Airport Cooperative Research Program

University Design Competition for Addressing Airport Needs
(2015-2016 Academic Year)

Title of Design:

Multi-Objective Simulation-based Optimization of Runway Operations Scheduling Using a Hybrid Metaheuristic Algorithm

Design Challenge Addressed:

IV. Airport Management and Planning

University:

Old Dominion University

Faculty Advisor:

Dr. Ghaith Rabadi
Old Dominion University
Department of Engineering Management and Systems Engineering

Individual Student (Graduate):

Bulent Soykan
Old Dominion University
Department of Engineering Management and Systems Engineering
Ph.D. Student, Engineering Management

Spring 2016

Executive Summary

Runways are commonly considered as the main bottleneck for airports in general, and developing effective and efficient methods for runway operations scheduling is important for both increasing runway utilization and decreasing runway related delays, and in turn reducing costs and environmental effects. Our design solution addresses the runway operations scheduling problem taking into account the stochastic and complex nature of runway operations as well as fairness among airlines. In 2015, air traffic volume accounted for 34.58% of all aircraft delays in the US (FAA, 2016); therefore, the problem is motivated by a clear evidence of a steady increase in air traffic congestion and delays at the major airports. .

Our approach started by reviewing the literature, engaging with airport operators and industry expert sources, and developing the research hypothesis. Then, we designed and implemented a multi-objective simulation-based optimization framework, which utilizes a discrete-event simulation model for accounting for uncertain conditions, and an optimization component for finding Pareto-optimal solutions. This framework takes uncertainty into account to decrease the real operational cost of the runway operations as well as stakeholders' preferences as part of the optimization process. Due to the problem's large, complex and unstructured search space, a scatter search-based hybrid metaheuristic algorithm is developed to find solutions by using a dynamic update mechanism to produce high-quality solutions and a rebuilding strategy to promote solution diversity. Computational experiments are conducted using real-life actual datasets for a major US airport to verify & validate the design. The results show that the implementation of the proposed design provides operational benefits compared to First-Come-First-Served approach without compromising schedule fairness. The proposed decision-making algorithm might be used as part of decision support tools to aid air traffic controllers in solving real-life runway operations scheduling problem.

Table of Contents

List of Tables

List of Figures

1. Problem Statement and Background

The air traffic control (ATC) system is very complex and difficult to handle as a whole; hence, it is partitioned into several services. The primary objectives of these services are to ensure the safety of the operations and to maximize the effectiveness of the system. In major airports, the air traffic controllers commonly consist of four groups: (1) gate controller (assigns gate to aircraft and grants pushback clearance), (2) ramp controller (provides clearance for ramp and sequences aircraft at the ramp), (3) ground controller (issues taxi clearances and arranges departure/arrival taxi queue), and (4) local (tower) controller (assigns runway and start times for landing/take-off of arriving/departing aircraft). The arrival and departure air traffic is controlled by the gate controller in the gate area, by the ramp controller in the ramp area, by the ground controller between the spot and the holding area / runway exit and by the local controller between the holding area / runway exit and the runway. In our design, we focused on local (tower) controller's task of runway operations scheduling, which is a challenging task due to its highly complex and dynamic nature.

Current air traffic flow operations heavily rely on local controllers to sequence aircraft, schedule the landing/take-off times and issue clearance instructions to each aircraft, and adjust the schedules when necessary to maintain minimum separations between aircraft. However, a paradigm shift has been pursued to change this reliance on air traffic controllers as part of complete Air Traffic Management (ATM) system transformation both in the US and in Europe. FAA's Next Generation Air Transportation System (NextGen) program in the US, and Eurocontrol's corresponding Single European Sky ATM Research (SESAR) program in Europe are still under development. These new systems adopt new technologies, such as 4-dimensional ATM (trajectory control), performance-based ATM, satellite-based navigation as well as collaborative decision-making (CDM) concepts. The primary benefit of these technologies is that aircraft are expected to fly more exact routes in a more automated manner, thus improving system predictability and reliability, and increasing runway throughput and efficiency under varying demand and weather conditions. Therefore, the new concept of operations envisaged by NextGen and SESAR for the near future presents new opportunities for dealing with air traffic flow management with respect to the current operational environment.

Currently, there exist several air traffic tower automation tools to assist air traffic controllers manage air traffic flow. The one that is a surface surveillance system, is Airport Surface Detection Equipment-Model X (ASDE-X), which is used in major airports to help air traffic controllers maintain safe separation of aircraft and vehicles on the airport surface. The other one that is developed for scheduling arrivals is Traffic Management Advisor, which is already in-service at many US major airports. As part of FAA NextGen efforts, the Traffic Management Advisor will be replaced by an advanced decision support tools suite called the Tower Flight Data Manager (TFDM). The primary aim of TFDM is to serve as a platform for air traffic controllers to manage aircraft operations on the airport surface and in the TMA. There exists some runway sequencing and scheduling related decision support tools considered under TFDM. The capabilities of these two tools should be integrated in order to better schedule runway operations.

In Mehta et al. (2013), a recent study performed by MIT Lincoln Laboratory and sponsored by the FAA, the decision support functions of the latest TFDM prototype is analyzed and evaluated, and several performance gaps are identified. This study indicates that the greatest potential operational benefits would come from decision support tools that facilitate managing runway queues and sequences. In addition, this study provides evidence that outputs of the optimization approaches proposed in the literature are not operationally feasible, and in practice air traffic controllers still rely on first-come-first-served (FCFS) strategy for scheduling aircraft. In the same study, two major issues have been identified to be addressed in order to make the proposed approaches applicable to practical runway operations including computational time and the impact of uncertainties on the resulting optimization algorithms (Mehta et al., 2013).

a. Formal Problem Definition

The prominent problem in operations research literature which deals with scheduling aircraft over multiple runways is generally referred to as a multiple runway aircraft scheduling problem, denoted as MRASP. MRASP can be briefly defined as follows. Given a set of departing aircraft and another set of landing aircraft, where each aircraft belongs to a weight class, we need to assign each aircraft to a runway, and then determine the start time for each aircraft's operation (landing or take-off) on the assigned runway, while considering operational constraints, such as minimum separation times, time windows etc. MRASP is a large-scale scheduling problem that consists of a three-step process. The first step involves allocating aircraft to different runways, the second step is sequencing the aircraft allocated to each runway, and the third step is determining the operation start times for each aircraft. This problem arises usually at busy airports where runway utilization needs to be optimized to prevent delay-related costs.

The MRAPS has a short planning horizon, usually 20-30 minutes. For landing, each aircraft is considered as soon as it arrives to the extended TMA, which is about 30-40 minutes before its target landing time, and scheduled landing time is assigned before it reaches the final approach path, (about 20-30 minutes in advance of landing). For take-off, each aircraft is considered as soon as it enters the holding area, and this time period varies among airports depending on the characteristics of taxiway, holding area, and runways. But, most of the time, take-off sequence for a runway should be determined before an aircraft enters the taxiway, since it is not possible to change the sequence during taxiing or at the holding area. Hence, take-off operations are scheduled approximately 20 minutes before target take-off time (Julia A Bennell et al., 2013). Therefore, from a practical standpoint, a solution method to the MRASP should generate a runway operations schedule within the mentioned short planning horizon.

One of the main challenges of MRASP is the minimum separation times between aircraft, which result from wake vortices. Wake vortices are turbulences of air which are caused by a leading aircraft as a result of its lift force. The Federal Aviation Administration (FAA) and other

Civil Aviation Authorities around the world specify a set of minimum separation requirements in units of distance or time. The FAA enforced minimum separation requirements are largely determined by the type of operation and weight class of the leading and trailing aircraft. In the presence of interdependent multiple runways, these minimum separation requirements are asymmetric (non-triangular), where the sequence of operations determines the actual separation time. Therefore, generating efficient aircraft schedules by exploiting the asymmetric separation times has the potential to increase runway utilization and delay reduction. However, the existence of asymmetric separation times between aircraft makes this scheduling problem a non-trivial one.

From an air traffic controller's point of view, the easiest to use scheme for scheduling aircraft over runways is through the FCFS order. At the same time, FCFS is perceived as reasonably fair by most airlines. For the landing operation, this order is based on the order they enter the radar range, and for the take-off operation, it is based on the order of the aircraft queuing at the holding area. Although FCFS is an efficient strategy in terms of implementation, it does not, most of the time, produce the best schedule for runway utilization (Caprì & Ignaccolo, 2004). For major airports, low level of runway utilization typically leads to traffic congestion and delays, which, in turn, leads to inefficiency and waste of resources and negative environmental effects.

The runway operations scheduling problem is usually modeled as a mixed integer programming (MIP) problem, a set partitioning problem or an asymmetric traveling salesman problem with time windows. The single runway operations scheduling problem has been shown to be Non-Deterministic Polynomial-time Hard (NP-Hard), which means there is no known algorithm for efficiently finding optimal solutions to real-life problem sizes in polynomial time (Garey & Johnson, 1979). MRASP is also NP-hard because it is a generalization of the single runway scheduling problem. In addition to the problem's inherent computational complexity, the magnitude of the problem's difficulty is exacerbated by considering uncertainties and multiple conflicting objectives. Therefore, exact (optimal) solution methods are not capable of solving practical problem sizes and one of the main alternative solution methods is to use heuristic or metaheuristic algorithms.

b. Operational Constraints and Typical Objectives

The commonly considered operational constraints and typical objectives for the runway operations scheduling problem are briefly presented below:

Time windows: Once an aircraft enters the radar range for landing or pushbacks from the gate for take-off, air traffic controllers assign a runway to it and a start time for landing / take-off. The start time has to be between the predetermined earliest and latest land / take-off time, so-called "time windows," which is a hard constraint. Also, there is a target time to land / take-off within this time window, which is the time that aircraft can land if it flies at its cruise speed for landing, and the most probable time for take-off considering the taxi-out and holding times for take-off.

Minimum separation requirements: This is the principal safety constraint that needs to be taken into account in a runway sequence, which is the spacing (time interval) between successive aircraft and it has to be equal or greater than the minimum requirement stated by the FAA. This spacing requirement is required for the wake turbulence to dissipate, and it is the main constraint that renders the problem challenging.

Limited flexibility in deviating from the FCFS order: In practice, air traffic controllers often simply depend on FCFS strategy, which is the most straightforward and widely used approach. Although FCFS order eases air traffic controllers' workload, maintains a sense of fairness among airlines and is easy to implement, most of the time it is not capable of providing the best schedule in terms of runway utilization. In practice, deviating from FCFS order is not common.

