. DFW, Love Field, Addison, etc.). The switch penalty is only used in current day (baseline) modeling as this delay is believed to be eliminated with reduced coordination uncertainty provided by automation and graphical displays in the Towers.

B. External Feedback Provided to the Scheduler by the Simulation Framework

The simulation framework provides feedback to the terminal departure scheduler in response to its guidance. This response seeks to model the response expected in the operational environment.

1. *Feedback mechanism*

The terminal departure simulation capability ensures minimal separation is enabled at the departure runway and departure fix. The purpose of this logic is to create a realistic environment in which to evaluate the terminal departure scheduler.

Minimal departure runway separations for large, well-equipped airports rely upon the separation logic from the SDSS. Each smaller airport surface scheduler has adapted separation that is used for minimal separation. For this research the runway separations for all airports was the same as used by the SDSS system that runs at DFW airport.

For departure fix separation, the routing or miles in trail constraint is enforced at the departure fix. For flights that would otherwise have insufficient separation at the departure fix boundary, the simulation places the flight in a *controller intervention* status. When a flight is placed in controller intervention status it is allowed to achieve the required amount of separation at the departure fix boundary. The amount of time that a flight is in controller intervention status is recorded. The purpose of recording the amount of time a flight is in controller intervention status in the simulation is to allow a method for evaluating controller workload associated with terminal departure scheduling.

2. *Error generation*

The error generation component applies a stochastic error to the time component in question as specified in the appropriate decision tree. Error can be applied to flight pushback time, taxi time, controlled OFF time assignment and terminal transit time. The size and distribution of error is controlled by the decision tree files as previously discussed.

C. Outputs

The terminal departure simulation environment produces output to both a flat file and a database. Database output is especially useful when executing a large number of Monte Carlo simulations like that performed in this research. The simulation output include details on each flight, simulation run, the error exerted on the flight, expected transit times, actual transit times, delay incurred on surface and airborne and the amount of controller intervention.

IV. Results

The results outlined in this section give insights into the effectiveness of the scheduling algorithm when exposed to a range of air traffic constraints, departure time uncertainty and flight time uncertainty. The primary metrics evaluated are size of departure delay, throughput and controller intervention.

A. Scheduler Performance under Varying Levels of OFF Time Compliance

Prior research on tactical departures¹ indicates that substantial improvements to OFF time compliance can be achieved with surface automation and reduced communication uncertainty. Improved OFF time compliance is expected for terminal departures for the same reasons. In addition, OFF time improvement is expected in the terminal environment due to greater situational awareness of upcoming flight departure times which is not possible in operations today due to the opaqueness of the schedule. This section analyzes terminal departure scheduler performance when exposed to a range of OFF time error.

1. Setup

To analyze the effect of OFF time compliance error on system performance, all experimental variables other than the OFF time error were held constant. The constraint used in the system was a 10 MIT restriction over departure fix SOLD0 with an expected crossing speed of 350 knots. The demand was 30 flights per hour for a duration of 80 minutes. This created a total of 40 flights that were evaluated in 500 Monte Carlo runs. For each Monte Carlo run, the OFF time error was varied according to the specified Gaussian distribution.

The OFF time error was varied from levels expected when terminal departure scheduling automation is available, to estimated levels with no automation, to one standard deviation greater than no automation levels. The OFF time compliance used for terminal automation simulation was a mean of -9 seconds and a standard deviation of 60 seconds for DFW flights. This compliance matches that seen in prior research for DFW when using surface automation.¹ The OFF time compliance used for all other D10 airports was slightly higher given the lack of surface automation available at these airports. For these airports, a mean of 0 seconds and standard deviation of 90 seconds was used. The highest OFF time compliance error was obtained by adjusting the standard deviation of the baseline estimate by a factor of two.

2. Results

As expected, the results indicate that better OFF time compliance leads to better terminal departure performance. Average ground delay per flight was 11.6 minutes, 16.3 minutes and 22.3 minutes for the automation, no automation and high error cases respectively. The 4.7 minute change in average delay between the automation and no automation case suggest that OFF time compliance is a significant factor in achieving reduced delay. The change in average delay between the no automation and high error case suggests that average delay will continue to increase as OFF time error increases.

