A Hybrid Metaheuristic Algorithm for Multi-Objective Runway Scheduling Problem in Simulation-based Optimization

Bulent Soykan, Ghaith Rabadi
Old Dominion University, Department of Engineering Management and Systems Engineering
Norfolk, VA
bsoyk001@odu.edu, grabadi@odu.edu

ABSTRACT

Even a moderately sized real-life runway scheduling problem tends to be too complex to be solved by analytical methods where the proposed mathematical models for this problem all belong to the complexity class of *NP*-Hard in a strong sense. Therefore, it is only possible to solve practical runway scheduling problems using mathematical programming methods by making a large number of simplifications and assumptions. As a result, most of the analytical models proposed in the literature suffer from too much abstraction and, in turn, not much applicability in practice. However, simulation-based methods have the capability to characterize complex and stochastic real-life runway systems in detail, as well as cope with several constraints and multiple objectives, which are important factors in practice. With a simulation-based optimization (SbO) approach where a discrete event simulation model is integrated with an optimization algorithm, the search for Pareto-optimal solutions can be done conveniently. Due to its large and unstructured search space, finding exact Pareto-optimal solutions to such multi-objective optimization problem is computationally intractable; given that such solutions need to be found in a reasonable timeframe, metaheuristic algorithms are the best option to pursue. In this study, a hybrid metaheuristic algorithm based on scatter search (SS), which takes advantage of the structural details of the problem, and uses a dynamic update mechanism to produce high-quality solutions and a rebuilding strategy to promote diversity, is proposed and presented. SS-based multi-objective evolutionary algorithms seem to be a promising research direction due to its efficiency and effectiveness in finding a set of non-dominated solutions in a SbO framework with multiple objectives. The experimental results that evaluate the proposed hybrid metaheuristic algorithm's performance in terms of both diversity of solutions and their proximity to the Pareto frontier are also presented.

ABOUT THE AUTHORS

Bulent Soykan is a Ph.D. candidate at Old Dominion University in the Department of Engineering Management & Systems Engineering (EMSE). He received his Bachelor's degree in Systems Engineering from the Turkish Military Academy (TMA) in 1997 and his Master's degree in Technology Management from the TMA Defense Sciences Institute in 2007. His main research interests include combinatorial optimization, simulation modeling and analysis, multi-objective optimization, metaheuristic algorithms, and airline/airport scheduling problems.

Ghaith Rabadi is an Associate Professor at Old Dominion University in the Department of Engineering Management & Systems Engineering (EMSE). He received his Ph.D. and M.S. in Industrial Engineering from the University of Central Florida (UCF) in 1999 and 1996 respectively, and his B.S. in Industrial Engineering from the University of Jordan in 1992. He has published numerous journal and conference papers and contributed several book chapters in edited volumes. His research and teaching interests include planning and scheduling, operations research, simulation modeling and analysis, analytics, and supply chain management and logistics. He founded the International Journal of Planning and Scheduling in 2010 and has been serving as its Chief Editor since then.

A Hybrid Metaheuristic Algorithm for Multi-Objective Runway Scheduling Problem in Simulation-based Optimization

Bulent Soykan, Ghaith Rabadi
Old Dominion University, Department of Engineering Management and Systems Engineering
Norfolk, VA
bsoyk001@odu.edu, grabadi@odu.edu

INTRODUCTION

The steady increase in air traffic is expected to continue in the coming years; however, this growth has not been supported by the increase in the capacity of major airports. As a result, air traffic delays are becoming inevitable in major airports when the demand for services exceeds its capacity. Due to the fact that runways are typically the primary bottleneck in air transportation system, the overall airport capacity is determined by the runways in use. Therefore, it is significantly important to effectively utilize resources, especially runways, to achieve significant improvements in air traffic delay and, in turn, to smooth the flow of air traffic. To accomplish this, effective and efficient algorithms are required for scheduling aircraft over runways as part of operational planning models and decision support systems used by air traffic controllers (ATCs).

