Contents

List of Figures

Introduction

Punctuality is a key element in aviation: the delays are a “failure” promise for the passenger and it carry a heavy cost and difficulty to the entire sector. The development of air transport in recent decades has occurred within a complex framework characterized by profound changes in technical, managerial and organizational likely to provide an adequate response to a growing demand. These changes were originated both from the aviation industry pressures and the need to liberalize a sector strong growth. After a collapse in air traffic growth due to the global economic crisis of recent years, a constant increase in demand has been, and forecasts for the coming years are positive.

To this steady increase in air traffic demand has not been matched by an adequate growth of the capacity of the system infrastructural networks. The main effect of this phenomenon are situations of congestion that creates delay on the ground and at departure and arrival queues, and causing difficulties to passengers and huge losses to airlines.

The capacity may be increased by providing the airport with a sufficient number of runways and parking bay; it is clear that if these structures are lacking or insufficient the infrastructural capacity on the ground has negative effects on the airspace’s capacity. In the short term, the best we can get from the system is to limit the impact of delays, caused by congestion, controlling the air traffic flow in order to match the demand with the available capacity. This activity is defined: *Air Traffic Flow Management* (or ATFM).

In Europe, a continent with many nations and airspaces. the air traffic and air flow control is a complex problem. The coordination and centralization task was entrusted to the ”Central Flow Management Unit” (or CFMU), that control the air traffic flow for 36 nations of the *European Civil Aviation Conference*[1]. The activity of flight assistance in Italy is provided by ENAV Spa that in addition to providing air navigation services and the development of new technology systems, it has as main objective to increase the ATC (Air Traffic Control) capacity, obviously treating the safety as a key component of the system.

Research on air traffic management began in the the late 80s, and they have mainly focused on models of optimization for the allocation of the delay to the ground. In this thesis In this work we introduce a mathematical model to improve the aircraft departures planning system. The objective is to maximize the airport performances, minimize delays in the runway operations and to support the air controller work. The followed approach is based on the combination of a one runway two stages algorithm with a multi-runway procedure to find the better departures scheduling. By means of the two stages algorithm, a complex problem dealing with multi-objective functions is split into two inter-connected one dimensional problems. In the first stage the aim is to minimize the throughput, defined as the number of aircraft in the time unit, subject to Wake Vortex Separations constraint. An ad-hoc control heuristic method is used to mix the pre-fixed landing arrivals slots with the departure slots outgoing from the first stage. In the second stage the class sequence, generated by the first one, is computed in order to minimize the delays between the actual and estimated take-off time of each departing aircraft, subject to fixed CTOTs and ETOTs, and considering some possible departing priority. Successively a multi-runway procedure is introduced, consisting of an heuristic methodology, which uses the two stage algorithm, to locate as better as possible the aircraft on each available runway. The result is the better feasible take-off sequence in a referred time window. Some simulations on typical flight strips from Milano Malpensa airport in Italy, having two runways, are shown.

This paper is organized as follows:

- in the first chapter we introduce the main players of the airport system and

[1]ECAC is an intergovernmental organization which was established by the International Civil Aviation Organization (ICAO) and the Council of Europe. It promotes the continued development of a safe, efficient and sustainable European air transport system. In doing so, it seeks harmonies civil aviation policies and practices amongst its Member States and promote understanding on policy matters between its Member States and other parts of the world

its terminology in use.

- in the second chapter we discuss some of the works presented in the literature concerning the problem of departure scheduling. In particular, we focus on the work of Anagnostakis used as a reference for this work.
- in the third chapter our model for solving the above problem is introduced.
- in the fourth chapter, we present some results obtained through simulations.

Chapter 1

Aeroportual's Domain

1.1 ATC - Basic Concept

The *Air Traffic Control* (ATC) is a set of rules and institutions that contribute to make safe and to regulate the flow of aircraft on both the ground and in the sky. The main task of this complex system is to prevent collisions between aircraft.

Eurocontrol, the European Organization for the Safety of Air Navigation, has established that every European state must have a regulatory entity giving, said that the in force rules (the *Regulator*), and an entity that provides services related to air traffic control, said *Air Navigation Service Provider* (or ANSPs). These entities should be separated: in particular, the Italian's role of regulator is entrusted to National Civil Aviation Authority (*ENAC*), while the *ANSP*s are *ENAV SpA* and the *Italian Military Air Force*, working in close coordination with each other, each one managing the air traffic Services within the airspace under its jurisdiction. In contrast, in the United States, the *FAA* (Federal Aviation Administration), acts both as a regulator and *ANSP*.

In Italy *ENAC* deals with many aspects of the regulation of civil aviation, the control and supervision of the adopted rules the regulation of administrative-economic aspects of air transport system. In particular, it must ensure the safety, that's mean safety and security, respect and application of International laws. Safety means the safety of operations with respect to possible malfunctions (ie mechanicals). The term security, however means security on the ground, in aircraft, inside and outside of airports against unlawful acts. Furthermore, this public entity represents Italy in the major international organizations of aviation civil: *International Civil Aviation Organization* (or ICAO), *European Civil Aviation Conference* (or ECAC), *European Aviation Safety Agency* (or EASA), and it maintains constant relationships, with them having a position of leadership. Instead, the ENAV SpA is the company, partly private and partly public, which the Italian state entrusts the management and control of civil air traffic in Italy.

Anyone who wants to cross the airspace, whether it's airline or private, must submit in advance to the attention of ENAV its own flight plan that collects all the essential information (identification of the aircraft and the pilot, take-off time, airport of departure and destination, etc.).

Airspaces are a finite and precious resource that must be managed with punctuality, security and business continuity. To implement the control function, the airspace has been divided into many smaller airspaces, called *Flight Information Regions* (or FIR), which have whether territorial and altitude limits.

In Italy there are three large FIR (see Figure 1.1), Milano, Roma and Brindisi, which are provided by the *Flight Information Services* (or FIS) and ALS *Alerting Service* (or ALS). In order to control the FIR, four *Area Control Center* (or ACC) located in Milan, Padua, Rome and Brindisi. The airspace of ACCs is in turn divided into sectors, whose shape and size are consistent with the flow of traffic to manage, since a single Air Traffic Controller cannot physically handle all aircraft present in a FIR.

Figure 1.1: Italian FIR

1.2 The airport in the international context

Eurocontrol is an European organization which ensures flight safety, the capacity, efficiency, Environment and security.

This civil and military organization, with has 28 member states, has as primary objective the development of an uniform and integrated pan-European system of air traffic management, according to the basic concept of the Single European Sky (SES), developing, coordinating and planning, with its member states, the implementation of strategies for pan-European air traffic management in the short, medium and long term.

At the international level, the rules for the operation and management of airports for civil use are, however, provided by ICAO[1] (International Civil Aviation Organization), in Montreal, Canada. ICAO is an independent agency of the United Nations, which was responsible to develop the principles and techniques of international air navigation, on routes and airports, and to promote the design and development of international air transport to make it more safe.

International Air Transport Association[2] (or IATA), also in Montreal, is rather an organization of airlines that has developed, such as the ICAO, codes for the identification of the airports in the world. The IATA codes consists of three-letter,and they are published every three years on the *IATA Airline Coding Directory*. The codes are used by airlines for schedules, reservations, and baggage handling.

1.3 The phases of departure and landing in the life cycle of a flight

Flight procedures, which an aircraft must follow in order to ensure safety are defined by international control entities (eg ICAO). These phases (Figure 1.2) are illustrated in brief below, although some events are optional (eg de-icing).

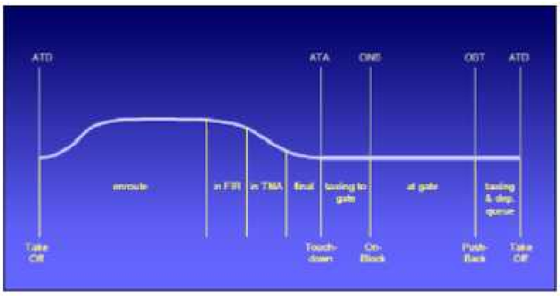

Figure 1.2: Flight Phases

The crew is on board and receives a series of information from ATC systems about the flight to be undertaken, such as the Flight Plan, the weather and so on. This phase is also called the *Control of Airport*: the aircraft is in contact with the Control Tower (TWR) authorizing it (at the time OFB[1]) to start the aircraft and move it from the parking zone to the assigned runway.

At the time *Actual Time of Departure*[2] (or ATD) the aircraft starts the stage of take-off and climb, passing to the stage of En-route, in which it maintains

[1] *Off Block Time*. The time at which the aircraft will commence movement associated with departure.

[2] The time that an aircraft takes off from the runway. (Equivalent ATOT – Actual Take Off Time)

altitude and speed indicated on its flight plan. After passing through various areas of control, phase of descent begins. Each FIR may contain several TMA (Terminal Manoeuvring Area), in which the aircraft is accompanied during the final approach from Control Tower. After landing, the aircraft crosses the assigned taxiway pointing to the gate of competence or to the parking area (Apron). Upon completion of these operations and in correspondence with *Actual of Block Time* (or OBT), the airplane is taxiing to the holding point of runway indicated on the flight plan, and then it starts taking off, taking the SID (Standard Departure Route) corresponding its route.

These flight procedures show a well-defined life cycle of the aircraft that is briefly described in the following sections.

1.3.1 Pre-flight, Taxi and Take-off

This is the stage before the take-off, during which the aircraft is stopped at the gate indicated on the Flight Plan to allow the boarding of passengers (Figure 1.3).

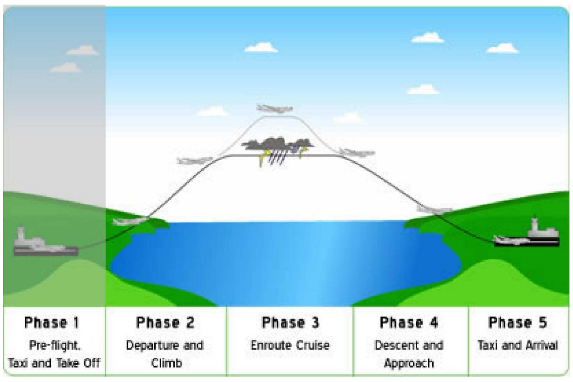

Figure 1.3: Flight Phases

The crew is on board and receives a series of information from ATC systems about the flight to be undertaken, such as the Flight Plan, the weather and so on. This phase is also called the Control of Airport: the aircraft is in contact with the Control Tower (TWR) authorizing it (at the time OFB - Off Block Time) to start the aircraft and move it from the parking zone to the assigned runway.

1.3.2 Climb

After the phase of movement on the ground, at the time OBT (Off-Block Time), the pilot is authorized, by the control tower, to take off which will take place at the time ATD (Actual Time of Departure) only when the safety distance (Wake Vortex Separation) from all other aircraft will be granted.

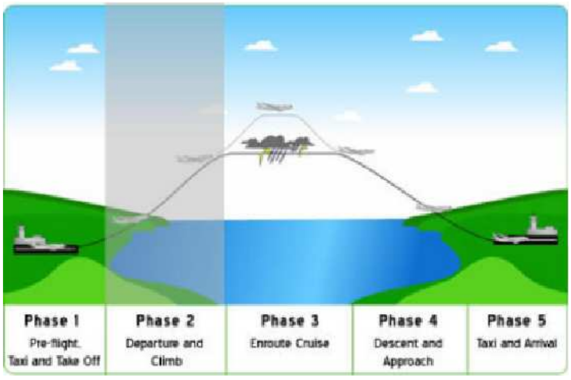

Figure 1.4: Climb

Once taken off the plane, that is always in contact with the Control Tower (TWR), it passes through the phase of Approach Control that ensures a safe route towards the assigned aerovia (air route between the airport of origin and destination).

1.3.3 En-Route

Once taken off (Figure 1.5) the aircraft goes on along the assigned airway and is taken over by the Area Control Centre (ACC) which manages the Route Control.

An altitude and the path to follow is assigned to the aircraft, so that it always respect the safety distance (called separation), both horizontally and vertically, from other aircraft.

Close the airport destination, the aircraft is in **Approach Control Phase** during which it is driven into the descent alignment with the runway.

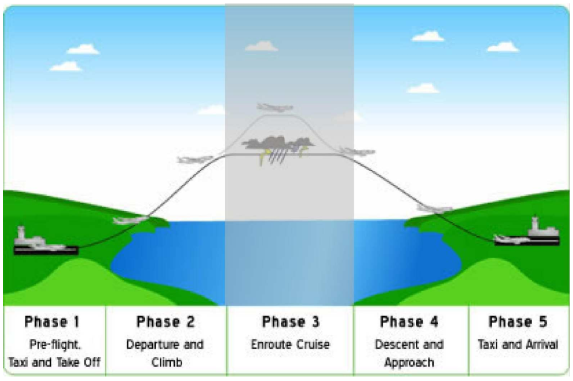

Figure 1.5: En-Route

1.3.4 Descent and approach

When the aircraft is on the path of landing (Figure 1.6) and the airport is close, it is managed by the Control Tower of the destination's airport, which guide the aircraft during landing until the parking or the Gate.

