COVER PAGE

Title of Design: A Tabu Search Approach to Tactical Runway Configuration Management

Design Challenge addressed: Airport Management and Planning

University name: College of William and Mary

Team Member(s) names: Jennifer Thorne

Number of Undergraduates: 0

Number of Graduates: 1

Advisor(s) name: Dr. Rex Kincaid

EXECUTIVE SUMMARY

Title: A Tabu Search Approach to Tactical Runway Configuration Management
Team: Jennifer Thorne, Graduate Student, College of William and Mary
Advisor: Dr. Rex Kincaid, College of William and Mary

Tactical Runway Configuration Management plans runway configuration (groups of runways) usage over a pre-specified time interval to minimize arrival and departure delays while taking into account different parameters including flight patterns, taxi plans, aircraft, weather, and airport usage. Currently, an exhaustive recursive search is used to test each runway configuration and determine the best possible management scheme. A tabu search routine has been implemented to improve the speed of the search for a high quality management scheme. When applied to a metroplex (a collection of airports) and evaluated several times during the day, the Tactical Runway Configuration Management optimization routine must run efficiently in order to provide timely information to air traffic managers. A challenge of Tactical Runway Configuration Management is adapting it to multiple airports. The fundamental goal remains the same for every airport: to select an airport configuration to maximize overall efficiency of the runways, airport surfaces, terminal airspace, and interaction of the airport with the National Airspace System. However, the way in which runway configuration decisions are made, airport surfaces are used, and terminal airspace is managed changes between airports. Despite these differences, a Tactical Runway Configuration Management optimization routine must be applicable to any airport (or metroplex) without much adjustment. Currently, the test cases are based on John F. Kennedy International Airport in New York City. We report the benefits of using a tabu search over a recursive one.

This work is supported through a National Aeronautics and Space Administration (NASA) grant, Airspace Systems Program through contract NNL09AA02B, with Mosaic ATM and by the Virginia Space Grant Consortium.

TABLE OF CONTENTS

I. PROBLEM STATEMENT AND BACKGROUND

A report by the Federal Aviation Administration (FAA) released in May 2007 provided an analysis of demand and future capacity at fifty-six U.S. airports. The report found that, even when factoring in planned improvements to increase capacity, four airports were operating at capacity in 2007 and by 2025 fourteen airports would be operating at capacity (Federal Aviation Administration & The MITRE Corporation, 2007). Air traffic in the United States is expected to triple by 2025, and airport improvements to increase capacity will be unable to fully meet this increased demand. In addition, airport expansion is often infeasible due to a number of reasons including space limitations, no-fly restrictions, and noise or environmental restrictions. Expansions can also be time-consuming and expensive.

Through minimizing delays and maximizing throughput, though, airport capacity can be increased without expansion. This can be achieved through appropriate airport and airspace management. There are many airport sub-problems that can be analyzed to improve airport throughput and increase airport capacity without expansion (Figure 1). One of these sub-problems is forecasting Runway Configuration Management decisions. Runway Configuration Management decisions determine what combination of runways is used at a given time. These decisions influence airport arrival and departure capacities. Deterministically forecasting Runway Configuration Management decisions proves difficult due to the dynamic and uncertain characteristics of the factors that govern such decisions; these factors include wind speed, wind direction, and the cloud cover ceiling. In addition, noise, no-fly restrictions, and other such environmental constraints play a central role in the Runway Configuration Management decision. Despite the system's dynamic nature, tools to schedule runway configuration changes throughout the day are necessary for the expansion of an airport's air traffic capacity. Runway Configuration Management models are an attempt to provide such a tool to air traffic controllers that will assist in the scheduling of runway configuration changes.

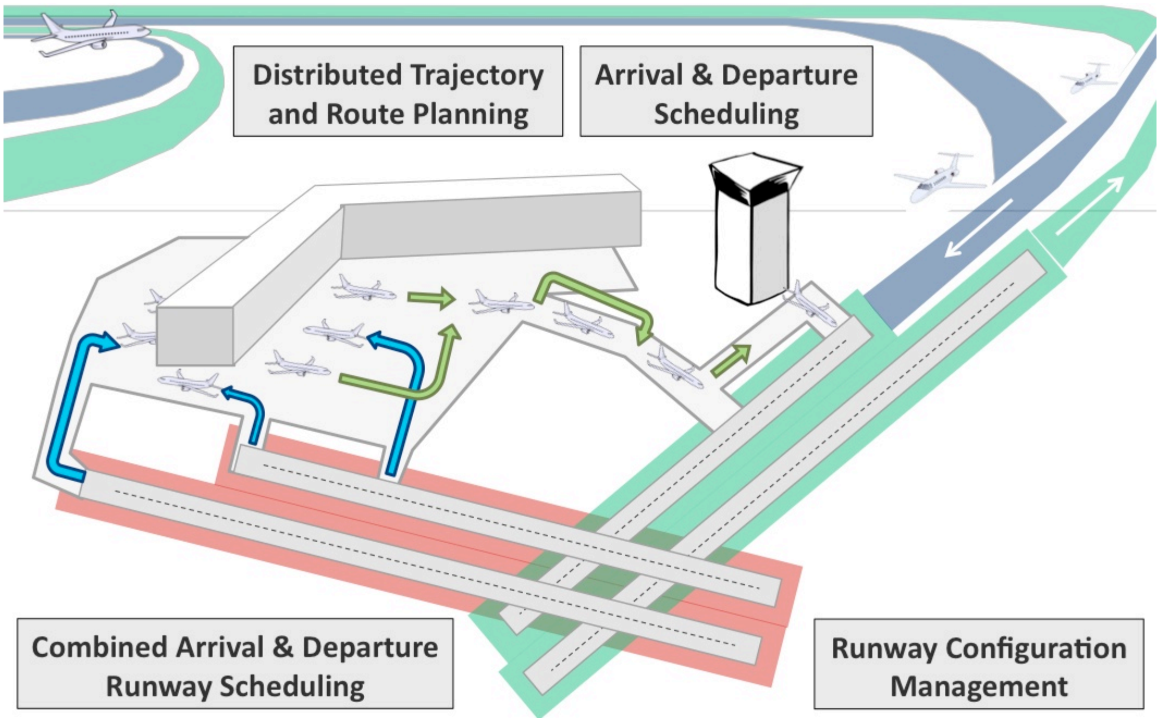

Figure 1: Airport Sub-problems

Currently, air traffic controllers make runway configuration decisions reactively, using current weather conditions, current arrival and departure demand, and imminent congestion issues (Provan & Atkins, 2011). Tactical Runway Configuration Management (TRCM) involves the development of a decision aid that will use current and future conditions to provide configuration management proposals to air traffic controllers (Provan & Atkins, 2011). TRCM is a real-time optimization tool that takes into account demand, weather, and the uncertainty inherent in both of these inputs

(Provan & Atkins, 2011). Such a system would result in improved configuration decisions by air traffic controllers, which would maximize airport profits, minimize delays in arrivals and departures, and maximize airport throughput. In addition, such a management system would improve airport configuration decisions and alleviate some capacity limitations that will plague airports as demand continues to increase. TRCM works within specified policy constraints to plan airport configurations that maximize the overall use and efficiency of airport surfaces and terminal airspace.

