

Optimization and Simulation of Airport Ground Equipment and Staff Scheduling Coordination

Yiyun (Cindy) Fei, Xuejian Gong, Shu Wang, Roger Jiao

School of Mechanical Engineering

Georgia Institute of Technology

Atlanta, Georgia, USA

yfei38@gatech.edu, xgong45@gatech.edu, swang737@gatech.edu, rjiao@gatech.edu

Abstract

Airports as major transportation hubs need to transport a large number of travelers every day. Effectively managing such a large transportation system without compromising safety has always been a huge challenge. One of the keys to enhancing overall airport management is improving operational efficiency, which includes optimizing the allocation and schedules of limited resources, such as workforce and equipment. This paper proposed a method for generating optimal job schedules for Ground Service Equipment and Ground Support Staff using a two-dimensional genetic algorithm and validating using discrete-event simulations. For the 2D-GA, the method for selecting the parents is the truncation based on a preset threshold, for crossover is both the horizontal and vertical swap by generating random exchange points, and for mutation is the random resetting. The iteration for generating new offspring, or new solutions, will stop once the stop condition is met which means the optimal solution(s) is found. This paper also presents the optimization of job scheduling using the two-dimensional genetic algorithm using real flight schedules from an airport. The schedule is generated in MATLAB and used as input to the Simio simulation model for validation.

Keywords

Predictive Maintenance, Real-time, Decision Support, Fleet Operation Scheduling, and Optimization

1. Introduction

Nowadays, with the ever-closer international relations, the demand for travel is also increasing. Airports as the major transportation hubs need to transport a large number of travelers every day. For a large-scale airport, such as the Hartsfield-Jackson Atlanta International Airport, which is the busiest airport in the world, annual passenger traffic could reach 110 million. Even under the influence of the Covid-19 in 2020, there were still more than 42 million passengers traveling through the Atlanta Airport.

Ground service equipment (GSE) is an essential part to perform a turnaround in daily airport operations. GSE provides various types of services to aircraft between the flights. It is usually kept in the apron area near the terminal for better accessibility. Common tasks of GSE contain aircraft maintenance and cleaning, aircraft maneuvering and refueling, airfield maintenance, de-icing and snow removal, emergency response, moving payloads, and restocking of provisions (KB Environmental Sciences, Inc. 2015). Since there are many types of GSE, the make, model, and capabilities of each GSE type are not always the same. Additionally, the same type of GSE might be required at various locations as soon as there is a service request, which would likely be after the landing of an aircraft. Therefore, managing the GSE fleet is a complex task. From regular job dispatching to equipment maintenance scheduling, different types of GSE have different frequencies and requirements for routine tasks and equipment maintenance. One way to improve efficiency from the airside operations side is to reduce the turnaround time, which is the time that an aircraft takes from landing to taking off for a new flight. In an ideal situation, all GSE and GSS (Ground support staff) should be available right away at the requested location to have a minimal turnaround time. However, it is not always the case in the real world. In some cases, the ground support staff is unable to obtain real-time status and conditions of the GSE. As a result, unexpected failures can occur, causing an increase in the turnaround time.

Effectively managing such a large transportation system without compromising safety has always been a huge challenge. One of the keys to enhancing the overall airport management is to increase the operational efficiency which includes optimizing the allocation and scheduling of limited resources, such as workforce and equipment. Airside operations as one type of airport operations involve the management of not only the aircraft's movements, but also the ground vehicles' movements. This paper focuses on optimization of job scheduling the Ground Support Equipment, which is used to provide services to the aircraft between flights, and the corresponding Ground Service Staff.

This research aims to address the joint job scheduling problem for both GSS and GSE by generating and optimizing the operational and maintenance job schedules for both GSE and GSS using a two-dimensional genetic algorithm. An experimental case study has been created and the feasibility of the generated optimal schedule is proved by a simulation model in Simio. The overall goal is to improve the operational efficiency in airports by reducing the flight turnaround time.

