Dynamic Airport Ground Crew Scheduling Using a Heuristic Scheduling Algorithm

B.Rodič, A.Baggia

Abstract**—**We describe the development of a heuristic scheduling algorithm developed for the purpose of workforce scheduling and shift construction at the Ljubljana JožePučnik Airport. The goal of the project was to develop a solution that would generate workforce shifts and allow rapid rescheduling and thus shorten the airport response time and improve the adaptability in conditions of dynamic flight schedules. Airport ground crew scheduling problem proved to be complex as there were conflicting demands in assigning personnel and equipment to tasks connected with aircraft arrivals and departures. We have attempted to use workforce algorithms described in literature to construct the solution, but due to specific requirements of the airport we have constructed heuristic algorithms that perform task schedule optimization, personnel and equipment requirements optimization and shift planning for airport ground crews and that may be also applicable to other airports of similar size and traffic.

Keywords**—**airport, flight information system, ground crew, scheduling, shift planning

I. Introduction

SCHEDULING problems and their solutions have been widely researched and discussed in recent years (e.g. [1], [2], [3] and [4]). Brucker [2] characterizes the theory of scheduling with a virtually unlimited number of problem types. Different types of resources can be scheduled (e.g. production, personnel, activities, aeroplanes, trains etc.). Researchers are trying to improve scheduling algorithms, e.g. Bester et al. [5] and Wassan et al. [6], but there is no general algorithm. With the wide range of predefined and custom-made scheduling solutions, deciding which scheduling solution to use is not simple.

Over decades, the scheduling community has been focused on developing algorithms that work well on a particular problem; therefore, it is very likely that those types of algorithms could not run on different types of scheduling problems. However, the research on algorithms that work well over a range of scheduling problems has been increased with the use of hyper-heuristics [7]. One of the main obstacles to applying AI planning techniques to real problems is the difficulty of modelling the domains [8].

Most sectors of the manufacturing and service industries require scheduling of several types of resources in addition to machine tools scheduling, and one of the most important resource types are human resources [3]. International airports are complex systems that require good functioning and coordination of all their parts. Due to a large number of tasks, connected with arrivals and departures of aircrafts, and frequent changes in flight schedules, the use of good workforce and equipment scheduling algorithms integrated into the airport information system is crucial for good performance of an airport as a system. In this manner we can drastically reduce the time needed to produce a ground crew work schedule, improve its quality, reduce the frequency of errors, and are able to quickly reschedule crews in case of changes in the flight schedule.

Work supported by Creative Core FISNM-3330-13-500033 'Simulations' project funded by the EU and the European Regional Development Fund.

Burke et al. [7] state that the responsibility for 50% of flight delays can be attributed to carriers, while 19% of delays occur due to problems with airport operations. According to Air Transport Association's data [9], every minute of delay costs approximately 100 USD (in year 2008), including fuel costs, airline and airport personnel, aircraft maintenance and depreciation, passengers' and cargo owners' time, and not including contract penalties for delays and damages paid to passengers. Due to frequent changes in flight schedules, the possibility to dynamically alter and adjust workforce and equipment schedules is therefore very important for airport operations [10]. Manual scheduling is too slow for the conditions that modern international airports operate in, and too prone to errors due to human errors caused by stress and insufficient time for schedule production, making computer supported scheduling a necessity.

In our research work for the Ljubljana JožefPučnik Airport we have developed algorithms that can compute the needs for individual types of workers and equipment per each minute within the working day, and produce work shift plans that take into account legal, economic and ergonomic requirement.

The goal of the project was to develop an automated workforce scheduling and shift generation system that would produce results comparable to manually produced schedules and shift in a fraction of the time needed for manual production, in order to improve ground crew scheduling response time to changes in flight schedules at the airport.

The basis for algorithm operation is heuristics, as the mathematical methods described in examined literature have been found to be inapplicable to the scheduling problem at this airport.

The main tool in the development of the heuristic algorithms was a simulation model that we have developed with the purpose of recording and verification of heuristics derived from the manual scheduling process. The simulation was then used to perfect the heuristic algorithms until they produced acceptable schedules and shifts.

The paper is organized as follows. In the second section we describe the theory of scheduling, the particulars of airport ground crew workforce scheduling. The next section describes the issues we have encountered in this ground crew scheduling project and our approach in the development of the scheduling algorithm. The fourth section describes the developed scheduling algorithm and its main parameters, while the final, fifth section discusses the main results and lessons from the project and the opportunities for further development.

II. Methodology

The theory of scheduling is characterized by a virtually unlimited number of problem types [11]. Spyropoulos [12] defines scheduling as the allocation of activities/actions over time to resource according to certain performance criteria. Different types of resources can be scheduled and for every type diverse solution or scheduling algorithms exist. Pinedo [4] emphasizes that every type of resource has attributes and parameters important for the process of planning and scheduling. As presented in Papler [13] the scheduling system is important for better capacity usage, better understanding of the problem, centralized information, for keeping the knowledge in the organization and nevertheless for efficient scheduling.

A scheduling problem may be described as a set of parameters which describes the set of machines, set of resources, set of tasks or operations, set of optimality criteria and the type of schedule. Not all of these parameters must have a value, but the class of schedule must always be described. Thus, a scheduling problem is in fact a set of instances obtained by assigning values to all the problem parameters [14] .

Pinedo [3] defines scheduling as a decision-making process with the goal of optimizing one or more objectives. Leung [15] defines scheduling as the allocation of scarce resources to activities with the objective of optimizing one or more performance measures. Optimization is usually first attempted with the use of mathematical methods such as fuzzy multiobjective linear programming [16], with heuristics being employed where mathematical methods can't be used to model certain aspects of the problem [17] . Further, Cesta et al. [18] describe scheduling as a set of predefined tasks, employing heuristics, which are capable of optimizing with respect to some criteria on the duration of the tasks and/or resource usage. Different types of scheduling problems are characterized by the existence of two sets of constraints, usually referred to as hard and soft constraints [19]. As we can see, depending the context of the problem they solved, different authors define scheduling in a slightly different way.