Mosaic Air Traffic Management, a company funded by National Aeronautics and Space Administration (NASA) to investigate improvements in air traffic management, has implemented a TRCM model that includes a recursive search routine to determine if and when a runway configuration change should be made during a specified planning period. This routine recursively iterates over the entire list of runway configurations available at an airport for each time interval in the planning period. While the search always finds a solution for a given time discretization, it can take an extremely long time to complete. The long runtime is an issue because TRCM also implements airspace and surface space planning routines, so the runway configuration decision is just a small part of the entire model. In addition, the model will be extended to a metroplex, a collection of airports in a metropolitan area, which means that the search will occur simultaneously for several airports in an efficient amount of time. With these two considerations, it becomes clear that the runway configuration search must be efficient in order to provide timely airport configuration recommendations to both airports and metroplexes.

II. SUMMARY OF LITERATURE REVIEW

Mixed integer linear programs (MILP) have been used quite extensively to model runway configuration management decisions at the strategic level, with a planning window of five or more hours. Capacity curves to represent the tradeoff between arrivals and departures for various runway configurations have also been developed to assist in such models. In these models, air traffic demand is aggregated into discrete time intervals, and the capacity curves are used to model different configurations. These models ignore surface and terminal airspace, allowing a string of scheduled flights determined by the aggregate demand to be completed according to the configuration capacities. (Gilbo, 1993; Gilbo, 1997; Bertsimas et al ,2005) The discrete time mixed integer linear program model has been expanded to include weather uncertainty in the configuration decision (Provan & Atkins, 2010; Durate et al, 2010; Zhang & Kincaid, 2011). The model including weather uncertainty served as the basis for an automated strategic planning tool prototype under NASA's System Oriented Runway Management concept (Provan & Atkins, 2011).

TRCM improves upon these MILP models in a variety of ways. The tactical planning problem in TRCM involves a planning window of less than five hours. The TRCM model also evaluates a shorter time interval than the previous models and includes individual flights rather than aggregate demand. In addition, TRCM includes airspace and surface space in decisions by considering factors such as taxi distance, fix distance, and gate assignments; the previous models were concerned solely with the available runways and their respective configurations. Finally, the TRCM model includes uncertainty in air traffic schedules and weather forecasts.

III. PROBLEM SOLVING APPROACH

III.1 Tabu Search Introduction

Tabu search is a heuristic procedure often used to find high quality solutions and avoid local optima in the solution space. It is a short-term memory process that uses a greedy search procedure subject to tabu restrictions. A greedy search algorithm makes the locally optimal choice at every stage of the search with the hopes of making it to the global optimal solution. Local searches procedures often find themselves stuck at local optima, so in order to avoid such local solutions, prevent cycling, and move toward a more global optimum, a list of attributes of recently visited solutions is maintained as a tabu list as part of the tabu search. (Glover, 1990)

III.2 Tabu Search Implementation

Both the original recursive search and the tabu search employ the same time discretization and use the same simulation routine to generate objective function values, which we seek to minimize. In many optimization problems, the objective function computation is inexpensive. However, the TRCM model relies on a time-consuming, scenario-based Monte Carlo simulation for objective function calculation. Therefore, one way to improve the runtime of the search routine is to limit the number of times this simulation is performed, which means reducing the number of configurations the simulation must evaluate.

A neighborhood of the current runway configuration is defined over the search interval, which is predefined based on the planning period and a set interval at which configuration changes can be made. The length of the search interval is defined in the parameters file for the search and is the same for both the tabu search and the recursive search. The neighborhood is defined using the flights that are scheduled to arrive and depart during the search interval and their respective taxi distances and gate assignments. For each flight in the search interval, the runway with the shortest taxi distance to or from the flight's gate is found. This runway is added to a list of good arrival or departure runways depending on the flight's designation as an arrival or departure. In addition to a gate assignment, each flight has a fix assignment; the fix distance for each flight is also checked, and if one runway has a fix distance above a certain threshold (currently set as the median distance for each fix) for more than half of the flights in the search interval, then that runway is removed from the list of good runways. Runways closest to the gate assignment and within the median fix distance for each flight in the search interval will limit the time a plane spends taxiing and the time a plane spends in the terminal airspace. Next, for each of the configurations at the airport, the arrival and departure runways of the configuration are checked against the list of good arrival and departure runways. If any arrival or departure runway of the configuration matches a runway on the respective list, that configuration is added to the neighborhood for the search interval. Once the neighborhood is established, the tabu search begins to look for improving configurations in the neighborhood.

Tabu search loops through the entire neighborhood of configurations. It begins by checking if a configuration in the neighborhood is on the list of previously simulated configurations. This list has a maximum length of six and is used to accelerate the search by eliminating the need to model a configuration change whose performance has recently been evaluated by the Monte Carlo simulation. When a configuration change is accepted and implemented by tabu search, the simulated list is reset to empty. Tabu search next checks to see if the configuration change is valid. A configuration change is valid if the configuration is different from the current one and the change time is allowable (configuration changes are only allowed after a certain amount of time has passed since the pervious change, this value is established by the user in the parameters file). If the configuration passes the validity check, the change is modeled and a new objective function is computed by sending the configuration change to the Monte Carlo scenario-based simulation. Then the configuration is added onto the list of simulated configurations. The new objective function value is compared to the current objective function value and the difference between these two values is compared to the best difference found so far. If the difference between the new objective value and the current value is greater than the best found so far, then the configuration change is checked to see if it is tabu. This check is performed by comparing the configuration's arrival and departure runways to the tabu lists of arrival and departure runways. If the configuration change is tabu, then the aspiration criterion is checked; this checks to see if the tabu move results in an objective function value that is better than the best objective found thus far. If the configuration change is not tabu or if the aspiration criterion is met, then the configuration change is accepted and implemented. The new objective function value is set to the value resulting from the configuration change. The tabu lists of arrival and departure runways are updated with the runways associated with the configuration change. If the tabu lists remain unchanged for three time intervals, then they are reset to empty to allow those configurations to be used again.

III.3 Alternative Searches for Runtime Improvement

Several additions to the tabu search and alternative searches have been implemented and tested for improved runtime and solution quality. To reduce runtime, the main goal of these alternative searches has been to reduce the search space in order to limit the number of calls to the scenario-based Monte Carlo simulation routine. A heuristic procedure has been developed as an alternative to both the recursive search and the tabu search. The heuristic creates a search neighborhood based on the number of arriving and departing flights in the search interval. If there are more arriving flights than departing flights, then configurations with the same or more arrival runways than departure runways are put into the search neighborhood. If there are more departing flights than arriving flights in the search interval, then configurations with the same or more departure runways than arrival runways are put into the neighborhood. The heuristic then proceeds by simulating every configuration in the neighborhood at each search interval and checking for objective function improvement. The configuration change that results in the best objective function is the recommended change.

A similar approach was also implemented to reduce the neighborhood size of the tabu search. Once the neighborhood is created, the number of arriving and departing flights in the search interval is compared to the number of arrival and departure runways. Any configuration that has more arrival runways than departure runways when there are more departing flights than arriving flights or vice versa is removed from the neighborhood. The search then proceeds in the same manner as the tabu search. This search is referred to as tabu search with neighborhood reduction.