Typical Objectives: Different objectives are utilized in the literature considering various stakeholders' point of view. It is not practical to address the interests of all the stakeholders at the same time. Hence, the most commonly used ones are minimizing the total delay (tardiness), minimizing the total deviation from the target time (earliness and tardiness), minimizing the average delay per aircraft, and maximizing the throughput (makespan which is the landing or take-off time of the last aircraft). Total weighted tardiness measures the cost of delay that is a function of the length of delay multiplied by a weight (penalty) value related to each aircraft, and it is capable of addressing different stakeholders' needs. This objective is also very important for airline companies since every second the aircraft waits to land or take-off increases operating cost.

c. Consideration of Practical Aspects

A comprehensive literature survey, a summary of which is presented in the next section, shows that there is a considerable gap between practitioners and academic researchers in the field of runway operations scheduling. Academic researchers are often not aware of the real-world complexities encountered by the industry practitioners, namely air traffic controllers, and, in turn, they usually do not take into account most of the practical aspects of the problem. Most of the academic research conducted have the following assumptions:

(a) They assume that all aircraft are already present in the holding area for landing or take-off, and they also assume to have precise and reliable data on the aircraft.

(b) They consider mostly the deterministic problem in which the presence of uncertainties in actual runway operations are ignored to limit the computational complexity of the problem.

(c) They highly focus on considering a single objective and do not consider interests of different stakeholders and Collaborative Decision Making (CDM) aspects.

However, air traffic controllers are faced with daily challenges where uncertainties are real and a trivial change in environmental conditions or small variations in implementation can be critical to operational safety and performance. In addition, finding the trade-offs between interests of different stakeholders is a key characteristic of the real-life problem. For example, concentrating only on runway utilization can cause unacceptable delays for individual aircraft, and in turn, this can impair the operational efficiency of the airline to which the aircraft belongs. Hence, the real-life runway scheduling problem involves several contradicting objectives that need to be satisfied simultaneously, the most important of which are maximizing runway utilization and maximizing fairness among all airlines.

Essentially, runway operations scheduling problem is an applied area of research and its benefits are ultimately derived from the results it achieves. However, by no means does this imply that the theoretical elements of this problem are not worthy of rigorous and careful treatment. Therefore, both theoretical and practical aspects of the problem are attempted in this design. To this end, in this design the following practical elements have been taken into account in order to bring the problem closer to the real-life practical applications: (a) uncertainties inherent to runway operations, (b) multi-objective nature of the problem, and (3) fairness among airlines that use the airport.

2. Summary of Literature Review

Since the 1960s, developing efficient methods for tackling runway operations scheduling problems has been of a great interest to both academic researchers and industry practitioners. Julia A. Bennell et al. (2011); (2013) have provided a comprehensive review of the problem. The solution methods for runway scheduling problems can be classified as exact and heuristic algorithms. Exact algorithms, such as branch-and-bound (B&B), dynamic programming (DP) etc., guarantee optimal solutions, but they are extremely computationally intensive for large problem instances. On the other hand, heuristic algorithms generate solutions which are not guaranteed to be close to the optimum and the performance of heuristics is often evaluated empirically. These algorithms are generally more time efficient.

a. Robust and Stochastic Approaches

Even though most of the previous research focused on deterministic runway operations scheduling, there are two robust and stochastic models in the literature that consider the uncertainties inherent to runway operations: NASA Ames Research Center (Chandran & Balakrishnan, 2007; Gupta et al., 2011) and Georgia Institute of Technology (Solveling & Clarke, 2014; Solveling et al., 2011).

NASA Ames Research Center researchers considered a runway schedule as “robust” if there is a high probability that an air traffic controller does not have to interfere once the schedule has been determined. They considered two conflicting objectives: maximizing runway throughput (or minimizing makespan) and maximizing reliability. They only considered the landing problem on a single runway and, therefore, assumed that the separation times satisfy the triangle inequality for all aircraft types. As a solution algorithm, they proposed a dynamic programming approach which is computationally efficient enough for a real-time application which schedules aircraft while limiting the number of positions an aircraft can move from its FCFS position. Their algorithm calculates a trade-off curve between throughput and the probability that random deviations of aircraft from the scheduled landing times that violate operational constraints. However, in their algorithm, the effect of uncertainties related to push-back times, wheels-on times and taxi predictions are not taken into account explicitly.

Georgia Institute of Technology researchers addressed the stochastic airport runway scheduling problem in which a set of aircraft are to be scheduled on one or multiple dependent runways. They developed a two-stage stochastic integer program and a solution method using scenario decomposition based on Lagrangian relaxation. Also, a stochastic branch-and-bound algorithm is proposed, which is a sampling-based approach in which the stochastic upper and lower bounds are generated. The proposed models of the stochastic runway scheduling problem correspond to single machine scheduling problem with probabilistic release times (and due dates) and sequence-dependent setup times.

In both of these approaches, actual operational distributions are not used but representative probability distributions are utilized. In addition, the proposed approaches are not applied to real-life large-scale problem instances. The fundamental consequence of such assumptions is that these approaches are not yet mature enough for operational deployment (Mehta et al., 2013).

b. Collaborative Decision Making and Fairness

CDM concept is a means to collaborate and share real-time operational information to improve situational awareness and decision making. It has the potential to improve TMA operations by allowing airlines to participate in air traffic decision-making that affects them. In the implementation of the NextGen, CDM concept is considered important for enhancing operational effectiveness through increased information exchange among stakeholders and consideration of desired intents. Even though final decision-making authorities are Air Navigation Service Providers (ANSPs), the involvement of other stakeholders, in particular, airlines has the potential for considerable benefits.

One of the main considerations within CDM concept is that ANSPs need to ensure that the outcome of runway scheduling is perceived by all airlines as fair. However, fairness is a significant challenge in terms of defining clearly what is considered as fair by airlines. Roger George Dear (1976) developed the heuristic methodology of Constrained Position Shifting (CPS), which limits the number of positions an aircraft can be moved from its FCFS ordered position, to make the schedule scheme fair. The maximum allowable number of position shifts is determined through a parameter called maximum position shifting (MPS). They examined and tested its effectiveness for several objective functions and concluded that by limiting the MPS to a small number, typically 2 or 3, it is possible to achieve most of the potential benefits.

Roger G Dear and Sherif (1991) and Venkatakrishnan et al. (1993) presented dynamic programming algorithms based on CPS. Recently, Balakrishnan and Chandran (2010) developed a shortest path problem algorithm based on CPS using a discretized network. Soomer and Koole (2008) demonstrated various definitions of fairness by using the aircraft landing problem, which includes absolute fairness, relative fairness, and fairness measured by delay. The proposed MIP formulations that include fairness is solved by local search heuristics. Also, computational experiments are conducted to assess how the fairness definitions and solution heuristics behave with real life problems. The results of these experiments demonstrate that it is possible to attain more fairness while still obtaining considerable cost compared to the FCFS schedule.

As a conclusion, based on the literature review and to the best of our knowledge, there is no work reported that deals with the multiple runway operations scheduling problem under uncertainty with utilizing a simulation-based approach. In addition, fairness is not taken into consideration during the optimization process as a second objective along with runway utilization, which converts the problem to a multi-objective optimization problem.

3. Problem Solving Approach and Design Details

Simulation is commonly considered as a cost-effective and powerful method for modeling stochastic and complex systems. Simulation studies are conducted to simply determine an estimate of the system output from a set of system input configuration. However, the simulation does not include the capability to search for a set of system inputs that can produce the optimal or near-optimal system output. Hence, an optimization procedure needs to be incorporated into simulation models in order to empower it with an optimization capability. Numerous cases have been reported in the literature in which simulation and optimization methods were combined successfully, and these efforts extended the growth of research in the field of Simulation-based optimization (SbO).

The main strength of SbO methods is that they can consider the dynamic and stochastic nature of the real-life problem while optimal or near-optimal solutions can be obtained without much of computational intractability. However, these methods still face challenges especially when there exist multiple and conflicting objectives, and they typically require costly development process and difficult verification & validation process. Due to the fact that simulation is not an optimization tool, in essence, simulation experiments require to be designed in a systematic way for analysts to understand the simulation model's behavior. The overall problem-solving approach is based on a SbO approach, as presented in Figure 1.

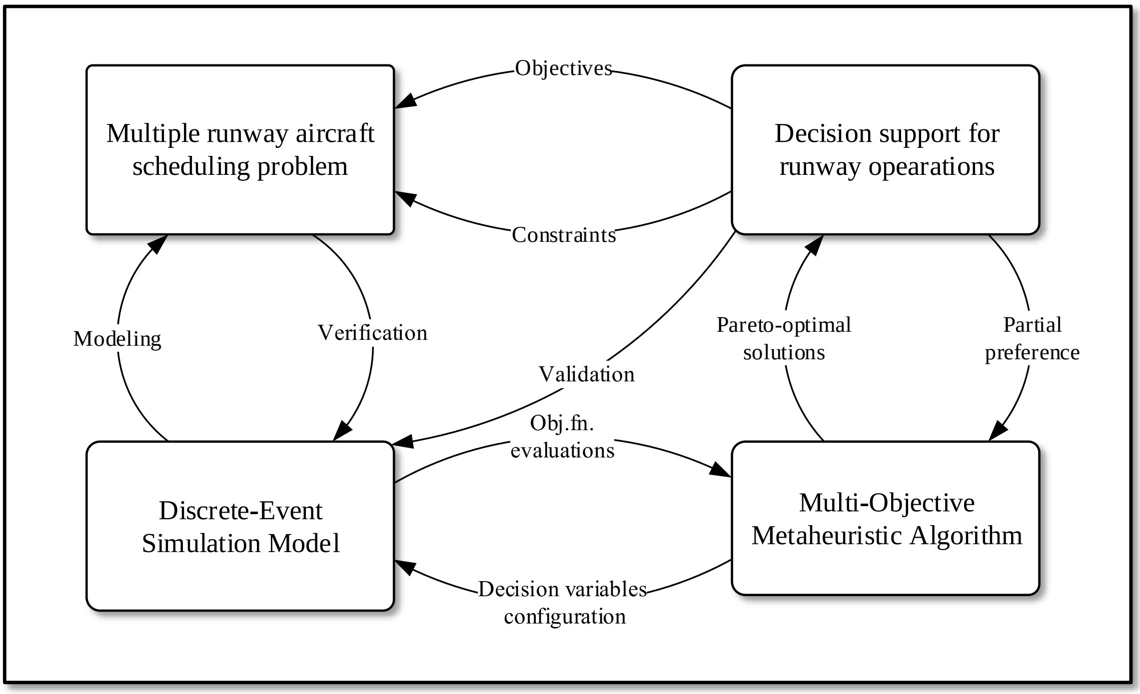

Figure 1: The Overall Problem SbO Approach

a. Simulation-based Optimization Framework

The concept model for the SbO framework is given in Figure 2. The SbO framework starts with a greedy heuristic algorithm to produce a relatively good initial solution compared to a random one. The simulation model outputs two performance measures from a single simulation run where it is treated as a black-box model that evaluates the performance of a particular configuration of system parameters and provides these performance measures as bi-objectives. The optimization component employs a metaheuristic algorithm that is discussed later to search for the values of system parameters. It is an iterative process initiated by the optimization algorithm starting with inputting the initial solution and generating a set of candidate solutions which act as input values for the simulation model. After receiving the input values from the optimization algorithm, the simulation model is executed to compute the performance measures (including optimization objectives and other output parameters of interest) which are then fed back into the optimization algorithm.