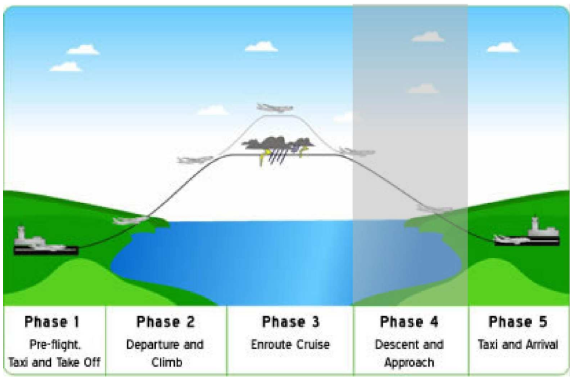

Figure 1.6: Descent and approach

1.3.5 Taxi and Arrival

The landing is completed at the time *Actual Time of Arrival*[3] (or ATA). During this phase the pilot may use, if available, instruments such as PAPI (Precision

[3] The time that an aircraft lands on a runway. (Equivalent to ALDT – Actual Landing Time).

Approach Path Indicator) and ILS (Instrumental Landing System), which guide the approach to the runway.

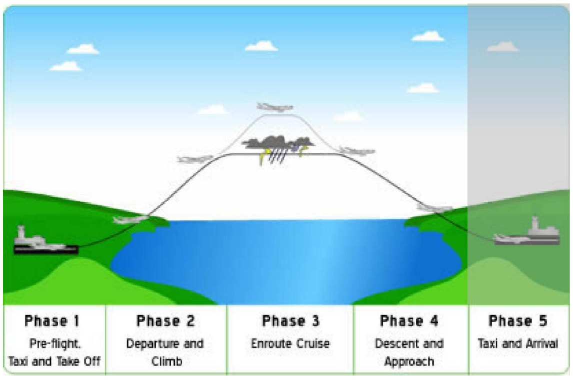

Figure 1.7: Taxi and Arrival

Chapter 2

Problem Statement and State of the Art

The airport system, for its role in public transport, must ensure that all operations (take-offs, landings, aircraft movements etc...) are completed safely. Appropriate authorities such as ENAC and ENAV, are delegates to ensure that these operations (both performed on the ground and in the air) are properly made. Such authorities have issued a series of regulations that govern the entire airport traffic (including departure and landing operations), from aircraft to means for the cargo. There are many modeling efforts and / or prototypes for scheduling such operations in order to maximize throughput, minimizing aircraft's delay and workload's controllers, etc. . The purpose of this chapter is to present the problem of departures planning and show how it has been treated in the literature by several authors.

2.1 Departure Flight Scheduling

The *Air Traffic Control* (ATC) is a set of rules and institutions that contribute to make safe and ordered the flow of aircraft both on the ground and in the sky. In Italy, the role of the Regulator, (the authority that issues the regulations) is covered by the Civil Aviation Authority, while the ANSP (Air Navigation Service Provider, a company that provides air traffic services) is ENAV SpA. The ENAV emanates data aeronautical information essential for the operation of air traffic in the form of publications **AIP Italia** [3] (*Aeronautical Information Publication*). These publications contain permanent aeronautical information concerning the national airspace, airports, organization of air traffic services and infrastructures. Examining *AIP Italia* publications one can identify a whole range of regulations that affect and limit the throughput (number of operations take off / landing in the unit of time, usually an hour), in order to ensure the necessary security to the aircraft flow.

The controllers in the control tower, which are responsible for the land traffic management, receive information on arriving aircraft a few minutes before they take off from the departing airport, so that they can prepare in advance a taxi plan from the landing runway to the gate. Compared to arrivals, planning for departing aircraft is processed directly in the *Control Tower*, and is ready in advance with respect to the departure time, so that accidents can be easily handled. For each departing flight, defined as "scheduled", a time slot assigned, consisting of a coordinated range of fifteen minutes, within which the aircraft must take-off. This slot, defined *Calculated Take Off Time* (or CTOT), is assigned by the *Control Flow Management Unit* (or CFMU), an operational unit within Eurocontrol, designated to air traffic control with headquarters in Brussels (Belgium), in order to maintain a flow of constant traffic in all European air regions, so as to avoid the occurrence of congestion. Until a few years ago, the process of planning the arriving and departing flights was exclusively managed by air traffic controllers, that in situations of heavy traffic, were assisted by other controllers.

For several years, tools to help and support the work of auditors in planning operations runway are under study and development. Their aim is to provide a possible "planning" in accordance to plane delays, queues at the holding point, destination, state of the taxiway and time needed to cover them, and other variables.

This schedule is processed trying to optimize the use of airport resources, reducing delays and minimizing waste.

In most airports, the runway is the bottleneck for an optimal planning of operations, since it is typically a shared resource between landing operations, takeoff and crossing. A possible solution to this problem is to build new runways, but this choice cannot be adopted for both the high cost that the airport should support, and due to the lack of suitable areas. In addition, the boundaries of the airports usually have highly developed urban centers, whose populations raise legitimate concerns to preserve the environment from noise and pollution due to aircraft engines, by limiting the number of arrivals and departures (hence throughput), particularly at night, resulting in the loss of revenue from airports.

The term DMAN (*Departure Manager*) means any tool that can optimize tail of take-offs, sharing the runway with landing and crossing operations. Stakeholders and various industrial partners gave the following definition of DMAN:

"*DMAN is a planning tool that on the basis of the constraints and preferences, aims to improve the flow of departing from an airport, calculating the Target Take Off Time (TTOT) and the Target Start-up Approval Time (TSAT) for each flight*".

The use of this tool in the airport context should give the following benefits:

- improved on-time departures;
- better use of runway capacity;
- better information service to passengers (about delays and quantification of the delay);
- reduction of the workload of the air traffic controller;
- reduction of emissions, due to the queues decreasing at the holding point;
- economic benefits.

The aim of this thesis is to define a mathematical model for the management of departures and implement a prototype of DMAN system.

2.2 Aeroportual Safety Rules and Goals

The management of airport traffic is, in general, subject to financial and operative interests of the various involved stakeholders, such as users of the airport (passengers, airlines, etc..) and providers of ATM services (airport authorities, air traffic controllers, etc..). In addition, there are legitimate concerns in order to preserve the surrounding environment from airport noise and pollution from engines on the aircraft. The objectives and interests can be mutually supportive, and rivals in some cases. A certain level of security should be guaranteed, and the workload of the *Air Traffic Controllers* must be sustainable.

In summary, the main objectives of an airport system are:

- **throughput (or capacity)**: optimize the use of shared resource (runways, taxiways, gates, etc.), minimizing the taxy and pushback time, and the delays in authorization (clearance) and pushback, implies an increase of the throughput;
- **aircraft delay**: minimizing any delays that may occur during the various stages prior to departure (boarding, refueling, cargo operation, etc.);
- **fairness**: treat airlines with equity;
- **workload**: minimize the effort of controllers in managing the runway's operation;
- **environmental pollution**: keep aircraft engines switched on as little as possible.

These objectives are subject to constraints in order to ensure that the runway's operations are completed in safety. These constraints are divided in two categories:

- **"universally recognized "**: common in all airport systems;
- **"local "**: local to the particular airport (eg state of the runs at night);

"Universally recognized" constraints, as resulting from regulations issued by local traffic control plan, are the following:

- Minimum separation between aircraft on the runway (Wake Vortex Separation);
- Departure routes (horizontal and vertical separation in air);
- Speed group;
- Calculated Time of Take-off (CTOT).

2.3 Minimum separation between aircraft

Aircraft taking off and landing on the back generate a wake vortex (Figure 2.1), originated from the wingtips, which although of limited duration, can interfere the operations of any aircraft that follows.

Figure 2.1: Vortex behind the aircraft

The *International Civil Aviation Organization* (or ICAO), based on the MTOM (Maximum Take-Off Mass), has divided aircraft into three categories[4] (Wake Vortex Category):

- **Light (L)**: MTOM of 7,000 kg or less;
- **Medium (M)**: MTOM exceeding 7,000 kg but less than 136,000 kg;
- **Heavy (H)**: 136,000 kg or more.

An aircraft of category **Light** (or L) can take-off only two minutes after the take-off of an aircraft of category **Heavy** (or H). Similarly are defined time separations between an operation of landing followed by a take-off and vice versa. In Italy, ENAV SpA give a different definition of *Wake Vortex Category* [5], grouping the aircraft into the following four categories, three of which according to MTOM, and the category **Super** (or J) depending on the type of aircraft:

- **SUPER (J)**: aircraft type A388;

- **HEAVY (H)**: MTOM exceeding 136,000 kg;
- **MEDIUM (M)**: MTOM exceeding 7,000 kg but less than 136,000 kg;
- **LIGHT (L)**: MTOM of 7,000kg or less.

The separation times for these weight categories are described in the AIP Italy of Enav SpA.

2.4 Departures routes

The Departures Routes are also known as *Standard Instrument Departure* (or SID). These are procedures the aircraft must follow immediately after take off, as long as it remains in the vicinity of the airport. In particular, these procedures have been designed to allow the aircraft to leave the airport without obstacles (artificial or natural) that can be found closely, and to reach a *Terminal Control Area* (or TMA).

If a same SID is assigned to more aircraft a separation among the aircraft must be imposed, so as to ensure that the security distance is maintained even in the phase of En-route (Figure 1.5). In particular, the minimum horizontal separation of vehicles, checked with radar equipment, is five miles, otherwise the procedural separation is applied, whose minimum extension is normally not less than twenty miles (approximately thirty-seven km).

2.5 Calculated Take-off Time

In order to ensure the efficient use of airspace, Eurocontrol has set up a task force whose role is to manage air traffic in Europe, so to avoid congestion. This operation of supervision is fulfilled assigning each aircraft, which have to take-off, one time slot *Calculated Time Of Take-off* (or CTOT), within which it is authorized to take-off. It is important that the aircraft take offs within this time window, and in case the estimated time of departure is not within this range a new CTOT should be asked. In Europe, this window is defined by the five minutes before $CTOT$ and ten minutes after.

2.6 State of the Art

The scheduling of departures planning for over a decade has become an important research topic in the context of *Air Traffic Management* (or ATM). Several researchers have focused their interest to these problems, as well as the most influential European authority in this field, Eurocontrol, that has invested significant resource in order to promote several studies on scheduling departures.

The *German Aerospace Research Centre* [6] (or DLR) has developed a tactical concept of departures management and functional prototypes of DMAN (Departure MANagement) system. NLR [7], the famous Dutch research institute specialized in avionics and aerospace, introduce an approach based on the technique of **Constraint Satisfaction Programming** (or CSP) to solve this problem.

In 2003, Eurocontrol commissioned DLR to develop a prototype of DMAN, configurable for a given set of airports and used as a demonstrator stand-alone or as an operational tool in a real simulated ATC scenario. But it should be noted that despite the different formulations and solutions, fundamental issues regarding the efficiency of a certain approach, the concept of use and the benefits that can be derived, are not fully understood. In addition, due to the different types of airports, a more general approach and a formulation of the problem to be quite general.

There are several scientific papers in the literature, but in this context of analysis, will focus our attention on some of these to show how the problem departure management has been addressed with different formulations. Anagnostatikis ([8], [9] and [10]), in departures planning considers also the eventuality of having to reschedule the sequences in relation to landing operations and crossing runway. In particular [8] and [10], the author presents a modeling based on the principles of linear programming, in which a multi-objective function is used in order to obtain a schedule of runway operations (methodology of single stage) . In [2] the same author deals the same problem dividing it into two subproblems, associated respectively with only two objectives, namely the throughput and the delay of the aircraft (dual-phase methodology). Atkin, Jason et al. in [11] and [12] present a mathematical model for the planning of departures based on the queues that are formed at the holding point. This approach has fallen on the specific Heathrow Airport, making it difficult to apply to other contexts airport. Instead, in [**?**], [13],

[14], [15] and [16] the problem of planning the departure is formulated with the innovative technique of *Constraint Satisfaction Programming.*

2.7 Single Stage Methodology

The problem of departures and arrivals planning is first formulated as an optimization problem based only on the starting sequence. Therefore define:

- D the set of n_D departures for which the optimal sequence s^* will be found;
- A the set of n_A landing aircraft;
- C the set of all known strong constraints (Wake Vortex Separation, SID, Priority, etc.) .

The optimization problem is expressed us:

$$s^* = \arg \min_{s \in S(D,A,C)} Q(s) \tag{2.1}$$

where the function $Q(s)$ measures the quality of sequence s and $S(D, A, C)$ is the set of all feasible sequences of arrivals mixed at departures.

The max cardinality of sequences in $S(D, A, C)$ is:

$$n_s \leq \frac{(n_A + n_D)!}{n_A!} \tag{2.2}$$

where n_s is the cardinality of the set S.

The relation (2.2) provides information about the space search that must be examined in order to find an admissible solution (or sequence). But as you can see, if you only need to plan take-off, maintaining constant the order of arrivals, the space search is still too large, and the search times are still computationally high. In addition, a method for the construction of the measurement function $Q(s)$, which should evaluate the adequacy of each possible sequence respecting the objectives and constraints of the planning, it is not yet known. Then according to these considerations it is evident that an approach in the resolution of such problem, based exclusively on sequences of aircraft without considering the variable "time", presents the disadvantage of being expensive in terms of resources and, moreover, remains unclear the methodology of calculation of landing times and take-off of such sequences.

In front of such difficulties and, noting that both objectives and constraints are based on the time variable, a most appropriate approach for these planning problems is to consider the time. For example, it is sufficient to map each taking-off operation exactly at a time, which could be the take-off time or the initial occupation time (we can imagine a sequence as an array of take-off time, each of which represents an occupation time), assuming that the width of these intervals is known.