2. Literature Review

Airport ground handling operations are a series of activities that process passengers, baggage, cargo, and flight supplies at the airside between two consecutive flights. It is typically divided into two categories, which are over the wing activities and under the wing activities (Tabares & Mora-Camino 2017). Those operations are usually performed by three different service providers: (a) third party handlers, (b) airline self-handlers, and (c) airport handlers, using different types of ground support equipment (Kolukisa 2011). Sometimes, each service provider would be responsible for a certain portion of the ground handling tasks. Thus, employment is more flexible and makes the workforce scheduling meaningful by reducing the operation cost as well as the labor cost. The turnaround task sequence and duration are affected by a number of factors, including but not limited to the aircraft type, ramp layout, and resource availability (Tabares & Mora-Camino 2017).

Bonilla et al. has studied a real-world case of United Airlines' operations at the ramp to develop a Ramp Operations Monitoring System in order to reduce flight delays (Bonilla et al. 2005). The system has also been validated with discrete-event simulation models. Schmidt has reviewed other research methods that have been used for turnaround modeling, including discrete event simulation, integer programming, virtual simulation, stochastic function, and agent-based simulation (Schmidt 2017). Recently, research has also been done on adopting a GSE booking system to reduce flight delays (Sagger et al. 2021). The results suggest that booking GSEs in advance has the potential to effectively reduce delays compared to using a first-come-first-serve method to fetch the GSE.

For many service industries such as airports, workforce scheduling is a challenging problem. In airport staff planning, there are usually four preparation stages (Herbers 2005). The first is the task generation using information about flight schedules, passengers and baggage, and engagement standards. The second stage is demand planning using the results from the task generation stage. The third stage is shift planning, which is designed to plan the starting time and duration of each shift to cover the demand. The fourth stage, rostering, defines the sequence and the pattern of the shift duties and days off for employees. Research has been done on optimization methods for shift planning and rostering to minimize operational costs. Ever since the integer programming techniques were proposed to solve real-world problems, it has received a lot of attention. The study of workforce scheduling using mathematical modeling and algorithms decreases the unnecessary complexity in planning manually. By integrating modular rules, metaheuristics are proposed to be valuable methods for coping with combinatorial optimization problems to efficiently implement shift planning (Clausen 2010; Lagodimos & Leopoulos 2000). Specifically, genetic algorithms have been emphasized as one of the efficient heuristics for solving optimal scheduling problems. Furthermore, the 2D genetic algorithm methods has been proposed for optimal shift planning (Gong et al. 2019).

3. Job Scheduling Optimization Using 2-Dimensional Genetic Algorithm

As a global heuristic search method, the Genetic Algorithm is commonly used to generate solutions for optimization and search problems, such as resource and workforce scheduling. It mimics the mechanism of natural selection and reproduction by probability to generate better solutions (Kumar et al. 2010). The initial population is the first generation of solutions, and each solution is a chromosome, where different genes, or elements, are combined. After creating the initial population, each solution will be evaluated by a fitness function and the reproduction process starts next. This process includes (1) Natural Selection, (2) Crossover, and (3) Mutation. The reproduction process will

continue until the solution met the predefined condition or converges to a specific value. Figure 2 shows the process flow of the Genetic Algorithm.

The joint job scheduling for GSE and GSS is basically a combinatorial optimization problem. Due to the large scale of resource types, including GSE with various capacities and functions and GSS with different skills, a one-dimensional genetic algorithm may increase the complexity of reproduction. The 2-dimensional genetic algorithm was proposed by Wang to overcome the drawback caused by the chromosome structure and validated using the application of airport shift planning (Wang 2019). By constructing a solution with a two-dimensional matrix, the selection of resources for job scheduling would be easier to manipulate. For the joint job scheduling using the 2-dimensional Genetic Algorithm, the process flow in Figure 1 is also applicable. Figure 2 shows the pseudocode for the implementation of 2D GA. The details of each process will be elaborated on in the later sections.