Figure 4 plots the distribution of ground delay associated with each OFF time error level as a function of time. In this diagram the delay for each error scenario was grouped in 10-minute increments and plotted as a function of minutes into the departure push. The distribution is plotted as a line

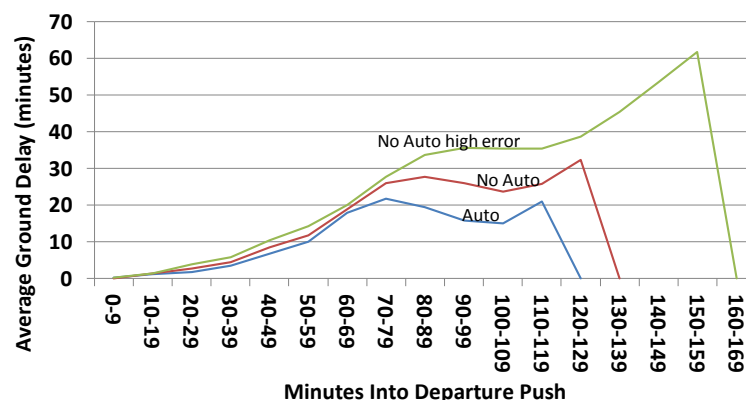


Figure 4. Lower OFF time error leads to lower average delay.

instead of histogram to aid in comparing the distribution amongst the error cases. The distribution of delay over time between the error cases is similar, however, there are two key differences. First, as the OFF time error increases the average amount of delay assigned to flights increase. This is visible in the separation between the lines which builds over time due to a slightly higher slope on higher error cases. Secondly, as the OFF time error increases the duration of the departure push is extended. All plots end with zero delay when the complete demand of 40 flights has been resolved by crossing the departure fix. In this case the duration of the departure push was 37 minutes longer in the high error case than with the automation.

Differences in the duration of the departure push for each OFF time compliance error case indicate a difference in departure throughput. To analyze the effect on throughput when varying OFF time error, the maximum departure rate metric was used. The maximum departure rate measures the highest number of flights that crossed the departure fix in a 10-minute window. This throughput measure is robust to changes in demand that can occur throughout the push, as well as push startup and shutdown variations. Figure 5 illustrates the departure rate of each error scenario over time. During the first 20 minutes all three scenarios show increasing departure rate as additional flights are injected into the simulation over time. Once the available capacity is saturated, the throughput difference between

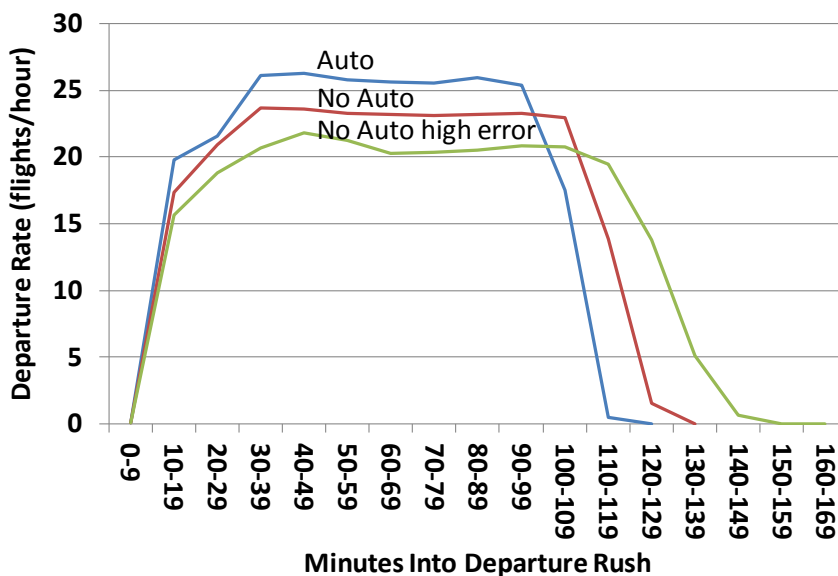


Figure 5. Departure throughput varied substantially between the lower and higher error scenarios.

the automation levels of OFF time error and other cases becomes more apparent. The highest difference in throughput is 5.4 flights per hour, which is seen in the 80-89 minute window between the automation and high error cases. The automation error case ends first, followed by the no automation error case and last is the highest error case. This suggests that OFF time error has a direct effect on departure throughput.

These findings underscore the benefits to terminal departure delay reduction and increased throughput that can be provided by greater OFF time precision from surface automation.

B. Scheduler Performance under Varying Levels of Flight Time Error

Scheduling a departure in the terminal environment in the NAS today requires two mental calculations by controllers, an OFF time estimate and a flight time estimate. In some cases the controller may not attempt to estimate the flight time but rather wait for the flight to clear a pre-determined airborne location prior to departing a trailing flight. Observation of the terminal scheduling process indicates that different methods may be employed by different personnel.

Flight time estimates are important for future automation as well. The terminal scheduler de-conflicts a departure with other flights on the runway and the departure fix. Thus, if the flight time is inaccurate the model upon which flights are being assigned delay can be incorrect. This experiment analyzes the sensitivity of departure scheduler performance to flight time error.