So, defined a vector t consisting of all take-off time of aircraft contained in the set D, the time optimums vector is given by:

$$t^* = \arg \min_{t \in T(C)} Q(t) \tag{2.3}$$

where Q(t) is an evaluation function that reflects the goals of planning and $T(C)$ is the n-dimensional space of solutions subject to strong constraints. But even this approach involves complications. In fact is noted that the set of constraints C contains both the demands of minimum separation between take-offs and information concerning arrivals, and therefore this implies that the space $T(C)$ is a non-convex set and a disjoint set. To better understand what is stated we consider for example the case in which the constraints are:

- minimum separation between two take-off operations t_1 and t_2 of 120 sec., denoted by $c_1 : |t_1 - t_2| \leq 120$;
- the occupation interval of the runway for an arrival is between 120 and 165 sec., denoted by $c_2 : t_{A1} = \langle 120, 165 \rangle$;
- departure marked "1" is supposed that, for some reason, can not reach the runway earlier than 100 sec..

The disjoint, non-convex and two-dimensional space $T(C)$ is shown in Figure 2.2.

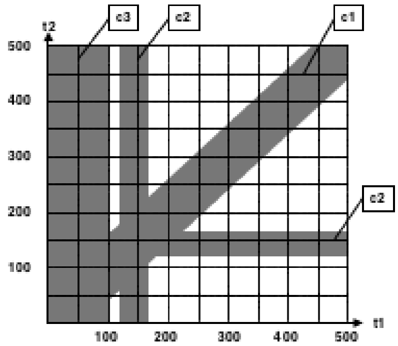

Figure 2.2: An example of a two-dimensional, non-convex, disjoint search space T(C) (white area)

It should be noted that, in addition to the difficulties related to the properties of $T(C)$, it could happen that the function $Q(T)$ is multi-modal, with several local minima.

It follows that the optimization problem is too complex and computation time too high. From what it has been seen, both the approach based on the sequences and on the time are not adequate for a planning problem, in fact the former is insufficient, the second is too complex.

A "mixed" formulation may provide, under certain conditions, a basis for resolving this problem. Then, for each sequence S an optimization based on the time could be considered; in this case the solution space is not defined only by the string constraints, but also by a series of inequalities each of which governs a pair of take off times:

$$U(s) = \{t_i \leq t_j \leftarrow ip_sj \qquad \forall i, j \in s\} \tag{2.4}$$

which indicates that the take-off time of aircraft i is lower than of the aircraft j if i precedes j in the considered sequence.

Therefore given the evaluation function $Q(t)$, the vector of optimum take-off times t^* for a specific sequence s is expressed by:

$$t^* = t^*(s) = \arg \min_{t \in T(C,U(s))} Q(t) \tag{2.5}$$

and the optimal sequence s^* is achieved minimizing the value of $Q(t)$ among all feasible sequences in the set $S(D,C)$:

$$s^* = \arg \min_{s \in S(D,C)} Q(t^*(s)) \tag{2.6}$$

The global optimal vector of take off time is then given by:

$$t^{**} = t^*(s^*) = \arg \min_{s \in S(D,C)} \min_{t \in T(C,U(s))} Q(t)$$

2.7.1 Multi-objective modeling

The goal of the system are transformed into objective functions indicated with $q_i(t)$. The latter must comply with the following properties:

$$q_i(t) \geq 0 \quad q_i(t) = \sum_{j=1}^{n} q_{ij}(t_j) \tag{2.7}$$

Furthermore each of these $q(t)$ is defined such that lower is its value and more the solution satisfies a certain goal or less is violated a certain weak constraint. Although you can define different objective functions for each of the goals of the system we will evaluate only the most influential: the throughput, defined as the ratio between the number of departures nor the time required to perform these operations and is related to the time of take-off of the last aircraft in optimal sequence S, impartiality (fairness) and the taxi times. As regards the capacity, it is natural, from the given definition, write it as $q(t) = \max(t)$. A more appropriate definition of this objective function could be the following:

$$q(t) = \sum_{j=1}^{n} b_j \tau_j^p \quad , \quad p > 1 \tag{2.8}$$

where $\tau_j = t_j - t_0$ is the time relative to the aircraft j, with t_0 the current time, b_j is a vector of weights and p is an appropriate parameter. As you can see, in the most general case, the objective function is not linear because of the presence of p.

A mathematical formulation of the objective function that expresses taxiing time based delays encountered at the end of the process of taxiing, is as follows:

$$q(t) = \sum_{j=1}^{n} \delta_{T,i}(t_j)^p \quad , \quad p > 1 \tag{2.9}$$

where $\delta_{T,i}(t_j) = t_j - t_{RTO,j}$ is the delay due to the delay of aircraft j and $t_{RTO,j}$ is initial time of take-off for the same aircraft.

With regard to fairness, that ensure equal treatment of passengers and airlines, can be formulated in terms of delay for the aircraft j, between time ready for pushback $t_{RPB,j}$, and the time *planned to pushback* $t_{PPB,j}$. In particular, since this delay does not affect the timing of taxiway, the objective function is given by a relative progressive evaluation of such delays, written as follows:

$$q(t) = \sum_{j=1}^{n} \delta_{P,B,j}(t_j)^p \quad , \quad p > 0 \tag{2.10}$$

where $\delta_{P,B,j}(t_j)$ means the delay of the aircraft pushback j.

A remark should be made on the weak constraints, defined as a function of time and which translate the constraints of flow to earth, whose violation in order to generate a solution is allowed (eg. CFMU). Such constraints, can be defined with appropriate functions that measure the degree of their violation. Therefore, an evaluation function of violations of constraints can be written as:

$$q(t) = \sum_{j=1}^{n} c(\tau_j)\delta_j(t_j, t_{c,j})^p \qquad p > 0 \tag{2.11}$$

where $t_{c,j}$ is the estimated time of departure (is $+\infty$ when a departure j is not subject to any constraint), and

$$\delta_j(t_j, t_{c,j}) = \begin{cases} 0 & t_j \leq t_{c,j} \\ \\ t_j - t_{c,j} & t_j > t_{c,j} \end{cases} \tag{2.12}$$

is the amplitude of the constraint violation.

2.7.2 Planning: solution

The objective functions may be useful for both evaluation of the goals and constraint violation. It can happen that some of these can contradict each other and therefore sometimes minimize one could result in the growth of the value of the other. In order to have different objective functions, the scheduling problem formulated as a vector optimization problem with the right compromise between these.

A general model for the planning function $Q(t)$ is represented by a linear combination of the individual objective functions $q_i(t)$:

$$Q(t) = a^T q(t) \tag{2.13}$$

where

$$q(t) = [q_1(t), ..., q_r(t)]^T \tag{2.14}$$

is a vector of r objectives functions and

$$a^T = [a_1, ..., a_r] \tag{2.15}$$

is the corresponding weight vector with:

$$a_i \geq 0 \quad \forall i \in N_r = \{1, 2, ..., r\}, \qquad \sum_{i=1}^{r} a_i = 1 \tag{2.16}$$

Therefore, the optimal solution becomes a function of the weight vector **a**:

$$t^*(a) = \arg \min_{t \in T(C)} Q(a, t) \tag{2.17}$$

$$s^*(a) = \arg \min_{s \in S(D,C)} Q(a, t^*(s)) \tag{2.18}$$

The given formulation of the problem of takeoff management refers to a "static" airport model, in which all the essential information are always available when required. However, such a problem is fallen in a real airport environment that is dynamic by nature, and where the information is not always available when required.

Therefore, the uncertainty created in these situations is reflected in formulated model negating the performance and operation.

To find the optimal solution s^*, the use of a search tree, and the technique of Branch & Bound appear to be the most appropriate, since only a subset of the possible sequences is examined.

2.8 Double Stage Methodology

The dual stage methodology divide the above problem into two sub-problems, said phases (or stages):

	S	L	H
S	60	60	60
L	60	60	60
H	120	120	90

Table 2.1: Example of Wake Vortex Separation matrix

- first stage: the goal is to maximize runway capacity (throughput), without violating the wake vortex separation and delay due to operations of runways crossing;
- second stage: the goal is to minimize the delay without violating the constraint imposed by CTOT (Europe) or EDCT[1](America).

2.8.1 First stage

The purpose of the first stage is to optimize the throughput generating sequences of take off, and considering as main time constraints wake vortex separations. Also are considered "mixed" runway operations, in the sense that in addition to departing aircraft, we take into account also arriving aircraft and "crossing" operations.

2.8.2 Constraints

The generated take off sequences in the first stage must comply the *Wake Vortex Separation* constraints, ie a minimum separation time between an operation and the next must be guaranteed. The set of the time intervals can be represented as a matrix (Table 3.1), whose rows correspond to the possible categories of weight of an aircraft that precedes another while columns express weight category. The value resulting from the intersection between row and column corresponds to the minimum separation time that must be guaranteed between the two operations.

Two additional constraints that influence further throughput sequencing are:

- the maximum number of landed aircraft that are waiting for a clearance to cross an active runway (where they are needed);

[1]EDCT is the runway release time ("Wheels Off") assigned to an aircraft in a GDP

- the maximum delay that an aircraft landed and waiting to cross an active runway can absorb.

The limit values of both constraints are input data to the planning system, and can be modified in real time by the planner.

2.8.3 Objective Function

Maximize capacity means minimizing the time in which the last operation are authorized take-off. Therefore, the formulation of the objective function is the following: let N_A and N_D respectively the total number of arrivals and departures, and $N = N_A + N_D$ the total number of mixed operations on the runway(s) during the current scheduling window (scheduling is performed dividing the time into smaller intervals of equal amplitude and considering operations that fall within that range).

Minimization of the time of the last takeoff is given by:

$$\min \max t_{D_i} \qquad 1 \leq i \leq N_D \tag{2.19}$$

2.8.4 Sequences's Classes

It is assumed that the schedule of arrivals is known a priori. Once the aircraft landing and decelerate, is sought authorization for the operation of "crossing" (if necessary) in the event that needs to cross an active runway, and the instant of time at which start the operation. These time instants are estimated on the basis of the weight of the arrival aircraft and taxiing constraints. Therefore maximizing requires careful planning departures and runway crossings (if they are required).

2.8.5 First Stage Output

The output of the first phase is a matrix containing all possible sequences of aircraft, ranked, with the property of maximizing the throughput of the runway.

The above matrix is namely "matrix CS" (matrix of *Classes of Sequences*).

$$CS = \begin{bmatrix} \text{class sequence 1} \\ \text{class sequence 2} \\ \dots\dots\dots\dots\dots \\ \text{class sequence m} \end{bmatrix} \tag{2.20}$$

The first class sequence in the matrix ensures the highest throughput and, for this reason, it is identified as **Target Class Sequence** (or TCS).

2.8.6 Second Stage

The second stage optimizes the delay of each aircraft. In the previous stage a matrix of *class sequences* has been generated and it ensures, in addition the best throughput, that the right separation time (according to the Wake Vortex Separation) are fulfilled. The first class Sequence of the matrix becomes the *Target Class Sequence* (or TCS), and it is used as a basis to generate the schedule. In fact, in this stage, the aircraft are assigned to each slot, trying to minimize the delay of take-off. If the selected TCS is not feasible solution (since the constraint of the second stage are not satisfied, the new TCS becomes the next row in the matrix. The decision variable X_{ij} used for the formulation of this second stage are defined by:

$$\begin{cases} X_{ij} = 1, & \text{if aircraft } i \text{ occupies slot } j \\ X_{ij} = 0, & \text{otherwise} \end{cases}$$

The basic requirement is that no aircraft is assigned to a slot that can not physically occupy, ie time slots earlier the time in which the aircraft can reach the runway. For example, if the air plane takes a time equal to 900 (seconds) to reach the runway and the time in the middle point of the fist two slots in the TCS is less than 900 (seconds), then the aircraft can not occupy the slots 1 and 2 in the final solution. In this case, this type of constraint is expressed by:

$$X_{ij} = 0, \quad \text{for slot } j = 1, 2$$

A further constraint is given by the same sequence of slots classes, ie if an aircraft belongs to the *Large* (or L) wake vortex class, then it can only occupy the slots of type **L** within the TCS. This constraint is translated as:

$$\sum_j X_{ij} = 1, \quad \forall \ \text{slot } j \in L$$

where L is the set of slots of type Large within the TCS. In addition, each aircraft must occupy only a single slot:

$$\sum_{j=1}^{N_S} X_{ij} = 1, \quad \forall \ \text{aircraft } i$$

where N_S is the total number of slots in the class sequence. Moreover each aircraft may be assigned at most one slot:

$$\sum_{i=1}^{N_D} X_{ij} = 1, \quad \forall \ \text{slot } j$$

where N_D is the numbers of aircraft planned for take-off.

The operational constraints, such as the estimated time for the authorization of a departure EDCT (Expected Departure Clarence Time) or DSP (Departure Sequencing Program), restrict the time available for the plane take-off:

$tEDCT_{i1} \leq t_{D_i} \leq tEDCT_{i2} \quad \text{or} \quad tDSP_{i1} \leq t_{D_i} \leq tDSP_{i2}$

where $tEDCT_{i1}$, $tEDCT_{i2}$, $tDSP_{i1}$ and $tDSP_{i2}$ are the times (as defined by ATC) that determine the time window of the EDCT (15 minutes) and the DSP (3 minutes) for the i-th plane.