Over the past several decades, researchers and practitioners have developed various models and tools to better schedule aircraft over runways. However, for major airports, it is still a challenge to schedule aircraft in Terminal Maneuvering Area (TMA) considering the complexity inherent to runway operations and uncertainties related to air traffic and weather conditions. In the United States (US), air traffic volume accounted for 34.58% of all aircraft delays in 2014 (FAA, 2016). One of the main reasons for this inefficiency is that currently used operational planning models and decision support tools do not take into account the stochastic nature of the runway operations and the interests of different stakeholders. Hence, there is an apparent need for improving these models and tools for planning and controlling the air traffic flow in runway operations to a level applicable for practical use.

The real-life combinatorial problem of scheduling aircraft over runways is commonly referred to as (airport) runway scheduling problem. This problem includes determining the optimal landing / take-off sequence and start times over each arrival / departure runway to improve the safety and efficiency of runway utilization, and ensure fairness among airlines. In practice, runway scheduling is one of the most challenging tasks ATCs have to carry out. Also, this scheduling process is significantly important for achieving efficiency and effectiveness in runway operations, improving passenger satisfaction, reducing air traffic delays and fuel burn, ensuring safety etc.

Due to the complexity of this real-life problem, in order to apply analytical methods, it has been inevitable to make numerous assumptions to reduce the complexity. As a consequence, the solutions acquired from the analytical models are far from practical. On the other hand, considering less simplifying assumptions for the sake of modeling closer to reality usually lead to intractable analytical models, where results can only be estimated via some heuristic or approximation techniques. Therefore, appropriate modeling techniques and solution algorithms need to be employed.

Nearly the entire literature addresses the runway scheduling as a deterministic problem with the assumption that all the input parameters are known with certainty in a deterministic environment. However, in practice, there are numerous sources of uncertainty that need to be considered during scheduling, such as inclement weather, airport congestion, equipment failure, unexpected delays in pushback or taxiing etc. And these uncertainties are often impossible to avoid in practice. Moreover, the other practical aspect that needs to be considered is the multi-objective nature of the problem which refers to finding the trade-offs between different stakeholders' interests. Hence, real-life runway scheduling problem involves several contradicting objectives that need to be satisfied simultaneously. As a result, the methods to solve this scheduling problem should be robust enough to consider uncertainties and interests of various stakeholders, and efficient enough to produce solutions in a reasonable time.

Given the complexity and practical requirements of the problem, the simulation-based optimization (SbO) approach seems to one of the best suitable methods for solving it. The main advantage of integrating simulation into optimization is that it can include less modeling assumptions, resulting in a more realistic and valid model and, in turn, leading to better decision-making process. In SbO approach, a simulation model is commonly utilized for evaluating the performance of a solution which provides a convenient means to capture more realistic aspects of runway operations. However, this approach still faces challenges in terms of optimization especially when there exist multiple and conflicting objectives. Due to the complexity of this multi-objective optimization (MOO), metaheuristics are a very practical and suitable technique where optimal or near-optimal solutions can be obtained without much computational intractability issues.

The overall aim of this paper is to develop a hybrid metaheuristic algorithm based on scatter search for approximating the Pareto frontier of the multi-objective runway scheduling problem in a SbO framework. The algorithm tries to evolve the reference set of solutions towards the Pareto frontier in each iteration and distribute it over the frontier in order to maintain a diverse set of solutions. To the best of our knowledge, this is the first attempt in the literature to employ a SbO approach for solving this real-life scheduling problem that considers both uncertainties and interest of different stakeholders where the existing algorithms are unable to achieve. Also, the proposed algorithm, which is the optimization component of the SbO framework, is a novel algorithm that takes advantage of the structural details of the problem, and can be distinguished from the current algorithms by the way in which it uses a dynamic update mechanism to produce high-quality solutions and a rebuilding strategy to promote diversity across the Pareto frontier.

The remainder of the paper is organized as follows. Next section provides an overview of the definition and characteristics of the specific problem at hand. Section 3 presents a brief overview of what was gleaned from the literature review focusing on the most recent research trends in runway scheduling problem, MOO, and SbO fields. Section 4 provides an overview of the SbO framework to solve the multi-objective runway scheduling problem, and describes the proposed hybrid metaheuristic algorithm. Section 5 summarizes the experimental design and the results of the computational experiments. Finally, the conclusions and future research areas are given in Section 6.