3.1 Problem Formulation

The job scheduling for the Ground Support Equipment and the Ground Service Staff requires various information inputs from both the flight schedules and the preliminary workforce and resource plans. First of all, by analyzing the flight schedules, the estimated turnaround time between flights will become available. Second, the turnaround tasks will be formed into a list, along with each task’s estimated starting time, duration, and ending time. The sequence of the list is sorted by the starting time of each task, from the earliest to the latest in one shift. In addition, GSEs will be grouped into various types based on their types of services and capacities. From the workforce scheduling perspective, preparation such as demand planning, shift design, and rostering need to be performed before dispatching the specific tasks. In this paper, it is assumed that the preliminary scheduling tasks are performed in advance, and therefore the number of available GSSs and their working hours are known and enough to cover all the task demands. Similarly, the number of types of GSE and the number of GSE in each type are also assumed to be known and sufficient to perform all the tasks required within the shift on time.

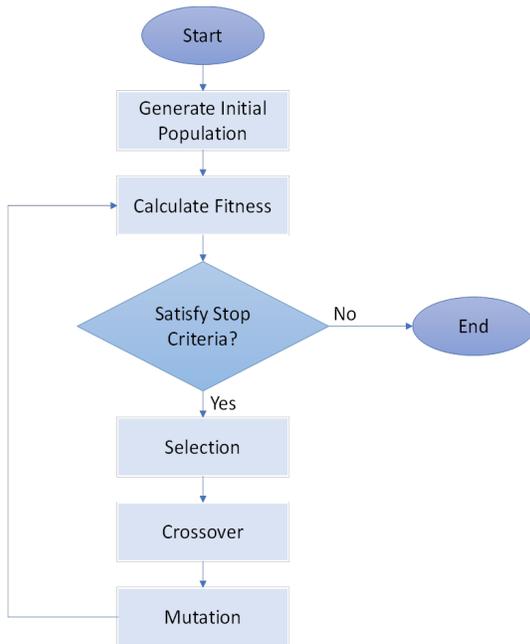


Figure 1. - Process Flow of the GA

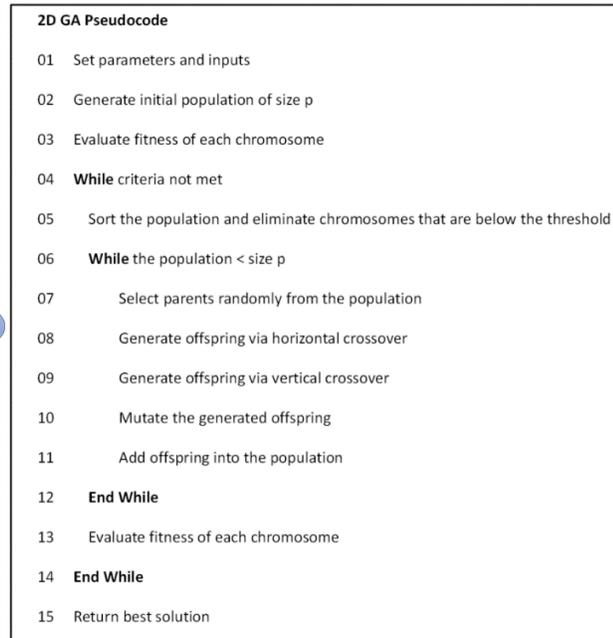


Figure 2. – Pseudocode for 2-Dimensional Genetic Algorithm

3.2 Genetic Encoding

By using the newly proposed 2D GA method, the chromosome is able to include the information for all types of GSEs, GSSs, and both the regular and maintenance tasks. Figure 3 shows an example structure of a chromosome. Each chromosome is divided into two regions, where the first region is in gray and blue, and the second region is in orange. For the first region, it is used to schedule the GSE and the corresponding GSS for all regular tasks. Each column represents one task from the input task table and each task requires a suitable type of GSE type and one GSS among

all available GSSs. The rows represent all GSE options and an intermediate table that lists all available GSEs and their types is required to increase the random scheduling efficiency by eliminating the possibility of selecting the wrong type of GSE. For each task, if a GSE is selected, the corresponding GSS will also be randomly selected and the GSS's ID will present in the cell, and all remaining GSEs will have a zero in their cells. For the second region, all GSEs that are estimated to have maintenance within the time range of the shift will be listed. Similar to region one, one GSS will be assigned to each maintenance task and the GSS's ID will be presented in the corresponding cell. The size of the chromosome matrix for each shift is dynamic because it is depending on the number of available resources and the predicted maintenance schedules.