A third search routine has been developed that allows only minor configuration changes for the first half of the planning period and then switches to tabu search for the second half of the planning period. For the first half of the planning period, only configuration changes that are in the same configuration group as the current configuration (minor changes) are simulated and checked for objective function improvement. Minor configuration changes involve the addition or removal of only one runway or a change of direction on one runway in the configuration. After the halfway point in the planning period, a tabu search is implemented using the closest runways to each flight's gate to develop the neighborhood of configurations to simulate, as was done before in the stand-alone tabu search routine. A search such as this was implemented because minor configuration changes are less expensive to implement than major changes; this is especially important early in the planning horizon where there may not be enough time to complete a major configuration change.

IV. SAFETY RISK ASSESSMENT

The automated TRCM system will reduce risks, increase safety, and provide a source of documentation of airport and runway configurations. An *Introduction Safety Management Systems for Airport Operators* (FAA Advisory Circular 150/5200-37) provides an overview of safety management systems and how airport operators implement such systems.

There are a few risks associated with the TRCM model that can easily be mitigated. One such risk is a poor configuration choice by the program, which could cause a back-up in arrival and departure traffic at the airport. However, air traffic controllers, who will analyze the output of the TRCM model for optimality, can mitigate this risk. The TRCM model with a tabu search for optimal runway configuration changes runs quickly and efficiently enough for runway configuration recommendations to be analyzed by air traffic controllers before the configuration change must be implemented. Another risk is bad weather inhibiting runway usage. The TRCM model contains a parameter for the probability that a pilot will balk for each runway at the airport, which is used by the model to make runway configuration decisions. This together with further analysis by air traffic controllers of the configuration choice made by the TRCM model will mitigate the risk of choosing a configuration that contains a runway whose use is inhibited by bad weather.

The automated TRCM system provides documentation of its runway configuration decisions, the times at which runway configuration changes should occur, and expected airplane delays, which allows for greater ease of safety reporting and auditing since the airport configuration choices and changes will be documented along with an explanation for the choice. Should an incident, such as a weather event or traffic diversion, occur, the automated TRCM system can be used to find the best new airport configuration given the altered circumstances in a short amount of time. The automated TRCM system could also be used in Safety Risk Management to analyze the airport configurations that would result from potential hazardous situations at the airport.

The *FAA Safety Management System Manual* provides guidance for the implementation of a Safety Management System and describes safety requirements of the Air Traffic Safety Oversight Service, International Civil Aviation Organization, and Air Traffic Organization. Safety engineers could use the automated TRCM system to analyze the results of different safety policies. For example, a runway closure could easily be implemented in the TRCM model and then flight lists could be analyzed using the model to see what configurations would result. The objective function values with and without the runway closure could be compared to analyze the effect of the closure on the number of delays in arrivals and departures.

Airplane separation tolerances for take off and landing are maintained in the TRCM search for the optimal airport configuration. The TRCM model is an error-tolerant system; some human errors in input to the model are detectable in error messages from the program. Errors in the flight list could alter the configuration recommendation without detection by the TRCM system, which is another reason why air traffic controllers must analyze the solution.

In the unlikely event that the TRCM software fails, air traffic controllers can still make runway configuration decisions; this provides redundancy for the system. To improve the ease of use and implementation of the TRCM model, a manual has been developed that provides a detailed explanation of the tabu search, including all variables and parameters used in the search and a description of each part of the search procedure. In addition, the code is extensively commented to explain usage and functionality.

In the event of a worst-case hazard such as extremely high levels of traffic or extreme weather disruption, the automated TRCM model could be used to find a new airport configuration given the new circumstances. The new heavy flight list could be input into the model, and the probability of a pilot balking could be adjusted to reflect the weather event. The tabu search operates in an efficient amount of time, so the TRCM model could output a new airport configuration for the given conditions quickly enough for the configuration to be analyzed and implemented. The airport could react in a more effective manner to such worst-case hazards with the automated TRCM system.

By having the TRCM model available on multiple computers the risk of software malfunction could be mitigated. There is also a risk of a common cause failure due to a bug in the software, but the model has been extensively tested with data from John F. Kennedy International Airport in New York City, and it appears to be quite robust and free of bugs. The automated TRCM model would also provide an easy way to input runway configuration changes into the Hazard Tracking System since solutions from the model could be easily copied and pasted into the System.

Overall the automated TRCM system will reduce risk because it will find more optimal runway configurations to reduce delays in arrivals and departures. Especially with arrivals, the reduction in delays will reduce risk since planes will spend less time in the air waiting to land. The reduction in airtime spent circling an airport will cut down on risk of fuel shortages, low airplane separation distances, and airplane flight path collisions.

V. TECHNICAL ASPECTS

The entire TRCM model is implemented in Matlab. Four different search routines were developed: a tabu search, a tabu search with neighborhood reduction, a tabu search with minor configuration changes, and a heuristic search. All four searches were compared to the original recursive search and to each other for runtime improvements and solution quality.

Test cases have been developed using John F. Kennedy International Airport (JFK) in New York City, New York. JFK has four runways and ten runway configurations (Figure 2). JFK is also one of the airports already operating at capacity as of 2007 according to an FAA Report (Federal Aviation Administration and The MITRE Corporation, 2007).