A certain similarity between scheduling procedures in the case of production resource scheduling and personnel scheduling process was found [20]. Scheduling procedures in the definition of production scheduling (S_1) and the definition of personnel scheduling (S_2) are presented in Table 1. The differences between presented procedures are in the approach and search procedures [20] . Procedure S1 searches for the most appropriate machine *Mi* for an order *Ni* in a predefined time frame, while considering constraints. The evaluation considers all constraints that need to be included when generating a schedule.

Procedure S_2 searches for the most appropriate person for a single activity in a predefined time interval. Constraints, hard or soft, are also considered in this case. When established that for activity A_1 persons W_1, W_2 and W_3 are appropriate, the search procedure finds a person with the highest value of criteria function considering the constraints. Person*Wi* is scheduled for the activity *Ai*.

The result of both procedures is the same; the difference is in the sequence of steps leading to the solution, where procedure S_1 searches for the machine for a certain order and procedure S_2 searches for the person for a specific job.

Table 1: Comparison of two different scheduling problem search procedures.

Production Scheduling problem S_1	**Personnel Scheduling problem S_2**
While ($t_1<t_k$)	While ($t_1<t_k$)
While (order)	While (activity)
While (machine)	While (person)
Evaluate order	Evaluate person
End-while	End-while
Find machine(order)	Find person(activity)
End-while	End-while
End-while	End-while

It is obvious that only Steps Two and Three of $S_1$and S_2 are interchanged due to the context of the domain. If we represent scheduling problem as a black box, Activity *a* or Order *n* is substituted with Input *x*(scheduling requirements) and the results of scheduling with Output *y* (results of scheduling) as presented in Figure 1.

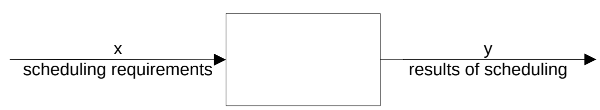

Figure 1: The scheduling problem as a black box.

As presented in Figure 1, scheduling requirements can be defined with activities or orders.

$$x_i \in \{a_i \in A, n_i \in N\} \quad (1)$$

The scheduling problem *Sn* could be written as presented in Table 2.

Table 2: The scheduling problem search procedure.

Scheduling problem S_n

While (t_1<t_k)
    While (x)
        While (v)
            Evaluate
        End-while
        y = find v(x)
    End-while
End-while

To summarize, the search procedure is general, disregarding the type of scheduling problem; the search procedures are equivalent:

$$S_1 \equiv S_2 \tag{2}$$

Only the variable *v*, machine or person is changed:

$$v_i \in \{m_i \in M, w_i \in W\} \tag{3}$$

In Expression (3) m_i states the machine and w_i a person. The search procedure of scheduling remains the same regardless of the type of scheduling problem. The differences between procedures are presented with variables for which optimal values are searched by the algorithm. The finding of an analogy in the scheduling procedures led to the attempts to describe the scheduling problem with a general model, regardless of the type of resource that is to be scheduled [20].

As presented in Expression (2), equivalence exists between different scheduling procedures. Every scheduling domain (SD) can be described by four basic elements: object types, syntax, parameter and algorithm:

$$SD \subseteq O \times P \times S \times A \tag{3}$$

Where

- *O* is an object type.
- *P* is a parameter.
- *S* is a syntax.
- *A* is an algorithm.

“Object type” represents the components that describe an object, schema or rule. Object types can be elementary or complex. Elementary object types are usually characterized by input, output and simple transparent rules. Complex object types constitute as a Cartesian product of elementary object types:

$$o_i \subseteq o_1 \times o_2 \times \ldots \times o_n \tag{4}$$

The syntax describes relationship rules between individual object types. Different types of relations between object types describe the behaviour of the scheduling problem. Every object type is described with parameters or attributes. Parameters define the specifics of an individual scheduling problem. Sometimes, relationships between object types also need to be specified in detail. Therefore, parameters can also be used in the syntax. Object types with their parameters and defined syntax represent the input for the scheduling algorithm. The scheduling algorithm defines the logic to generate a schedule.

Scheduling problems in the air transport industry are more demanding than traditional machine scheduling problems. While machine scheduling research has several hundred years of history and a standard terminology for problem description, there is less history and research behind air transport scheduling, and the terminology is less unified [10]. For example, despite some similarities the ground crew scheduling problem differs in several ways from the scheduling algorithms applicable to multiple machine scheduling problems as described in Kofjač and Kljajić [21]:

- the sequence of operations per job is not fixed, and the operations (e.g. cleaning, passenger guidance) can be scheduled independently to some extent,
- the sequence of orders (i.e. flight schedule) is known well in advance, however it changes often, sometimes shortly before the operations are to begin.

Since the sequence of operations per job is not fixed, it is possible to treat the scheduling of individual ground crew teams (e.g. stewardesses, baggage handlers) as single machine problems such as described in Vakhania [22], with ground crew teams representing operations or “machines”, but there is a very significant difference:

- the time window for the execution of operations is foremost defined by the presence of the aircraft, therefore there is very little scheduling flexibility.