In order to generate the chromosomes, rules have been set to increase the efficiency of this 2-dimensional genetic algorithm. First, GSE for each task will only be selected randomly from the required GSE type as listed in the input task table. Second, only one GSS will be selected for each task.

		Tasks						
GSE Type1		0	0	0	0	0	0	0
		0	x	0	0	0	0	0
		0	0	0	0	0	0	x
GSE Type2		0	0	0	0	0	0	0
		x	0	0	0	0	0	0
		0	0	0	0	x	0	0
GSE Type3		0	0	0	0	0	x	0
		0	0	0	x	0	0	0
		0	0	0	0	0	0	0
GSE Type4		0	0	0	0	0	0	0
		0	0	x	0	0	0	0
GSE That Need Maintenance		x	0	0	0	0	0	0
		0	0	0	0	0	x	0
		0	x	0	0	0	0	0

Figure 3. An Example of a Chromosome Matrix

3.2 Fitness and Stop Criteria

As stated in the previous section, the purpose of job scheduling is to find the optimal combination of resources to improve operational efficiency by eliminating potential flight delays. In order to prevent potential delays during the turnaround process, it is necessary to have minimal conflicts when dispatching the tasks, which means that one resource cannot be assigned to two or more tasks if the tasks' time overlaps.

Equation 1 is the objective function of this optimization problem, where the first half is the number of conflicts for GSE, and the second half is the number of conflicts for GSS. n_E and n_S are the total number of GSE and GSS. t_{x_E} and t_{x_S} are the total numbers of tasks assigned to the x_E th GSE and the x_S th GSS, respectively. G_i is the GSE type, n_{G_i} is the number of GSEs in each type, and k is the number of types of GSE. Equations 2 and 3 are the constraint functions that calculate whether there are any conflicts for every GSE and GSS. T is the starting time in minutes and D is the duration in minutes. Equation 2 calculates the time difference between the ending time $T+D$ of the y_E th task that is assigned to the x_E th GSE and the starting time T of the next task for the same GSE. Equation 3 does a similar calculation for GSS. Either one of these conflicts may result in delays. Therefore, the fitness, or the object function, of this application is the total number of conflicts in each solution. Moreover, the optimal solutions will have minimal conflicts and once the optimal solutions have been generated, the iteration in the algorithm will stop.

$$\min \sum_{x_E=1}^{n_E} \sum_{y_E=1}^{t_{x_E}-1} 1_{g_1} + \sum_{x_S=1}^{n_S} \sum_{y_S=1}^{t_{x_S}-1} 1_{g_2} \quad (1)$$

$$g_1(x_E, y_E) = T_{x_E, y_{E+1}} - (T_{x_E, y_E} + D_{x_E, y_E}) < 0 \quad (2)$$

$$g_2(x_S, y_S) = T_{x_S, y_{S+1}} - (T_{x_S, y_S} + D_{x_S, y_S}) < 0 \quad (3)$$

3.3 Selection

There are many existing methods for selecting the parents to generate the offspring, and they can be classified into three types, which are fitness proportionate methods such as the Roulette Wheel Selection, ordinal based method such as the Tournament Selection, and threshold-based methods such as the Truncation Selection. In this application, the Truncation Selection is used by setting a threshold value for the number of conflicts and removing all the chromosomes that have more conflicts than the threshold value. For the remaining chromosomes, each one has the same probability to be selected as a parent. The selection step is in an iterated loop, which will end if the sum of the remaining chromosomes and the generated offspring reach the initial population size. In each iteration, two parents will be selected randomly and start reproduction by crossover and mutation.