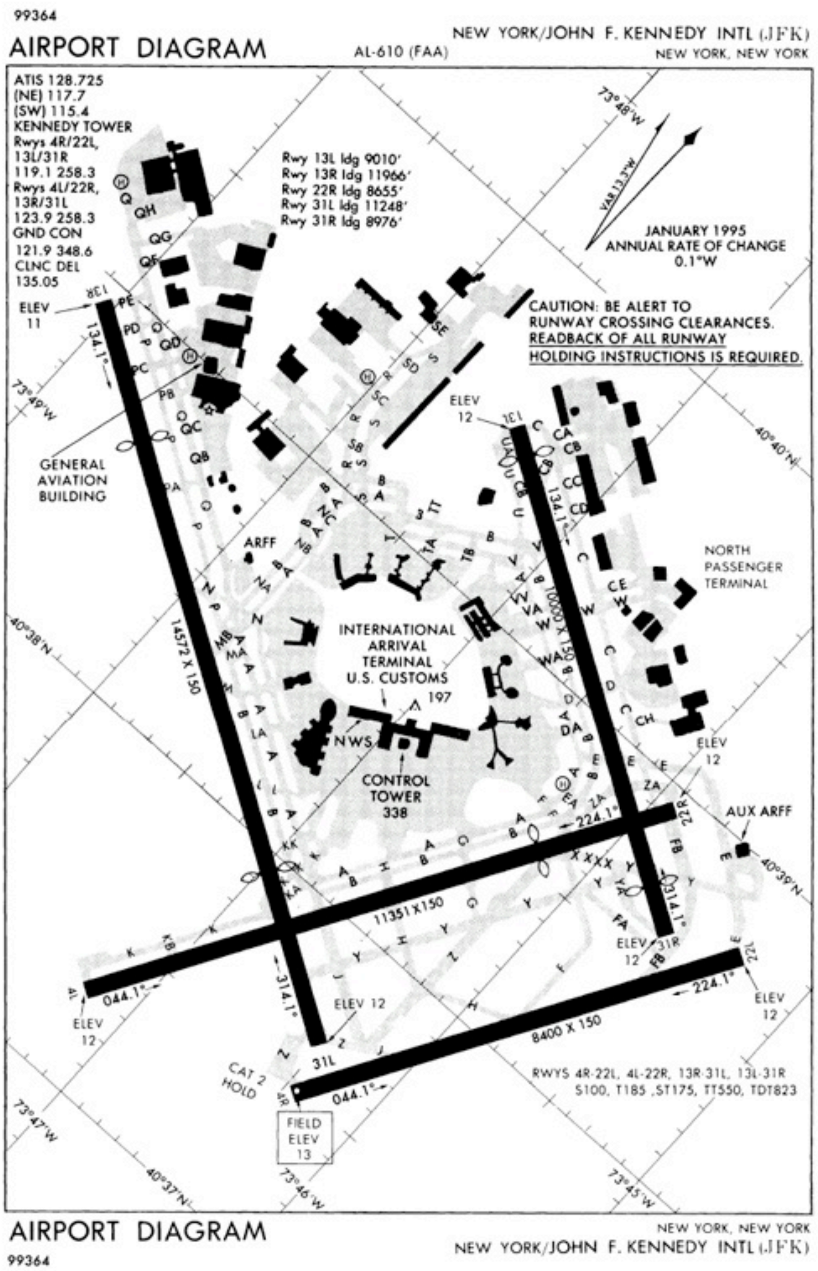

Figure 2: Airport Diagram of John F. Kennedy International Airport

The input parameter file for the TRCM model includes a planning horizon, a list of flights with their respective properties, the current runway configuration, and any runway configuration changes that have already been scheduled. The model outputs any recommended configuration changes with the configuration name and the time at which it should be performed. Configurations are combinations of runways with the arrival runways listed first, separated from the departure runways by a vertical bar.

The model also outputs the objective function value achieved by the recommended configuration change. This objective function value is computed by the scenario-based Monte Carlo simulation and is based on the schedule of flights that results from implementing the configuration change. The objective function value reflects the effective travel time of all flights in the input flight list. Lower objective function values are better since the effective travel time reflects any delays in arrivals and departures that result from the runway configuration schedule.

V.1 Test Case 1

The main scenario used for developing test cases consisted of an hour-long flight list with 79 flights. The planning period was 45 minutes long, and the search interval was five minutes. Configuration changes were allowed once every 30 minutes. These test cases were used to compare the tabu search to the recursive search, the heuristic search and minor search (the tabu search with minor configurations only allowed in the first half of the planning period) (Table 1). The starting configuration was held constant as 22L, 22R | 22L, and the configuration change set in the parameter file for 30 minutes from the start of the scenario was changed for each test case.

Config Change	Search	Run Time	Objective Value	Change Time	Config Name
13L,22L|13R	Tabu	*12.5275*	**105590**	3600	4R|4L,31L
	Recursive	19.5230	**105590**	3600	4R|4L,31L
	Heurisitic	13.7119	**105590**	3600	4R|4L,31L
	Minor	5.799	109490	5100	22L|22R,31L
31R|31L	Tabu	10.6907	104130	3900	4R|4L,31L
	Recursive	19.5298	**103610**	4500	22L|22R,31L
	Heurisitic	*10.6241*	**103610**	4500	22L|22R,31L
	Minor	5.2373	104190	None	None
4R|4L	Tabu	14.5119	**107910**	3600	4R|4L,31L
	Recursive	24.4490	**107910**	3600	4R|4L,31L
	Heurisitic	*12.0436*	**107910**	3600	4R|4L,31L
	Minor	8.0718	109670	4200	22L|22R,31L
22L|22R	Tabu	25.3524	**104510**	3300	31R|31L
	Recursive	39.0731	**104510**	3300	31R|31L
	Heurisitic	*21.7537*	**104510**	3300	31R|31L
	Minor	5.7201	109920	None	None
13L|13R	Tabu	*11.0383*	**113210**	3600	22L|22R,31L
	Recursive	22.1920	**113210**	3600	22L|22R,31L
	Heurisitic	17.6139	**113210**	3600	22L|22R,31L
	Minor	5.4739	118340	None	None
31L,31R|31L	Tabu	35.3495	125950	3600	4R|4L,31L
	Recursive	46.4063	**117870**	2700	31R|31L
	Heurisitic	31.9530	**117870**	2700	31R|31L
	Minor	*24.6349*	**117870**	2700	31R|31L
22L|22R,31L	Tabu	*9.8057*	**101800**	3600	4R|4L,31L
	Recursive	19.5723	**101800**	3600	4R|4L,31L
	Heurisitic	14.8119	**101800**	3600	4R|4L,31L
	Minor	5.6566	103090	4800	13L,22L|13R
4R|4L,31L	Tabu	10.2365	**104070**	None	None
	Recursive	17.5969	**104070**	None	None
	Heurisitic	14.2765	**104070**	None	None
	Minor	*6.951*	**104070**	None	None
None	Tabu	38.7276	**109640**	3000	4R|4L,31L
	Recursive	50.8127	**109640**	3000	4R|4L,31L
	Heurisitic	*31.9847*	**109640**	3000	4R|4L,31L
	Minor	12.8606	117240	None	None
4L,4R|4L	Tabu	24.5905	**116170**	3600	4R|4L,31L
	Recursive	33.5783	**116170**	3600	4R|4L,31L
	Heurisitic	*23.0248*	**116170**	3600	4R|4L,31L
	Minor	14.9237	119600	3000	4R|4L

Table 1: Results from a set of test cases where the start configuration was held constant as 22L 22R | 22R and configuration changes at 1800 seconds from the start of the scenario were altered. Bold-face indicates the best objective function value for each case and italics represents the best runtime of the searches that found the best objective function value.

Table 1 shows the results of ten test cases where the configuration change was rotated between each of the ten configurations at JFK.

The first column of Table 1 gives the configuration change 30 minutes into the scenario that was used in the test case. The second column displays the name of the search: tabu search, recursive search, heuristic search, or minor search. The third column shows the runtime in seconds of each search for each test case. The best runtime for the best objective function value is italicized for each test case. The objective function value is displayed in the fourth column, and the best objective function value found in each test case is shown in bold. The last two columns of Table 1 show the time and name of the configuration change that was recommended by each search for the test cases.

Averaging over the ten test cases in Table 1, the tabu search has a runtime of 19.2831 seconds, the recursive search has a runtime of 29.2733 seconds, the heuristic search has a runtime of 19.1804 seconds, and the minor search has a of 9.5239 seconds. The tabu search resulted in a runtime reduction of 34.13% compared to the recursive search. The tabu search found the same objective function value in eight of the ten test cases. This proves that the tabu search results in high quality solution and a reduction of runtime when compared to the recursive search for this test case.

The heuristic search has an average runtime that is very close to the tabu search, and it found the same solution as the recursive search in all ten test cases. Therefore, the heuristic search is shown to also be a high quality alternative to the recursive search routine. The minor search had the fastest average runtime of only 9.5239 seconds, but it only obtained the same solution quality as the recursive search in two of the ten test cases, which means that this search is not a viable option for a search routine to replace the original recursive search.

V.2 Test Case 2

Additional flight data was obtained from Mosaic Air Traffic Management to form test cases based on longer flight lists. A subset of a 24-hour flight list from JFK from November 11, 2008 was used to create a new scenario of just over two-and-a-half hours with 148 flights. Test cases based on this scenario were developed in a similar manner to the development of the test cases based on the original one-hour scenario provided for initial testing.