Most papers on air transport scheduling are focused on flight scheduling [11], [10] or aircraft crew scheduling [14], [23], [3] and less on airport crew scheduling problems. Chu [24] described scheduling of baggage service employees at the Hong Kong airport. An goal programming based algorithm determines workforce needs per hour, per day, and generates daily schedules. Herbers [25] presented models and algorithms for airport ground staff in his doctorate thesis. He describes the optimization problems in different phases of planning and proposes procedures for requirement planning, shift planning and schedule assembly. Broggio et al. [26] divided the rules and demands in airport ground crew scheduling into two groups: hard rules and soft rules. They described the scheduling problem as a whole number programming problem and used an optimum oriented polyhedral algorithm and robustness oriented local search heuristics. By combining these two methods the optimality and robustness of the solution is to be ensured. Hasselberg [27] described a two-step solution for the scheduling problem. First step defined the blocks, while step two assigns the blocks to individuals. In addition to worker competences and work hours demands they also take into account transitions of workers between units and the costs of schedules. Bazargan [28] in the chapter of workforce scheduling presents several mathematical methods for airport ground crew scheduling. The case of the JFK airport and one airline is examined. The required number of workers is calculated, but the rules for these calculations are not described. The goal of the model is to determine the minimum required number of workers and set their work schedules in a way to respect the limitations such as working hours, number of shifts, and number of working days in a row.

The approaches to shift planning and crew assembly found in the literature often use assumptions with strongly limited validity, or deal with simplified problems, thus limiting the approaches wider practical applicability [25]. The case we describe in this paper involves a small national airport, where the scheduling problem did not fit any of the examined mathematical methods, requiring a heuristic approach to shift and work schedule planning. The expert knowledge of the heads of departments that produce work schedules for the airport ground crews was used as a basis for the heuristics, and the scheduling problem was described.

III. The Ground Crew Scheduling Problem

Resources are among many important factors of a successful organizational system. Organizational resources can be divided into three basic classes; human resources, means of work and work objects. Although the ground crew operations naturally require means of work, i.e. technology, the assumption in our case was that work means are always available, and we have focused on human resource scheduling.

The arrivals and departures of aircrafts require the execution of a set sequence of ground crew tasks. Each task has a planned duration, and can be executed only within a set time frame delimited by the earliest possible task start time of the task and the latest possible end time of the task. The time frame is determined by the physical accessibility of the aircraft and the required task sequence. Air transport delays are very expensive, requiring a strict adherence to the planned duration of tasks. The main scheduling problem is therefore how to guarantee the execution of the required sequence of tasks within the planned duration with the minimum number of workers.

The requirements for different types of staff assigned to individual tasks depend on several criteria. In manual scheduling, the heads of departments also took into account the special requirements of individual airlines, flight destinations, and types of flights (charter, scheduled, transfer). The heuristic algorithm was developed to take into account the following criteria:

- Arrival or departure,
- Flight type,
- Aircraft type,
- Airline,
- Destination.

Different tasks tied to specific aircraft types require varying numbers of workers and have varying durations. Certain tasks can be also postponed in case of a long stay (aircraft departs several hours after arrival), allowing the airport to assign some of the staff to more urgent tasks on other flights.

The process of airport ground crew and equipment scheduling is shown on Figure 2. Several departments and individuals cooperate in the development of a work schedule: the planner, the human resources (HR) department, and the IT department. The HR department provides up to date information on available employees, while the IT department provides the flight schedules.

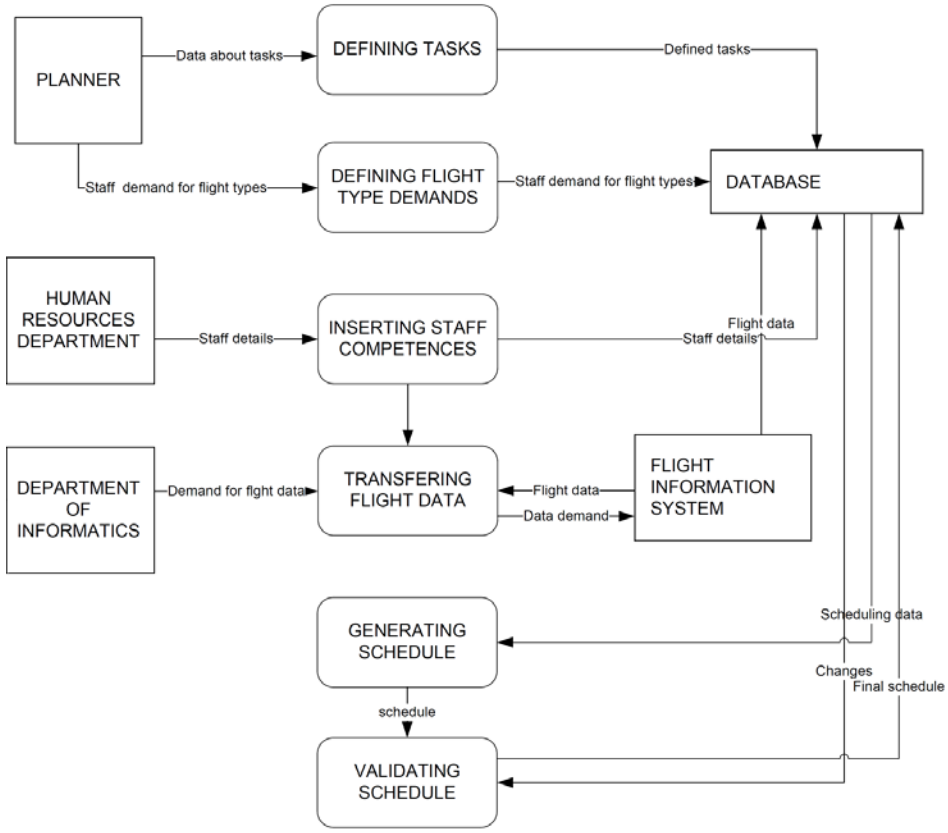

Figure 2: Scheduling process at the airport

In the process of several meetings we have recorded the rules used by operators preparing schedules and shifts. To verify the accuracy of recorded rules and develop the heuristic algorithms, we have developed a simulation model using Microsoft Excel VBA (Visual Basic for Applications) that was used to visualize the schedules produced by the heuristic algorithms on the basis of flight schedules. Several iterations of algorithms modification and verification were necessary to produce acceptable (i.e. comparable to manually produced) schedules and shifts.