PROBLEM DEFINITION AND CHARACTERISTICS

This section is dedicated to present the background of the study focusing specifically on the problem definition and fundamental characteristics in order to set up the foundation.

Since the 1960s, developing efficient methods for tackling runway scheduling problem has been of great interest for both academic researchers and practitioners. This scheduling problem basically deals with assigning each aircraft to a runway, sequencing assigned aircraft for each runway and then calculating each aircraft's start time for landing or take-off. Several techniques have been formulated to solve various forms of the problem and Bennell et al. (2013) identified the main solution techniques as dynamic programming (DP), branch-and-bound (B&B), and heuristics.

Typical objectives utilized for the problem in the literature are minimizing the total delay (tardiness), minimizing the average delay per aircraft and maximizing the throughput while considering certain operational restrictions. The commonly considered constraints are detailed below:

Time windows: Once an aircraft enters the radar range for landing or pushbacks from the gate for take-off, ATCs assign a runway and a start time to land / take-off for it. 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 Federal Aviation Administration (FAA.) This spacing requirement is associated with the wake turbulence that is the disturbance in the air left after the passing of an aircraft. This is the main constraint that makes the problem a non-trivial one.
Limited flexibility in deviating from the First Come First Served (FCFS) order: In practice, ATCs often simply depend on FCFS strategy, which is the most straightforward and widely used approach. Although FCFS order eases ATCs' workload, maintains a sense of fairness among airlines and easy to implement, most of the time it is not capable of providing the best schedule in terms of runway utilization. Hence, usually deviating from FCFS order is limited.

There are several dimension of classifying runway scheduling problems and most prominent ones are listed below:
Static vs dynamic: This characteristic of the problem relates to time. Static case is solved before actual operations with known or predicted information while dynamic case is solved to generate schedules in real time as aircraft arrive in real time.
Different runway characteristics: These characteristics are related to the number of runways that are considered (single or multiple), the interaction between the runways (interacting or independent), and the mode in which each runway is operated (mixed mode or segregated mode). Arriving and departing aircraft inevitably interact through the common use of taxiways and runways. Thus, managing the air traffic flow successfully requires considering both landing and take-off aircraft in the TMA.

The following are the characteristics that taken into account in this research to make the problem closer to practice:

- Static (offline) case of the problem is considered.
- Multiple interacting runways that operate in mixed mode operations are taken into account.
- Several uncertainties that are inherent to practical operations and multi-objective nature of the problem is also taken into consideration, and detailed in the rest of this section.

While academic researchers have been highly focused on solving deterministic runway scheduling problems and do not consider collaborative decision-making (CDM) aspects, ATCs are faced with daily challenges where uncertainty and interests of different stakeholders are the key characteristics of the real-life problem. These two aspects of the problem are detailed below:
Uncertainty: In practice, there are numerous sources of uncertainty that need to be considered during the scheduling process, such as inclement weather, airport congestion, equipment failure, ground speed variations caused by the wind, piloting indecisions, unexpected delays in pushback or taxiing etc. In such cases, the quasi-optimal schedules become far from optimal in practice because of challenges posed by uncertainty impacts. Uncertainty in runway operations usually manifests in the TMA in the form of traffic queues, and such queues typically result in operational inefficiencies, additional costs, such as fuel costs and environmental consequences.
Interests of different stakeholders: The viewpoints of the various stakeholders who affect or be affected by the scheduling of aircraft over runways differ substantially. Moreover, each stakeholder is usually concerned with multiple performance measures. For air navigation service providers (ANSPs), safe flow of air traffic and runway utilization are the primary concerns. On the other hand, airlines are mainly concerned with resource utilization, punctuality, air traffic and on-time performance etc. As a relatively recent concept, CDM has been proposed to enhance collaboration between stakeholders, especially between ANSPs and airlines, to increase the efficiency of air traffic flow and satisfaction of the airlines. CDM concept is a means to deal with challenges at major airports and has a potential to improve runway operations by allowing airlines to participate in air traffic decision-making that affects them. Therefore, in order to integrate CDM concept into models, different conflicting objectives of the practical problem have to be taken into consideration, and the key challenge for this is the fact that most of these objectives are non-commensurable and it is hard to aggregate them into one synthetic objective.