The planning period was 45 minutes long, and the search interval was five minutes. Configuration changes were allowed once every 30 minutes. The planning period began 45 minutes after the start time of the scenario. These test cases were used to compare the tabu search, referred to as Tabu 1 in Table 2, to the recursive search, the heuristic search and the tabu search with neighborhood reduction, referred to as Tabu 2 in Table 2 (Table 2). The starting configuration was held constant as 22L, 22R | 22L, and the configuration change set in the parameter file for 30 minutes from the start of the scenario was changed for each test case. Table 2 shows the results of ten test cases where the configuration change was rotated between each of the ten configurations at JFK.

Averaging over the ten test cases shown in Table 2, the tabu search had a runtime of 25.2633 seconds, the tabu search with neighborhood reduction had a runtime of 10.1506 seconds, the heuristic search had a runtime of 24.9373 seconds, and the recursive search had a runtime of 31.1387 seconds.

Config Change	Search	Run Time	Objective Value	Change Time	Config Name
13L,22L|13R	Tabu 1	22.0660	**114820**	3900	31R|31L
	Tabu 2	8.1676	115830	4200	31R|31L
	Heuristic	*20.7028*	**114820**	3900	31R|31L
	Recursive	26.7383	**114820**	3900	31R|31L
31R|31L	Tabu 1	18.7270	**112470**	None	None
	Tabu 2	*5.5766*	**112470**	None	None
	Heuristic	20.2933	**112470**	None	None
	Recursive	25.0680	**112470**	None	None
4R|4L	Tabu 1	22.7573	**115430**	3600	31R|31L
	Tabu 2	9.6333	115720	3900	31R|31L
	Heuristic	*21.6691*	**115430**	3600	31R|31L
	Recursive	26.9874	**115430**	3600	31R|31L
22L|22R	Tabu 1	33.5101	**113210**	3000	31R|31L
	Tabu 2	*10.8748*	**113210**	3000	31R|31L
	Heuristic	32.0480	**113210**	3000	31R|31L
	Recursive	41.4067	**113210**	3000	31R|31L
13L|13R	Tabu 1	22.8772	**117250**	3900	31R|31L
	Tabu 2	8.9151	118250	4200	31R|31L
	Heuristic	*21.4733*	**117250**	3900	31R|31L
	Recursive	27.4724	**117250**	3900	31R|31L
31L,31R|31L	Tabu 1	31.9523	125110	3300	31R|31L
	Tabu 2	12.4491	125110	3300	31R|31L
	Heuristic	*32.8593*	**121000**	2700	31R|31L
	Recursive	40.3871	**121000**	2700	31R|31L
22L|22R,31L	Tabu 1	*19.1748*	**112760**	3900	31R|31L
	Tabu 2	10.3162	112860	3600	31R|31L
	Heuristic	20.7993	**112760**	3900	31R|31L
	Recursive	23.5859	**112760**	3900	31R|31L
4R|4L,31L	Tabu 1	*20.6591*	**112880**	3600	31R|31L
	Tabu 2	9.5278	113240	3900	31R|31L
	Heuristic	21.3967	**112880**	3600	31R|31L
	Recursive	24.5782	**112880**	3600	31R|31L
22L,22R|22R	Tabu 1	34.9576	**115440**	3000	31R|31L
	Tabu 2	*11.2767*	**115440**	3000	31R|31L
	Heuristic	33.8469	**115440**	3000	31R|31L
	Recursive	43.7022	**115440**	3000	31R|31L
4L,4R|4L	Tabu 1	25.9515	**123870**	3600	31R|31L
	Tabu 2	*14.7683*	**123870**	3600	31R|31L
	Heuristic	24.2905	**123870**	3600	31R|31L
	Recursive	31.4607	**123870**	3600	31R|31L

Table 2. Results from test cases with a new scenario (a subset of JFK November 11, 2008) where the start configuration was held constant as 22L, 22R | 22R and configuration changes at time 1800 were altered. Bold-face indicates the best objective function value for each case and italics represents the best runtime of the searches that found the best objective function value. Tabu 1 refers to the original tabu search with no neighborhood reduction. Tabu 2 refers to the tabu search with neighborhood reduction by comparing the number of arriving and departing flights to the number of configuration arrival and departure runways.

For this scenario, the tabu search reduced the runtime by 18.87% when compared to the recursive search, and in only one of the ten test cases did the tabu search find a lower quality solution than the recursive search. The tabu search with neighborhood reduction had a much faster average runtime than the stand-alone tabu search, but it only found the same solution as the recursive search in half of the test cases. The heuristic search had a similar average runtime when compared to the tabu search, and it found the same solution as the recursive search in all ten test cases.

The test cases based on the longer flight list scenario confirm the high solution quality and runtime improvement of the tabu search and the heuristic search over the recursive search and show that the tabu search with neighborhood reduction is not a viable option for a routine to replace the recursive search.

V.3 Test Case 3

A third set of test cases was developed using the two-and-a-half hour scenario of 148 flights from November 11, 2008. These test cases were used to compare the tabu search, the minor search (tabu search with minor configuration changes for the first half of the planning period), the recursive search, and the heuristic search (Table 3). In these test cases, the start configuration was held constant as 22L, 22R | 22L and no configuration change was set in the parameters file. A sixty-minute planning period was shifted along the two-and-a-half hour scenario to create five test cases (Table 3). The planning period for each test case is shown in the first column of Table 3 and is given as a window of time in seconds, where the first number shown is the number of seconds that elapse from the start of the scenario to the start of the planning period and the second number is the number of seconds that elapse from the start of the scenario to the end of the planning period. The search interval was five minutes, and configuration changes were allowed every 30 minutes. Table 3 shows the results of the five test cases.

Planning Period	Search	Run Time	Objective Value	Change Time	Config Name
3100-6700	Tabu	167.3732	**188170**	3100	31R|31L
	Minor	71.1039	190410	5500	31R|31L
	Recursive	214.7264	**188170**	3100	31R|31L
	Heuristic	*130.3250*	**188170**	3100	31R|31L
3200-6800	Tabu	162.0301	**188010**	3500	31R|31L
	Minor	68.0013	190570	5000	31R|31L
	Recursive	203.9231	**188010**	3500	31R|31L
	Heuristic	*129.5150*	**188010**	3500	31R|31L
3300-6900	Tabu	158.6293	**188220**	3900	31R|31L
	Minor	65.7017	190550	5100	31R|31L
	Recursive	199.8760	**188220**	3900	31R|31L
	Heuristic	*124.0946*	**188220**	3900	31R|31L
3400-7000	Tabu	151.8619	**188180**	3700	31R|31L
	Minor	62.3074	190410	5500	31R|31L
	Recursive	192.1243	**188180**	3700	31R|31L
	Heuristic	*118.7478*	**188180**	3700	31R|31L
3500-7100	Tabu	147.3299	**188010**	3500	31R|31L
	Minor	59.8415	190700	5600	31R|31L
	Recursive	186.4582	**188010**	3500	31R|31L
	Heuristic	*115.2694*	**188010**	3500	31R|31L

Table 3: Results from test cases with a new scenario (a subset of JFK November 11, 2008) where the start configuration was held constant as 22L, 22R | 22R and the planning period was altered. Bold-face indicates the best objective function value for each case and italics represents the best runtime of the searches that found the best objective function value.