A heuristic method may be used to transform the time window in a take-off slots. Let us consider the take-off position of each plane as function of the variable decision $\sum_{j=1}^{N_s} j * X_{ij}$ then the previous constraints are formulated as an acceptable range of slots:

$sEDCT_{i1} \leq \sum_{j=1}^{N_s} j * X_{ij} \leq sEDCT_{i2} \quad \text{or} \quad sDSP_{i1} \leq \sum_{j=1}^{N_s} j * X_{ij} \leq tDSP_{i2}$

where X, Y, Z and J are the final values of the take-off slot (defined by ATC) for the i-th flight, which define the windows of the slot of take-off EDCT or DSP.

Other types of constraints (such as in airplanes for rescuing) are modeled in the form of upper limit on the position of the take-off sequence, ie:

$$\sum_{j=1}^{N_s} j * X_{ij} \leq X_{\max TO_i}$$

or in terms of inequality constraints between different flights:

$$\sum_{j=1}^{N_s} j * X_{ij} \leq \sum_{j=1}^{N_s} j * X_{kj}$$

The operational constraints "Miles In Trail" (MIT) and "Minutes In Trail" (Mint) require separations between aircraft en route, and are defined in terms of separation of time at the point of take-off:

$|t_{D_i} - t_{D_j}| \geq \Delta T_{ij}$

where ΔT_{ij} is the minimum separation time of the take-off point between the airplane i and j that have a restriction "In Trail".

The constraints *MIT* (or MinT) can be defined as a function of the minimum separation required in the sequence of take-off ΔX_{ik} between the flights i and k and the spatial position of the slots of the class sequence:

$$\left| \sum_{j=1}^{N_s} j * X_{ij} \cdot \sum_{j=1}^{N_s} j * X_{kj} \right| \geq \Delta X_{jk} \Rightarrow \begin{cases} \sum_{j=1}^{N_s} j * (X_{ij} \cdot X_{kj}) \geq \Delta X_{ik} \\ \sum_{j=1}^{N_s} j * (X_{ij} + X_{kj}) \geq \Delta X_{ik} \end{cases}$$

An huge task for air traffic controllers is to ensure the "fairness" between airport users; for this purpose a constraint called "impartiality" is introduced. It is defined as the maximum displacement of the take-off position (MPS – Maximum take-off Position Shifting) misleading the adoption of a policy of "First Come First Serve". The value of MPS can be predetermined by ATC or by airlines and gives the acceptable positions in the take off sequence for each departure. For each aircraft i, indicated with X_{PBi} and X_{TOi}, respectively the position in the sequence of pushback and the position in the sequence of take-off, one has the following relationship:

$$|X_{PBi} \cdot X_{TOi}| \leq MPS \Rightarrow \begin{cases} -\sum_{j=1}^{N_s} j * X_{ij} \leq MPS - X_{PBi} \\ -\sum_{j=1}^{N_s} j * X_{ij} \leq MPS + X_{PBi} \end{cases}$$

where the values MPS and X_{PBi} are known a priori.

2.8.7 Objective Function

The objective of this second phase is to minimize the delay for each aircraft, comparing the first time available in the class sequence with the estimated time of departure issued by the airline to which it belongs.

In the general case of use of a runway, TO_{ni} the original times of arrival (when the aircraft touches the ground), TX_i runway crossing times of these arrivals, and $TOff_j$ times 'objective' departure (take off authorizations), as the values of the midpoint of the slot class, are defined.

Similarly for each aircraft delays (intended as time differences) between the actual landing, the crossing, time of take-off and the corresponding values closest possible (the temporal midpoint of each corresponding slot), respectively defined as EO_{ni} , EX_i and $EXOff_i$. The delay value for each operation indicates how it deviates from the nearest time to perform this operation. The total delay is formulated as:

$$(\sum_{i=1}^{N_D} |TOff_i(x_i) - EOff_i|^{k_D} + \sum_{i=1}^{N_A} |TOn_j - EOn_j|^{k_A} + \sum_{i=1}^{N_A} |TX_m - EX_m|^{k_X}) \quad (2.21)$$

where $1 \leq i \leq N_D$ and $1 \leq j,\ m \leq N_A$ with N_D the total number of departures and N_A the total number of arrivals; x_i is the position of the aircraft i in the slot and k_A, k_D and k_X are parameters used to penalize delays of specific flight with k_A, k_D, $k_X \geq 1$.

Therefore we aim to minimize the objective function in (2.21). In particular, if we aim to minimize only the delays associated with departures we obtain:

$$\min \sum_{i=1}^{N_D} |TOff_i(x_i) - EOff_i|^{k_D} \quad \text{dove} \quad 1 \leq i \leq N_D \qquad (2.22)$$

The output of the second stage is a matrix AS (Aircraft Schedule) in which each row represents a scheduling of the aircraft. The matrix is sorted for delay. So the first line indicates the best schedule:

$$AS = [aircraft\ \ schedule\ \ 1, aircraft\ \ schedule\ \ 2, ..., aircraft\ \ schedule\ \ N]$$

2.8.8 Approach based on CSP

A number of problems in Artificial Intelligence are classified as *Constraint Satisfaction Problem* (or CSP) problems. The classical combinatorial, resource allocation, planning and time reasoning problems, which are exactly solved by techniques of CSP.

Formally a problem of CSP is defined by a set of variables, indicated by $X_1, ..., X_n$, and a set of constraints $C_1, ..., C_n$.

Each variable X_i belongs to a non-empty domain of possible values. Instead a constraint $C_i = C(X_{i1}, ..., X_{ik})$, between k variables, is a subset of the cartesian product $D_{i1} \times D_{i2} \times .... \times D_{ik}$ that specifies the values of compatible variables.

A statement of the problem is given by an assignment of values to some or all of the variables, say X, which is called 'consistent' if any of the constraints are not violated.

When the assignment involves all the variables is called "full": therefore, consequently, we defines "solution of the CSP" a complete assignment that satisfies all constraints.

When a goal is added to a CSP, the problem obtained is defined *Objective Constraint Programming* (or COP), in which the aim is to find an optimal solution according to a preassigned evaluation criterion.

Therefore a resolution algorithm for any CSP problem can be used in an equivalent manner to solve a COP simply adding a further variable that represents the objective function. Each time a solution is found, the solver imposes the constraint that any further solution found has the value of the better objective function, and if this procedure leads to a failure then the current solution is the optimal one. The technique of resolution of the CSP makes use of a decision tree that is obtained by matching at every level the assignment of a variable, according to a predetermined order, and at each node the choice of the value to be assigned to the variable corresponding to the level where the node is located. It is therefore evident that each leaf of the tree represents a different combination of values assigned to the variables. The leaves associated with assignments compatible with all the constraints of the problem are solutions of the problem. The search for a solution is therefore equivalent to the exploration of the decision tree associated with the problem, in order to find a leaf with an assignment compatible with the imposed constraints.

CSP formulation of the problem of scheduling of take-offs

For the formulation of the scheduling problem in terms of CSP first identify the variables, their domains and constraints in the domain context. The basic object considered in the model is the flight. An aircraft, its crew and its flight plan that contains general information such as departure airport, destination, etc. are associated to each flight.

The set of flights for planning is $\{F_1, F_2, ...F_N\}$. For every flight F_j the following date are known:

- gate and parking bay;
- destination point, ie the exit point (TMA – Terminal Manoeuvring area);
- CTOT (Calculated Take-Off Time). There exist two typologies of flight, ie scheduled and not. The first need to take-off within the time given by the CFMU, the latter are not the subject of study;
- flight plan;
- the technical characteristics of the aircraft.

The output for every flight F_j we want to calculate:

- the take-off time (TTOT – Target Take-Off Time), which is the point in time at which the aircraft should start the race on the runway;
- a sequence number, which indicates, for a specific track, the order of departure of the aircraft;
- a route SID that allows the aircraft to reach the exit point.

The set of constraints specifies restrictions for a single flight or between two of them. In fact, precisely, the constraints are modeled through the relationship between the characteristics of a single flight and its time position or by relationships between two or more flights. Hard Constraint (cannot be violated) as the Wake Vortex Separation, the Departure Routes (SID) and the Speed Group are obliged to be satisfied in order to avoid dangerous situations, but instead the weak constraints (can be relaxes), such as the pilot's flight plan and priorities due to the long delay in aircraft may be violated, even if it is not, usually, the preferred solution.

As an example of a formulation of constraint through the technique CSP consider the case in which an aircraft class *Light* should be scheduled at least three minutes after the previous (because of the wake vortex). We can say that the aircraft associated with the flight F_i is heavier than the one associated with F_j and the only situation in which the F_j could start before F_i is keeping them on separate runways. The corresponding constraint, here indicated with C_k, is written as a set of four conditions to be avoided:

$$C_k = \begin{cases} \forall F_i, \forall F_j \quad \text{con} \quad F_i \neq F_j \\ NOT(R(F_i) = R(F_j)) \\ OR \;\; (t_{takeoff}(F_i) > t_{takeoff}(F_j)) \\ OR \;\; (w(F_i) \leq w(F_j)) \\ OR \;\; (t_{takeoff}(F_i) + 3 \leq t_{takeoff}(F_j)) \end{cases}$$

where F_i and F_j are two flights to be scheduled and $R, w, t_{takeoff}$ are functions that return respectively the runway allocated, the weight class of aircraft and the take-off time. Considering the routes SID as a set of "nodal" points, the corresponding constraint, understood as separation between two aircraft (in order to prevent overload situations on these routes of "fitting" as well as for safety reasons) can be written as following:

$$C_s = \begin{cases} \forall P_i, \forall P_j \quad \text{con} \quad P_i \neq P_j \\ NOT \;\; (sameSID(P_i, P_j)) \\ OR \;\; NOT \;\; (follows(P_i, P_j)) \\ OR \;\; (t_{over}(P_i) + flying\ time(P_i, P_j) = t_{over}(P_j)) \end{cases}$$

Here P_i and P_j are the points related to the two aircraft, same SID and follows are two functions that control, respectively, if the two points belong to the same SID, and if the two points follow each other, t_{over} is a function that returns the instant of time in which a flight crosses a "nodal" point.

The *National Aerospace Laboratory* (or NLR) has developed a tool to generate the sequences of the departures, known with the name of *Mantea Departure Sequencer* (or MADS) [17]. MADS ensures that each aircraft take offs in its CFMU slot, respecting the separation and not overloading the sectors of airspace.

MADS was made by examining five categories of constraints:

- separation constraints (Wake Vortex Separation);
- constraint on the use of the runway. For example in airport with more runways, for each of them are indicated: availability, size, surface conditions, equipment and weather conditions.

- constraints of *Terminal Manoeuvring Area*[2] (or TMA) and En-Route. It is necessary to ensure the constraints of separations in air (such as the separation SID).
- time constraints: we must ensure that each aircraft take offs on its time slot (CFMU slot).

The technique used to develop this tool is already consolidated and is known in Operations Research as Constraint satisfaction Programming (CSP). This technique allows you to describe the problem using a set of variables, each of which is associated with a set of possible values. Constraints define the combinations of values of variables that are not allowed (for example two aircraft cannot take-off at the same time and on the same runway). A solution consists of a sequence of values associated with the variables, such that all constraints are satisfied. This solution, if it exists, is determined starting from an initial search space, with the technique of backtracking: the idea is to restrict as much as possible the domains of all the variables through the propagation of constraints. At the end of this operation the following situations can occur:

- a solution was found;
- one or more variables have no values: there are some constraints violated;
- the search space may still contain solutions, therefore the algorithm continues in the search for a solution (propagation of constraints).

This type of research involves a solution or a failure. Sometimes while using the CSP search space is too much reduced, it may happen that the number of valid sequences is still huge, and therefore it becomes necessary to create heuristics to speed up the search process. It is also important to initialize the algorithm, or choose the variables right place to start the search. In the literature there are several strategies, including that to independent domains, very fast, and that in domains dependent that in the case of the problem of departures requires a special heuristics to obtain a fast convergence to the optimal solution.

[2]Terminal manoeuvring area in Europe, is an aviation term to describe a designated area of controlled airspace surrounding a major airport where there is a high volume of traffic

2.9 Approach based on queueing theory

The following approach, proposed in [11] and [12] has been designed for Heathrow Airport in London, composed by two runways, which is among the busiest in Europe and is situated in a very small portion of land. It aims to increase the runway capacity according to a set of constraints and ensuring airport security.

2.9.1 Application context

A schematization of the two runways of Heathrow is shown in Figure 2.3:

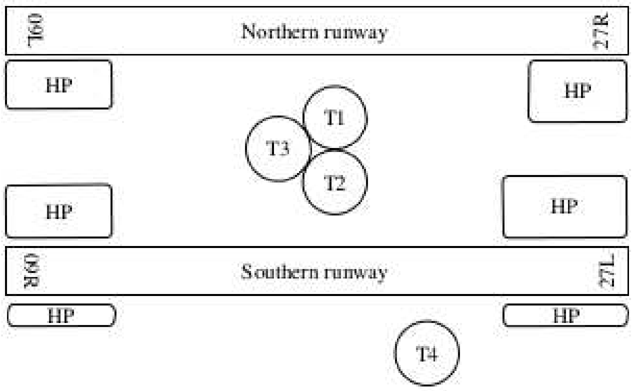

Figure 2.3: The Layout of London Heathrow Airport

In this figure we highlight four "terminal" of which the first three, indicated by $T1$, $T2$, $T3$, are located between the two runways, while the fourth, $T4$ is south of the track labeled 27L. The objects labeled with HP are called "holding points". Within these physical structures the runway controller can reorder the aircraft before they reach the runway.