LITERATURE REVIEW

This section describes the existing work done on runway scheduling problem that considers uncertainties and multiple objectives concisely. A brief background and literature review on MOO and SbO that supports the assumed methodology is also provided. The full details of these methods are relegated to references cited to avoid getting immersed in unnecessary detail.

Runway Scheduling Problem

Even though there is a large body of research on deterministic runway scheduling in the literature, there is a few research that considers the stochastic and multi-objective nature of the real-life problem.

Chandran and Balakrishnan (2007) considered a runway schedule “robust” if there is a high probability that an ATC does not have to interfere once the schedule has been determined. They tried to optimize two conflicting objectives: maximizing throughput (or minimizing makespan) and maximizing reliability. As a solution algorithm, they proposed a DP approach which is computationally efficient enough for a real-life application.

Montoya et al. (2014) proposed a multi-objective DP algorithm that minimizes the total delay of aircraft and the makespan of a sequence in order to find a set of Pareto-optimal solutions that completely represent the non-dominated frontier. The simulation results to validate the proposed algorithm are also provided.

Solveling and Clarke (2014) addressed the stochastic version of the 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 B&B algorithm is proposed, which is basically a sampling-based approach where the stochastic upper and lower bounds are generated.

Multi-Objective Optimization

MOO refers to finding decision variable values that correspond to optimizing more than one conflicting objectives. A solution is considered to be Pareto-optimal if there does not exist any other solution that is at least as good regarding all objectives and better regarding at least one objective. When all solutions are non-dominated, the solution set is called a non-dominated solution set. Coello et al. (2007) identified two essential goals for MOO: (1) find a set of solutions as close as possible to the true Pareto-optimal front, and (2) find well-distributed Pareto-optimal solutions that cover the entire Pareto-optimal region in order to ensure a good set of trade-off solutions. MOO problems can be tackled by either exact or metaheuristic methods. In either case, the concept of Pareto-optimality is generally embraced.

For more than three decades, population-based metaheuristic algorithms, such as evolutionary strategy, genetic algorithm (GA) etc., have been widely adopted for solving MOO mainly because of its capability to exploit the diversified and comprehensive set of Pareto-optimal solutions simultaneously in one run. Lately, the enhanced version of the Non-dominated Sorting Genetic Algorithm (NSGA-II) is generally considered as one of the principal algorithms in the domain, which implements a non-dominated sorting of a combined population with an elitist mechanism that helps to enhance efficiency of the algorithm significantly (Deb, 2001). The term "elitist" commonly refers to an algorithm that utilizes the combined population rather than just replacing the old population with the new offspring, which basically prevents the loss of promising solutions. Another feature that is often used in NSGA-II applications is "crowding distance," which refers to the density of solutions neighboring a particular solution in the population that is typically computed by the average distance between two points on either side of this solution along each of the objectives.

Although there has been a pervasive interest in applying GA to MOO problems in the literature, there has been some attempt to propose MOO algorithms based on scatter search (SS). Rahimi-Vahed et al. (2007) suggested a non-dominated sorting procedure called MOSS (Multi-Objective Scatter Search), which ranks every solution of the reference set. In order to maintain non-dominated solutions uniformly dispersed along the Pareto front, the NSGA niching method is employed. Nebro et al. (2008) proposed a hybrid metaheuristic algorithm called AbYSS (Archive-Based hYbrid Scatter Search) which adapts the traditional SS template but utilizing mutation and crossover operators coming from the field of evolutionary algorithms. The proposed algorithm is built on integrating the ideas of Pareto dominance, external archiving, and two different density estimators. Recently, there are many open research lines on metaheuristic-based MOO methods, which include developing more efficient algorithms, defining new performance measures, and integrating with simulation models.