1. Setup

All experimental variables other than the flight time error were held constant. The April 10th, 2013 scenario mentioned in the previous section was utilized, however, for all scenarios the OFF time error remained at expected levels with future automation.

The flight time error was varied from a mean error of 0 seconds to a mean error of 25 seconds. The standard deviation of flight time was varied from 15, 30, 60 and 240 seconds. The flight time error level used to estimate

future automation was a mean of 25 seconds with a standard deviation of 30 seconds for DFW. This flight time error was chosen because it matches that seen in prior research.¹ In the automation scenario, flight time error from smaller airports was slightly higher due to variance from less frequent demand from departure airports to departure fixes in non-standard terminal constraint situations. Small airport flight time error for future automation is expected to be a mean of 35 seconds and standard deviation of 40 seconds.

2. Results

Consistent with intuition, the simulation results indicate that lower flight time error leads to less controller intervention. As indicated in Table 4, in the lowest flight time error case the percentage of flights that are estimated to require controller intervention of one minute or greater are 22%. As previously discussed a flight is considered in controller intervention status when the simulation evaluation harness determines that inadequate separation will exist at the departure fix boundary. The flight remains in controller intervention status until enough simulation time has transpired to achieve the required amount of separation. Controller intervention percentage grows a modest 1% in the automation scenario but increases substantially to 37% of flights in the largest error scenario. In addition to

Table 4. Results of Flight Time Error Variation on Departure Scheduler Performance.

Scenario Name	Mean Error (s)	Standard Deviation Error (s)	% Required Controller Intervention	Controller Intervention Duration (s)	Average Ground Delay (m)	Maximum Effective Throughput per hour (% total demand)	Duration Longest Push (m)
Low Flight Time Error	0	15	22	115	12.2	88	115
Automation (expected)	25	30	23	117	13.8	84	117
Automation w/2sigma	25	60	29	119	13.9	84	119
Automation w/4sigma	25	240	37	135	15.8	79	135

increased need for controller intervention, the duration of the intervention is also longer. In the low flight time error case average controller intervention is 115 seconds, while in the largest flight time error scenario average controller intervention is estimated to be 135 seconds.

Perhaps a less obvious effect of increased flight time error is the transitive effect which may lead to ground delay. Given short flight times in the terminal area, unexpected airborne delay can ripple back to departing airports that are scheduling into this environment. As indicated in Table 3, the average ground delay changes by 3.6 minutes from the lowest flight time error scenario to the highest.

The maximum effective throughput listed in Table 3 is defined as the highest percentage of the hourly rate achieved during a 10 minute window. This throughput metric is useful to analyze max throughput despite demand variations that can occur due to clumping of demand at the beginning or end of the push. The highest throughput was demonstrated by the low flight time error case in which 88% of its given demand was resolved. The lowest throughput was demonstrated by the highest flight time error case, with a maximum effective throughput of 79%.

Given 500 Monte Carlo simulation runs were performed for each scenario, it was possible to compare the shortest and longest length of a departure push. This metric can be used to estimate the best and worst case scenarios from a system performance standpoint. The difference between the longest and shortest departure push was 135 minutes versus 115 minutes for the low and highest flight time error cases.

While the variations to departure performance are not as significant as those demonstrated by OFF time compliance, they do suggest a strong correlation between improved flight time prediction and better system performance. It is worth noting that the flight time results discuss in this section all use the east gate. Given D10's predominant use of south flow configuration, terminal transit times departing the north gate are generally longer. Longer terminal transit times often increase flight time variance. Additional analysis would be required to determine the effect airspace geometry has on the metrics measured in this section.

C. Analyzing the Switching Penalty in Baseline Operations

A less obvious benefit from terminal departure scheduling automation is that associated with loss of throughput due to coordination timing between departing facilities. This time is referred to in this research as switching time, which has an associated switching penalty. Based upon observations of current day terminal departure scheduling, the primary reason that a switching penalty exists today is the inherent opaqueness in the schedule. During terminal constraints, key personnel are often so busy with tasks that they are not able to coordinate with all the required parties in a manner that allows adequate lead time to prepare the next flight in sequence for departure. Inadequate lead time can result in unutilized departure demand. This phenomenon is known as a switching penalty. However, with automation available to all required parties and an indication of the forthcoming flight's departure time, the

switching penalty is expected to be removed. This experiment assesses the impact of a switching penalty in baseline operations. Later in this paper, the switching penalty is combined with expected improvements to OFF time and flight time error to estimate benefit of automation over the current day baseline.