The holding points can be identified as queues, ie points where aircraft are queued before entering the runway. If there are many different queues revenue, the ground controller attempts, starting from an initial entry order of the aircraft queues in the holding points, to direct the next aircraft to the HP "tail cheaper" (then running load balancing operation, so as to maximize the capacity of the track). Such operation is not only very difficult, but it can be repeated a limited number of times: this is said constraint of holding points.

As for other proposed formulations, constraints considered for airport security, in this approach, are still the Wake Vortex Separation, the safety distance for routes SID, Speed Group for the initial calibration of the separations and CTOT.

2.9.2 Heuristics for the assignment of path

In reference to the runway 27R shown in Figure 2.3, a typical graph of the waiting point (HP) is shown in Figure 2.4. Each node represents a valid location

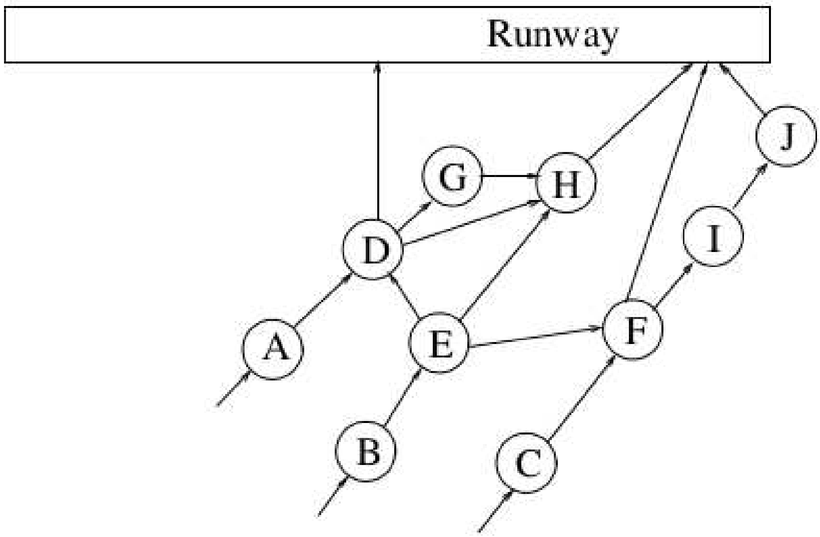

Figure 2.4: An example of holding point network structure

for the aircraft, while the edges represent their transitions through nodes within the holding points.

A valid transverse path consists of a sequence of nodes, interconnected by arcs; the first node of the sequence is the input in the waiting point, the end node is instead the runway. Therefore paths within the holding point are determined using an appropriate heuristic based on how fast an aircraft needs to arrive at the runway.

2.9.3 Departure Scheduling Algorithms

The Basic Search Algorithm

Rather than modeling the movement within the queue positions, the algorithm finds solutions that indicate only the order of take off. Once fixed the order of take off, the paths of the aircraft through the structure of the HP are assigned heuristically. A scheduling is considered "feasible" when all aircraft arriving on the track in the correct order. All of the search heuristics that has been investigated had the same basic format but differed in the details. The full algorithm for the basic search is as follows:

The search algorithm of solutions consists of the following steps:

1. start from the initial solution determined by the order in which the first aircraft arrives in the holding point. This solution is already "feasible" as it does not require the adjustment;

2. assigns heuristically paths for each aircraft within the holding points;

3. through an algorithm of "feasibility" ensure that the order of take-offs is correct;

4. evaluate the cost of the solution;

5. accept or reject the candidate solution. In this step, you can apply different methods of meta-heuristic search (local search algorithms as the first descent, the steeper descent, tabu search, simulated annealing) whose solution becomes the new current solution if it is accepted;

6. if the number of possible evaluations has been reached then the algorithm stops and the best solution hitherto found; otherwise it selects the one close to the current solution and the algorithm repeats starting from step 2.

Search Algorithms **First Descent Algorithm**: this is the most simplistic algorithm. Each new solution is accepted only if it is better than the current solution.

Stepeer Descent Algorithm: The steeper descent algorithm selects fifty candidate solutions at a time. Each candidate is evaluated and the best of the feasible candidates is adopted. The best candidate is adopted even if it is worse than the current solution, which means this is more than a strict descent algorithm. This gives the algorithm a limited ability to move out of local optima but no method to avoid it moving straight back to the local optima it just left. Evaluations of candidates are expensive so the searches are limited to a number of evaluations rather than a number of iterations. This means that the first descent algorithm runs for fifty times as many iterations as the steeper descent algorithm.

Tabu Search Algorithm: The tabu search algorithm is similar to the steeper descent algorithm except that it maintains a list of tabu moves. When a move is made, the reverse move is added to the tabu list to ensure that the search does not go back to where it came from. The reverse move that is recorded will stop any move which would put all of the aircraft that moved back into the absolute positions they previously occupied.

Simulated Annealing algorithm: If the cost of the new solution is less than the current one, then the new will always be accepted, while if it is greater there is still a chance of being accepted. So if we denote by D_{curr} the cost of the current solution and D_{cand} that of the candidate solution, then it will be accepted if $D_{cand} < D_{curr}$.

2.9.4 Formal mathematical model

Let n be the number of considered aircraft, $i \in \{1, ..., n\}$ an integer that represents an individual aircraft, a_i its position in the order of arrival at the waiting point (so that if i is the first to arrive then $a_i = 1$) and c_i its position in the take-off (so that if i is the second aircraft to take off then $c_i = 2$).

We introduce the following integers, referring to the generic aircraft:

- t_i, the internal path to the waiting point;
- v_i, weight class;
- s_i, speed group;
- r_i, the SID;

- h_i, the instant of time in which it enters the waiting point;
- d_i, the scheduled time of departure;
- b_i and l_i, interval bounds corresponding to the time CTOT;
- $\max(0, c_i - a_i)$ indicating the positional delay accumulated by each aircraft that i is overtaked;
- $d_i - h_i$ that expresses the delay to the waiting point, ie the time spent by the aircraft i within the waiting point.

We define two functions $V(v_i, v_j)$ to calculate wake vortex separation, based on the weight classes v_i and v_j , and $R(r_j, s_j, r_i, s_i)$ to calculate the required separation based on the routes SID, r_i and r_j, and the group velocity, s_i and s_j, for aircraft i and j. Both functions V and R give standard values of separation in accordance with the current regulations. Note that the ground controller has a certain flexibility, in the case there are clear weather conditions, to reduce the separations given by $R(r_j, s_j, r_i, s_i)$.

Unlike the functions for the routes SID and speed group, the function related to wake vortex separations satisfies the triangle inequality, ie $V(v_i, v_j) + V(v_j, v_k) \geq V(v_i, v_k)$ for aircraft taking off in the order i, j, k; therefore it is not possible to guarantee that all separations are maintained only by ensuring sufficient separation between adjacent take-offs. For example, consider a scenario in which a start "slow" aircraft is directed to north, followed by another "faster" directed to south, in turn followed by an other "faster" direct to north. As the two trajectories diverge between north and south, departure that consecutively follow these trajectories require a separation of one minute, due mostly to a difference in weight classes. A fast aircraft headed north following a slow also directed it towards the north however, requires a separation of three minutes.

e_i is defined as the lowest take-off time for which all the separations are guaranteed and is given by:

$$e'_i = \begin{cases} 0 & \text{if } \; c_i = 1 \\ \max_{j \in \{1,\dots,n|\} c_j < c}(d_j + \max(V(v_j, v_i), R(r_j, s_j, r_i, r_j))) & \text{if } \; c_i \geq 2 \end{cases} \tag{2.23}$$

For each aircraft a transverse path t_i is assigned heuristically , through the structure of the holding points and defining a suitable function $T(t_i)$, returning the minimum time in which an aircraft i transversely crosses the holding points along the route t_i .

Given the time h_i, in which the aircraft arrives at the waiting point, the lowest time it can reach the runway and take off is given by $h_i + T(t_i)$; in particular the time of take-off can be predicted through the following relation:

$$d_i = \max(e_i, h_i + T(t_i), b_i) \tag{2.24}$$

For the constraint on the time CTOT defines an evaluation function $C(d_i, b_i, l_i, h_i)$ that penalizing those schedules that put planes at the ends of the time interval defined by CTOT. The function $C(d_i, b_i, l_i, h_i)$ has a complicated expression, it takes into account different functional cost which evaluate the delay and introduces factors that allow deviations of scheduling the time of predicted takeoff.

The objective function to be minimized takes into account the total accumulated delay to the holding point, the delay of positioning and eventual failure of the time constraint imposed by CTOT, and is given by:

$$\sum_{i=1}^{n} (W_1(d_i - h_i) + W_2(\max(0, c_i - a_i))^2 + C(d_i, b_i, l_i, h_i)) \tag{2.25}$$

where W_1 and W_2 are appropriate weights.

Chapter 3

Two Stage Algorithm: Formal Model

3.1 Introduction

The proposed modeling approach was inspired by the works of I. Anagnostakis ([8] and [9]), in which some techniques of classical operations research are used. In particular, in [8] it is shown the "Single Stage" methodologie in which the planning function $Q(t)$ is a linear combination of individual objective functions $q(t)_i$, each of one schematizes any goal of the system, while in [9] the original scheduling problem is splitted in two sub-problems or stages. In the first one the goal is to maximize the runway throughput (or capacity) while in the second one it tries the minimize the delay of each aircraft departure.

By analyzing in detail the differences of the two previous methodologies we decided to follow "Two Stages" approach, since the computation times are lower than those obtained with previous approach, and the space of feasible solutions has lower cardinality. Moreover, this kind of methodologie allows us to treat the objectives and constraints separately, increasing maintainability and allowing to manage more efficiently and dynamically insertion of new constraints.

Here, initially a mathematical formulation for the case of single-runway airport was defined, then an algorithm for multi-runway management, considering the special case of independent runways, has been planned and integrated.

3.2 Two Stage Algorithm

The goals commonly recognized by the airport system are:

- maximization of throughput, defined as the number of aircraft that can take off in the considered time window;
- minimization of the delay of individual aircraft: the planned time of take-off (Estimated Take-Off Time) shall not differ much from the target take-off time — TTOT (Target Take-Off Time) — calculated by DMAN;
- minimization of the workload of the controllers in the management of the runway operations;
- fairness, i.e. to treat all the airport users (airlines, passengers, etc.) in the same way;
- minimization of environmental pollution,keeping the engine running as little as possible;

However, since it is difficult to formulate some of these objectives we focused attention only on the following objectives:

- maximization of throughput;
- minimization of aircraft delays, considering the below constraints:
  - Wake Vortex Separation;
  - Calculated Take-Off Time (or CTOT);
  - Estimated Take-Off Time (or ETOT)[1];
  - aircraft priority, i.e. the requirement that aircraft take off as soon as possible.

Both steps of the algorithm are formulated through a linear integer programming model. In a generic problem of *Integer Linear Programming* (or ILP) the objective and constraints are expressed in terms of real linear functions, whose

[1]The first time at which the aircraft is ready for departure.

variables may assume only integer values. Generally, this kind of problems is NP-hard[2]. It means that (unless it is worth P = NP) many problems of PLI require, at the worst case, an exponational time for the computation of the solution compared to the size of the input data. However, some of these problems have been thoroughly investigated and today there are techniques that allow you to solve them in a reasonable time in most common applications.

In the first stage the goal is the maximization of throughput without violating the *Wake Vortex Separation* constraints, while in the second one the delay of each aircraft is minimized, considering as constraints the assigned CTOT, $ETOT$ and priority.

The optimal take off time (TTOT) and the *Target Start uo Approval Time* (or TSAT) of each aircraft are determined. In particular, TSAT is given from the difference between TTOT and the taxiing time employed to reach the runway by aircraft.

$TSAT$ is associated with a change of status of the flight. In fact, if the current time is subsequent to $TSAT_i$, the corresponding $TTOT_i$ can no longer be changed, since the i-th aircraft is taxiing. Therefore, this flight is considered as "*freezed*".

In fact, for each instant of time, a check ion all $TSAT$ is made in order to determine which aircraft can not be rescheduled.

We can summarize, with a high abstraction level, the flow of data between the two stage as shown in figure 3.1.

The flight plans of take-offs and arrivals are properly combined by the *Input Management* module in order to be compatible with the data format used by algorithm. The output is the optimal departures planning.

A more detailed vision of the modules and the related data flow is shown in figure 3.1, in which the core for the *First Stage* is represented by the *Class Sequencer*, while for the second one is the *Aircraft Planner*. The module *Class Sequencer* generates sequences of slots for take-off (class sequences) sorted by decreasing throughput. Consequently, the output is computed by the module "*Merger*", which mixes the sequences with the set of landing operations and "freezed" flight. These se-

[2]The complexity class of decision problems that are intrinsically harder than those that can be solved by a nondeterministic Turing machine in polynomial time. When a decision version of a combinatorial optimization problem is proved to belong to the class of NP-complete problems, then the optimization version is NP-hard.

quences are then stored in a matrix named *Class Sequences Mixed with Arrivals* (or CSMA), shown below.

$$CSMA = \begin{bmatrix} Class \;\; Sequence \;\; Mixed \;\; with \;\; Arrivals \;\; - \;\; 1 \\ Class \;\; Sequence \;\; Mixed \;\; with \;\; Arrivals \;\; - \;\; 2 \\ \ldots\ldots\ldots\ldots\ldots\ldots\ldots \\ Class \;\; Sequence \;\; Mixed \;\; with \;\; Arrivals \;\; - \;\; n \end{bmatrix}$$

Every i-th class sequence — *Class Sequence Mixed with Arrivals* — can be seen as a sequence of slots (”container for aircraft”) in which each individual slot is associated with a weight category whose values can be: *Light* (or L), *Medium* (or M), *Heavy* (or H) or *Super* (or J). Then, the next step consiste of assigne each flight to the compatible slot.