3.4 Crossover

The crossover operations aim to produce offspring by exchanging the genes between two parent chromosomes. By using the substring crossover method, an exchange point that represents the column number will be generated randomly to define the segments for gene exchange. The crossover operations of the two-dimensional chromosome matrix can divide the parent chromosome from the exchanging point horizontally and vertically (Wang, 2019). In this application, both the one-point and the two-point crossover methods will be used. Because there are two regions in each chromosome and the limitation of the GSE type, different methods will be applied accordingly.

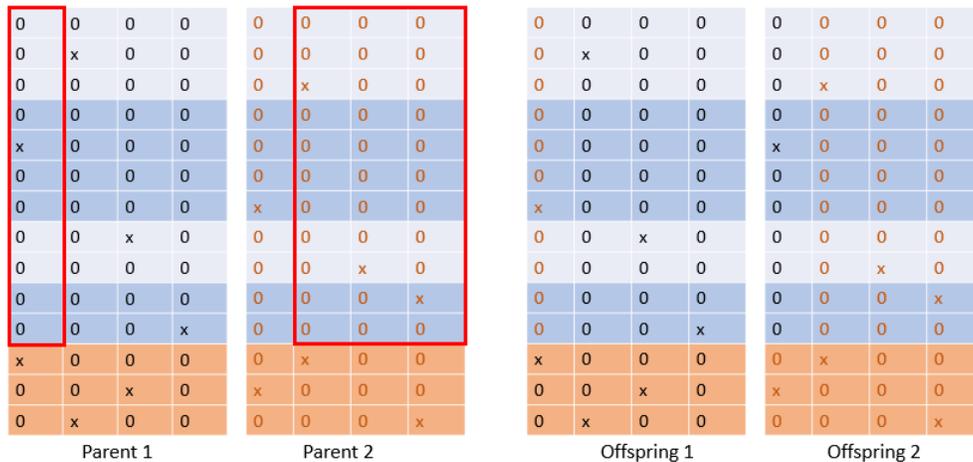


Figure 4. Horizontal Crossover in Region 1

After the selection process, two parents will be generated, and each pair of parents will produce 3 pairs of offspring. For the first pair, a horizontal crossover will happen at the first region, which contains the assignment for regular tasks, at a random exchange point R1. Figure 4 shows an example of the parents and offspring for horizontal crossover in the first region. The second pair of offspring are also generated in the first region. Instead of a one-point horizontal crossover, a two-point vertical crossover will be applied. Figure 5 shows an example of the vertical crossover in the first region. The last pair of offspring is generated in the second region by a one-point vertical crossover. Figure 6 shows an example of the generation of the third offspring pair.