1. Setup

To analyze the effect of switching time penalty on system performance, all experimental variables other than the switching time penalty variable were held constant. The switching time penalty variable is specified in the simulation as the number of seconds of delay incurred when switching from one departure airport to another. In terminal departure operations today this delay is generally experienced over time while waiting for the tower that just received authorization to depart flights communicate with the pilots and prepares them for departure. While this waiting is occurring, delay at other airports continues to build. The switching penalty is only imposed if the flight is ready to depart. For instance, if a 30 second switching penalty is enforced but a flight is 60 seconds late due to OFF time error compliance, then no penalty is enforced. However if the flight was ready to depart, a 30 second delay would be added to that flight and any other flights that were immediately trailing the flight.

When switching from one departure airport to another, the departure controller must first recognize the readiness for this activity by observing the departing flight from the preceding airport. Then the terminal departure controller communicates with the airport departure controller to allow the flight to depart. Finally, the airport departure controller then communicates with the pilot to clear the flight for departure. Based upon estimates from prior research¹ which analyzed response times to controller commands, the entire switching process is estimated to take at a minimum 30 seconds. Thus, the switching penalty values analyzed were 0 seconds, 30 seconds, 60 seconds and 120 seconds. The OFF time error was held constant at future automation levels to minimize the effect of multiple error sources on the switch penalty analysis.

2. Results

As expected, the results indicate that as the switch penalty increases, so does the average delay and push duration. No switch penalty resulted in a 13 minute delay average, while a 120 second switch penalty resulted in a 15.9 minute delay average. Increased switch penalty also has an effect on throughput. The longest duration departure push occurred in the highest switch penalty case, which was 14 minutes longer than no switch penalty scenario. These results indicate the switching time period encountered in current day terminal operations has a substantial impact on flight delay and throughput. Increased visibility into the departure schedule from automation is expected to reduce or eliminate this shortfall.

D. Scheduler Performance with Varying Miles in Trail Constraints

Terminal departure simulation was used to investigate the effect of increasing MIT restrictions on terminal departure performance. This section gives insight into how terminal departure delay grows as MIT increases and what the expected benefit of terminal departure automation is as a function of MIT constraint.

1. Setup

A MIT constraint over a single departure fix was used for this experiment. The size of the MIT constraint was varied from 10 to 30 miles in trail in 5 mile increments. The demand for all scenarios was 30 flights per hour for a duration of 80 minutes. Two scenarios were analyzed at varying MIT, one representing a current day without automation and the other terminal automation. The no automation scenario used 30 seconds switching penalty, a mean OFF time error of 15 seconds and a standard deviation of 115 seconds. The automation scenario had no switching penalty and used a mean OFF time error of -9 seconds with standard deviation of 60 seconds for DFW, and a mean of 0 seconds with standard deviation of 90 seconds for other airports.

2. Results

Results indicate that MIT has a strong relationship to average ground delay and throughput. As illustrated in Fig. 6, as MIT increases, so does the average ground delay assigned. In all cases evaluated the automation scenario outperformed the no automation case. In terms of percentage improvement over no automation, the greatest benefit is seen at lower MIT values. Specifically, at 10 MIT the automation scenario demonstrates a 35.2% reduction in average departure delay over the no automation case. There are believed to be two reasons for the decreased percentage of delay as a function of increased MIT. First, the portion of delay that is saved due to removal of the airport switching penalty stays the same across all MIT values. As a percentage of the total ground delay this portion is higher in the 10 MIT case than in larger MIT cases. Secondly, as MIT values continue to grow, the demand reaches a point of saturation such that flights with high OFF time compliance error that would not have made their departure time now do. Thus, the benefit of improved OFF time compliance error is proportionally smaller as the delays grow larger.

To analyze the effect on throughput when varying MIT, the maximum departure rate metric was used. The maximum departure rate measures the highest number of flights that crossed the departure fix in a 10 minute window. This throughput measure is robust to changes in demand that can occur throughout the push, as well as push startup and shutdown variations. Figure 7 illustrates the departure throughput by MIT. As expected, the

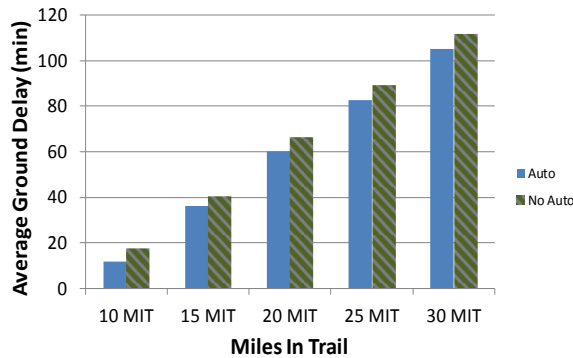


Figure 6. Average Ground Delay Varies by MIT.