A “start time”, which is calclualted according to the minimum separation time (Wake Vortex Separation) between consecutive movements, is assigned to each slot. In particular, this last operation consists of the following steps:

1. retrieval of all $ETOT$ from the input departures file (the cardinality of $ETOT$ must be equal to that one of the slots);

2. ascending sort of $ETOT$;

3. the first “start time” of the sequence is set exactly equal to the minimum identified $ETOT$;

4. the next *start time* is calculated by adding to the previous one the relative spacing interval where such operation is allowed or the $ETOT$ referred to the current slot is temporally lower to the calculated *start time.* In the case that the last condition is not satisfied, the *starttime* of the current slot is fixed exactly to the value of $ETOT$. For example, let consider two adjacent slots of a generic row of the matrix $CSMA$. We assume that they are compatible with the category M and successively separated of $120 seconds$ to satisfy the Wake Vortex separations. Let also suppose that $ETOT$s are respectively 9:00 and 9:04. The first *start time* is fixed exactly to the first $ETOT$, i.e. 9:00. The second *start time* is obtained by adding to the first slot the separation time between the two classes, which is 120 seconds; therefore t_2 would be

equal to 9:02, but since $t_2 < ETOT_2$ then $t_2 = ETOT_2$. By iteraring this procedure we are able to define all the "start times".

Then, an i-th *Class Sequence Mixed with Arrivals* (or CSMA), including four *Medium-class* and one *Heavy-class* mixed with two landing operations, is shown in Table 3.1.

OPERATION	D	D	D	A	D	A	D
WVC	M	M	M	M	M	M	H
ETOT	9:00	9:01	9:04	9:06	9:06	9:06	9:08

Table 3.1: Departure ready to schedule. D = Departure, A = Arrival

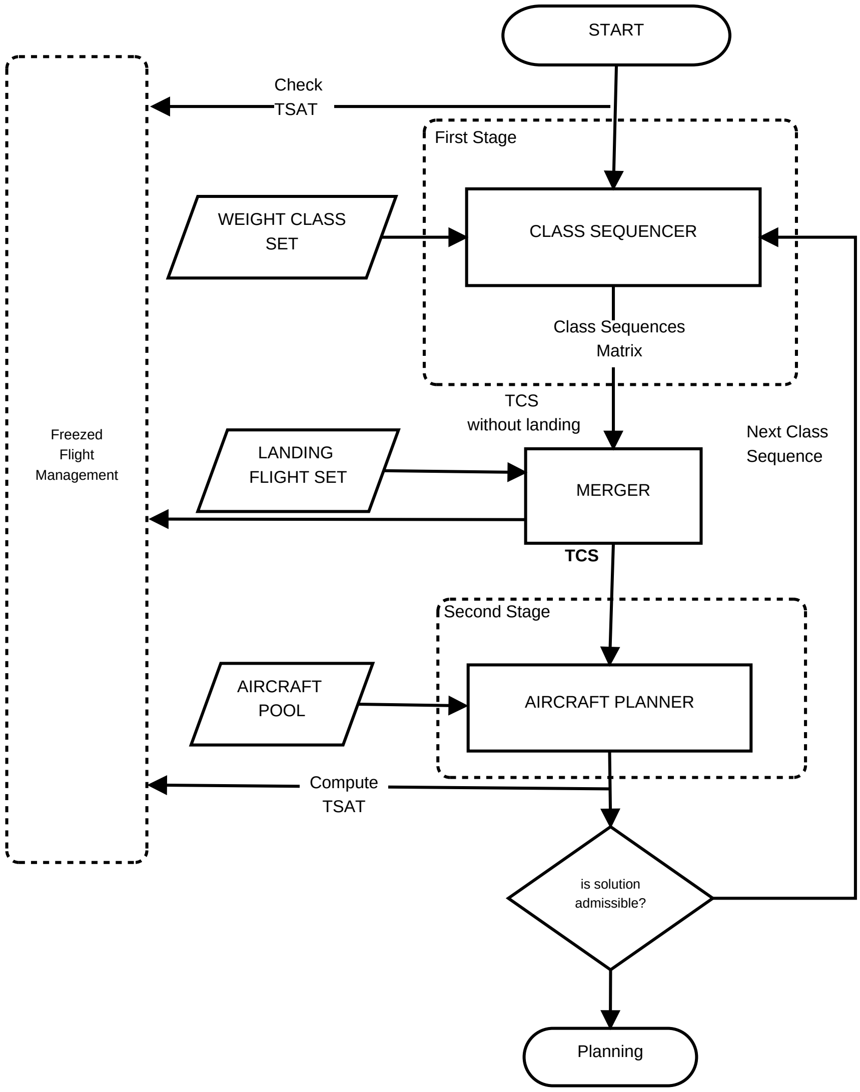

Figure 3.1: Flowchart of algorithm Two Stage

At the end of *First Stage* a tipycal $CSMA$ is shown in the Table (Table 3.2).

OPERATION	D	D	D	A	D	A	D
WVC	M	M	M	M	M	M	H
Start Time	9:00	9:02	9:04	9:06	9:07	9:08	9:09

Table 3.2: Departure ready to schedule. D = Departure, A = Arrival

The Aircraft Planner module is responsible of the assignment of specific aircraft to the slot. In this case we can observe that an output solution can be not admissible according to the following reasons:

- there is no possible allocation of aircraft in the *Class Target Sequence* (or TCS) which satisfies the constraints of the *Second Stage*, ie the $CTOT$ and $ETOT$; in this case the current TCS is replaced with the next available in the matrix CS in order to find a compatible schedule;
- the air traffic controller does not consider suitable planning generated by the algorithm, in which case a new schedule compatible with constraints and the needs of the controller is generated;

3.3 First stage

In order to maximize the runway's *throughput*, we minimize the runtime of the last authorized take off. In particular, sequences of departures slot (class sequences) are generated with the aim of minimizing the time interval that elapses between the first and the last take off operation, without violating the constraints of *Wake Vortex Separation.* If N_A and N_D denote the total number (ie cardinality) of landings and take-offs, respectively, the total number of mixing runway operations is given by: $N = N_A + N_D$. In general the objective function, as said can be written as:

$$\min \max t_i, 1 \leq i \leq N_A + N_D \tag{3.1}$$

where t_i is three occurrence time of the i-th operation of runway.

Here we define throughput as the sum of the time intervals (separations) necessary to ensure that the operations of takeoff / landing are happen safely. Therefore, defined with T_{TOTAL} the time needed to complete the sequence of departures, and

$T_{m,n}$ the time slot between two successive departing operations, m and n, since the arrival time slots are fixed, maximizing departure throughput can be expressed as:

$$\max throughput = \min T_{TOTALE} = \min \sum_{m,n \in N_D} \tag{3.2}$$

where $T_{m,n}$ are obtained from the *Wake Vortex Separation* matrix (Figure 3.2), whose elements indicate the time (in seconds) that must elapse between two take-offs (identical or different weight class) adjacent so that they will be made in safety.

Front \ Following		Departures				Arrivals			
		L	M	H	J	L	M	H	J
Departures	L	60	60	60	60	60	60	60	60
	M	120	60	60	60	120	60	60	60
	H	120	120	120	120	120	120	60	60
	J	180	180	120	120	180	180	60	60
Arrivals	L	60	60	60	60	-	-	-	-
	M	120	60	60	60	-	-	-	-
	H	120	120	120	120	-	-	-	-
	J	180	180	120	120	-	-	-	-

Figure 3.2: Separation times between weight class

In this first phase the class sequence generated will be free of slots to host landings that will be inserted subsequently by *Merger*. This design choice is dictated by the nature of DMAN, whose aim is to develop a schedule of take-offs with fixed slots for incoming flights (whose planning is task of the system AMAN).

The output of the first phase is an mxn matrix of class sequence, in which m represents the number of sequences generated and n the number of slots of each of them. The output of the first stage is mxn matrix of class sequences, ordered from maximum to minimum throughput, where m represents the number of generated sequences and n the number of slots for each sequence. The term class sequence indicates a sequence of time slots based on weight classes of all aircraft considered in the predefined time window.

3.4 Formal model for the first stage

A class sequence is a sequence of slots with the property to maximize the throughput of the runway. To calculate this sequence we conceived a model of *Mixed Integer Linear Programming* (or MILP) on the *Asymmetric Traveling Salesman Problem* (or ATSP) archetype. The problem was first formulated as a mathematical problem in 1930 and is one of the most intensively studied problems in optimization. It is used as a benchmark for many optimization methods. Even though the problem is computationally difficult, a large number of heuristics and exact methods are known, so that some instances with tens of thousands of cities can be solved.

In the theory of computational complexity, the decision version of the *Traveling Salesman Problem* (or TSP) — where, given a length L, the task is to decide whether any tour is shorter than L — belongs to the class of *NP-complete problems.* Thus, it is likely that the worst-case running time for any algorithm for the TSP increases exponentially with the number of cities.

In the symmetric TSP, the distance between two cities is the same in each direction, forming an undirected graph. This symmetry halves the number of possible solutions. In the asymmetric TSP, paths may not exist in both directions or the distances might be different, forming a directed graph. Traffic collisions, one-way streets, and airfares for cities with different departure and arrival fees are examples of how this symmetry could break down.

For the resolution of the ATSP can use traditional algorithms for NP-hard problems:

- **Algorithms to find the exact solution**: reasonably fast for problems with a relatively low number of nodes. The most commonly used algorithms in this area are: *Branch & Bound* (B&B) and *Branch & Cut* based on linear programming.

- **Approximation algorithms**: they have a high probability of producing a "good" solution "quickly". Modern methods can find solutions in an acceptable time for extremely large problems (millions of nodes), with a high probability that differ by 2-3% from the exact solution.

As mentioned previously, in order to formulate the model of mixed-integer linear programming for this first stage we brought back to ATSP problem considering instance of our problem (for the first stage) and transforming it into ATSP, so that an optimal solution for the latter is also an optimal solution for the problem of the first phase. As mentioned previously, in order to formulate the Mixed Integer Linear Programming model for this first stage was made a reduction to the ATSP problem , so that an optimal solution for the latter is also an optimal solution for the problem of the first stage.

The reduction operation, simplified, to a ATSP problem is described below:

- determine the weight category for each aircraft which is expected to take-off in the time interval considered and assign a unique label (eg for the first aircraft of class *Light* (or L) assign the label $L1$);
- each label detected becomes a node on the graph of the ATSP.
- for each pair of nodes in the graph add a directed arc, the weight of which is the time of separation to avoid the interference caused by the wake vortex or a function which takes account of these separations and other parameters;
- add a dummy node to the graph labeled "*Start*", connected with directed arc at each node of the graph that identifies a distinct class of weight (in relation one by one). This node represents the last category aircraft cleared to take off in the time frame immediately preceding it. The weight on the arc is the time of separation to avoid disturbances of the wake vortex. In the particular case in which there is no earlier than the current time window, on the arcs coming out from the node "*Start*" arises a weight equal to zero;
- add the node "*Next*" that has the meaning to represent the first plane of the next sequence (ie the time window of the next). This node is connected with directed arc at each node of the graph that identifies a distinct class of weight (in relation one by one);

A possible graph that is obtained from this reduction operation is represented in the Figure 3.3.

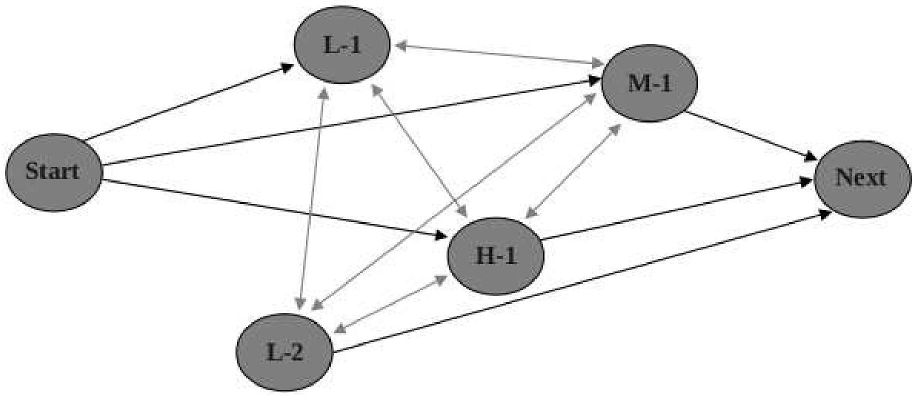

Figure 3.3: Example of reduction to ATSP problem

The reduction shown in Figure 3.3 is just an unoptimized version of that actually modeled, because ignoring the node *Start* and *Next*, the resulting graph is complete.