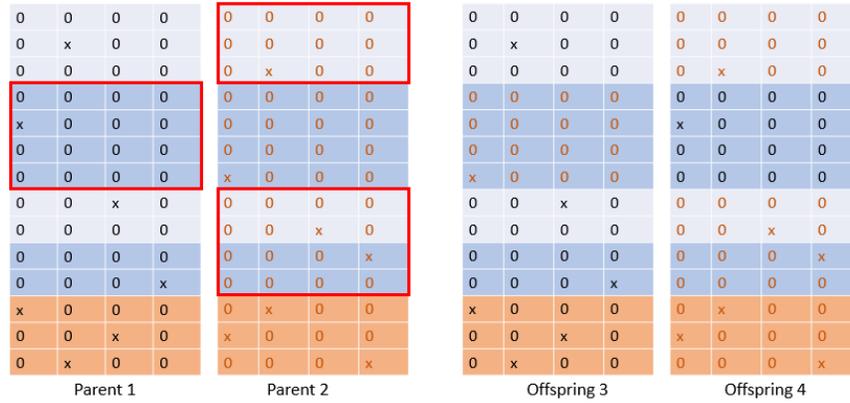


Figure 5. Vertical Crossover in Region 1

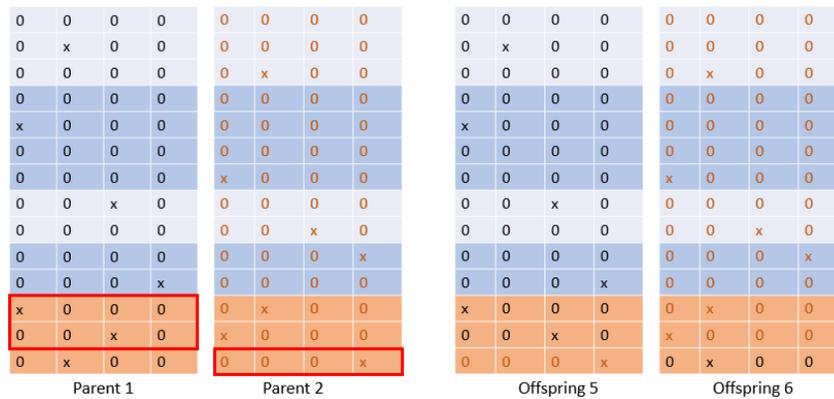


Figure 6. Vertical Crossover in Region 2

3.5 Mutation

The mutation operations are applied to add diversities to the populations. The methods for mutation include the bit flip method, random resetting method, swap, scramble, and inversion mutation at arbitrary positions. In this application, the random resetting method is used, where a random value from a set of permissible values replaces a randomly selected original gene. For each chromosome, the algorithm will loop over each gene. If the value of a gene is zero, no mutation will be applied. On the other hand, if the value of a gene is non-zero, it will have a preset mutation rate to replace the assigned GSS with another one from the provided GSS list. Therefore, this mutation operator only affects the GSS assignment. Figure 7 shows an example of the mutation process. After the mutation, the original GSS who has an ID number 2 has been replaced with the GSS who has an ID number of 5 in row two and column two.

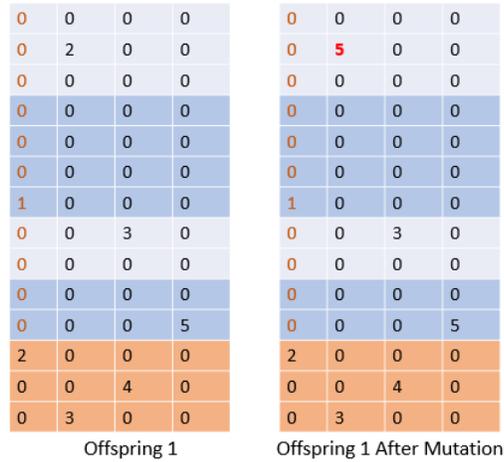


Figure 7. Mutation

4. Case Study and Experiment

In order to validate the proposed methods for solving job scheduling problems, an experiment has been done using real-world data. Table 1 is a part of the real flight schedule at the Harrisburg International Airport in Pennsylvania. The flight schedule contains the flight information, such as the previous flight arrival times, the next flight departure times, the locations, and the turnaround times, for the American Airline from 11 am to 9 pm on March 17th, 2022. Since the airport has a relatively small size, the aircraft that go to that airport also have small capacities. Therefore, the duration of turnaround tasks decreases, causing the average turnaround time to be around 30 minutes. In this experiment, it is assumed that each flight will have four types of ground service, which are baggage loading/unloading, aircraft refueling, cabinet cleaning, and aircraft pushback. Each type of service requires a different GSE type, and the tasks may start parallel. Figure 8 presents an example of the Gantt Chart showing the four tasks between the flights. In addition, in order to ensure that there are enough resources to cover the task demand, both the numbers of available GSS and GSE are set to 8. Based on the given information, an input task list can be generated, as shown in Table 2 in black, and the highlighted cells are the outputs after optimization.