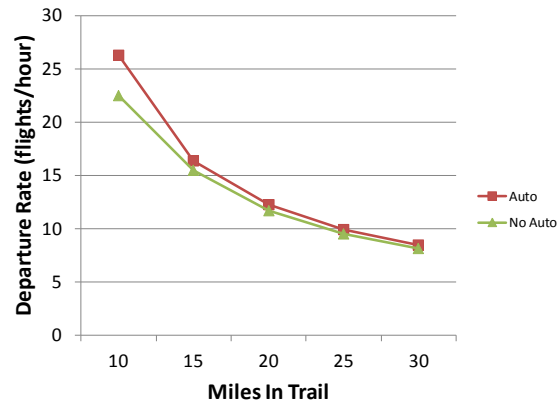


Figure 7. Departure rate as a function of MIT.

departure rate decreases as MIT increases. The largest gains in throughput between the no automation and automation case are seen at the lowest MIT. At 10 MIT a 3.7 flights per hour departure rate improvement is observed by the automation scenario over the no automation scenario. For the traffic levels analyzed in this experiment, that is approximately 16.6% improvement in departure throughput.

E. Analysis of Schedule versus Sequence Freeze Performance

A shortfall of the current day manual terminal departure scheduling process is the lack of visibility into the scheduled departure time. Given the short notice flights may receive prior to departure, efficiency is lost due to time required for staging the departure for takeoff. Future terminal departure automation is expected to provide the departure information with sufficient lead time for controllers to prepare the flight for departure. However, when communicating this information it is important that as few changes to the communicated departure time occur as possible. Changes to controlled departure time can increase controller workload and decrease efficiency. Given the expectation that the time will remain unchanged, the process of communicating a controlled time to operational personnel is often referred to as ‘freezing’ the flight.¹⁸ This section discusses the evaluation of two distinct terminal departure freeze methodologies.

Observations of D10 terminal departure scheduling indicate that operational personnel adhere to a specified departure sequence, however, they are not required to adhere to a specified departure time. The process of specifying and communicating the *order* of departing flights that will not change barring a re-plan is called a *sequence freeze*.¹⁸ The process of specifying both the *order* and *departure time* for departing flights is referred to as a *schedule freeze*.

In this prototype terminal departure scheduler the user must specify either a sequence or schedule freeze capability. If schedule freeze is enabled and a flight misses its assigned departure time by a specified number of seconds, the terminal scheduler will reschedule the flight. When a flight is rescheduled it will be assigned the next available time which does not impact other frozen flight times. If sequence freeze is enabled, the specified flight order is maintained without regard to departure compliance. This prototype terminal departure scheduler requires the user to specify a freeze horizon window for use in the scheduling process. The freeze horizon window is the time prior to departing the airport that a flight becomes frozen. The freeze horizon must be large enough to allow adequate time for controllers to prepare a flight for departure. However, the challenge with extending this time period too far prior to departure is that uncertainty can be frozen into the schedule which might otherwise be resolved more efficiently later using more accurate departure time information. For the analysis discussed in this section, the freeze horizon value was held constant at 180 seconds.

A goal of this analysis is to analyze the tradeoffs of sequence freeze versus schedule freeze capability in terminal departure scheduling. Given substantial uncertainty that exists in terminal departure operations, the overall stability and efficiency of the departure schedule may be compromised if a large percentage of flights have to be re-planned

due to missing their scheduled departure time. The balance between greater visibility into the schedule and departure planning stability is analyzed with simulation using realistic assumptions for expected departure time compliance.

1. Setup

The variables modified in this experiment were OFF time error and rescheduling time window. OFF time error was varied between the three levels described in earlier sections. The rescheduling time window specifies the number of seconds past the expected OFF time that a flight must be rescheduled. The rescheduling time window value was varied from 30 seconds through 360 seconds.

2. Results

Figure 8 illustrates the effect of OFF time error on the schedule freeze capability. These results were obtained by using a 60 second reschedule time window for the schedule freeze capability. The OFF time error was varied from levels expected with future terminal departure automation, to estimated levels in today's operations without automation, to twice the standard deviation of the no automation estimate. As seen with previous results, average delay incurred by flights generally grows as the departure push continues. Varying the OFF time error has a visible effect on the distribution of delay over time. Consistent with previous results, lower OFF time error yields lower delay. A difference from prior results is that in this case a number of flights were required to be rescheduled due to missing their assigned OFF time by greater than 60 seconds. Out of a 40 flight scenario, the number of flights that required rescheduling were 5, 14 and 16 for the automation, no automation and high error cases respectively. Thus, as OFF time compliance error grows, the number of flights that missed the required departure window also grew.