An example of task schedule and workforce needs for individual tasks is shown in Figure 3, where we can see that the preparations of a size B aircraft (e.g. Airbus A320) for departure require the execution of 12 distinct tasks, with the peak workforce requirement of 17 workers. For departures, the tasks are timed in hours and minutes to planned departure time. Tasks execution can start up to two hours before planned departure. Thus the tasks in graph in Figure 3 start at 2:00, with 0:00 being the planned departure time.

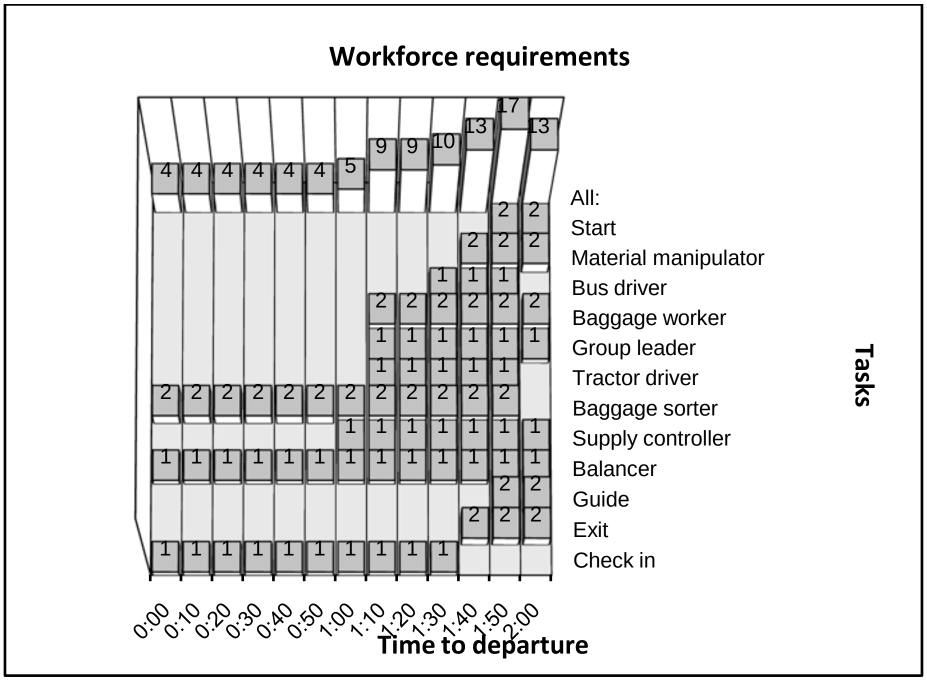

Figure 3: Tasks schedule and workforce requirements for departure of a size B aircraft

The tasks at the airport are divided into fixed tasks and operative tasks. Fixed tasks do not depend on the airport traffic, while for operative tasks the required strength of workforce depends on the traffic at the airport.

The tasks are assigned to one of the three service departments:

- Aircraft supply service,
- Passenger service,
- Technical/fire brigade service.

Each of these three service department has specific rules that are taken into account by the heads of department in charge of daily workforce schedule generation. For example, during a working day, some workers can move between different tasks within their service departments, allowing a degree of flexibility and reducing the total required number of workers. Schedules are prepared every 14 days, for the next 14 days. These schedules usually require very few changes during their execution, and the changes affecting a given work day are incorporated into schedules at least one day before. However, there are also changes within a given day due to extraordinary events. Extraordinary events or disruptions can be caused by various factors. Schaefer et al. [14] classify the disruptions as frictional (short term; e.g. waiting for the passengers, minor faults, local weather) and serious disruptions (long term; serious fault, major storm systems). The response to extraordinary events needs to be fast, and heads of departments at the airport are normally successful in implementing dynamic schedule alterations.

Until the completion of this project, the heads of departments at the airport performed the scheduling manually, using heuristics developed through experience, without formalized algorithms and without automation. They obtained flight scheduled information from the information system and used it to manually construct a work schedule. IT support was limited to office tools such as spreadsheets and did not include specialized scheduling tools. The input into schedule construction included requirements derived from flight information and informal and formal competences of the available staff.

Preparation of data for the scheduling algorithm

The knowledge and procedures used by the heads of service departments had to be recorded, formalized, and presented in a generalized form. The first phase in algorithm development was the definition of scheduling criteria. The data required for scheduling are obtained from two sources: the information system (flight information and HR data) and the expert knowledge of the planners (heads of service departments).

The fundamental data used for the scheduling algorithm is the flight data. This data is obtained the Flight Information System (FIS), which contains the data on scheduled arrivals and departures of aircraft. FIS is the basic information system of the airport and is linked with information systems of other airports. Flights can be either arrivals or departures. The most important attribute in FIS data is flight number. Using the flight number we can determine how long the airplane will stay on the ground. This time is important for the scheduling of variable tasks, i.e the tasks that have a movable execution window.

Apart from the flight type, the scheduling algorithm also uses the data on traffic type, aircraft type, and airline. These data are also obtained from the FIS. All of these data are used as criteria that influence the list of tasks and workforce and equipment requirements per task. The criteria are used in the algorithm sequentially and have different priorities: some criteria can override other criteria. The sequence and priorities of criteria is predefined for all service departments and can be changed by the operator in charge of the scheduling application.

In the first phase of our project the heads of departments have defined workforce strength requirements, start of task and end of task for each of the listed criteria, with separate specifications for arrivals and for departures. In the next phase we used an approach similar to Mason and Ryan [29], and used a simulation of heuristic schedules to help the heads of departments at the airport determine the minimum workforce requirements. Simulation can be a powerful tool for group decision support and can help a decision group reach better results faster [30], [31].