Multi-Objective Simulation-Based Optimization

Simulation modeling is commonly considered as a successful and effective method for analyzing and evaluating the dynamic and stochastic nature of complex systems, such as runway systems. Integrating this capability with an optimization model is typically called as SbO, where these two methods are traditionally considered as separate and alternative. In general, SbO approaches try to determine the exact combination of system parameters which produce the optimal or near-optimal performance measure(s). Existing SbO methods include response surface methodology, metaheuristic methods, and statistical selection methods. However, metaheuristics prevail the practical applications to a great extent, and most of the commercial simulation software packages utilize metaheuristic-based optimizers.

Multi-objective SbO is an evolving area of research that integrates optimization techniques into simulation modeling and analysis, where optimization deals with more than one conflicting objectives simultaneously. Joines et al. (2002) proposed a GA-based multi-objective SbO optimization approach by adapting NSGA-II and applied it to a real-life supply chain optimization problem with two objectives. Eskandari et al. (2005) incorporated a simulation model with a stochastic non-domination-based MOO method and GA. New operators, which include elitism, dynamic expansion, importation operators, for GA are introduced to improve the performance of the algorithm in terms of both effectiveness and efficiency.

SIMULATION-BASED OPTIMIZATION FRAMEWORK

This section gives an overview of the SbO framework and highlights each component to provide a high-level understanding of the design. By and large, SbO approach tries to determine the exact combination of system parameters which produce the optimal or near-optimal performance measures.

Conceptually the SbO model resembles the Figure 1. The SbO framework starts with a greedy heuristic algorithm to produce a relatively good initial solution compared to a random one. 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 the proposed metaheuristic algorithm to search for the values of system parameters.

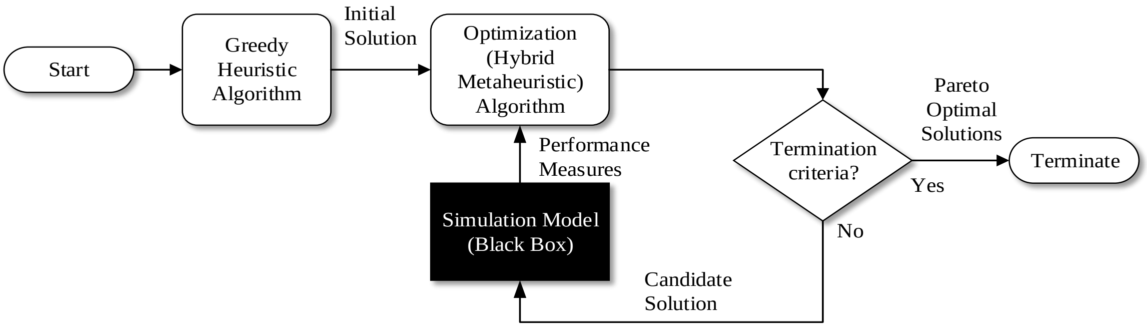

Figure 1. Simulation-based Optimization Framework

Greedy Heuristic Algorithm

The greedy heuristic algorithm 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 (r_j), latest (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, which principally scales the contribution of each part of the composite dispatching rule relative to the total. This parameter is determined empirically and validated in terms of its suitability for practical problem instances to get high-quality solutions. 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.

Simulation Model

The simulation component of the SbO is a discrete event simulation model based on an aircraft trajectory model and runway operations are approximated to fit the model. The airside of the airport is represented as a network of nodes and links, on which aircraft follow prescribed paths, and it mimics the movement of individual aircraft with an acceptable level of accuracy. It also simulates the stochastic processes using random variables to reflect the uncertainty in runway operations. The data that the entities (aircraft) need as they move through the model include: release times into the system, separation times, runway occupancy times (ROTs), probability distributions associated with release times and ROTs. Entities carry some of the data with them as they move and some of the information are shared across entities, which belongs to the system as a whole. Aircraft release times into the system and ROTs are considered to be the random variables. The simulation model consists of several modules: Aircraft Generate module generates aircraft for landing and take-off according to the probability distributions based on the airport's fleet mix; Arrival and Departure modules simulates the landing and take-off, respectively, through reasonable trajectory states at key points along the path of the aircraft, where landing and take-off are broken into segments and each segment forms a node of the simplified runway operations. The states are calculated from the average speed of that class of aircraft can fly with respect to the constraints.