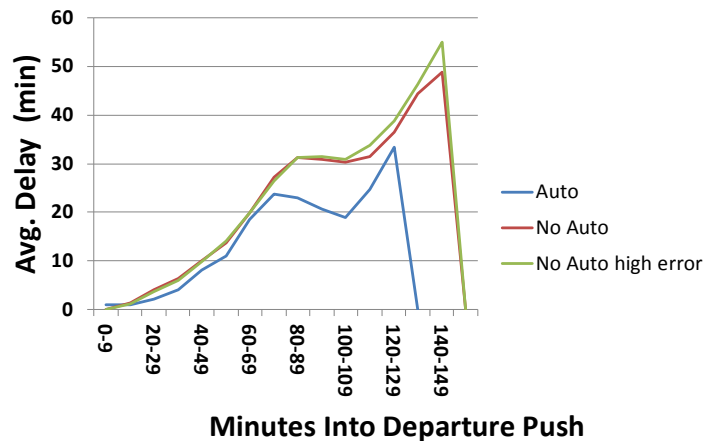


Figure 8. Delay distribution of 60 second reschedule time window at various OFF time error levels.

To compare the performance of the sequence freeze capability against the schedule freeze capability in a fair manner, OFF time error was held constant. Since this capability is targeted at future automation, the expected OFF time error associated with that environment were used. Figure 9 illustrates the delay distribution over time of sequence freeze capability against schedule freeze capability at varying rescheduling time windows. The performance of the schedule freeze capability improves as the rescheduling window is raised. The worst performance of all freeze scenarios is the 30 second rescheduling time window. The reason for this is the number of flights that require rescheduling are higher given the low threshold for OFF time compliance performance. The process of rescheduling a flight creates more demand that must be resolved. This in turn takes more time which leads to higher delay. As the rescheduling time window gets larger fewer flights are required to be rescheduled, resulting in improved performance of the schedule freeze capability. As such, the best performance of the schedule freeze capability is seen when the rescheduled window is at 360 seconds. Even at 360 seconds the schedule freeze time window's average delay is slightly larger than that of the sequence freeze average delay. Thus, at the error levels expected in future terminal departure automation, the sequence freeze capability performs better than the schedule freeze capability.

The results shown in Fig. 9 are based upon the expected levels of OFF time error with future automation. As Fig. 8 demonstrates, the size of the OFF time error has an effect on the delay distribution of scheduled freeze capability. The assumption on OFF time error made in this research is lesser-equipped airports will have a 50% larger standard deviation of OFF time error than well-equipped airports. If the OFF time error is more disproportionate than assumed given higher OFF time error from lesser-equipped airports, then the schedule freeze may become a more attractive option to ensure lesser-equipped airport delay is not propagated to well-equipped airports. Additional research is needed to study the sensitivity of schedule freeze parameters to varying and disproportionate OFF time error levels.