Across the five test cases, the average runtime for the tabu search is 157.4449 seconds, the average runtime for the minor search is 65.3912 seconds, the recursive search had an average runtime of 199.5216 seconds, and the heuristic search had an average runtime of 123.5904 seconds. In these test cases, the heuristic search had a much faster runtime than the tabu search, where in the previous two sets of test cases the tabu search and heuristic search had similar runtimes. The runtime reduction of the tabu search over the recursive search was 21.09% and the runtime reduction of the heuristic search over the recursive search was 38.06%. In all five test cases, the tabu search, recursive search, and heuristic search found the same solution. The minor search resulted in lower solution quality than the other three searches for all five test cases.

V.4 Test Case 4

Twelve hours of flight data was extracted from the daylong flight list from JFK on November 11, 2008. This data was used to create a scenario file and a longer test case to compare the tabu search and recursive search. For the tabu search, the search was run four times over scenarios of three hours from the twelve-hour flight list and a ninety-minute planning period. The entire flight list was used as the input file for the test cases with the scenario period adjusted for each case. For the recursive search, the search was run once over the entire twelve hours with a ninety-minute planning period. The recursive search was run twice, once with one configuration change allowed and a second time with two configuration changes allowed. All test cases were run with a starting configuration of 22L, 22R | 22R.

Tabu search was run with four consecutive three-hour intervals because the tabu search does not have the ability to make more than one configuration change at a time. The recursive search does have this capability, and consecutive configuration changes can be made. But the tabu search performs only one search for a single configuration change rather than recursive iterations to find (potentially) multiple changes. Therefore, the tabu search was run four times, and each time a configuration change was made, it was added to the parameters file of the next run.

Over the four scenarios, the tabu search found two configuration changes: one 181 minutes after the start time to configuration 22L | 22R, 31L and the second 275 minutes after the start time to configuration 31R | 31L. The four scenarios of the tabu search ran for a total time of 16.251 minutes and the total objective function value was 344,574.

The recursive search with one configuration change allowed ran for 57.9767 minutes and found one configuration change 180 minutes after the start time to configuration 22L | 22R, 31L. The objective function value was 373,190. The recursive search with two configuration changes allowed ran for 43.46 hours and found only one configuration change: 180 minutes after the start time to 22L | 22R, 31L. The same objective function value, 373,190, was found in this recursive search. Clearly, the tabu search in this scenario is preferred over the recursive search as the runtime is much improved (16.251 minutes versus 43.45 hours), and the tabu search found a better solution than the recursive search (objective function value of 344,574 versus 373,190).

V.5 Conclusions

Tabu search results in runtime improvement and the same or high solution quality when compared with the exhaustive recursive search originally implemented by Mosaic Air Traffic Management. The average runtime reduction over the three sets of test cases at John F. Kennedy International Airport of the tabu search over the recursive search is 24.69%. The heuristic search provides the same solution quality in all three sets of test cases when compared to the recursive search and reduced the runtime by an average of

30.82%. Tabu search with neighborhood reduction and the tabu search using minor configuration changes for the first half of the planning period both resulted in very fast runtimes but poor solution quality.

VI. INTERACTION WITH INDUSTRY EXPERTS

College of William and Mary professor Rex Kincaid provided constant support to the design and implementation of the tabu search and other search routines to improve upon the original recursive design. Dr. Kincaid has extensive experience with air traffic management and has advised students working on runway configuration management projects for the past three years. Weekly meetings between Dr. Kincaid and myself provided guidance for the search and test case development. Specifically, Dr. Kincaid supported the work by helping set goals and recommending new avenues for research.

Extensive communication with Mosaic Air Traffic Management occurred throughout the development of the design. Dr. Stephen Atkins is Vice President of Mosaic Air Traffic Management and a Principal Analyst. He has sixteen years of experience in aviation and air traffic management research and development. Christopher Provan, ABD, is a Senior Analyst at Mosaic Air Traffic Management and has extensive experience with air traffic and runway configuration management. Mr. Provan was responsible for the development of the original Matalab implementation of the TRCM model with the recursive search.

Dr. Kincaid and I corresponded with Dr. Atkins and Mr. Provan frequently during the course of the TRCM tabu search project. We had monthly teleconferences and frequent discussions over e-mail regarding design points and code inquiries. Mr. Provan provided the twenty-four hour flight data from John F. Kennedy International Airport for the design testing and assisted in the conversion of the flight data files into test cases for the various search routines.

Members of the Mosaic Air Traffic Management team corresponded with air traffic controllers during the course of the three-year contract from NASA to obtain firsthand knowledge of current runway configuration management practices. They spent several days at John F. Kennedy International Airport observing air traffic controllers and discussing current practices. This information has been vital for the development of an automated runway configuration management system because such a system must be easily integrated with current policies and easy for air traffic controllers to adapt to and use. Feedback from these conversations was used in the implementation and design of tabu search and other searches for an optimal runway configuration change. The TRCM model and search for an optimal configuration was written in a comprehensive and simple manner, and the runtime of the search has been reduced significantly by the tabu search and heuristic search so that configuration change recommendations are efficiently provided to air traffic controllers. In order for the TRCM model with tabu search to be implemented and used by air traffic controllers, it must be efficient and effective, the fast runtime and high solution quality shown in the test cases prove that the model meets these criteria.

VII. PROJECTED IMPACTS OF DESIGN

Implementation of a tabu search for an optimal runway configuration change improves the runtime performance of the original recursive search without significant detriment to solution quality. Runtime improvement is vital for real world application of the TRCM model because configuration decisions must be performed in real time and efficiently enough for air traffic controllers to analyze the recommendation and implement it at the airport or metroplex. As air traffic continues to increase, airport capacities will be reached, which creates significant concerns for airport operation. TRCM allows more efficient use of airport resources without airport expansion, which decreases the number of delays and increases the number of flights that can pass through the airport. In order to meet the increasing demand for air travel with fewer delays, a system based on the TRCM model is vital to assist air traffic controllers in managing surface and air space.

The TRCM model with tabu search allows for more efficient and optimal use of existing airport resources to better meet the current and future demand of air traffic. The model also provides a tool for air traffic controllers and other airport managers to plan runway configuration changes for upcoming airport and air traffic needs. Furthermore, the TRCM model can be used in situations when a deviation from normal airport operations occurs. If, for example, a weather incident occurs at a nearby airport and traffic from that airport is rerouted to another airport, the airport receiving the rerouted traffic can use the TRCM model with tabu search to quickly find an optimal runway configuration for the new increased traffic. The search runs quickly enough that a runway configuration change can be implemented to reflect the rerouted traffic within a sufficient amount of time to increase the airport's capacity so that delays in arrivals and departures will be minimal. As another example, if a runway must be closed for maintenance or due to wind speeds over its threshold, the TRCM model with tabu search can be used to find a configuration that does not include that runway but still optimally satisfies the current arrival and departure demand. An automated system for optimal runway configuration management such as the TRCM model with tabu search will result in more efficient use of airport resources and fewer delays in arrivals and departures.

Implementing the tabu search and TRCM model commercially would be relatively simple. The process would only require the alteration of the airport object within the Matlab code for the specific airport at which the model is to be implemented, which can be done by simply changing the code to reflect the runways and runway configurations available at the airport. After that alteration, the model could be implemented at any airport. Air traffic controllers would also have to learn how to use the model, but it is relatively simple and there are extensive comments regarding usage in the Matlab code. All one needs to do to run the TRCM model with tabu search is update the parameters file with a flight list (which would be readily available to an air traffic controller), the time horizon over which the search is to run, the planning horizon over which configuration changes can be made, and the frequency with which configuration changes can be made. Any previously scheduled configuration changes can also be input into the model through the parameters file.