Input Data Analysis: For each stochastic activity in the simulation model an appropriate probability distribution is chosen based on the historical data obtained from the publicly available FAA Operations & Performance Database. Also different weather conditions are taken into account due to the fact that one of the main results of runway operations in inclement weather is the reduced runway capacity due to increased ROT.

Verification and Validation Study: Verification study sought to show that simulation model performs as expected and provides an accurate logical representation of the conceptual model. On the other hand, validation study established that model behavior validly represents the real runway system being simulated. The subject matter expert judgment constituted a crucial component of the validation process. Statistical comparison of numerical values of the output performance measures with the real-life runway system. The total waiting time and the landing/take-off time of each aircraft are used as an index of the simulation accuracy. Given the complexity of runway operations, validating the discrete event simulation model was a challenge, and several possible scenarios are simulated to overcome this challenge.

Hybrid Metaheuristic Algorithm Based on Scatter Search

Scatter Search (SS) is a population-based metaheuristic search method which is introduced by Glover (1977) as a heuristic for integer programming. SS has been successfully applied to a wide range of real-life hard optimization problems and it is commonly considered as a flexible and adaptable metaheuristic due to the fact that it offers various implementation alternatives by exploiting its foundational strategies. SS starts with generating an initial population of candidate solutions, reducing this population to a reference set of solutions, and then builds, maintains and evolves this reference set throughout the search where preferred subsets of solutions in the reference set are combined to generate new trial solutions by weighted linear combinations. The reference set is updated by selecting the promising solutions from the trail solutions. Glover (1994) extended the basic SS by combining it with adaptive memory structures of Tabu Search to balance search intensification and diversification, and Glover (1998) provided a simplified SS template that has been serving as the main reference for recent SS applications.

The proposed hybrid metaheuristic algorithm's general framework is similar to that of the traditional SS template. The primary differences between two include the following procedure and mechanisms:

- A dynamic update procedure is employed with the intention to produce high-quality solutions, where non-promising solutions are replaced immediately and future combinations are done with more promising ones.
- 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.
- A rebuilding mechanism is adopted to enhance and maintain the diversity of the final Pareto front.

The general 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. The pseudocode describing the proposed algorithm is provided in Algorithm 1. The proposed algorithm has two main loops: (1) a "while loop" that controls the generation of the *P*, and (2) a "while loop" that *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.)

Representation: 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.

Diversification Generation Method: This method creates a starting set of solutions systematically to guarantee a critical level of diversity.
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 depends on neighborhood search and restricted to local search procedure, where “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. This method is applied to all solutions present in the set *P* initially, and then, to new solution generated by Solution Combination method.
Subset Generation Method: This method generates subsets from *RefSet* that will be used for creating new solutions.
Solution Combination Method: This method utilizes the generated subsets to combine the elements of each subset to create new trial solutions. This method is a problem-specific procedure and its implementation is systematic rather than a probabilistic one as in GA. 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.
Reference Set Update Method: This method is utilized to generate and maintain the *RefSet*. 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 current *RefSet.* This mechanism reinitializes the Diversification Generation method to generate diverse solutions with respect to high-quality solutions in the current *RefSet.*

Algorithm 1 Hybrid Metaheuristic Algorithm based on Scatter Search

```
Initialization
$P=\{\}$
while $|P|<Psize$
    generate a solution $x$ with Diversification Generation Method
    improve $x$ with Improvement Method
    if $x \notin P$ then $P=P \cup x$ else discard $x$
end while
build RefSet with $b$ ($b_1$ high-quality and $b_2$ diverse) solutions from $P$
sort the solutions in RefSet according to their ranking value in ascending order
while the termination criterion is not met
    update the iteration counter, $iter=iter+1$
    generate subsets with Subset Generation Method
    do no more new subsets
        select the next subset
        combine the solutions immediately with Solution Combination Method
        improve the generated new solution with Improvement Method
        Reference Set Update Method
            if solution quality is not sufficient to displace current RefSet
            then Rebuilding Mechanism
     while
end while
```

COMPUTATIONAL EXPERIMENTS 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 to SbO as part of the validation study of the whole SbO framework. Retrospective tests are employed which use historical data to measure how well the proposed approach would have performed if they had been used in the past. After the experiments are conducted and output data collected, state-of-the-art data analysis and visualization methods are utilized to identify patterns interactively. Additionally, a sensitivity analysis is performed on several dimensions, including operational benefits, solution robustness under various traffic and weather conditions.