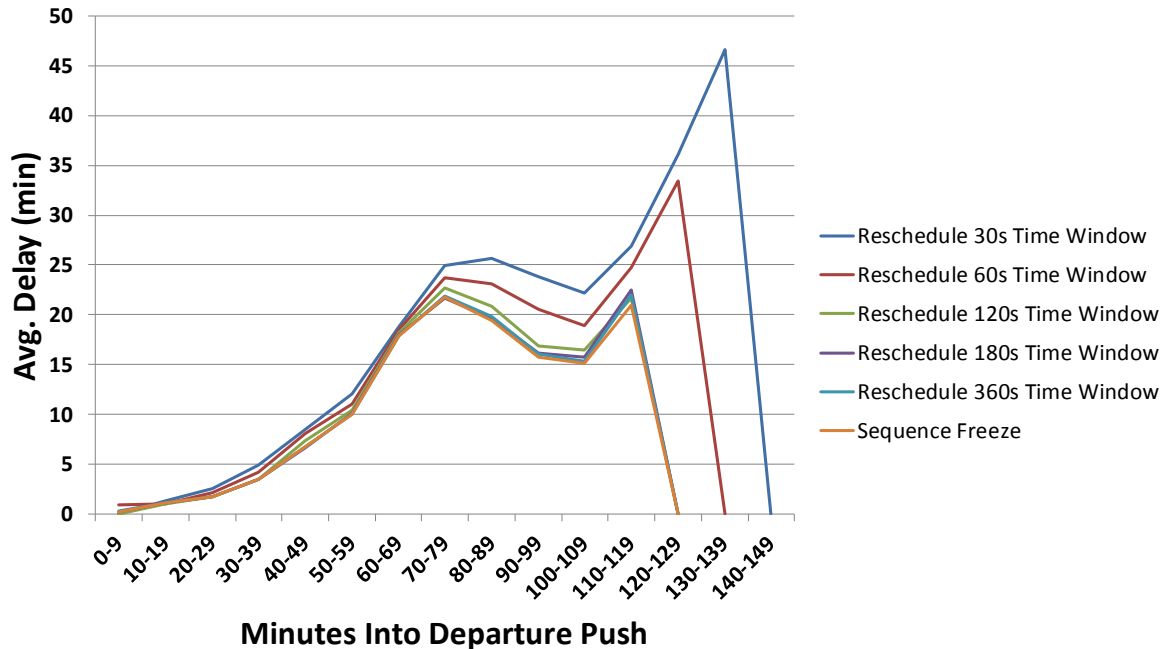


Figure 9. Comparison of delay distribution of sequence freeze and scheduled freeze capability at varying rescheduling time windows.

V. Discussion

The simulation results described in previous sections illustrate the cumulative nature of terminal departure delay. Terminal departure delay builds upon itself until the demand is resolved or the constraint is removed. This finding underscores the importance of reducing the duration of the terminal departure restriction to the greatest degree possible. To support this objective the terminal departure solution should aim for simplicity to reduce the amount of time required to set up the constraint and communicate it to all required parties. Equally, if not more important, is the need to ensure that a terminal departure restriction does not remain in place unnecessarily. This suggests the tactical departure scheduling capability would benefit from close integration with future automation geared toward automatic detection of local flow imbalances like the Integrated Departure Route Planner (IDRP).¹¹

Results indicate a direct relationship between OFF time compliance and departure scheduler performance. Improved compliance demonstrates a notable improvement to delay and throughput. High OFF time compliance error may also lead to increased controller workload and airborne fuel utilization. This underscores the importance of leveraging newer technologies like that demonstrated in prior tactical departure scheduling research¹ as well as focused efforts to improve departure compliance at lesser equipped airports.

Results also indicate a direct relationship between terminal transit time prediction and departure scheduler performance. In addition to creating higher controller workload and greater fuel utilization, flight time error can result in delay being propagated back to the airport surface. The terminal departure scheduling solution should seek to build upon improvements to predictive accuracy of terminal transit time made in prior work.¹

The departure scheduler used in this evaluation demonstrated robustness to terminal transit prediction error up to twice the levels expected in today's operations. However, at prediction error levels 4 times the variation expected,

substantial controller workload and additional ground delay is experienced. These error levels may occur during inclement weather scenarios in which the nominal departure route is blocked. More research is needed to further assess the sensitivity of performance to flight time error and identify approaches to resolve this challenge.

Schedule freeze capability allows greater transparency into flight's departure plan than does sequence freeze capability. Additionally, schedule freeze can help ensure that uncertainty at one airport does not impact departing flights at a separate airport. However, implementing a schedule freeze requires a rescheduling methodology for those flights which do not make their controlled departure time. The results of this analysis indicate that, at the OFF time error levels expected with terminal departure automation, the additional demand caused by schedule freeze rescheduling creates higher average delays and longer pushes than sequence freeze. Additional research may be warranted to more fully evaluate freeze options in the terminal departure environment.

VI. Conclusions

Fast time simulation modeling of terminal departure traffic was used to assess performance of a new terminal departure scheduler. The prototype terminal departure scheduler was exposed to a range of air traffic constraints, departure time uncertainty and flight time uncertainty to better understand its sensitivity to these variables. Simulation was used to assess the tradeoffs of sequence and schedule freeze methodologies in the terminal departure environment. Both freeze capabilities were evaluated under a range of possible OFF time errors. Sequence freeze capability demonstrated lower average delay than schedule freeze capability for expected levels of OFF time compliance in future automation.

Simulation results of D10 airspace indicate delay reductions of 35 percent over current-day scheduling practices are possible for commonly used terminal departures constraints, as well as an increased departure throughput of 17 percent. This benefit is derived via a combination of improved OFF time compliance, reduced flight time error and removal of the airport switching penalty associated with lack of automation in terminal departure operations today. Results indicate modest decreases to controller workload and airborne fuel utilization are possible.

The results of this study were used to establish design considerations for a terminal departure scheduler which will undergo evaluation at NASA's North Texas Research station. The results are also used to inform the concept of operations (ConOps) document being developed on the future terminal departure scheduling process.

Acknowledgements

The authors would like to acknowledge the essential support provided by FAA personnel at Dallas/Fort Worth terminal Radar Approach Control (TRACON) facility, DFW ATCT and DAL ATCT. Finally, we wish to thank our colleagues at NASA/FAA North Texas Research Station (NTX) and NASA Ames whose support was critical to the success of terminal departure research objectives.