In the next phase of our project, the specifications for workforce requirements were used to develop an algorithm prototype using Microsoft Excel and Visual Basic for Applications. The prototype used FIS data to produce workforce requirements per each minute within a given time span, determined by the time span of flight schedule data. The time span ranged from several days to several weeks. The workforce requirements generated by the prototype were excessive, as the required number of workers according to the algorithm exceeded the actual number of workers, as defined by manual schedules, during traffic peaks several times per each working day. This has led to development of additional heuristic rules by the developer staff and the heads of service departments at the airport, which took into account that a worker can move from task to task in very short time periods, and changes in workforce requirements that allowed partial assignment of workers to tasks, allowing workers to be assigned to several tasks within a time frame.

IV. Result: Algorithm for Workforce Requirements and Shift Generation

The developed algorithm has two distinct phases. The first phase determines workforce needs for variable tasks, and the second phase generates shifts, based on workforce needs and HR data. Both phases allow for manual correction of results. The shift generation algorithm was developed using the data model of a general staff scheduling tool. Fixed tasks have predefined workforce needs, and a separate shift generation algorithm is used to produce shifts for the fixed tasks.

The main problem that had to be solved in the generation of workforce needs difficult are the daily traffic peaks i.e. occasional significant increase of the frequency of arrivals and departures that last less than the minimal shift duration, i.e. less than a few hours. During these daily peaks an additional number of workers are theoretically needed, however the requirements are increased only for a very short time, e.g. 10 minutes. Figure 4 shows the difference between workforce requirements as calculated by the heuristic algorithm (in red) and the number of workers that would be normally (manually) scheduled (in blue). The challenge for the heuristic workforce requirement calculation and shift optimization is how to smooth these peaks without causing flight delays due to workforce overload. Seasonal peaks make no difference to daily scheduling as their duration is longer, and can be thus dealt with by additional shifts, i.e. additional workers.

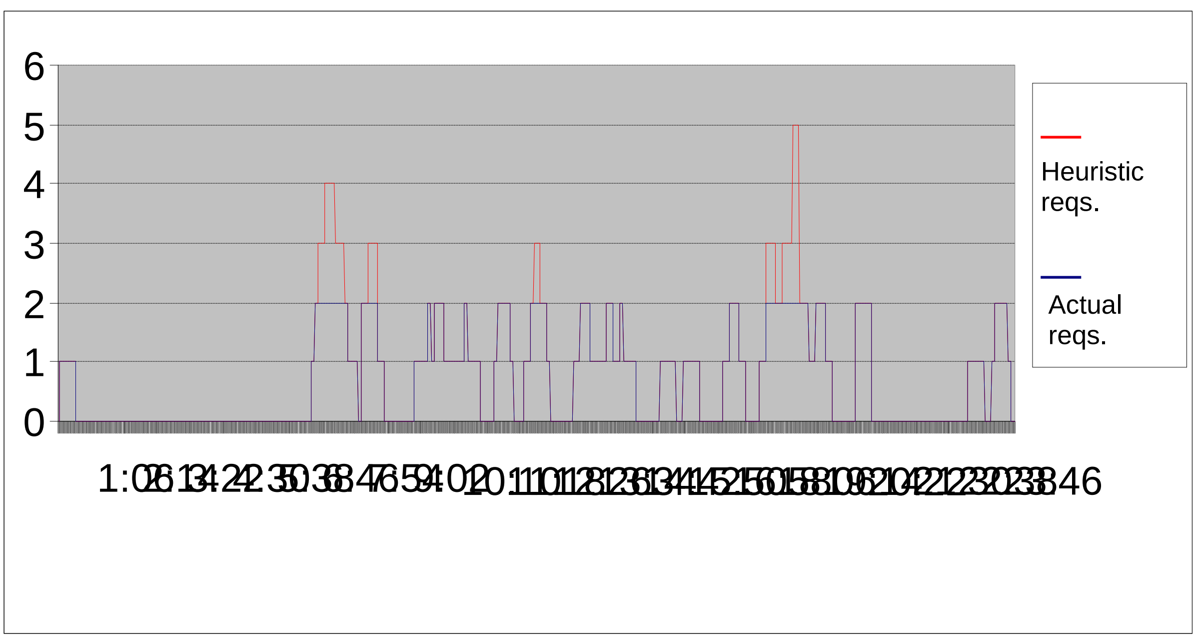

Figure 4: daily workforce requirements for Cargo balancer skill group showing short term peaks in requirements

To smooth the workforce requirement peaks the algorithm needs at least one degree of freedom (the possibility to change at least one schedule parameter). The available degrees of freedom in the given case are:

- The possibility for transfer of workers between task types within a working day, limited by worker skills (classified as “skill groups”), and
- The possibility to move the execution time of individual tasks within the allowed time frame.

The following sections present the scheduling algorithm for the operative tasks that are more complex from the scheduling aspect.

First phase of the algorithm

In the first phase the algorithm examines the flight data and generates a list of required tasks per flight. Per each required task, a default workforce requirement and default start and end time is entered into a database table. Then the detailed flight data are examined to tune the exact flight task requirements. We have defined a criterion for each relevant type of flight data has been defined, and these criteria are used by the algorithm to compute the detailed task requirements for each flight. The equipment requirements are derived from workforce needs.

Several criteria types are used in the scheduling. These criteria are divided into absolute criteria, which set new values for individual parts of requirements regardless of the current values of task requirements for a given flight, and relative requirements, which modify (add or subtract) the current values of task requirements for a given flight.

The criteria define how to adjust the number of workers and the start and end time of their tasks to the requirements defined by flight data. Each criterion sets/modifies one or several requirements (number of workers, start time of task, end time of task). Relative criteria can have positive (increase requirements) or negative values (decrease requirements). The final requirements are computed by using all relevant flight data and comparing it to the criteria. The criteria priorities define the sequence of criteria in the algorithm. The first criterion examined is flight type (arrival or departure). Arrivals have a higher priority than departures, and thus the workforce (and related equipment) requirements for arrivals are determined first. This criterion defines the default (i.e. starting) list of tasks and workforce requirements and default start and end time per task. The criteria that are examined after that can either increase, decrease, or set new values for the requirements. As several criteria change the workforce needs by fractions, the final calculated workforce requirements are rounded up to the whole value (e.g., if a final workforce requirement is 2.1, it is rounded up to 3.0).