3.5 First Stage Mathematical Model

We divide the set of all take-offs in several subsets representing the flights belonging to a specific weight class:

- $N_D = \{1, ..., n\}$,the set off take-offs to schedule;
- $N_L = \{1, ..., l\}$,the set of *Light* (or L)weight classes;
- $N_M = \{l + 1, ..., m\}$,the set of *Medium* (or (M) weight classes;
- $N_H = \{m + 1, ..., h\}$, the set of *Heavy* (or H) weight classes;
- $N_j = \{h + 1, ..., j\}$, the set of *Super* (or J) weight classes;
- $N_{Start} = \{1, l+1, m+1, h+1\}$, the set of nodes with ingoing arcs from *Start* node;
- $N_{Next} = \{l, m, h, j\}$, the set of nodes with outgoing arcs to *Next* node.

We note that $N_d = N_L \cup N_M \cup N_H \cup N_J$. The decision variables of the problem are:

$$X_{a,b} = \begin{cases} 1, \text{if, in the sequence, aircraft } \mathbf{a} \text{ follows aircraft } b \\ 0, \text{otherwise} \end{cases}$$

$$X_{Start,a} = \begin{cases} 1, \text{if, in the sequence, aircraft } \mathbf{a} \text{ follows aircraft } Start \\ 0, \text{otherwise} \end{cases}$$

$$X_{a,Next} = \begin{cases} 1, \text{if, in the sequence, aircraft } \mathbf{Next} \text{ follows aircraft } a \\ 0, \text{otherwise} \end{cases}$$

The objective function can be formulated as:

$$\min \sum_{\substack{a,b \in N_d \\ a \neq b}} T_{a,b} X_{a,b} + \sum_{a \in N_d} T_{a,Next} X_{a,Next} \tag{3.3}$$

The constraints are given by

$$\sum_{b \in N_d - N_K} X_{a,b} + X_{a,a+1} = 1, \forall a \in N_k - N_{Next} \ , K = L, M, H, J \tag{3.4}$$

$$\sum_{b \in N_d - N_{Start} - N_k} X_{a,b} + X_{Start,b} = 1, \forall b \in N_{Start},\ K = L, M, H, J \tag{3.5}$$

$$\sum_{b \in N_d - N_{Next} - N_k} X_{a,b} + X_{b,Next} = 1, \forall b \in N_{Next},\ K = L, M, H, J \tag{3.6}$$

$$\sum_{a \in N_{Start}} X_{Start,a} = 1, \sum_{a \in N_{Next}} X_{a,Next} = 1 \tag{3.7}$$

The MTZ Formulation. To ensure that all the aircraft of N_d are in the computed solution (to exclude subtour in the graph), we can use extra variables $u_i (i = 1, ..., n)$ and following constraints:

$$\begin{gathered} u_1 = 1, \\ 2 \leq u_1 \leq n \ \forall i \neq 1, \\ u_1 - u_j + 1 \leq (n-1)(1 - x_{ij}) \forall i \neq 1, \forall j \neq 1 \end{gathered} \tag{3.8}$$

The relations in (3.8), which is an arc-constrain, is known in literature as *Miller-Tucker-Zemlin* (or MTZ) formulation. It indeed excludes subtours, the arc-constraint $for(i, j)$ forces $u_j \geq u_i + 1$, when $x_{ij} = 1$; if a feasible solution of ATSP with MTZ contained more than one subtour, then at least one of these would not contain node 1, and along this subtour the u_i values would have to increase to infinity. This argument, with the bounds on the u_i variables, also implies that the only feasible value of is the position of node i in the tour. The advantages of the MTZ formulation [18] are:

- its small size (we need only n extra variables and roughly $n^2/2$ extra constraints),
- if it is preferable to visit, say, city i early in the tour, one can easily model this by adding a term $-\alpha u_i$ with some $\alpha > 0$ to the objective.

3.6 Heuristic for Mixing Arrivals and Departures

The *Departure Management* (or DMAN) requires in input the sequence of landing aircraft for planning an optimal scheduling of departures. Safety regulations require that arrivals have priority over departures. In addition, you must ensure that between successive arrivals and successive departure and arrivals the right separation time are observed to prevent the formation of wake vortex.

To manage the minimum separation between arrivals and departures, and vice versa, was designed and developed heuristics “ad hoc” represented by macroblock *Merger* of Figure 3.1. The first operation, this block does, is to identify within the class sequences the correct position for the arrival flight based on the value of its *Estimated Landing Time* (or ELDT) and add a new time slot (forbidden departures) with start time equal to its ELDT. As mentioned above regarding the high priority of arriving aircraft, in order to limit the time that they spend in flight above the airport (to avoid waste of fuel, clogging of the TMA, noise emissions, etc..), If the time ELDT an arrival is the same (overlapping) to ETOT of a departure, the latter is delayed by the time necessary to ensure the separation between landings and take offs.

To get an admissible and optimized planning of departures, we adjust every class sequence taking into account arrivals, i.e. separations between successive arrivals and departures, in the selected time window. The mean steps for this operation are:

1. find the right position of each arrival inside the class sequence based on ETA (Estimated Time of Arrival);
2. add a new time slot for each arrival as a "fixed"" (the "start time" is fixed) slot for departures, with start time equal to *Estimated Time Arrival* (or ETA);
3. to compute by means of heuristic the new start time of departure time slot.

Heuristic method

We indicate with t_{d_j} the start time of the j-departure time slot, t_A the time of some arrival, $\triangle t$ the separation slot between departure and arrival and $slot(j; j+1)$ the separation time between two consecutive aircraft j and $j+1$. The new start time is computed as follows:

- if $|t_{d_j} - t_A| \leq \triangle t$, then $t_{d_j} = t_A + \triangle t$, $t_{d_{j+1}} = t_{d_j} + slot(j, j+1)$;
- if $|t_{d_j} - t_A| > \triangle t$, then t_{d_j} remains unchanged.

The intervals of separations ($\triangle t$) that exist between successive arrivals and departures and vice versa are shown in the Table 3.2.

3.7 Second Stage Mathematical Model

In the second stage we will generate an "*Aircraft Schedule*" (physical assignment of specific aircraft to the slots) admissible to minimize the sum of all delays (total delay), or the sum of all the intervals between the take-off time (scheduled) for each specific aircraft (ETOT) and the *start time* associated with each slot of the *Target Sequence Class* (or TCS), taking into account the possible constraints of CTOT, if present, and priority constraints.

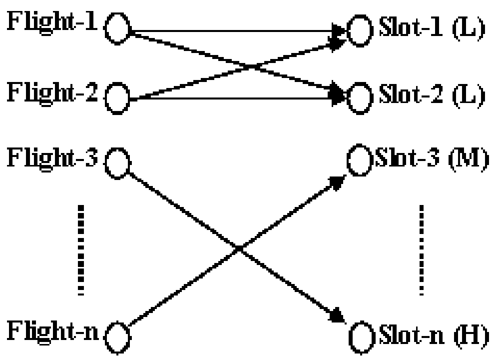

Figure 3.4: Second Stage graph

This can be achieved by formulating a model of operational research, in which we need to assign different activities in an optimal way, this class of problem is known in the literature as *Assignment Problems* and generally belongs to the class of *NP-hard problems*.

Even at this stage, as in the above, we make use of *Graph Theory* to make the understanding of the problem more immediate. In fact, the second stage may be schematized as in Figure 3.4, in which each aircraft is assigned to a slot of TCS, compatible with its weight class, taking into account the delay (weight on arcs) that aircraft incur if assigned to that particular slot.

As already mentioned, the constraints considered at this stage are the $CTOT$, the $ETOT$ and *priority*. To better understand the basic idea of this discussion we consider an example. Suppose that the TCS output of the first stage (input to the second) is that shown in the Table 3.3, then in the Table 3.4 is shown the likely aircraft allocation processed of the *Second Stage Algorithm*.

TCS	L	M	L	M	H
START TIME	09:00	09:01	09:03	09:04	09:05

Table 3.3: TCS in output from the first stage

Call Sign	**AZ235**	**CB253**	**D784**	**ZZ453**	**AA112**
Weight Class	M	L	M	H	L
ETOT	09:01	09:03	09:05	09:02	09:00
pCTOT_ETOT	[2-4]	[1-3]	[2-4]	[5]	[1-3]
Prt	1	0	0	0	1

Table 3.4: Summary the elaboration of the Second Stage

We introduce the following notation:

- **pCTOT_ETOT** is a suitable function which converts each interval bounded below by ETOT and upwardly by $CTOT + 10$ in a set of positions inside the TCS (for example since the aircraft AZ235 belongs to class M can only occupy the slots 2 or 4);
- **Prt** is a function which, depending on the priorities indicated in the flight plan, restricts the scope 'time' allocation, thus forcing the aircraft with high priority to occupy the first slot compatible with its weight class, and without violating the $ETOT$ and $CTOT$.

The assignment problem is represented by the graph as in Figure 3.5. Here the arc weights indicate the del delay (expressed in seconds) of the generic aircraft respecting the possible slot allocation (i.e. the absolute value of difference between the $ETOT$ of each aircraft and the "*start time*" of the corresponding assignable slot).

The only feasible solution to the above problem is shown in Figure 3.6.

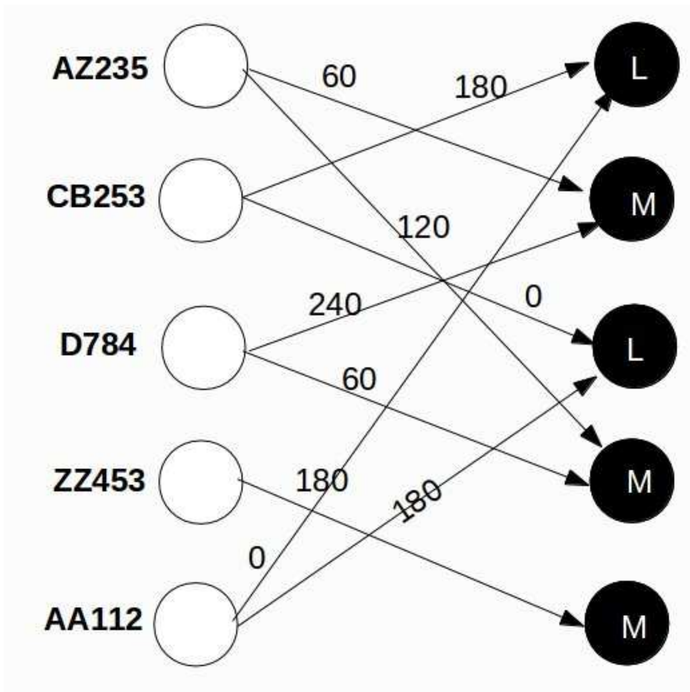

Figure 3.5: Input graph for the Second Stage Algorithm

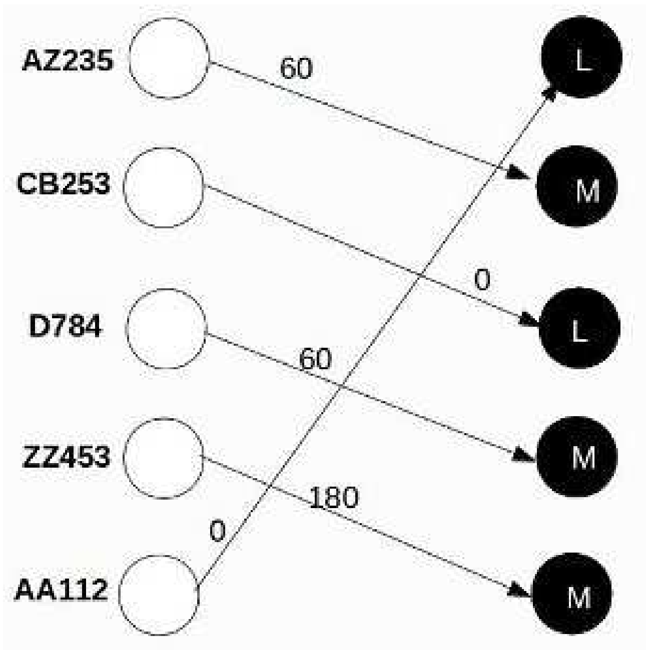

Figure 3.6: Output graph for the Second Stage Algorithm

We have to observe that in the sequences of flight plans, there may be aircraft without assigned $CTOT$. These flights are handled with low priority.

3.8 Second Stage Mathematical Model

We introduce the following notations:

- $S = \{s_1, s_2, ..., s_n\}$ represents the solution given by the algorithm in output of the first stage and each s_1 with $i = 1, ..n$, indicates the slots;
- f_i $i = 1, ..., n$ represents the identification number of the aircraft;
- $F = \{f_1, f_2, ..., f_n\}$ is the set of aircraft f i to assign to the slot;
- $SC_j = \{f_i \in F \; |s_j$ is compatible with $f_i\} \; \forall \; j \in S$ the set of each aircraft f_i compatible with time slot s_j ;
- $FC_i = \{s_j \in S \mid f_i$ is compatible with s_j $\}$, is three set of all slots s_j compatible with the aircraft f_i;
- $ETOT(f_i)$ is the Estimated Take-Off Time of the generic f_i ;
- $TOff(s_j)$ is the start time of the generic slot s_j ;

Let us define the decision variables
$X_{f_i}, s_j = \begin{cases} 1, \text{if aircraft } f_i \text{ can be assigned to the slot } s_j \\ 0, \quad \text{otherwise} \end{cases}$
and two functions:

- $maxCTOT(f_i)$, $\forall f_i \in F$ which returns the value $CTOT + 10$ (in minutes) referred to the aircraft with Identifier Code f_i ;
- $Prt(f_i)$, $\forall f_i \in F$ which provides the allocation limit according to the specific priority value.