The discrete event simulation model is implemented by using 14.70 version of Arena commercial simulation software package for its flexibility in constructing complex models, and greedy heuristic and optimization algorithms are implemented by using Visual Basic for Applications (VBA) Editor embedded in Arena model that enables to transfer relevant information from Arena model to VBA, and vice versa. The Arena 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 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.

Parameter Settings

Simulation experiments are expensive both in terms of time and resource consumption. Thus, not only the simulation model but also its utilization requires being carefully planned and designed. The traditional design of experiments methods is not very well suited for complex and large-scale simulation experiments due to the fact that they can investigate only limited number of factors and they usually make unrealistic assumptions with regards to the responses. Therefore, a design of simulation experiments method, particularly Latin hypercube sampling (LHS), is used to reduce the number of experiments required and obtain a structured way to understand the system behavior. The choice of LHS does not necessarily depend only on the power of the design and available resources, but also depend on the sample size (number of replicates), selection of a suitable run order for the experimental trials, and determination of whether or not randomization restrictions are involved.

Tuning the 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, which is a well-known and widely used design of experiments method, is used to determine the optimal values of the parameters. 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 (b_1) and remaining 10 are selected because of their diversity (b_2). The number of improvement iterations is set to 25, and the maximum allowable iterations is fixed to 10000.

Experimental Setup

In the experiments, the performance of the proposed approach is studied based on real data belong to Washington Dulles International Airport (IAD). IAD is one of the busiest airport in the US 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. The FAA enforced minimum separation times are shown in Table 1, and ROTs are given in Table 2.

Table 1. Minimum Separation Times in seconds (FAA standard)

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

Table 2. Runway Occupancy Times in seconds

Operation / Aircraft Type	Heavy	Large	Small
Landing	40	35	30
Taking-off	50	40	30

In computational experiments, 20 independent runs are performed on each iteration. Measuring the performance of MOO algorithms is more difficult than in the case of single-objective optimization, because of the two simultaneous goals (convergence and diversity) that any MOO algorithm should able to achieve. The observed performance measures are convergence to the Pareto-optimal front and adequate diversity among solutions. Furthermore, a visual comparison of the two objectives can provide unbiased insights related to the both convergence and diversity performance.

Experimental Results

The hypervolume 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 hypervolume measure (Deb, 2001). In experiments, the origin of the objective space is used as the reference point. Figure 2 presents the graphics of the Pareto frontier obtained in one instance.

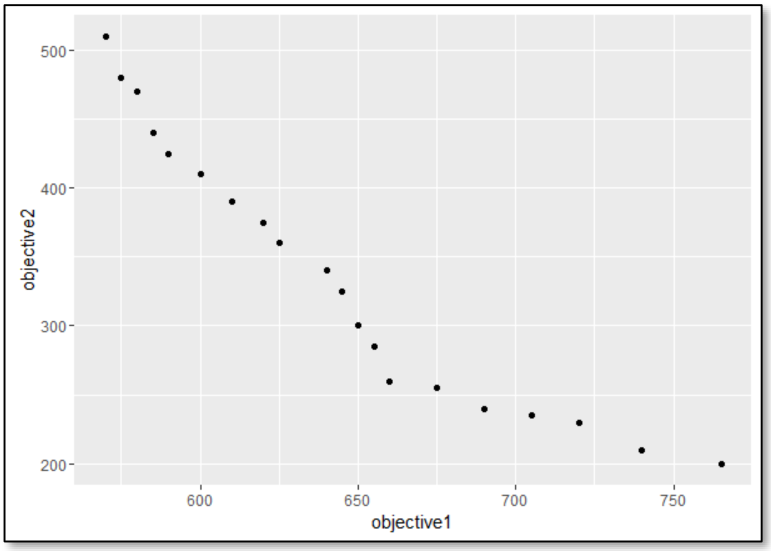

Figure 2. Experimental Results

The above computational experiments led us to the following results:

- Initial solution generated by the greedy heuristic algorithm that utilizes problem-specific knowledge lead to performance improvement compare to a randomly generated initial solution. So that, Diversification Generation method can able to generate initial set of good Pareto front.
- The dynamic update mechanism in the solution combination method seems to be an efficient mechanism to generate 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 on maintaining diversity over the Pareto-optimal front.