Second phase of the algorithm

In the second phase of the algorithm the workgroups and shifts are constructed using the task requirements from the first phase, the data on worker categories, available equipment and allowed shift timing (start and end hours). The shifts are constructed per skill group.

Optimizing workforce needs: workgroup generation

A skill group is defined as a set of tasks that can be performed by any worker that belongs to the skill group. This allows two subsequent tasks from a single skill group to be performed by the same individual, reducing the required number of workers within a shift. The degree of freedom is therefore the possibility of transition of workers between different types of tasks within a shift. Each worker is assigned only one skill group, and each shift is also assigned only one skill group.

If several workgroups shares the same skill group, the workers can be moved to move from one workgroup to another. Table 1 shows some of the tasks and skill groups that allow this type of flexibility.

Table 1: some of the skills and skill groups

Task	**Skill group**
Balancer	Balancer
Supply controller	Supply controller
Cleaner	Cleaner
Fecalist	Fecalist
Baggage worker	Baggage worker
Check in	Stewardess
Gate	Stewardess

Normally the workgroup sequence is constructed in this sequence:

- For each flight the algorithm checks if a skill from a skill group is required,
- For each workgroup the algorithm assigns the required equipment,
- Equipment has to be available during the activity of a workgroup,
- If different skills are present within a workgroup, they are merged, allowing transition of workers between different tasks.

Optimizing workforce requirements: variable task

Tasks that can be moved on the time axis are called variable tasks. The start time of a variable task is changed by the algorithm if the required equipment is not available, or if the move results in a reduction of the total number of workers required that day. The variable tasks can be moved within a time interval calculated on the basis on arrival and departure data, as shown in Figure 2. The basic limitation is the physical presence of an aircraft, i.e. the time between its arrival and departure. However there are further limitations, which depend on task type and prescribed task sequence:

- ESAA (earliest start after arrival),
- LEAA (latest end after arrival),
- LEBD (latest end before departure) and
- ESBD (earliest start before departure).

In calculating the exact time window for task execution, the stricter limitations, resulting in a shorter time window, are used. Due to varying time between arrival and departure, absolute time (day:hour:minute) has to be used to calculate the time window. The equations (eq. 5-8) used are:

$$T_{ESBD} = ST\_D + ESBD \quad (5)$$

$$T_{ESAA} = ST\_A + ESAA \quad (6)$$

$$T_{LEBD} = ST\_D + LEBD \quad (7)$$

$$T_{LEAA} = ST\_A + LEAA \quad (8)$$

Where ST_D: time of departure, ST_A: time of arrival

To determine the exact time window, we compare the times defining the start of time window (TESBD and TESAA) and the times defining the end of the time window (TLEBD in TLEAA), and choose the later of the start times, and the earlier of the end times. Therefore the rules can be described as following:

IF $T_{LEBD}>T_{LEAA}$ THEN USE T_{LEAA}, ELSE USE T_{LEBD}

IF $T_{ESBD}>T_{ESAA}$ THEN USE T_{ESBD}, ELSE USE T_{ESAA}

An example of the time window and relations between the times are shown in Figure 3. Here, the time window where we can place the task is the time interval between T_{ESAA} and T_{LEBD}.

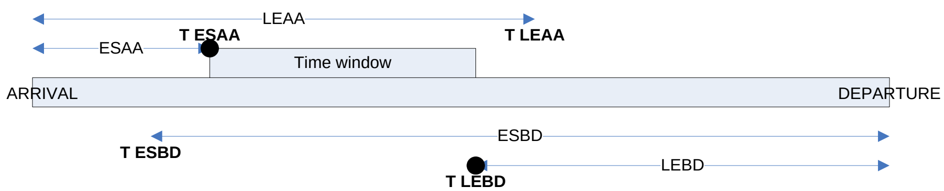

Figure 5: Example of the time window for the execution of a task

Another degree of freedom in the optimization of workforce requirements are temporary overloads. These temporary overloads allow us to handle daily traffic peaks without additional shifts and additional workers by temporary assigning more than one task to a worker, and are possible only for certain skills (worker types). An example of such a skill is the Load Balancer, which is in charge of planning the distribution of cargo and fuel on an aircraft. For a short period of time, a Load Balancer can handle several tasks, i.e. several aircrafts at once. Overloads are planned by fragmenting a workgroup into several workgroups (down to single worker), assigning tasks to them, and then finding the workgroup that has the least task load during the overload interval. In case all workgroups have equal task loads during the overload interval, the algorithm looks for the workgroup that is relieved the soonest. Every skill in the database has the maximum overload attribute defined, however most skills have the value set to 1.0, allowing no overload.

Shift construction

After all the workgroups are created, we can link the workgroups into shifts. The algorithm is workgroup oriented – it examines the available workgroups and tries to find the most suitable shift to assign them to. Shifts are constructed separately for each skill group. A shift construction starts by finding an available workgroup that starts first in the work day, and assigning the workgroup to a new shift. The workgroups in the same skill group that follow are then added to the shift, until the maximum length of shift is reached, or until there are no more available workgroups. The number of workers in a shift is equal to the number of workers in the first workgroup in the shift, and is constant. Therefore only workgroups that have the same or higher number of workers as the first workgroup in a shift are assigned to the shift under construction. As there may be a time gap between workgroups in a shift, breaks are created for every gap.