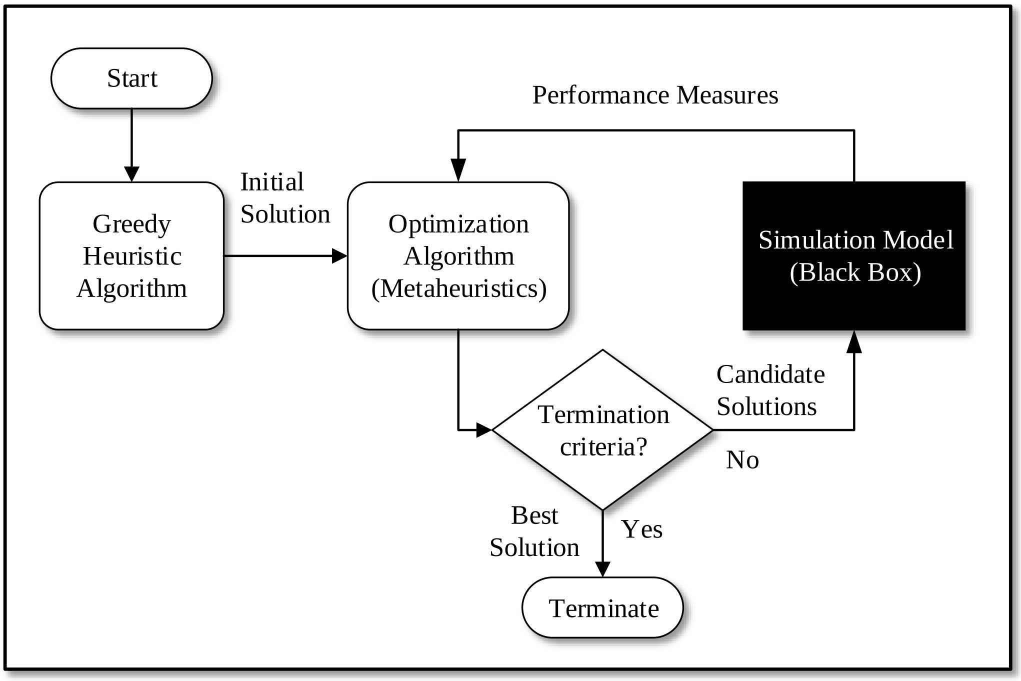

Figure 2: Schematic Representation of SbO Framework

The proposed SbO method is comprised of two main components: an optimization model for managing the search process and a simulation model for evaluating the performance of candidate solutions. The results of the performance evaluation are used to refine the optimization process and to return the best solution at the end. In this method, a simulation model is utilized as a replacement for an analytical fitness function in order to better mimic the behavior of the real-world runway system as well as to account for uncertainty.

This process is iterated until a termination criterion is met. Therefore, the whole process treats the simulation model as a black box where the optimization algorithm feeds candidate solutions to the simulation model which then generates the performance measures back to the optimization model. Since metaheuristic algorithms do not make explicit assumptions about the underlying structure of the objective function, the black box nature of the simulation model does not create any difficulty. In this research, a metaheuristic method namely Scatter Search, is used as the optimization component.

b. Greedy Heuristic Algorithm for Initial Solution Generation

In MRASP context, an elementary dispatching rule is a function of attributes of an aircraft, such as earliest, latest and target time. The overall priority of an aircraft is influenced by an attribute of the aircraft that is mainly determined by a look-ahead (scaling) parameter. A look-ahead parameter scales the contribution of each part of the composite dispatching rule relative to the total. This parameter has to be suitable for the problem instance at hand to get good quality solutions and it is determined empirically.

The greedy heuristic algorithm for initial solution generation is based on a composite dispatching rule and has been initially presented in Soykan and Rabadi (2016). It exploits the structure of the problem and includes attributes of aircraft (i, j), such as earliest time (r_j), latest time (d_j) and separation times (s_{ij}) as parameters. The overall priority of an aircraft is influenced by an attribute of the aircraft that is mainly determined by a look-ahead (scaling) parameter discussed earlier. This parameter is determined empirically and validated in terms of its suitability for practical problem instances to get high-quality solutions (Hancerliogullari et al., 2013). In this composite dispatching rule, aircraft are scheduled one at a time, i.e. when a runway becomes free, a priority index is computed for each remaining aircraft and the aircraft with the highest priority index is then selected to be scheduled next. The priority index, that is a function of the time t, is defined as follows:

$$\eta_{jr} = w_j \exp\left(\frac{-\max(d_j - t, 0)}{k_1}\right) \exp\left(\frac{-s_{ij}}{k_2 \bar{s}}\right) \exp\left(\frac{-\max(r_j - t, 0)}{k_3}\right) \qquad (1)$$

where j is the aircraft to be scheduled; i is the previous aircraft; r is the runway; w_j is the weight parameter for j, and k_1, k_2 and k_3 are the scaling parameters.

The main idea in the greedy heuristic algorithm is to generate a better quality initial solution in which the aircraft are sorted by the priority index and are considered to be landing or take-off on the runway at its best available time one after another. The pseudo-code of the greedy heuristic algorithm is given below.

Algorithm 1 Greedy Heuristic Algorithm for Initial Solution

Input: List of aircraft and number of runways, M

begin
    Initialization
    sort aircraft in accordance with the look ahead priority index (1 to N)
    for $i = 1$ to N
        for $r = 1$ to M
            calculate E_{ir} (Earliest feasible time that aircraft i can land on or take-off from runway r)
        end for
        calculate start time $s_i = min\ \{E_{ir} \mid r \text{ in } M\}$
        assign aircraft i to the runway related to calculated s_i
    end for
    calculate the fitness value (obj.fn.value)
end

Output: A feasible solution consists of sequence of aircraft over runways with a fitness value and start time for each aircraft

c. Discrete-Event Simulation Model

Discrete event simulation modeling is a powerful and widely accepted method for analyzing airport and runway operations. There exist several simulation models that simulate runway operations, and these models are usually utilized for evaluating a performance measure, such as throughput, resource utilization etc. However, these existing simulation models and tools for quantifying runway operations' performance necessitate a vast amount of detailed data. Also, they are not open source and cannot be used as a simulation component in a SbO framework. Therefore, a new discrete event simulation model is developed that is capable of capturing the essential interactions of key system components, representing the system with sufficient detail, and reflecting the uncertainty associated with the runway operations.

In order to realistically represent the components of the real runway system and their complex interactions, system approach is adopted in developing the simulation model, and an object-oriented architecture is employed. This architecture is designed to simulate the flow of air traffic for both arrivals and departures at the macro level, where aircraft are generated as objects that move through the runway threshold and airspace segments. The simulation model is built to compute the performance measures, which are the outputs for optimization component, by tracking the flow of aircraft in specific points and the time duration between these points.

In the simulation model, the TMA is represented by a network of nodes and arcs. Aircraft move on this network along prescribed trajectories that are made up of strings of nodes and arcs, where each arc can be occupied by a single aircraft at a time. Accordingly, whenever an aircraft tries to use an arc that is already occupied by another aircraft, delay takes place. The primary nodes that are located in this network representation are shown in Figures 3 and 4.