References

¹Engelland, S.A., Capps, A., Day, K., Kistler, M., Gaither, F., and Juro, G., "Precision Departure Release Capability (PDRC) Final Report," NASA/TM-2013-216533, June 2013.

²Engelland, S.A., Capps, A., and Day, K., "Precision Departure Release Capability (PDRC) Concept of Operations," NASA/TM-2013-216534, June 2013.

³Engelland, S.A., Capps, A., Day, K., Robinson, C., and Null, J.R., "Precision Departure Release Capability (PDRC) Technology Description," NASA/TM-2013-216531, June 2013.

⁴FAA, "NextGen Mid-Term Concept of Operations for the National Airspace System," version 2.1, September 2010.

⁵MITRE CAASD MTR110240R1, "A Concept for Integrated Arrival, Departure, and Surface (IADS) Operations for the Mid-Term", January 2012.

⁶Swenson, H. N., Thippavong, J., Sadovsky, A., Chen, L., Sullivan, C., & Martin, L. (2011). "Design and Evaluation of the terminal Area Precision Scheduling and Spacing System." Ninth USA/Europe Air Traffic Management Research and Development Seminar. Berlin, Germany.

⁷Martin, L., Swenson, H., Thippavong, J., Sadovsky, A., Chen, L., & Seo, Y. (2011). "Effects of Scheduling and Spacing Tools on Controllers' Perceptions of the Their Load and Performance." Digital Avionics Systems Conference (30th DASC). Seattle.

⁸Thippavong, J., Swenson, H., Lin, P., Seo, A., & Bagasol, L. (2011). "Efficiency Benefits Using the terminal Precision Scheduling and Spacing System." AIAA Aviation Technology Integration and Operations (ATIO). Virginia Beach

⁹Jung, Y.C., Isaacson, D. "Development of Conflict Free, Unrestricted Climbs for a terminal Area Departure Tool", AIAA's 3rd Annual Aviation Technology, Integration and Operations Conference, Denver, Colorado, 17-19 November 2003

¹⁰Jung, Y.C., Isaacson, D.R., "Design Concept and Development Plan of the Expedite Departure Path," Proc. of the AIAA's Aircraft Technology, Integration, and Operations (ATIO) Forum, Oct 1-3, 2002, Los Angeles, CA.

¹¹Wang, J.J., Chang, Paul, and Datta, K, "Life-Cycle Cost/Benefit Assessment of Expedite Departure Path (EDP)," Prepared for NASA Ames Research Center, Contract No. NAS2-98074, August 2003.

¹² Federal Aviation Administration Air Traffic Organization Operations Planning Research & Technology Development Office Air Traffic System Concept Development, AJP-66, "Integrated Arrival/Departure Control Service (Big Airspace) Concept Validation", September 2007.

¹³ DeArmon, J., Taber, N., Bateman, H., Song, L., Masek, T., Gilani, D., "Benefits Analysis of a Departure Management Prototype for the New York Area", Tenth USA/Europe Air Traffic Management Research and Development Seminar (ATM2013).

¹⁴ Tuinstra, E., Haschke, K., "Generic Operational Concept for Pre-departure Runway Sequence Planning and Accurate Take-Off Performance", 09 July 2009.

¹⁵Kistler, M., Capps, A., Engelland, S., "Characterization of Nationwide TRACON Departure Operations", 14th AIAA Aviation Technology, Integration, and Operations (ATIO) Conference, Atlanta, GA, June 16-20, 2014.

¹⁶Wanke, C. R., Callaham, M. B., Greenbaum, D. P., and Masalonis, A. J., "Measuring Uncertainty in Airspace Demand Predictions for Traffic Flow Management Applications," Proceedings of the AIAA Guidance, Navigation, and Control Conference, AIAA, Austin, Texas, August 2003.

¹⁷Krozel, J., Rosman, D., Grabbe, S., "Analysis Of En Route Sector Demand Error Sources", AIAA-2002-5016, AIAA Guidance, Navigation, and Control Conference and Exhibit, Monterey, California, August 5-8, 2002.

¹⁸Wong, G., "The dynamic planner: the sequencer, scheduler, and runway allocator for air traffic control automation", NASA/TM-2000-209586 (2000)

¹⁹Brinton, C. and S. Atkins, "A Probabilistic Modeling Foundation for Airport Surface Decision Support Tools," 2009 ICNS Conference, 13-15 May 2009.

²⁰Boyer, J., "Best practices for designing and implementing decision services, Part 1: An SOA approach to creating reusable decision services", June 13, 2012