Table 1. Flight Schedule for American Airline on March 17th

Airline	Arrive	Depart	Destination	Gate	Duration
AA 6135	11:19	11:45	PHL	C3	0:26
AA 6014	14:35	15:06	CLT	C3	0:31
AA 6025	16:15	16:51	BOS	B2	0:36
AA 5241	16:27	16:57	CLT	C3	0:30
AA 3073	17:30	18:01	ORD	C1	0:31
AA 5258	19:16	19:45	CLT	C3	0:29

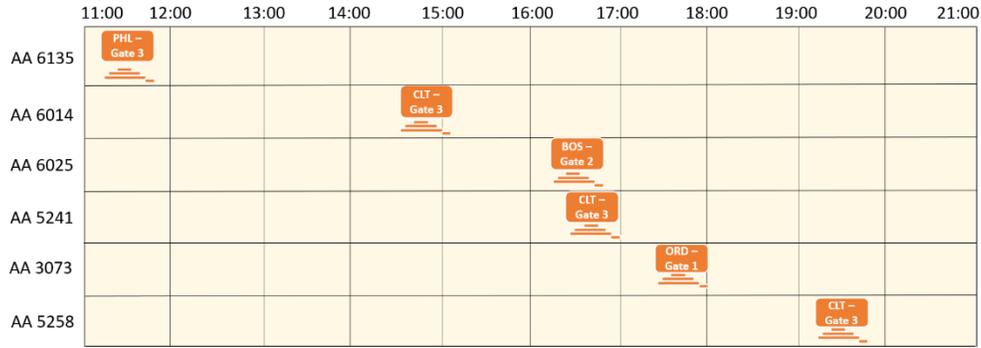


Figure 8. Example of Turnaround Task for the Selected Time

The 2D GA is written in MATLAB version R2021b. It retrieves the information in black from the input task list and starts processing the algorithm in iterations. After the termination condition is met, it outputs the information in the highlighted cells in Table 2. This updated task list is then to be used as the input for the discrete event simulation. Figure 9 shows an example of a part of the job schedule that has not been optimized in the form of a Gantt chart comparing with the optimal job schedule generated by the 2D-GA. It proves that the 2D-GA can generate a job schedule with minimal conflicts.

Table 2. Generated Task Schedule

Task ID	Task Start Time	Duration	Task End Time	GSE Type	Location	Description	Assigned GSS	Assigned GSE
1	20	23	43	1	3	Baggage	6	102
2	22	15	37	2	3	Refuel	1	202
3	24	8	32	3	3	Clean	5	302
4	43	2	45	4	3	Pushback	4	402
5	216	23	239	1	3	Baggage	5	102
6	218	20	238	2	3	Refuel	1	202
⋮								
25	403	30	430	1	Maintenance	Maintenance	7	202
26	525	120	675	4	Maintenance	Maintenance	5	401



Figure 9. Example of a section of the original job schedule vs. Optimized schedule

A discrete event simulation has been built to simulate the experiment system. It contains eight workers, or GSSs, eight GSEs, three gate servers, and one maintenance server. Each task in the list will be treated as an entity that is distributed by the source at the given task starting time to the given destination server. The task will not start until the server has seized both the required GSS and GSE, which will happen only if they are available at the moment. Figure 10 is the layout of the simulation model. The goal of the simulation is to verify that the job schedule generated by the two-

dimensional genetic algorithm is feasible and does not cause delays for the upcoming flight. Therefore, the actual task durations from the simulation are compared with the estimated task durations.