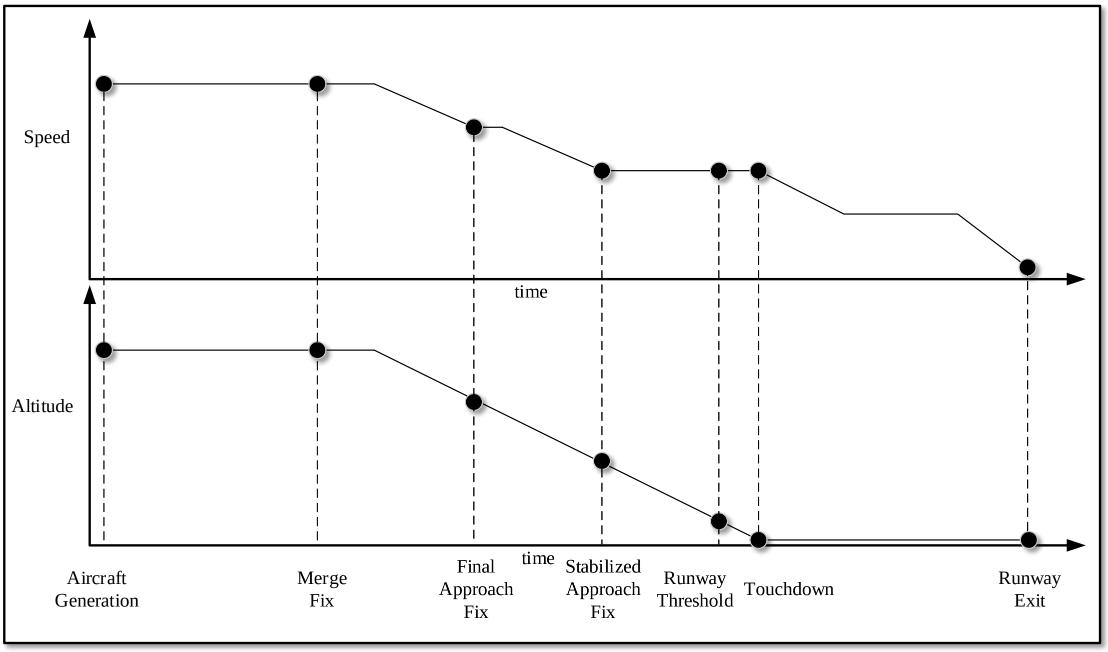

Figure 3: Primary Nodes for Arrival

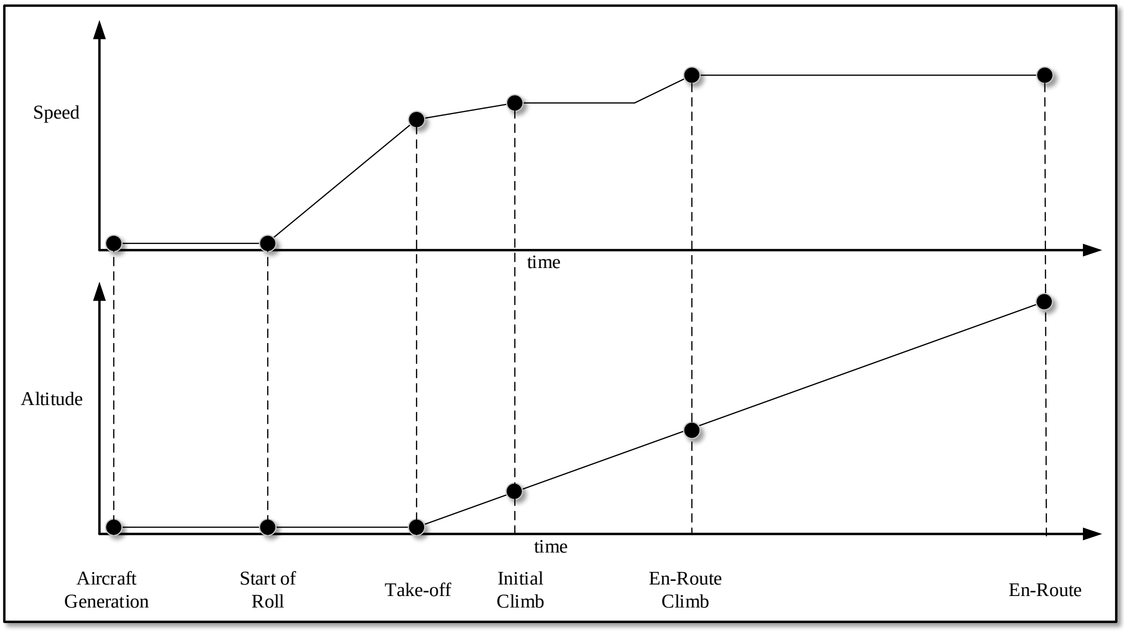

Figure 4: Primary Nodes for Departure

The primary advantage of this network structure is the detailed representation of the TMA and convenience in collecting various statistics. Essentially, the state of the simulation model changes only in discrete points in times, which is typically referred to as event times. The design parameters for the discrete event simulation model are listed in Table 1.

Table 1: Simulation Model Design Parameters

Design Parameter	Description
Runway layout	Number of runways and configuration
Runway procedures	The way runways are operated
Fleet mix and operating sequence	The percentage of operations among all aircraft weight classes and their arrival/departure sequence
Weather	Visibility
Minimum separation requirements	The required minimum distance/time between leading and trailing aircraft
Final approach length and speed	The distance of the final approach and the speed of the aircraft

Due to the fact that minimum separation times are the primary requirement that needs to be enforced between aircraft, it is fulfilled by adjusting the speed of aircraft accordingly. In each node that is located in this network representation, the separation time between the aircraft is checked based on time-based distance. If the separation time is not obeyed, then the following aircraft will be slowed down until the separation time is obeyed.

The holding pattern, which is an airspace section for aircraft waiting to continue the final approach to the runway, is modeled as the queue for the landing aircraft where aircraft fly at a certain speed. In the simulation model, the holding pattern is used as an indicator for infeasibility. If more than a specific number of aircraft exist in the holding pattern at the same time or if any aircraft spends more than a threshold of time in the holding pattern that schedule is considered infeasible.

Performance measures to be collected are runway utilization and fairness. *Runway utilization*: Utilization of runways is calculated for every 15-minute interval as a percentage of time in each interval for which runways are being used for active runway operations. The types of active runway operations are listed as follows: (a) final approach, which is the time an aircraft enters to final approach fix to the time of touchdown, (2) runway occupancy for landing and take-off, which is the time between touchdown and leaving the runway for arriving aircraft, and the time between start of takeoff roll to wheels off for departing aircraft. *Fairness*: This performance measure is motivated by social justice, where scheduling using the arrival order is considered fair. Therefore, this metric attempts to measure the deviation from FCFS order.

As part of simulation model development, an iterative verification and validation process is used to determine whether the simulation model is valid to an acceptable level. Several versions of the simulation model were developed until a valid simulation model was obtained. The simulation model's validity was gradually improved through the process by increasingly building confidence in the accuracy of the model by applying verification and validation tests. Verification study sought to show the simulation model performs as expected and provides an accurate logical representation of the conceptual model. On the other hand, validation is established when the model's behavior validly represents the real runway system being simulated. Subject matter expert judgment constituted a crucial component of the validation process. Statistical comparison of numerical values of the output performance measures to the real-life runway system was conducted. The total waiting time and the landing/take-off time of each aircraft were used as an index for the simulation accuracy. Given the complexity of runway operations, validating the discrete event simulation model is a challenge, and several possible scenarios are simulated to overcome this challenge.

A high-level (framework) block diagram of the discrete event simulation model is shown in Figure 5.

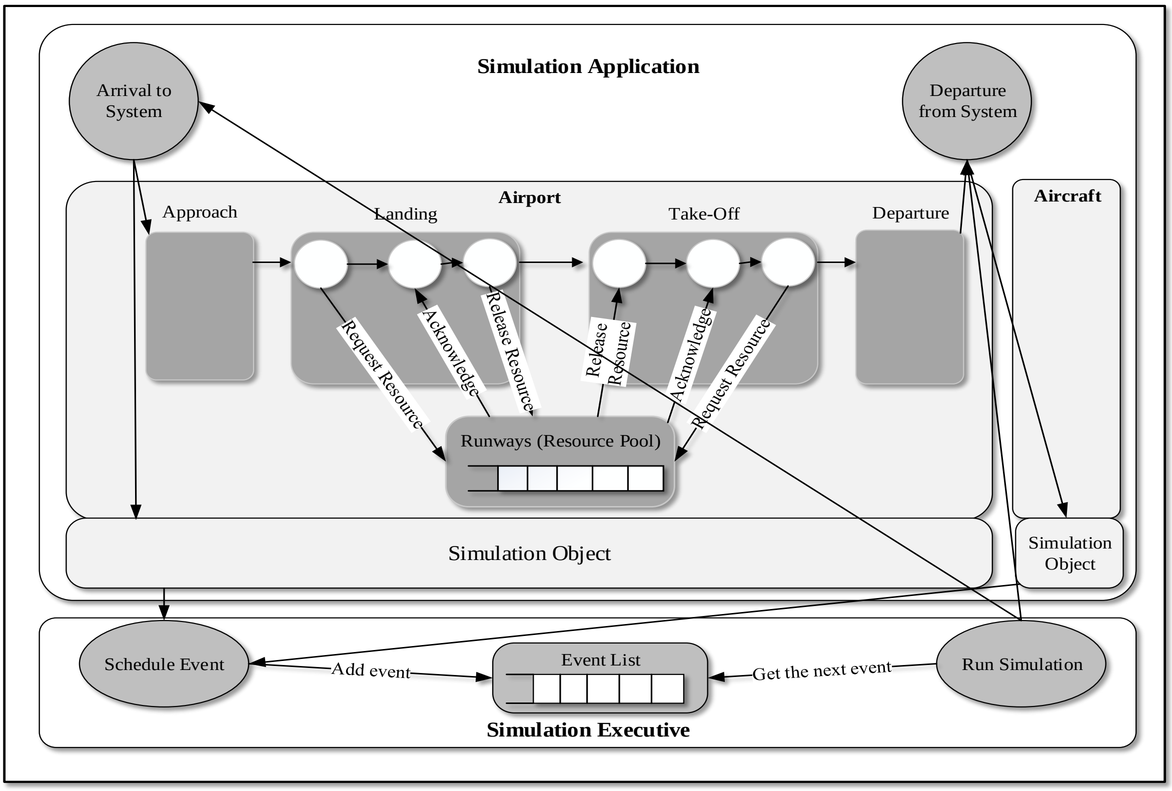

Figure 5: High-level Block Diagram of the Simulation Model

d. Hybrid Metaheuristic Algorithm

Over the last several decades, interest in metaheuristic algorithms in solving multi-objective optimization (MOO) problems has risen considerably among researchers and they have become more widely accepted as a viable alternative to exact methods. The behavior of a metaheuristic algorithm is largely determined by the intensification and diversification mechanisms for the search. Intensification is the mechanism for exploring intensely the most promising search areas and it is commonly implemented with local search techniques. On the other hand, diversification is the mechanism for diversifying the search process in order to move towards new areas of the search space and it is commonly implemented with tracking the search history such as long-term memory utilization. In metaheuristic algorithm design, it is important to find a good trade-off between these two mechanisms. In recent years, hybrid metaheuristics have been widely used to solve large-scale real-world MOO problems due to the fact that systematic combination of different metaheuristics have the potential to provide more efficient and flexible solutions (Talbi, 2015).

Scatter Search (SS) is one of the most promising population-based metaheuristics for SbO, which utilizes adaptive memory principles of Tabu Search (TS). Also, both algorithms have a shared history, since their basic principles are suggested by Glover (1977). A SS-based hybrid metaheuristic algorithm is developed as the optimization engine for the SbO framework for the following reasons:

(a) It generates and maintains a reference set of solutions at each iteration rather than a single solution, and this mechanism gives the ability to search for multiple Pareto-optimal solutions concurrently in a single run, without repeatedly finding each Pareto-optimal point one at a time.

(b) It improves the solutions increasingly at each iteration and this facilitates evaluating and improving the candidate policies through simulation.

(c) It is capable of handling non-differentiability and discontinuity that often appear in a SbO approach.

The proposed hybrid metaheuristic algorithm's general framework is similar to that of the traditional SS template. The primary additional procedure and mechanisms integrated into the SS template are listed below:

(a) Adaptive memory structures are utilized explicitly to store complete solutions. After a new trial solution is created with the solution combination method, the memory structure ensures that this trial solution has not visited previously. Then, it is sent to the simulation model for performance evaluation. Since computational time is the limiting factor in SbO approaches, integration of adaptive memory structures is very important for efficiency.