The algorithm is completed when all workgroups are assigned to shifts. If necessary, workgroups are split into smaller workgroups to fill the shifts. If the algorithm runs out of workgroups to add before the shift length reaches the minimum shift length limit, the shift is split into two parallel shifts, with smaller workgroups (the limit is one worker per shift), and the algorithm tries to find suitable smaller workgroups to extend these shifts. The algorithm adjusts the shift start time to the earliest half hour interval. If e.g. a workgroup starts its work at 6:23, the shift starts at 6:00, and a 23 minute break is created to fill the gap. The construction of shifts also follows the following criteria: The share of breaks in a shift is below a set limit. Each break's duration is below a set limit (break duration maximum), Preferred shift duration is defined. Preferred shift start and end times are defined. Shifts that respect these preferences are constructed first. The limit for the share of breaks is calculated according to this equation (eq. 5):

$$MaxBreakShare =< 1 - \frac{total\ duration\ of\ all\ workgroups\ in\ the\ shift}{current\ shift\ duration} \quad (9)$$

If a workgroup can be assigned to more than one shift, the most appropriate shift is found using a weighted criteria function (eq. 10).

$$ShiftSuitability = (c_1 * w_1) + (c_2 * w_2) + \cdots + (c_n * w_n) \quad (10)$$

Four criteria are defined in the start of the algorithm: time usage efficiency, last break duration, shift duration and task matching. The weights of the criteria can be adjusted by the operator (eq. 11).

$$ShiftSuitability_A = (c_1 * w_1) + (c_2 * w_2) + (c_3 * w_3) + (c_4 * w_4) \quad (11)$$

where:

c_1 → *time usage efficiency*

c_2 → *last break duration*

c_3 → *shift duration*

c_4 → *task matching*

The shift with currently (i.e. from the start of working day until the start of the currently considered workgroup) worst time usage efficiency is found using the following equation (eq. 12):

$$c_1 = 1 - \frac{total\ duration\ of\ all\ workgroups\ in\ the\ shift}{start\ of\ current\ workgroup - start\ of\ first\ workgroup\ in\ the\ shift} \quad (12)$$

The shift with the longest last break is found using this equation (eq. 13):

$$c_2 = \frac{start\ of\ current\ workgroup - end\ of\ last\ workgroup\ in\ the\ shift}{max.\ break\ duration} \quad (13)$$

The shortest shift is found using this equation (eq. 14):

$$c_3 = 1 - \frac{start\ of\ current\ workgroup - start\ of\ first\ workgroup\ in\ the\ shift}{max.\ shift\ duration} \quad (14)$$

Task matching is verified using the following conditions:

```
IF Skill(WG(n)) = Skill(WG(n-1)) THEN C4 = 1
IF Skill(WG(n)) ≠ Skill(WG(n-1)) THEN C4 = 0
```

Where WG(n) is the current workgroup and WG(n-1) is the previous workgroup.

If there are several shifts with the same value of the criteria function available, the shift is selected at random. In order to evaluate and improve the selected criteria and weights, the algorithm records the number of randomly selected shifts.

V. Conclusion

We can conclude from our experience in this project that scheduling problems in the air transport industry are more demanding than traditional machine scheduling problems, as the mathematical scheduling models from previous research could not be utilized, and heuristic algorithms had to be developed instead.

This research has resulted in algorithms for computation of workforce requirements for several types of workers and equipment requirements for every minute in a working day, and for construction of shifts for airport ground crew at the Ljubljana JožePučnik Airport. The basis for algorithms is heuristics. The algorithms perform the optimization of task schedule, workforce requirements schedule, and shift schedule at the airport. The output of the algorithms is a shift schedule including specified skills of workers in shifts. This serves as an input into a separate HR department application that generates the detailed work schedule, which specifies the names of workers and each shift and the equipment to be used. This application uses the skill information to find the most suitable workers according to their skills, training and personal preferences.

The developed scheduling algorithms are flexible and adaptable, and will be applicable also in case of enlargement of the airport and opening of a new terminal, as these changes will only affect the data that the algorithms use: the criteria, their priorities, and their values.

The optimization parts of the algorithms are limited by the list of tasks that allow the transition of workers between tasks and the criteria defining the workforce requirements per tasks. A more detailed task and skill categorization, and a more detailed differentiation between the requirements of different aircraft types and airlines would allow a more precise workforce requirement and task timing criteria and thus a better optimization of workforce requirements.

The algorithms are based on the heuristics and structures of the departments, specific to Ljubljana JožePučnikAirport, however a similar organization of work is used by other airports of similar size and traffic intensity, e.g. the Salzburg airport, Austria. In the implemented software solution, the structure and the number of airport departments, flight categories and the sequence and requirements of tasks are entered as data within a relational database, and not as a part of the data model or the algorithms; therefore the solution could be relatively easily adapted to the needs of other airports.

Another opportunity for the optimization of the airport logistics is integration of the solution in the Flight Information System, which would allow prediction of flight delays due to ground crew overload or unavailability. An example of an integrated scheduling and information system is described in Greiner and Volek[32].

References

[1] R. W.Conway, W. L.Maxwell, and L. W. Miller,*Theory of Scheduling*. New York: Dover Publications, Inc.,2003

[2] P. Brucker,*Scheduling Algorithms*. Berlin: Springer-Verlag, 2001.

[3] M. Pinedo,,*Scheduling Theory*, Algorithms and Systems. Upper Saddle River: Prentice Hall,2002.

[4] M. Pinedo,*Planning and Scheduling in Manufacturing and Services*. New York: Springer, 2005.

[5] M. Bester, I. Nieuwoudtand J. H. Van Vuuren,“Finding Good Nurse Duty Schedules: A Case Study”, in*Journal of Scheduling*, 10, 2007, pp. 387–405.

[6] N. Wassan, G. Nagy andS.Ahmadi, “A heuristic method for the vehicle routing problem with mixed deliveries and pickups”,*Journal of Scheduling*, 11(2), 2008, pp. 149-161.