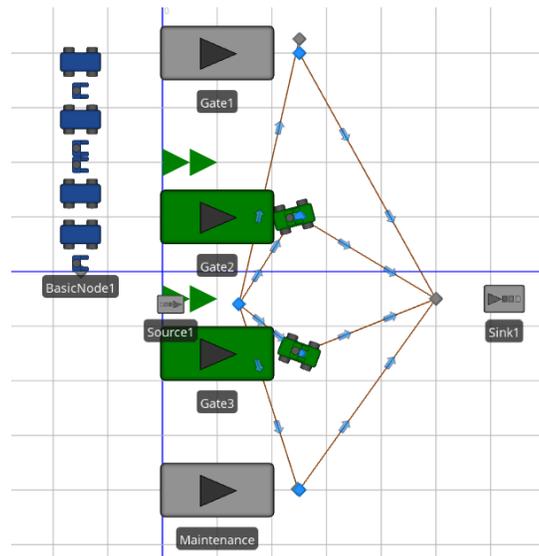


Figure 10. Job Simulation Model In Simio

5. Conclusion

This paper presents the solution of joint job scheduling for the ground support equipment (GSE) and the ground service staff (GSS) using the two-dimensional Genetic Algorithm. Both the flow chart and the pseudocode of the 2D GA are provided. The required information, such as the flight engagement schedules and the preliminary resource planning, are also discussed in this paper. In summary, the method for selecting the parents is the truncation based on a preset threshold, for crossover is both the horizontal and vertical swap by generating random exchange points, and for mutation is the random resetting. The iteration for generating new offspring, or new solutions, will stop once the stop condition is met which means the optimal solution(s) is found. This paper also presents a case study of the optimization of job scheduling using the two-dimensional genetic algorithm using real flight schedules from the airport and simulation by a discrete-event simulation model. The schedule is generated in MATLAB and used as input to the Simio model. The simulation results also suggested that the optimized schedule is feasible.

For limitations and future works, although the optimal job generation uses real flight schedules from the MDT airport, assumptions have been made on the type and duration of turnaround tasks and also the number of resources. For future work, real operational information can be used as input for job schedule generating. Furthermore, the 2D-GA presented in this thesis does not take into consideration the travel time between tasks and resource utilization. Future work can be done to improve the 2D-GA by evaluating the travel time and resource utilization and including those factors in the fitness calculation.

References

- Anderson, C. A., Jones, K. F. and Ryan, J., A two-dimensional genetic algorithm for the Ising problem, *Complex Systems*, vol. 5, no. 3, pp. 327-334, 1991.
- Bonilla, J., Mohammed, H., Norris, J., Phung, T. and Zegada-Frias, I., Ramp operations monitoring system, *IEEE Design Symposium, Systems and Information Engineering*, pp. 127-133, 2005.
- Clausen, T., A Rule-based local search algorithm for general shift design problems in airport ground handling, *DTU Management*, no. 16, 2010.
- Gonçalves, J. F., A hybrid genetic algorithm-heuristic for a two-dimensional orthogonal packing problem, *European Journal of Operational Research*, vol. 183, no. 3, pp. 1212-1229, 2007.
- Gong, X., Wang, S. and Jiao, R., An efficient 2D genetic algorithm for optimal shift planning considering daily-wise shift formats: A case of airport ground staff scheduling, *IEEE International Conference on Industrial Engineering and Engineering Management*, pp. 1440-1444, 2019.

- Herbers, J., Models and algorithms for ground staff scheduling on airports (Doctoral dissertation, Bibliothek der RWTH Aachen), 2005.
- KB Environmental Sciences, Inc. and National Academies of Sciences, Engineering, and Medicine, Improving ground support equipment operational data for airport emissions modelling, *Washington, DC: The National Academies Press*, 2015.
- Kolukisa, A., Evaluation aircraft turnaround process in the framework of airport design and airline behaviour (Doctoral Dissertation, University of Porto), 2011.
- Kumar, M., Husain, M., Upreti, N. and Gupta, D., Genetic algorithm: review and application, 2010.
- Sagger, S., Tomasella, M., Cattaneo, G. and Matta, A., Enhanced operational management of airport ground support equipment for better aircraft turnaround performance, *Winter Simulation Conference*, pp. 1-12, 2021.
- Schmidt, M., A review of aircraft turnaround operations and simulations, *Progress in Aerospace Sciences*, vol. 92, pp. 25-38, 2017.
- Tabares, D. A. and Mora-Camino, F., Aircraft ground handling: analysis for automation, *17th AIAA Aviation Technology, Integration, and Operations Conference*, 2017.
- Tsai, M. W., Hong, T. P. and Lin, W. T., A two-dimensional genetic algorithm and its application to aircraft scheduling problem, *Mathematical Problems in Engineering*, 2015.
- Wang, S., Workforce scheduling with large-scale mixed integer programming using column generation and 2D genetic algorithms: an application to airport ground staff scheduling (Master Thesis, Georgia Institute of Technology), 2019.