(b) A dynamic update procedure is employed with the intention of producing high-quality solutions, where non-promising solutions are replaced immediately with more promising ones.

(c) The fitness of each solution is computed with a non-dominating sorting approach, and a ranking procedure is utilized to classify solutions over the bi-objective domain, where both the objective value of the solution and its proximity to other solutions are considered.

(d) A rebuilding mechanism is adopted to enhance and maintain the diversity of the final Pareto-frontier.

(e) A two-step approach that includes a Tabu Search and a local search step is applied to improve solutions in the improvement method.

(f) An elitism mechanism is adopted where both dominated and non-dominated solutions are stored in a fixed-size archive. Also, a truncation procedure is employed based on density assessment by measuring the Euclidean distance in order to restrict the number of stored solutions.

The high-level scheme of the proposed hybrid metaheuristic algorithm is as follows: SS procedure is initiated by constructing a population of solutions (P) by using the initial solution obtained from the greedy heuristic algorithm as its starting point (seed) and, then a reference set (*RefSet*) is selected from the population. During the procedure, *RefSet* is evolved through subset generation, combination, and improvement sub-procedures. *RefSet* consists of two distinct subsets H and D, representing the high-quality and diverse solutions subsets respectively (*RefSet* $= H \cup D$). *RefSet* is updated from iteration to iteration by Reference Set Update method. *RefSet* is always maintained in order, where x^1 is the best solution and x^b is the worst one. Hence, in each iteration, *RefSet* is updated by assigning the incumbent trail solution to x^b and reordering the

RefSet. The schematic representation of the proposed algorithm is provided in Figure 6. The proposed algorithm has two main loops: (1) a "while loop" that controls the generation of the *P*, and (2) a "while loop" in which *RefSet* is evolved until a termination criterion is met (when the current iteration is equal to the maximum allowable iterations or the maximum allowable CPU time.)

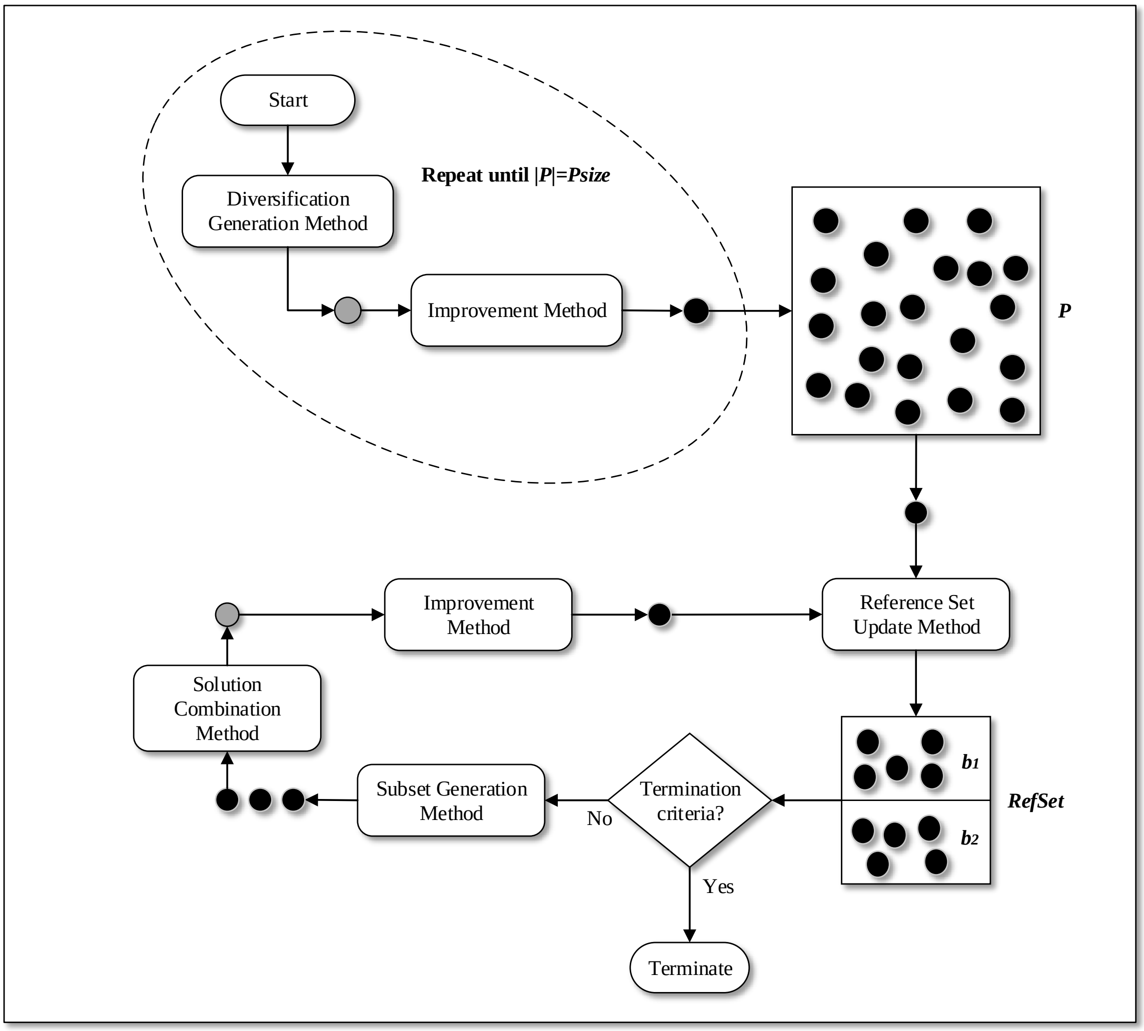

Figure 6: Schematic Representation of Hybrid Metaheuristic Algorithm (Adapted from Laguna and Marti (2012))

Representation and Search Operators: For effective and efficient application of any metaheuristic algorithm, it is essential to find an appropriate representation for a candidate solution that largely depends on the nature of the problem at-hand and search operators that conform well to the characteristics of the representation, and SS is no exception. The main requirement for the representation is the fact that it has to be capable of covering all candidate solutions in the search space, since the design of appropriate search operators is closely related to representations. However, there are not many theoretical models available that explain how different types of representations impact algorithm’s success and to what extent. The related properties of representation that impact solution quality, convergence time, and diversity for the multi-objective problems have to be identified. The most commonly accepted properties are redundancy, scaling, and locality (Franz, 2006).

A solution in the reference set corresponds to a set of decision variables for the optimization problem that is going to be simulated. Each iteration contains different input parameters which have to be experimented by the simulation model. Due to the fact that generating and maintaining diversification effectively depends on the solution representation, a permutation encoding is employed where a solution is represented by a sequence of integers corresponding to the index of the aircraft, and each row corresponds to a runway. In the literature a number of representation types are proposed for encoding permutations, where integer numbers are utilized to represent a sequence directly. However, this representation type requires additional repair mechanisms in order to apply the solution combination method, which will yield infeasible permutations with duplicate elements.

The five SS methods of the proposed hybrid metaheuristic algorithm are detailed below:

1. Diversification Generation Method: The algorithm starts with an initial set of trial solutions which are required to be diverse. Hence, a systematic procedure is used to generate those trial solutions. When a termination criterion is met, the algorithm provides the best solution found during any iteration. SS utilizes a reference set by combining the solutions in the reference set to generate new solutions, where the reference set is the core element. In a case such that all solutions in the reference set are similar, then whole procedure will probably not be capable of improving the best solution found so far. The Diversification Generation method creates a starting set of trial solutions systematically to guarantee a critical level of diversity.

2. Improvement Method: This method is an important intensification method to further transfer the incumbent solutions into a set of enhanced solutions of reasonable quality and diversity. This method is comprised of two steps: a simple Tabu Search, and a local search (neighborhood search) procedure. In the Tabu Search step, only the solutions that have a rank greater than a threshold value are considered. This threshold value is an assumed parameter for the algorithm. In local search step, all trail solutions are considered. In this step, "insertion" technique is used for moving from one solution to another. This procedure terminates when exploration of the neighborhood fails to find an improving move.

3. Reference Set Update Method: This method is utilized to generate and maintain the *RefSet*. During the first application of this method (initial generation of *RefSet* from the population), a minimum diversity test is utilized, which operates as follows:

Algorithm for Minimum Diversity Test for Initial *RefSet* Generation

Input: A population of improved trial solutions (*P*)
begin
    find the best solution according to *Obj.Fn.* value in *P*
    select this solution to become x^1 in the *RefSet*
    delete this solution, x^1, from the *P*
    while ($|RefSet| < a$) **do**
        find the next best solution *x* according to *Obj.Fn.* value in *P*
        select this solution, *x*, to be included in the *RefSet* only if $distance_{min}(x) >= tresholdDistance$
        delete this solution, *x*, from the *P*
end
return *RefSet*

The minimum diversity test procedure for the initial *RefSet* generation starts with choosing the best solution according to the objective function (*Obj.Fn.)* value in the population, selected as the best solution in the *RefSet*, and this solution is extracted from the population. Then, at each iteration, the next best solution in the population is selected only if the minimum distance between the selected solution *x* and the solutions currently in *RefSet* ($distance_{min}(x)$) is at least as large as the threshold value (*tresholdDistance*). Additionally, an elitist sorting mechanism for the non-dominated solutions is utilized to rank the solutions in the *RefSet* and sort them according to their rank. These solutions are then compared to each other to identify the distribution of solutions in the current Pareto front. Decision for accepting a candidate solution to *RefSet* is made based on the dominance relation and the density of the *RefSet* (whether it improves the diversity of the set). The distance between solutions in *RefSet* is calculated based on the crowding distance from each member of *RefSet.* Also, a rebuilding mechanism is employed to partially rebuild the *RefSet* when the Solution Combination and Improvement methods are not able to provide solutions of sufficient quality to displace the current *RefSet.* This mechanism reinitializes the Diversification Generation method to generate diverse solutions with respect to high-quality solutions in the current *RefSet.* It consists of the b_1 best solutions from the preceding step (solution combination or diversification generation). It also consists of the b_2 solutions that have the largest Euclidian distance from the current solutions in the *RefSet*. At each iteration a set of high-quality solutions replaces less promising solutions to improve the quality of the *RefSet.*

4. Subset Generation Method: This method generates subsets from *RefSet* that will be used for creating new solutions where the subsets are constructed by including all pairs of *RefSet* solutions except the pairs that have already been included in previous iterations.