The cost of implementing the TRCM model with tabu search at an airport will be relatively low. The only requirements for running the model are a computer and access to Matlab. A Matlab license can be expensive to obtain; for government or commercial use an individual license costs $2,150.00. However, the only other cost associated with implementation of the TRCM model with tabu search is training air traffic controllers to use the tool, and, as previously shown, the tool is relatively easy to learn, adapt to the airport, and use, so it should only take a few hours to implement. The 2010 median pay for an air traffic controller was $51.94 per hour (Bureau of Labor Statistics, 2012). Given this 2010 hourly wage and assuming the TRCM model would require eight hours for an air traffic controller to adapt the model to the airport and learn to use the software effectively, it would cost the airport $415.52 for proper training and adaptation. The total cost of implementation would then be $2,565.52.

There are many reductions in cost that would result from the implementation of the TRCM model with tabu search at an airport. The projected number of air traffic controllers to be hired by the FAA through 2021 is 11,747 (Federal Aviation Administration, 2011). The average cost of a developmental air traffic controller in fiscal year 2012 is expected to be $97,500 (Federal Aviation Administration, 2011). Naturally, this cost will increase over the years, but assuming it remains constant, the total cost of the FAA hiring projection would be $1,145,332,500 over the next 10 years. If the TRCM model was used to replace one new hire of an air traffic controller, then the cost savings would be $94,934.48 based on the expected average cost of a developmental controller in fiscal year 2012 and the total cost of implementing the TRCM model based on the 2010 hourly wage of an air traffic controller and the cost of a Matlab license.

Additional cost savings would be found in the TRCM model's minimization of delays in arrivals and departures. A recent report by researchers at the University of California, Berkeley found that the total cost of flight delays in 2007 was $32.9 billion (Ball, Barnhart, Dresner, Hansen, Neels, Odoni, Peterson, Sherry, Trani & Zou, 2010). This cost was broken down into several components including a $8.3 billion cost to airlines, a $16.7 billion cost to passengers, and a $3.9 billion cost due to lost demand

(Ball, Barnhart, Dresner, Hansen, Neels, Odoni, Peterson, Sherry, Trani & Zou, 2010). Clearly, the cost of flight delays for airlines and airports is significant, and the ability of the TRCM model with tabu search to efficiently find runway configuration changes that minimize the number of delays in arrivals and departures would provide an effective tool to reduce these costs.

The current method for runway configuration management is reactive and performed by air traffic controllers based on current air traffic demand. The TRCM model provides an automated system that can more accurately analyze current and future needs of an airport. In addition TRCM takes into account the uncertainty inherent in both air traffic and weather patterns, which allows for more optimal runway configuration management decisions. Such a system with a fast runtime and high solution quality provided by a tabu search implementation will become a vital tool in air traffic management as air traffic increases and airports are unable to expand to meet the increasing demand.

APPENDIX A – CONTACT INFORMATION

Student:

Jennifer Thorne
jathorne@email.wm.edu

Advisor:

Rex Kincaid
rrkinc@wm.edu

APPENDIX B – DESCRIPTION OF THE COLLEGE OF WILLIAM AND MARY

Chartered on February 8, 1693 by King William III and Queen Mary II of England, the College of William and Mary in Virginia is the second oldest college in the United States of America. Phi Beta Kappa, the nation's first academic Greek society, was founded at the College of William and Mary in 1776. The College of William and Mary hosts 6,071 undergraduate students and 2,129 graduate students from 49 states, the District of Columbia, and 61 foreign countries. There are 591 full-time faculty members across the undergraduate, graduate and professional programs at the College of William and Mary. The College also boasts a 12:1 student-faculty ratio. The College of William and Mary is well known for its liberal arts emphasis and for the fostering of undergraduate and graduate research. Among the many notable alumni of the College of William and Mary, there are three U.S. presidents: Thomas Jefferson, James Monroe, and John Tyler. The College of William and Mary is one of only eight U.S. institutions of higher education to earn the designation of "Public Ivy," a state-assisted institution that offers superior education at a lower cost than that of Ivy schools. (William and Mary, 2012)

Operations Research has been taught at the College of William and Mary for over thirty years. The current Computational Operations Research program has been in place since 1999 and involves faculty across many departments, including Mathematics, Computer Science, and Applied Science. Through the Computer Science Department at the College of William and Mary graduate students have the opportunity to earn a Masters of Science in Computer Science with a specialization in Computational Operations Research. In this two-year program, students gain knowledge in linear programming, discrete optimization, simulation, reliability, discrete-event simulation, deterministic and stochastic models, and statistics.

APPENDIX C – DESCRIPTION OF NON-UNIVERSITY PARTNERS

Mosaic Air Traffic Management is a small business that was founded in 2004 to improve the efficiency and safety of air transportation and air traffic systems. The company also seeks to advance the science and application of unmanned aircraft systems. Mosaic Air Traffic Management's team of engineering professionals works on a variety of government-funded, commercial, consulting, and internal development projects. Mosaic Air Traffic Management has its headquarters in Leesburg, Virginia.

Dr. Stephen Atkins
Vice President and Principal Analyst
Mosaic Air Traffic Management
atkins@mosaicatm.com

Christopher Provan
Senior Analyst
Mosaic Air Traffic Management
cprovan@mosaicatm.com

APPENDIX E – EVALUATION OF EDUCATIONAL EXPERIENCE

For Student:

1. Did the FAA Design Competition provide a meaningful learning experience for you? Why or why not?

Yes, the FAA Design Competition did provide me with a meaningful learning experience. Through the Competition I learned to analyze a design challenge presented by the FAA and apply my knowledge from coursework to a real world problem. I also learned how to design an effective algorithm that addresses the needs of air traffic controllers and runway configuration management.

2. What challenges did you and/or your team encounter in undertaking the Competition? How did you overcome them?

The main challenge I faced in undertaking the Competition was understanding and altering Matlab code written by another programmer. Employees at Mosaic Air Traffic Management wrote the original recursive search that we sought to improve. At times, it proved difficult to understand the inner workings of the search and the procedures it called. I overcame this challenge by asking many questions of the programmers at Mosaic Air Traffic Management and tracing the function calls in the program. Another challenge was obtaining new test cases to further analyze the tabu search and compare its results to the recursive procedure. The test cases had to be obtained from Mosaic Air Traffic Management and then reconfigured so that they could be input into the program. The test cases sent to us by Mosaic Air Traffic Management were designed for a different program, so the reconfiguration proved rather difficult.

Again, I overcame this challenge by frequent correspondence with Mosaic Air Traffic Management and deeper analysis of the entire set of Matlab function routines to understand the inner workings of procedures written by the programmers at Mosaic Air Traffic Management.

3. Describe the process you or your team used for developing your hypothesis.

To develop the hypothesis that we could improve the runtime of the recursive search for an optimal runway configuration using a tabu search procedure, my advisor and I thought of different heuristic techniques that are known to produce high quality solutions. We chose tabu search because it is quite simple to implement and involves creating a limited solution space in which the search is carried out. In addition, tabu search includes the maintenance of a tabu list, which prevents cycling through solutions that have recently been checked. These aspects of limiting the solution space with a rather simple heuristic search were very appealing and led us to develop the hypothesis of implementing a tabu search in place of the recursive one.