[7] E. Burke, P. De Causmaecker, G. De Maere, J. Mulder, M. Paelinck, and G. VandenBerghe, (2009) “A Multi-Objective Approach for Robust Airline Scheduling” in *Computers and Operations Research* [Online]. Available: http://www.cs.nott.ac.uk/~ekb/Publications/klm.pdf

[8] M. D.R-Moreno, D.Borrajo, A. CestaandA. Oddi, "Integrating Planning and Scheduling Workflow Domains", in*Expert Systems with Applications*, 33(2), 2007, pp. 389-406.
[9] Air Transport Association (2008): "Costs of Delays", in *Air Transport Association*[Online], Available: http://www.airlines.org/economics/specialtopics/ATC+Delay+Cost.htm
[10] X. Qi, I. Yang, andG. Yu, "Scheduling Problems in the Airline Industry".In: Leung, J.Y-T. (ed.), *Handbook of Scheduling: Algorithms, Models, and Performance Analysis*. Boca Raton: Chapman & Hall/CRC, 2004.
[11] L. Bianco, P. Dell'Olmoand S. Giordani, "Scheduling models for air traffic control in terminal areas,"*Journal of Scheduling*, Vol. 9, No. 3, 2006, pp. 223-253.
[12] C.D. Spyropoulos, "AI planning and scheduling in the medical hospital environment,"in *Artificial Intelligence in Medicine*, Vol. 20, 2000, No. 2, pp. 101-111.
[13] A. Papler, *Interaktivnovečkriterijskorazporejanjeproizvodnje*. Master Thesis .Kranj: Univerza v Mariboru, 2001.
[14] A. J. Schaefer, E. L. Johnson, A. J. Kleywegt, G. L. Nemhauser, "Airline Crew Scheduling Under Uncertainty," in*Transportation Science*, Vol. 9, No. 3, 2005, pp. 340-348.
[15] J. Y.-T.Leung,*Handbook of Scheduling: Algorithms, Models and Performance Analysis*. Boca Raton: Chapman & Hall/CRC, 2004.
[16] D. Peidro, M.Díaz-Madroñero, J.Mula, »An interactive fuzzy multi-objective approach for operational transport planning in an automobile supply chain«, in *WSEAS Transactions On Information Science And Applications*, Issue 2, Volume 7, February 2010.
[17] K. Cafuta, I. Klep, J. Povh,"Constrained polynomial optimization problems with noncommuting variables," in *SIAM j. optim.*(Print), vol. 22, no. 2, 2012, pp. 363-383
[18] A. Cesta, G. Cortellessa, F. Pecora and R. Rasconi, R., "Intelligent Supervision for Ambient Intelligence: Customizing Scheduling Technology", in*Proceedings of the IEE Workshop on Intelligent Environments,* Colchester: University of Essex, 2005, pp. 235-242.
[19] S. Petrovic, "Towards the Benchmarks for Scheduling Problems," in*Proceedings of the Workshop "Scheduling a Scheduling Competition"* held in conjunction with the 17th International Conference on Automated Planning & Scheduling (ICAPS '07). Rhode Island: ICAPS,2007.
[20] A. Baggia, "Meta-Language Framework for Personnel Scheduling," in M. Jaškova (Ed.), *ECON'04*,Ostrawa: Technical University of Ostrava, 2004, pp. 23-30.
[21] D. Kofjač, M.Kljajić, "Application of genetic algorithms and visual simulation in a real-case production optimization," in *WSEAS transactions on systems and control*, vol. 3, no. 12, Dec. 2008, pp. 992-1001.
[22] N. Vakhania, "A Binary Search Algorithm for a Special Case of Minimizing the Lateness on a Single Machine", *International journal of applied mathematics and informatics*, Issue 3, Volume 3, 2009.
[23] F.Bian, "Measuring the Robustness of Airline Fleet Schedules," in V Kendall, G., Burke, E., Petrovic, S., Gendreau, M. (eds.), *Multidisciplinary Scheduling: Theory and applications*. New York: Springer, 2005.
[24] S.C.K. Chu, "Generating, scheduling and rostering of shift crew-duties: Applications at the Hong Kong International Airport," in*European Journal of Operational Research*, Vol. 177, No.3, 2007, pp. 1764-1778.
[25] J.Herbers (2010), "Models and Algorithms for Ground Staff Scheduling on Airports,"PhD Dissertation. Aachen: Rheinisch-WestfalischenTechnischenHochschule Aachen [Online]. Available:http://deposit.ddb.de/cgi-bin/dokserv?idn=974659916&dok_var=d1&dok_ext=pdf&filename=974659916.pdf, 2005.
[26] G. Broggio, B. Paoletti, G. Felici and C. Gentile(2009), "Solution Algorithms for Ground Staff Roster Planning (ALITALIA)", [Online],Available: http://www.agifors.org/document.go?documentId=1229&action=download, 1999.
[27] E.Hasselberg(2010), "Ground Staff Scheduling at Göteborg-Landvetter airport," in*AGIFORS Operations Control Study and Ground Resources Joint Study Group Meeting* [Online],Available: http://www.agifors.org/document.go?documentId=1332&action=download 1999.
[28] M.Bazargan,*Airline Operations and Scheduling*. Cornwall: MPG Books Ltd, 2004.
[29] A. J. Mason, D. M. Ryan, "Integrated simulation, heuristic and optimisation approaches to staff scheduling," in*Operations Research*, Vol. 46, No. 2, 1998, pp. 1-15.
[30] N. Damij, "Business process and health care simulation" in *WSEAS Trans. Syst.*, Mar. 2006, vol. 5, iss. 3, pp. 526-531.
[31] M. KljajićBorštnar, M. Kljajić, A. Škraba, D. Kofjač, V. Rajkovič, "The relevance of facilitation in group decision making supported by a simulation model," in *Syst. dyn. rev.*, vol. 27, no. 3, 2011, pp. 270-293.
[32] K. Greiner, J. Volek, "Distributed system for train route ordering" in *International journal of applied mathematics and informatics*, Issue 2, Volume 4, 2010.