5. Solution Combination Method: This method utilizes the generated subsets to combine the elements of each subset to create new trial solutions. The input for this method is limited to *RefSet*, and a dynamic update strategy is utilized where a new solution is included in the *RefSet* as quickly as possible before the next combination is performed. Hence, an intermediate pool of solutions is not utilized in the implementation to enhance efficiency. Also, an intensification strategy is integrated into this method to improve the search toward the Pareto-optimal front. Solution Combination method also tracks the subsets of *RefSet* solutions that have already been exposed to this method in each iteration. Whenever a new trial solution is created with this method, by using memory structures, this trial solution is checked whether it has not been visited previously. Then, it is sent to the simulation model for performance evaluation.

The proposed algorithm has two important multi-objective components: (1) *Fitness Assignment*: Due to the low-dimension (only two conflicting objectives) of the problem at hand, a Pareto-based fitness assignment method is employed to converge the solutions in a direction normal to the Pareto-optimal region and, at the same time, to promote diversity among solutions. This assignment method is applied together with a density measure, which is incorporated in such a way that adopts a two-stage process where first solutions are compared based on Pareto-fitness, then the density measure is applied. The main strength of this approach is that at the initial stages the force for diversity is higher, on the other hand, when the solutions begin to move to the Pareto-frontier, convergence force becomes dominant as most of the solutions that are equally fit. (2) *Diversity Preservation*: A diversity assessment scheme is adopted as the core element of diversity preservation component. Since the main goal of the proposed algorithm is to obtain a diverse Pareto-frontier, this diversity assessment scheme is applied in the objective space. And a distance-based assessment, in particular niching (niche sharing) is employed, which promote diversity in the *RefSet*.

4. Computational Experiment and Results

The main objective of the computational experiments for the implementation of the SbO approach is to study the quality and efficiency of the solutions generated by the proposed hybrid metaheuristic algorithm incorporated in the SbO as part of the validation study of the whole SbO framework. After the experiments are conducted and output data collected, state-of-the-art data analysis and visualization methods are utilized to identify patterns. Additionally, a sensitivity analysis is performed on several dimensions, including operational benefits, solution robustness under various traffic and weather conditions. The main quantifiable benefit metrics include increase in the runway capacity and delay savings.

A two-stage approach is utilized for the implementation of the proposed design. First the core data structures of the algorithm are created, and then the algorithmic structure is built on them. Since adaptive memory structures heavily depend on memory structures, object oriented techniques are employed sensibly. The statistics are collected, analyzed and visualized by using R Statistical Software Package.

a. Experimental Design

Both of the simulation and optimization (metaheuristic) components of the proposed SbO approach need some necessary parameter setting to adapt to the problem at hand and the choice of parameter values has a significant effect on the quality of the solution. Unfortunately, there is no one-size-fits-all parameter setting for any given simulation or optimization model. For this reason, optimal values for the parameters need to be determined carefully in a timely manner. Due to the fact that one-factor-at-a-time (OFAT) method does not consider the interactions between the parameters, which may significantly affect solution quality and performance, Design of Experiment (DoE) techniques are utilized, and experimental designs are established according to DoE’s formal procedures DoE methods are employed for two reasons: (1) to determine the various parameters’ main and interaction effects on the solution quality and algorithm efficiency, (2) to identify the optimal combination of parameter levels. Experiments conducted several replications to avoid artefactual results and enhance experimental validity.

Tuning the optimization (metaheuristic) algorithm to the specific problem being considered is very important for achieving high performance in terms of both solution quality and CPU time. In this regard, Central Composite Design (CCD), which is a well-known and widely used design of experiments method, is used to determine the optimal values of the parameters. CCD allows us to estimate all full second-order models (i.e., main effects, two-way interactions, and quadratic effects) with a reasonable amount of experiments. As a result, the size of the population (Psize) is set to 100 and the size of the RefSet (b) is set to 20, where initially 10 solutions are selected because of their high-quality (b1) and the remaining 10 are selected because of their diversity (b2). The number of improvement iterations is set to 25, and the maximum allowable iterations is fixed to 10000.

The major challenges regarding determining the parameter values for the simulation model include the following: large number of potential factors, multiple performance measures, complex response surface, time-varying correlated output streams, and inclusion of simulation-specific factors (Kleijnen et al., 2005). Considering these challenges and due to the fact that the simulation model is very complex and involves many variables with complicated interrelationships, Latin Hypercube Sampling (LHS) is chosen for determining the parameters of simulation model. LHS maximizes the minimum distance between design points but requires even spacing of the levels of each factor.

The simulation model is set up to implement two important variance reduction techniques: (1) common random numbers method is utilized to generate the sequence of pseudo-random numbers for uncontrollable factors in simulation experiments, and (2) antithetic variates to generate antithetic samples between successive pairs of replications. These variance reduction techniques are employed primarily to enhance the optimization of the simulation model.

b. Experimental Setup

Simulation Inputs and Data Sources: Multiple types of data are needed for the simulation model. Data are collected from a combination of the Aviation System Performance Metrics (ASPM) and the Airline Service Quality Performance (ASQP) databases. These databases provide the times of aircraft pushback from the gates, their takeoff and landing times, and the gate-in times, as reported by the airlines. ASPM also provides airport-level aggregate data, which enumerates the total number of arrivals and departures in 15-minute increments. Real-world historic data based on historic FAA-based Aircraft Situation Display to Industry (ASDI) and airline-reported Bureau of Transportation Statistics (BTS) data is utilized for Input Data Analysis.

In the experiments, the performance of the proposed approach is studied based on real data that belong to Washington Dulles International Airport (IAD). IAD is a busy US airport with more than 300 take-offs and landings per hour in visual weather conditions, and it has four runways. IAD handles both domestic and international flights, and the traffic volume is relatively unstable throughout a day. IAD operates in either an arrival or departure priority mode, as opposed to a single balanced operation between arrivals and departures to maximize capacity. Table 2 shows the capacity rates for arrivals and departures operations at IAD airport, presented as a range depending on the configuration priority mode (Jennifer Gentry et al., 2014).

Table 2: Runway Operations (Landings and Take-offs) per hour in IAD

Configuration	**Weather Conditions**		
	Visual	**Marginal**	**Instrument**
Arrival Priority	150-159	112-120	108-111
Departure Priority	156-164	136-145	125-132

The fleet mix is grouped by wake category, which is the same as separation requirements that are primarily based on weight class. Table 3 presents the fleet mix data for IAD airport whose fleet mix does not change with the weather (i.e., IAD do not have substantial numbers of Visual Flight Rules (VFR) operations) (Jennifer Gentry et al., 2014). In order to be consistent with actual traffic data at IAD, the fleet mix ratio is assumed as in Table 3.

Table 3: Annual Fleet Mix percentage by Wake Class in IAD

Heavy	**B757**	**Large**	**Small**
10.1	3.8	74.3	11.8

The FAA wake separation standards at the threshold are given in Table 4. Since time-based separation requirements are expected to be implemented, distance-based separations are converted to time-based separations assuming a 5 NMs final approach path and a nominal approach speed. Also, super and B757 weight classes are not considered for the sake of simplicity, which does not affect the validity of the design. The minimum separation requirements for operations on the same runway are given in seconds in Table 5a, where the leading aircraft is given by the rows, and the trailing aircraft is given by columns. In Table 5b the separation requirements for parallel runways depending on their spacing are shown.

Table 4: FAA Minimum Separation Standards in NMs
(Source: FAA (2014))

Leader/Follower	Super	Heavy	B757	Large	Small
Super	MRS	6	7	7	8
Heavy	MRS	4	5	5	6
B757	MRS	4	4	4	5
Large	MRS	MRS	MRS	MRS	4
Small	MRS	MRS	MRS	MRS	MRS

MRS: Minimum Radar Separation

Table 5: Minimum Separation Times in seconds

Departure → Departure				**Departure → Arrival**			
Lead / Trail	Heavy	Large	Small	Lead / Trail	Heavy	Large	Small
Heavy	90	120	120	Heavy	60	60	60
Large	60	60	60	Large	60	60	60
Small	60	60	60	Small	60	60	60
Arrival → Departure				**Arrival → Arrival**			
Lead / Trail	Heavy	Large	Small	Lead / Trail	Heavy	Large	Small
Heavy	75	75	75	Heavy	96	157	196
Large	75	75	75	Large	60	69	131
Small	75	75	75	Small	60	69	82

(a) Minimum separation times for operations on the same runway

Runway spacing	Departure → Departure	Departure → Arrival	Arrival → Departure	Arrival → Arrival
up to 2500 ft (up to 760 m)	As on single runway	As on single runway	Independent	As on single runway
2500 ft – 4300 ft (760 m – 1310 m)	Independent	Independent	Independent	40
more than 4300 ft (more than 1310 m)	Independent	Independent	Independent	Independent

(b) Minimum separation times for operations on parallel runways

We divide the data set into two disjoint subsets. One subset is used for simulation-based optimization, whereas the other subset is used as the test set for evaluating and validating the performance of the model.

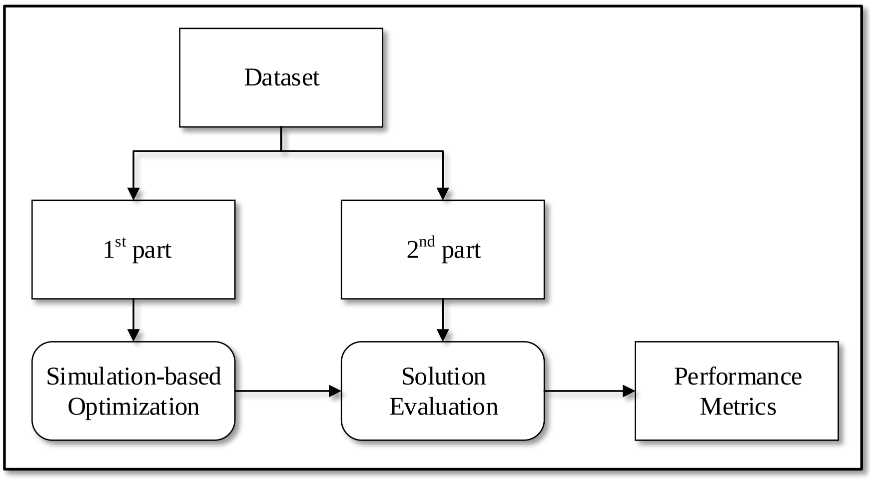

Figure 7: Utilization of Dataset

c. Experiments and Results

We analyze the computational performance of the SbO approach using the realistic IAD data by finding how optimized runway schedules can be used to gain additional runway capacity compared to an FCFS scheduling discipline. Fairness among aircraft is calculated as the absolute difference between the position of an operation in the optimized sequence and its position in the respective FCFS sequence.