4. Was participation by industry in the project appropriate, meaningful and useful? Why or why not?

Participation by industry in the project was very appropriate, meaningful and useful. Mosaic Air Traffic Management, a company dedicated to improving the efficiency and safety of air transportation, sponsors the research. Without the support of Mosaic Air Traffic Management, we could not have accomplished this work. Frequent teleconferences with Mosaic Air Traffic Management employees provided insight into runway configuration management decisions and the original recursive search from which we built the tabu search procedure.

5. What did you learn? Did this project help you with skills and knowledge you need to be successful for entry in the workforce or to pursue further study? Why or why not?

This project definitely helped me develop the skills and knowledge I need to be successful for entry in the workforce and in my continuing studies. I gained greater skills with Matlab and algorithm development. I also gained hands-on experience with the application of a tabu search procedure. In addition, I have presented my work at two conferences and have written abstracts and papers for each of those conferences. I learned how to apply the theoretical knowledge in heuristics and algorithm development learned through my coursework to a real-world problem and how to effectively convey my work through oral and written methods.

For Faculty Member:

1. Describe the value of the educational experience for your student(s) participating in this Competition submission.

Jennifer joined my runway configuration project when it was entering its third, and final, year. Jennifer met weekly with me for 1 year. Each week she reported on her research activity and we discussed any problems that arose. Twice a month telephone conference calls were held with Mosaic ATM staff in which we reported on our progress and they made suggestions about our work.

Jennifer had completed my “Discrete Optimization” class. As a result, she understood the conceptual framework for the heuristics (tabu search) we proposed to use to improve upon the recursive search algorithm developed by Mosaic ATM for the tactical runway configuration management (TRCM) problem. Implementing tabu search for a specific airport (JFK) provided an important educational opportunity. She learned that there is a large fixed cost associated with learning and understanding a specific problem (TRCM in this case) and the details surrounding a specific data set (2 years of data for JFK). Classroom homework assignments are not so “messy” when all of the needed application details are provided in a short paragraph. Jennifer is now prepared to face “real” problem instances.

In addition, Jennifer learned how to find answers to difficult technical questions. The Mosaic ATM staff provided expertise in how air traffic control and runway configuration decisions are made, but they were not experts in the tabu search heuristic that Jennifer was coding. It is critical for analysts to be able to capture technical details with language that others can understand. Jennifer was able to bridge the language gap between experts who had key insights that she needed and her understanding of what these insights meant in tailoring the heuristic for this specific application.

Lastly, Jennifer’s presentation skills improved. She had the opportunity to present her research results at an academic conference, at NASA Langley Research center, in phone conversations with Mosaic employees, at the Virginia Space Grant Consortium, and weekly to me. As a result, she gained confidence in her ability to convey difficult technical information.

2. Was the learning experience appropriate to the course level or context in which the competition was undertaken?

The experience pushed Jennifer in several areas: computer programming, technical writing, and conveying technical material in a non-technical verbal format. These experiences included weekly progress reports, bi-monthly telecons, and multiple technical presentations. I believe all these experiences were appropriate and highly beneficial.

3. What challenges did the students face and overcome?

The initial challenge Jennifer faced was joining a project that had been underway for 2 years. I carved out a new direction for her so that the learning curve was not as steep as it might have been. I decided to have her attempt to improve upon the solution procedure that Mosaic ATM had developed for the Tactical Runway Configuration Management (TRCM) problem. Mosaic handed off the Matlab code they had used to prototype TRCM. As is common with prototyping code, there were not many comments and the role of several of the functions was unclear. Jennifer waded right in, but was forced to rely on email and telecon responses from Mosaic. In the end, Jennifer was able to understand the Matlab code, write a new subprogram for our tabu search heuristic, and produce good computational results.

4. Would you use this Competition as an educational vehicle in the future? Why or why not?

Yes, I would. However, since I am in a Mathematics Department I rely on NASA

Langley Research Center as a venue for finding projects amenable to the FAA competition.

5. Are there changes to the Competition that you would suggest for future years?

This is my first year to sponsor a submission. I think I should wait and see how my first submission turns out before offering suggestions.

APPENDIX F – REFERENCE LIST

Ball, M., Barnhart, C., Dresner, M., Hansen, M., Neels, K., Odoni, A., Peterson, E., Sherry, L., Trani, A.A., Zou, B. (2010). *Total delay impact study: A comprehensive assessment of the costs and impacts of flight delay in the united states*. University of California, Berkeley, Berkeley, CA. Retrieved from http://its.berkeley.edu/sites/default/files/NEXTOR_TDI_Report_Final_October_2010.pdf

Bertsimas, D., Frankovich, M., Odoni, A. (October 2009). *Optimal Selection of Airport Runway Configurations*. INFORMS Annual Meeting, San Diego, CA.

Bureau of Labor Statistics. U.S. Department of Labor. (March 2012). *Occupational Outlook Handbook, 2012-13 Edition. Air Traffic Controllers*. Retrieved from http://www.bls.gov/ooh/Transportation-and-Material-Moving/Air-traffic-controllers.htm.

Duarte, M., Weld, C. Kincaid, R. (April 2010). *A Runway Configuration Management (RCM) Model with Marginally Decreasing Transition Capacities*. Systems and Information Engineering Design Symposium, University of Virginia.

Federal Aviation Administration. (2011). *Air Traffic Controller Workforce Plan 2012-2021*. Retrieved from http://www.faa.gov/air_traffic/publications/controller_staffing/media/CWP_2012.pdf

Federal Aviation Administration. (February 28, 2007). *Introduction to Safety Management Systems for Airport Operators.* Retrieved from http://aviation.osu.edu/SMS/Content/Pdfs/Advisory%20circular.pdf

Federal Aviation Administration Air Traffic Organization. (May 2008). *Safety Management System Manual.* Retrieved from http://faadesigncompetition.odu.edu/Apps/FAAUDCA.nsf/SMSManual.pdf

Federal Aviation Administration and The MITRE Corporation. (May 2007). *Capacity Needs in the National Airspace System 2007-2025.* Retrieved from http://www.faa.gov/airports/resources/publications/reports/media/fact_2.pdf.

Gilbo, E. P. (September 1993). *Aiport Capacity: Representation, Estimation, Optimization.* IEEE Transactions on Control Systems Technology, Volume 1, Number 3, pp. 144-154.

Gilbo, E. P. (September 1997). *Optimizing Capacity Utilization in Air Traffic Flow Management Subject to Constraints at Arrival and Departure Fixes*. IEEE Transactions on Control Systems Technology, Volume 5, Number 5, pp. 490-503.

Provan, C.A., Atkins, S.C. (September 2010) *Optimization Models for Strategic Runway Configuration Management Under Weather Uncertainty*. Proceedings of the 10th AIAA Aviation, Technology, Integration, and Operations (ATIO) Conference, Fort Worth, TX.

Provan, C.A., Atkins, S.C. (May 2011). *Tactical Airport Configuration Management.* 2011 Integrated Communications Navigation and Surveillance (ICNS) Conference.

William and Mary. (2012). *College of William and Mary Website*. http://www.wm.edu/

Zhang, R., Kincaid, R. (2011). *Robust Model for Runway Configuration Management.* Working paper.