CONCLUSIONS AND FUTURE RESEARCH DIRECTIONS

The main contribution of this paper is in the field of developing a hybrid metaheuristic algorithm and applying it to the multi-objective runway scheduling problem, which basically tries to generate an aircraft sequence and a time

schedule for each runway in use, based on a SbO framework. The main reason for employing a SbO framework is to capture the relevant intricacies of the practical operational environment, particularly the stochastic nature of this real-life problem and interests of different stakeholders.
The results obtained from computational experiments indicate that the use of an SS-based metaheuristic algorithm within a SbO framework is promising, given that problem's computational complexity is *NP*-Hard, runway operations are dynamic and stochastic in nature, and two conflicting objectives need to be optimized concurrently. SS has a potential to exploit knowledge obtained from the search space, utilize strategic designs, and construct search paths from a reference set as compared to randomization used in GA. This allows SS to be an efficient optimization component in a SbO approach, where simulation runs require more time. Moreover, SS's reliance on the concepts of quality and diversity of solutions helps it to approximate the Pareto frontier of MOO problems efficiently.

In conclusion, integrating multi-objective metaheuristic algorithms into SbO frameworks as the optimizer component seems to be an encouraging direction for future research and practice. Additionally, although most of the research on metaheuristic algorithms for MOO is dedicated to evolutionary algorithms, aforementioned capabilities of SS justify further work in this field. Finally, this research is currently being extended by improving the implementation of the overall proposed approach so that it can produce solutions not only effective enough but also efficient enough in terms of computation time for real-life applications. The results from the initial experiments will be used to guide refinements to the implementation of the optimization algorithm as well as the simulation model, and to incorporate all of the SbO components into a robust and powerful application with a modular object oriented design.

REFERENCES

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

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

Coello, C. C., Lamont, G. B., & Van Veldhuizen, D. A. (2007). *Evolutionary algorithms for solving multi-objective problems*: Springer Science & Business Media.

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

Eskandari, H., Rabelo, L., & Mollaghasemi, M. (2005, 4-7 Dec. 2005). *Multiobjective simulation optimization using an enhanced genetic algorithm.* Paper presented at the Simulation Conference, 2005 Proceedings of the Winter.

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

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

Glover, F. (1994). Special Volume Viewpoints on OptimizationTabu search for nonlinear and parametric optimization (with links to genetic algorithms). *Discrete Applied Mathematics, 49*(1), 231-255.

Glover, F. (1998). A template for scatter search and path relinking. *Lecture notes in computer science, 1363*, 13-54.

Joines, J., Gupta, D., Gokce, M. A., King, R. E., & Kay, M. G. (2002). *Supply chain multi-objective simulation optimization.* Paper presented at the Simulation Conference, 2002. Proceedings of the Winter.

Montoya, J., Rathinam, S., & Wood, Z. (2014). Multiobjective Departure Runway Scheduling Using Dynamic Programming. *Intelligent Transportation Systems, IEEE Transactions on, 15*(1), 399-413. doi: 10.1109/TITS.2013.2283256

Nebro, A. J., Luna, F., Alba, E., Dorronsoro, B., Durillo, J. J., & Beham, A. (2008). AbYSS: Adapting Scatter Search to Multiobjective Optimization. *Evolutionary Computation, IEEE Transactions on, 12*(4), 439-457. doi: 10.1109/TEVC.2007.913109

Rahimi-Vahed, A., Rabbani, M., Tavakkoli-Moghaddam, R., Torabi, S. A., & Jolai, F. (2007). A multi-objective scatter search for a mixed-model assembly line sequencing problem. *Advanced Engineering Informatics, 21*(1), 85-99.

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

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.