The proposed algorithm and simulation model are implemented in C++ and complied with Microsoft Visual Studio 2013 environment. All the experiments are performed on a standard PC machine with a 64-bit Intel(R) Core(TM) i5-3210M CPU 2.50 GHz processor and 8 GB of RAM running Microsoft Windows 10 operating system. The computational times increased with increasing runway load. Since take-off delays are less costly compared to landing delays, the average take-off delay resulted higher than the average landing delay. Considering the fairness objective, it is also observed that the number of position shifts and the maximum position shifts to earlier or later positions increased with increasing runway load.

In order to evaluate the impact of the planning horizon and, in turn, the effect of computational times on runway operations, we considered three different planning horizons of 20, 25 and 30 minutes. As expected, when the planning horizon increased, the quality of the solution improved. SbO approach produced schedules with increased runway utilization compared to FCFS approach. Also, in terms of fairness among aircraft, the maximum position shift was calculated as 3.

For evaluating the performance of the metaheuristic algorithm, hyper-volume measure is utilized as the performance measure to assess the non-dominated solution set obtained in each iteration. This measure is the area of the dominated region by a non-dominated solution set and a reference point is needed for calculating hyper-volume measure (Deb, 2001). In the experiments, the origin of the objective space is used as the reference point. Figure 9 presents the graphics of the Pareto frontier obtained after 3315 iterations.

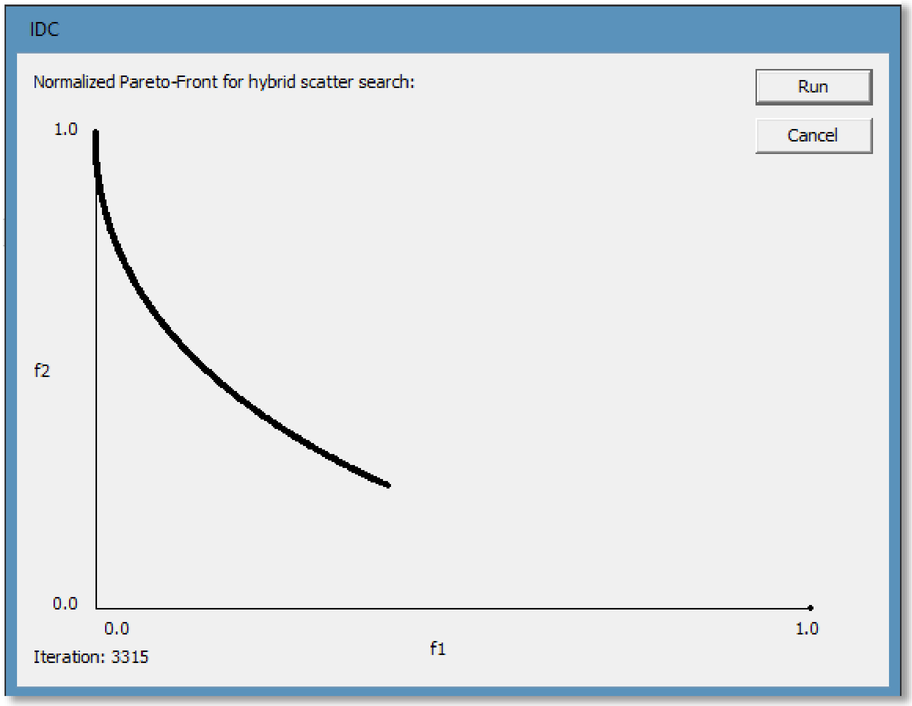

Figure 8: Visualization of Pareto-frontier Obtained After 3315 Iterations (*f1*: runway utilization, *f2*: fairness)

The above computational experiments led us to the following results:

(a) *Metaheuristic algorithm*: The initial solution generated by the greedy heuristic algorithm that utilizes problem-specific knowledge leads to performance improvement compared to a randomly generated initial solution. Therefore, the Diversification Generation method might be able to generate initial set of good Pareto-frontier. The dynamic update mechanism in the solution combination method seems to be an efficient mechanism for generating new Pareto-optimal solutions. The rebuilding strategy, which partially rebuilds the *RefSet* when the Solution Combination and Improvement methods do not provide diverse solutions, seems to be effective in maintaining diversity over the Pareto-frontier. Utilizing adaptive memory structures is important for creating SS algorithms for solving practical MOO problem instances. This strategy observed to be systematic in the sense that it progresses towards to Pareto-optimal rather than revisiting the earlier developed solutions too many times. In general, the algorithm has the ability to converge to multiple solutions at the same time by encouraging competition between solutions within the same local optimum neighborhood.

(b) *Simulation model*: The primary computational bottleneck in SbO frameworks is the simulation model, i.e. solution evaluation. Also, it is concluded that evaluation of the impact of uncertainties with a simulation model is important for validation for operational use.

(c) *SbO approach in general*: Not all the airports are likely to benefit from these advanced runway scheduling optimization approaches compared to FCFS, but if the air traffic is dense and aircraft fleet mix is diverse, then there is potential to achieve benefits.

5. Safety Risk Assessment

The FAA Air Traffic Organization (ATO) Safety Management System Manual is one of the main safety documents that provides the procedures and guidance to manage safety risk, and tries to establish a mature and integrated Safety Management System (SMS). Considering the

SMS detailed in this document, the proposed design poses a low-level of safety risks as well as mitigates the safety risk to some extent.

There is a potential trade-off between capacity and safety. As a result of applying this design, the number of aircraft operations per unit time will increase, and therefore, air traffic controller workload can potentially increase with a chance that the actual separation between aircraft to be violated compared to current practices. However, in computational experiments the proposed design did not lead to changes in air traffic control decision-making process and operational procedures. If any application of the proposed design result in changes in air traffic control decision-making process and operational procedures, this application should be accompanied by a safety risk assessment documented in accordance with the policy outlined in the ATO SMS Manual.

In addition, we have identified a few other low-level safety risks associated with the proposed design that can easily be mitigated. One such risk is a poor runway operations schedule generated by the proposed design, which could cause long delays in landing and take-off air traffic at the airport. In Visual Meteorological Conditions, the main risk transfer strategy is to change the responsibility of maintaining the minimum separation requirements from the air traffic controller to the pilot. In Instrument Meteorological Conditions, the air traffic controller can mitigate the risk by analyzing the performance measures generated by the simulation model.

On the other hand, consideration of uncertainties in runway operation scheduling mitigates a level of safety risk, since any schedule that does not respect minimum separation requirements as a result of an unexpected event are considered as infeasible during the optimization process. Minimum separation requirements are also important for controlling the risk of simultaneous runway occupancy. Therefore, the proposed design has a great potential to reduce safety risks to some extent, since it is capable of generating near-optimal schedules to reduce landing and take-off delays.

6. Interactions with Industry Professionals

We have contacted numerous aviation professionals from airports, airlines, consulting firms, and external experts throughout the development of the design to seek advice and feedback. In addition, we communicated with other faculty members who are knowledgeable in runway operations scheduling.

Both the faculty advisor (Dr. Ghaith Rabadi) and the graduate student (Bulent Soykan) successfully completed the “runwaySimulator Training Course” organized by the MITRE Corporation. There were more than 20 participants and most of the participants were aviation professionals from airports, consulting firms, or external experts. During the course, we got the opportunity to network with the participants, and to ask the questions related to practical aspects of runway operations scheduling.

We communicated with Mr. Thierry Sarr AAAE C.M. (Aviation Planner) to learn about the data sources that are necessary for the experimental study. Mr. Thierry Sarr provided us the sources of data that are necessary for the experiments as well as simulation model development.

We also communicated with Dr. Alan Bell and he reviewed our design from an applicability to practice perspective, and assessed its practical contribution and whether the results contribute to the actual problem in practice. We have received very helpful input from Dr. Bell, which eventually improved the design from a practical standpoint.

7. Summary, Conclusions and Recommendations

Despite the growing body of literature on Simulation-based Optimization (SbO), little effort has been devoted to applying this method to real-life scheduling problems. The possibilities of linking an optimization method with a simulation model are so vast. By the same token, the comprehensive literature review on runway operations scheduling shows a room for improvement in the literature on methods that address the challenges of multiple runway aircraft scheduling problem (MRASP) under uncertainty. This design develops a SbO approach that tackles such challenges by incorporating simulation and optimization. The advantage of this approach is in its robustness at incorporating complexity of the system to the required level of detail, by application of simulation as well as employing optimization algorithms to find high quality solutions in a timely manner. In addition, a greedy heuristic algorithm reinforces the proposed approach by delivering promising initial solutions obtained from solving the deterministic version of the problem.

As the optimization component for the SbO approach, a novel hybrid metaheuristic algorithm based on Scatter Search (SS) is developed, which do not guarantee finding the Pareto-frontier, but capable of finding reasonably good and diverse Pareto-optimal solutions in a relatively short time. The applicability of the proposed approach in large-scale real-world problem instances is investigated by using data obtained from a major international airport. As the experimental results have shown, an appropriate consideration of problem-specific knowledge is highly relevant for efficiency.

As a conclusion, this design has shown that considering the uncertainties during runway operations schedule optimization has the potential to increase airport's operational effectiveness as well as its capacity. It is also shown that consideration of fairness among airlines during runway operations schedule optimization improves airlines' on-time performance. The evidence obtained from the experiments illustrated that potential operational benefit can be achieved by research on closing the performance gaps between current and potential capabilities. These operational benefits include an increase in runway utilization while not sacrificing safety, which in turn, might lead to significant fuel and carbon reduction benefits.

Appendix A - List of Complete Contact Information

Faculty Advisor:

Dr. Ghaith Rabadi
Associate Professor
Old Dominion University
Department of Engineering Management and Systems Engineering
Email: grabadi@odu.edu

Individual Student (Graduate):

Bulent Soykan
Ph.D. candidate, Engineering Management
Old Dominion University
Department of Engineering Management and Systems Engineering
Email: bsoyk001@odu.edu

Appendix B - Description of the University

Old Dominion University

Old Dominion University, located in the City of Norfolk in the metropolitan Hampton Roads region of coastal Virginia, is a dynamic public research institution that serves its students and enriches the Commonwealth of Virginia, the nation and the world through rigorous academic programs, strategic partnerships, and active civic engagement. It was established in 1930 as the Norfolk Division of the College of William & Mary and is now one of the largest universities in Virginia.

The discovery of new knowledge through research and creative endeavor is a central function of Old Dominion University, which values and supports faculty participation in the discovery, synthesis, application and creation of new knowledge and art forms. The institution shall promote and preserve excellence in basic and applied research as a Carnegie Foundation Doctoral Research-Extensive University which is a key production and coordination force in technology development.

Department of Engineering Management & Systems Engineering

The Department of Engineering Management & Systems Engineering provides its graduates with the necessary skills, knowledge, and abilities required to design and manage the technology-based, project-driven enterprise. Fundamentally, the engineering management discipline addresses the problems, design, and management of projects and complex operations. The programs are grounded in solid principles of systems science and systems engineering while exploiting the tools of management science and project management.

Appendix C - Description of Non-University Partners Involved in the Project

External Expert - Dr. Alan Bell

Dr. Alan Bell
alan.bell@wings-of-gold.com

Dr. Alan Bell is an aviation professional with three decades of experience as a leader, operator, and analyst in the industry.

After graduating from the US Air Force Academy, Alan cross-commissioned into the Navy to begin a career as a combat pilot, eventually flying the both the S-3 and F/A-18 aircraft operationally. During twenty years of service, he accumulated over 4,000 hours of flight time, 1,100 arrested carrier landings, and was progressively assigned to positions of increasing rank and responsibility, including assignment to lead the US Navy's school for training carrier-based tower and flight deck air traffic controllers. He also holds civilian ratings as an Airline Transport Pilot and Certified Flight Instructor.

Following retirement from active duty, Alan began supporting the Federal Aviation Administration, providing safety and systems engineering support for a number of acquisition initiatives. Retirement from active duty also provided an opportunity to return to academia where he pursued a combination of business and engineering curricula leading to an eventual PhD in Engineering Management.

Appendix E - Evaluation of the Educational Experience Provided by the Project

Faculty Advisor - Dr. Ghaith Rabadi

l. Describe the value of the educational experience for your student participating in this competition submission.

Bulent had successfully completed my "ENMA 803 - Optimization Methods" class previously and grasped the design and implementation specifics of metaheuristic algorithms. However, during the class, we dealt with benchmark problems that can be found in the literature. With this design challenge, he got a great opportunity to deal with a real-life problem, which has unique challenges. As a result, he has become experienced in abstracting and modeling a real-life problem, designing and implementing a simulation model and a metaheuristic algorithm with an object-oriented approach, interfacing these two components seamlessly, and visualizing the relevant information.

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

The learning experiences was appropriate and very beneficial. This experience allowed and motivated Bulent to develop and enhance the following skills and knowledge which are all essential to be successful in industry or academic workforce: abstracting and modeling a real system, designing and implementing a computer simulation model, designing and implementing an optimization algorithm, visualization and statistical analysis.

3. What challenges did the students face and overcome?

The initial challenge Bulent faced was finding and engaging with the right expert who is knowledgeable in the area of runway operations scheduling, but after attending the "runwaySimulator Training Course" organized by the MITRE Corporation, he was able network with experts who provided practical information related to runway operations scheduling. Also, initially he faced some technical challenges related to the simulation modeling tool. At that time, we planned for him to use Arena Simulation Software package, but after implementing the simulation model in Arena, we realized that it is not efficient enough for practical problem sizes. He then switched to a low level programming language, particularly C++, and was able to implement both simulation and optimization components in C++, which made the integration of these components much easier.

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

I will definitely use this competition as an educational vehicle in the future, since it provides a valuable venue and motivation for forcing students to work on a real-life problem.

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

My only suggestion for future year competitions is that it would be more appropriate if the requirement for the length of the report is extended in order to allow detailed explanation of the design.

Graduate Student - Bulent Soykan

1. Did the Airport Cooperative Research Program (ACRP) University Design Competition for Addressing Airports Needs provide a meaningful learning experience for you? Why or why not?

ACRP University Design Competition for Addressing Airports Needs provided a meaningful learning experience for me due to several reasons. First of all, I got the chance to develop skills that are necessary for solving a complex real-life problem, and I had the opportunity to practice and strengthen the knowledge acquired in my previous studies. Second, it served as a means to accomplish several learning outcomes including, but not limited to the following: (a) deepen my knowledge on designing and implementing a simulation-based optimization framework, which is not a trivial task, (b) improve my programming skills by solving a complex and large-scale problem, and (c) enhance my statistical analysis and data visualization abilities. Lastly, it has been definitely a unique learning experience from that activated my self-confidence.

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

The primary challenges we encountered in undertaking the competition include the following: In the initial stage of the design project, it was not easy to engage with industry practitioners and experts. Another challenge that we encountered was obtaining the data we required for computational experiments. In order to overcome these challenges, we contacted numerous professionals, experts, practitioners, knowledgeable faculty members and at the end, we were able obtain the necessary information and data.

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

The process we used for developing our hypothesis is as follows: We first determined a general area of interest and reviewed the literature in this area to identify the knowledge gap. Then, mainly based on the literature review, we developed the research hypothesis, which is a specific, testable claim about what we expect to observe given a set of circumstances.

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

The participation by industry, specifically with subject matter experts, in the project is definitely appropriate, meaningful and useful. These engagements helped us quite a lot to in terms of understanding the potential operational benefit that can be achieved by research on closing the performance gaps between current and potential capabilities. Also, interactions with experts helped us to accomplish the project objectives in a timely manner.

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 increased my self-confidence and helped me with skills and knowledge I need to be successful to pursue further study. I have garnered experience in multi-objective metaheuristic algorithm analysis, design as well as in algorithm implementation in a low-level programming language. In addition, I learned how to design and implement a discrete event simulation model with an object-oriented architecture, and how to integrate a simulation model into a simulation-based optimization framework.

Appendix F - Reference List

Balakrishnan, H., & Chandran, B. G. (2010). Algorithms for scheduling runway operations under constrained position shifting. *Operations Research, 58*(6), 1650-1665.

Bennell, J. A., Mesgarpour, M., & Potts, C. N. (2011). Airport runway scheduling. *4or, 9*(2), 115-138. doi: 10.1007/s10288-011-0172-x

Bennell, J. A., Mesgarpour, M., & Potts, C. N. (2013). Airport runway scheduling. *Annals of Operations Research, 204*(1), 249-270.

Caprì, S., & Ignaccolo, M. (2004). Genetic algorithms for solving the aircraft-sequencing problem: the introduction of departures into the dynamic model. *Journal of Air Transport Management, 10*(5), 345-351.

Chandran, B., & Balakrishnan, H. (2007). *A dynamic programming algorithm for robust runway scheduling.* Paper presented at the American Control Conference.

Dear, R. G. (1976). The dynamic scheduling of aircraft in the near terminal area

Dear, R. G., & Sherif, Y. S. (1991). An algorithm for computer assisted sequencing and scheduling of terminal area operations. *Transportation Research Part A: General, 25*(2), 129-139.

Deb, K. (2001). *Multi-objective optimization using evolutionary algorithms* (Vol. 16): John Wiley & Sons.

FAA. (2014). *Advisory Circular (AC 90-23G) - Aircraft Wake Turbulence*. Retrieved from https://www.faa.gov/regulations_policies/advisory_circulars/index.cfm/go/document.information/documentID/1023467.

FAA. (2016). Airline On-Time Statistics and Delay Causes, from http://www.transtats.bts.gov/OT_Delay/OT_DelayCause1.asp?pn=1

Franz, R. (2006). Representations for genetic and evolutionary algorithms: Springer–Verlag, Berling, Heidelberg, Netherlands.

Garey, M. R., & Johnson, D. S. (1979). A Guide to the Theory of NP-Completeness. *WH Freemann, New York.*

Glover, F. (1977). Heuristics for integer programming using surrogate constraints. *Decision Sciences, 8*(1), 156-166.

Gupta, G., Malik, W., & Jung, Y. (2011). Effect of uncertainty on deterministic runway scheduling. *11th AIAA's Aircraft Technology, Integration, and Operations (ATIO)*, 20-22.

Hancerliogullari, G., Rabadi, G., Al-Salem, A. H., & Kharbeche, M. (2013). Greedy algorithms and metaheuristics for a multiple runway combined arrival-departure aircraft sequencing problem. *Journal of Air Transport Management, 32*, 39-48.

Jennifer Gentry, Kent Duffy, & Swedish, W. J. (2014). Airport Capacity Profiles - Final Report from www.faa.gov/airports/planning_capacity/

Kleijnen, J. P., Sanchez, S. M., Lucas, T. W., & Cioppa, T. M. (2005). State-of-the-art review: a user's guide to the brave new world of designing simulation experiments. *INFORMS Journal on Computing, 17*(3), 263-289.

Laguna, M., & Marti, R. (2012). *Scatter search: methodology and implementations in C* (Vol. 24): Springer Science & Business Media.

Mehta, V., Reynolds, T., Ishutkina, M., Joachim, D., Glina, Y., Troxel, S., Taylor, B., & Evans, J. (2013). Airport Surface Traffic Management Decision Support: Perspectives Based on Tower Flight Data Manager Prototype

Solveling, G., & Clarke, J.-P. (2014). Scheduling of airport runway operations using stochastic branch and bound methods. *Transportation Research Part C: Emerging Technologies*.

Solveling, G., Solak, S., Clarke, J.-P., & Johnson, E. (2011). Runway operations optimization in the presence of uncertainties. *Journal of Guidance, Control, and Dynamics, 34*(5), 1373-1382.

Soomer, M., & Koole, G. (2008). Fairness in the aircraft landing problem. *submitted paper, VU University, De Boelelaan 1081a, 1081*.
Soykan, B., & Rabadi, G. (2016). A Tabu Search Algorithm for the Multiple Runway Aircraft Scheduling Problem. In G. Rabadi (Ed.), *Heuristics, Metaheuristics and Approximate Methods in Planning and Scheduling* (pp. 165-186). Cham: Springer International Publishing.
Talbi, E. G. (2015). Hybrid Metaheuristics for Multi-objective Optimization. [Article]. *Journal of Algorithms & Computational Technology, 9*(1), 41-64. doi: 10.1260/1748-3018.9.1.41
Venkatakrishnan, C., Barnett, A., & Odoni, A. R. (1993). Landings at Logan Airport: Describing and increasing airport capacity. *Transportation Science, 27*(3), 211-227.