The objective function can be stated as:

$$\min \sum_{f_i \in F, s_j \in S} |TOff(s_j) - ETOT(f_i)| X f_i, s_j \tag{3.9}$$

minimized the delay for each individual aircraft in relation to its ETOT.

With constraints

$$\sum_{s_j \in FC_i} X f_i, s_j = \Pr t(f_i), \forall f_i \in F, \tag{3.10}$$

$$\sum_{f_i \in SC_j} X f_i, s_j = \Pr t(f_i), \forall f_i \in F, \tag{3.11}$$

$$ETOT(f_i) \leq \sum_{s_j \in C_i} s_j X f_i, s_j \leq \max CTOT(f_i),\ \forall f_i \in F, \tag{3.12}$$

$$\sum_{s_j \in FC_i} s_j X f_i, s_j = \Pr t(f_i), \forall f_i \in F \tag{3.13}$$

The constraint 3.10 requires that for each aircraft f_i can be assigned to a single slot s_j, meanwhile the constraint 3.11 requires that at each slots s_i can be assigned to a single aircraft f_i. The constraint 3.12 identifies for each aircraft f_i in F , a whole range of slots within the set SC_j which can be assigned. The last constraint 3.13 restricts the allocation for aircraft with high priority.

3.9 Freezed Flight

With freezed we indicate those aircraft which already left the parking bay and are taxiing to the allocated runway. In these cases the assigned slot is said "*freezed*", as it can no longer be changed inside the starting sequence. The check is performed by calculating, for each aircraft, the *Target Start-up Approval Time* (or TSAT), which is the time they leave their parking bay.

The algorithm for the management of "freezed" flights consists of the following steps:

1. for each departure, after execution of the *Second Stage*, $TSAT$ are calculated as $TTOT - average\_taxitime$[3] ;

2. aircraft are stored in a dynamic buffer; the corresponding slots are frozen ("freezed") and deleted from the input sequences of the *First Stage*;

[3]The average time for taxiing the airplane from the parking bay to the holding point of the runway.

3. at the beginning of the *First Stage*, a check for aircraft with $TTOT \leq current\_time$ is done; in this case those aircraft are deleted from both the input sequence;

   After the *First Stage* the Merger module mixes the sequence of arrivals with flights stored in the buffer;

3.10 The multi-runway procedure

Since in most cases, airports have more runways, the problem is to find the better assignment of each departure to each runway in order to maximize the performance of the airport, i.e. to minimize the total delay, defined as the sum of each aircraft delay, in compliance with safety and security rules.

Between the general allocation criteria in this context we use only:

- runway status: ONLY ARRIVAL, ONLY DEPARTURE, BOTH, CLOSED;
- aircraft weight class associated to the category of each runway;
- capacity of each runway, i.e. the maximum number of operations that the runway can contain.

Other criteria are the wind direction, destination airport, the first fix point en route, taxi-out time, some specific airport characteristics.

In Figure 3.7 on the left a macroscopic vision of the multi-runway approach is shown. The multi-runway macroblock receives as input the departures and arrivals flight plan and the configuration parameters of each runway, i.e., status in the temporal horizon, capacity and category. The macroblock consists of three internal sub-modules (Figure 3.7, right), i.e. the Configuration Procedure, Sequencing Procedure and Capacity Control. The second module recalls the Two Stage Algorithm in order to control and establish if each runway current scheduling allows to get the minimum possible total departing delay, while the last module control if the capacity of each runway is respected.

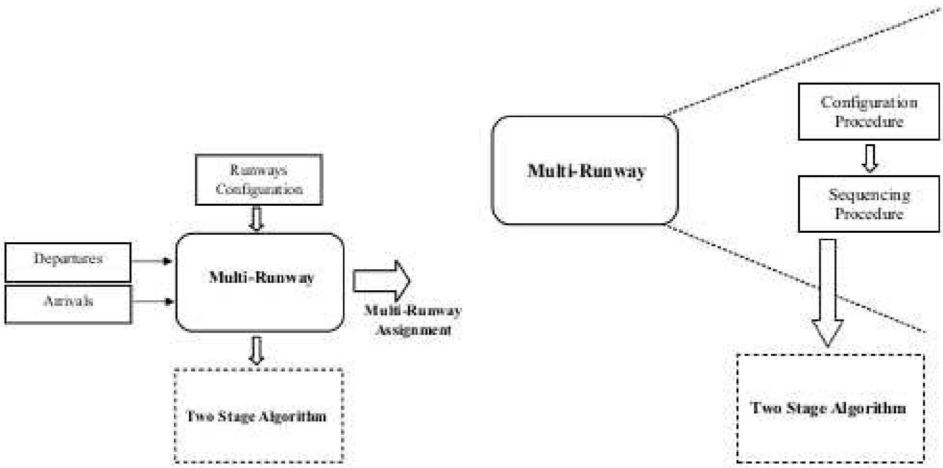

Figure 3.7: Two stages algorithm dataflow

3.10.1 Configuration Procedure

Starting from the information about the properties and characteristics of each runway, this procedure consists of several steps to locate preliminarily aircraft on the available runways:

Step 1. Sorting of the runway considering the following order: Only Departure (or Both), Only Arrival, Closed;

Step 2. Separations of aircraft in some configurable time windows (in minutes);

Step 3. Alternate positioning of the separation aircraft class (found to the previous step) on each runway *OnlyDeparture* or *Both*;

Step 4. For each time window, verify if aircraft are compatible with the runway category; if not, move that specific aircraft on the compatible runway;

Step 5. Put the arrivals inside each time window (if they exist) for the runways *Both*;

Step 6. Call the *SequencingProcedure* module.

3.10.2 Sequencing Procedure

This module allows to reduce the total delay, using the *Two Stage Algorithm* applied for each runway and successively adopting a heuristic process of swapping between the runways. First, the *Two Stage Algorithm* computes the delays on each runway and a queue, formed by delayed aircraft, is created.

Then, for each queue, if the compatibility between the runway category and the aircraft weight class is strictly verified, this aircraft is removed from the queue, while the more delayed one is moved on another compatible runway and using again the *Two Stage Algorithm*, the new delays are computed. If the total delay, given by the sum of the delays of each runway, decreases, then the swapping is confirmed, otherwise the aircraft goes back in the original position and it is removed from the queue. This process is iteratively repeated until each queue will be empty, so that the current planning is finally accepted.

We can summarize, with the following steps, this operations on two runway airport:

1. using the *DMAN* you calculate the delays of each runway;
2. with each runway is associated a queue that contains all aircraft with a delay nonzero;
3. for each runway, if the associated queue contains aircraft that can not be moved (only compatibility with the current runway), then such aircraft are eliminated;
4. for each runway, sort the queue in descending order (on delay) and then move the first aircraft to the other compatible (Annex B) runway. Then apply the *Two Stage Algorithm* for computing the new delays;
5. for each runway, if the total delay (sum of the delay of each runway) was decremented, then the displacement is confirmed, otherwise the aircraft is returned to its position of origin and deleted from the queue delays;
6. for each runway, if the total delay (sum of the delay of each runway) was decremented, then the displacement is confirmed, otherwise the aircraft is returned to its position of origin and deleted from the queue delays;

7. repeat the first step until the queue delays do not empty;
8. capacity control step: construct an instance of the problem (*Integer Linear Programming*) as defined in the Section 3.10.3;
9. build flight plan from the solution of previous step.

3.10.3 Capacity Control

In order to ensure that each runway capacity is respected, starting from the final solution of Sequencing Procedure, an Integer Linear Programming problem is solved.

Mathematical Model

We indicate with $F = \{1, ..., n\}$ the set of the aircraft to assign, and $R = \{1, ..., m\}$ the set of the available runways.

Let us define

- $E = \{E_1, E_2, ..., E_n\}$ where $Ei \subseteq F$ has $ETOT$s inside a pre-fixed time window;
- $U_E = \{E_1, ..., Et\}$ such that $E_j \subseteq F$ and $|ETOT_{E_t} - ETOT_{E_1}| \leq 60$ min;
- $U = \{U_{E_1}, ..., U_{E_t}\}$;
- $PC_{i,j} = \{C_1, C_2, ..., C_n\}$ with $C_j = \{c_1, c_2, ..., C_x\}$ the capacity's vector for each time windows x and runway j.

The decision variable is

$$X_{i,j} = \begin{cases} 1 & \text{if aircraft } i\text{can be assigned to the runway } j \\ 0 & \text{otherwise} \end{cases}$$

Introduce a probability function $f_{i,j}(\bar{D})$ which gives the probability to assign each aircraft i to the runway j, where $\bar{\mathrm{D}}$ is the vector of assignment criteria.

For example for an airport with two runway in *Both* mode, the probability function is expressed as

$$f_{i,j}(\bar{D}) = \begin{cases} 1 & \text{if aircraft } i \text{ is assigned to the runway } j \\ 0 & \text{if aircraft } i \text{ is not assigned to the runway } j \\ 0.5 & \text{if aircraft } i \text{ can be assigned to each runway } j \end{cases}$$

For all U_{E_s} belonging to U the problem to compute the better aircraft planning subject to the capacity of the available runways can be formulated as

$$\max \sum_{i=1}^{n} \sum_{j=1}^{m} f_{i,j}(\bar{D}) X_{i,j} \tag{3.14}$$

subject to

$$\sum_{j=1}^{m} X_{i,j} = 1, \forall i \in F \tag{3.15}$$

$$\sum_{i=1}^{n} X_{i,j} \leq c_j, \forall j \in U_{E_s}, \forall c_j \in PC_U \tag{3.16}$$

$$X_{i,j} \in \{0, 1\} \tag{3.17}$$

with constraints (3.15) ensure that each aircraft is assigned almost one runway and constraints (3.16) impose that for each time window the maximum capacity of runway is not exceeded.

Chapter 4

Simulations and Results

4.1 Introduction

The *Multi-Runway Procedure* and the *Two Stage Algorithm* were implemented and tested on some real data on Milano Malpensa Airport supplied us by the researchers of SESM s.c.a.r.l. (a Finmeccanica Company).

The airport has two runways, 35L (left) and 35R (right) and we consider various time windows in which each runway can be in mode:

- ONLY ARRIVAL: runway accepts only landing flight;
- ONLY DEPARTURE: runway accepts only departure flight;
- CLOSED: all operations on runway are inhibited:

A flight plan contains the following main fields, (others fields are omitted for simplicity):

- Call Sign: an ID that identifies a flight;
- Type: aircraft model;
- Category: wake vortex category;
- ADEP: departure airport:
- ADES: destination airport;

- ETOT: estimated take-off time;
- CTOT: calculated take-off time;
- Priority: priority;
- TA: time of arrival.

The tests were conducted with the aid of a tool (hereinafter called DMAN), specially designed, which implements the Two Stage and Multirunway Algorithm.

4.2 Scenario #1

In this scenario, prove that the DMAN ensures the safety requirements considering the constraints of separation between (incoming and outgoing) flights due to the category of wake vortex.

We test the following constraints:

- Wake Vortex Separation;
- CTOT - Calculated Take-Off Time;
- Landing flight priority;
- One runway active (P35R);
- Priority between departure.

In Tables 4.1 and 4.2, respectively, are shown *Flight Strips* for departure and landing aircraft.

Call Sign	Category	ADEP	ETOT	CTOT	Prt
AZ700	J	LIMC	10:10	10:10	H
AZ800	L	LIMC	10:10	10:10	N
AZ701	J	LIMC	10:20	10:20	H
AZ801	M	LIMC	10:20	10:20	N
AZ702	J	LIMC	10:30	10:30	H
AZ802	H	LIMC	10:30	10:30	N
AZ703	J	LIMC	10:40	10:40	H
AZ803	J	LIMC	10:40	10:40	N
AZ900	L	LIMC	11:00	11:00	H
AZ901	M	LIMC	11:10	11:10	N
AZ902	H	LIMC	11:20	11:20	H
AZ903	J	LIMC	11:30	11:30	N
AB100	L	LIMC	11:40	11:40	H
AB101	M	LIMC	11:50	11:50	N
AB102	H	LIMC	12:00	12:00	H

Table 4.1: Flight Strips for departure aircraft

TA	Call Sign	Category	ADEP	ADES
11:00	ARR100	L	EDDM	LIMC
11:10	ARR200	L	EGKK	LIMC
11:20	ARR300	L	LIRA	LIMC
11:30	ARR400	L	LIRF	LIMC
11:40	ARR101	M	LSZH	LIMC
11:50	ARR201	M	EKCH	LIMC
12:00	ARR301	M	LIRN	LIMC

Table 4.2: Flight Strips for landing aircraft

In Table 4.3 the schedule given by DMAN is shown.

Operation	**Call Sign**	**TCT**	**ETOT**	**TTOT**	**TA**	**TSAT**
DEP.	AZ700	J	10:10	10:10		10:09
DEP.	AZ800	L	10:10	10:13		10:12
DEP.	AZ701	J	10:20	10:20		10:19
DEP.	AZ801	M	10:20	10:23		10:22
DEP.	AZ702	J	10:30	10:30		10:29
DEP.	AZ803	H	10:30	10:32		10:31
DEP.	AZ703	J	10:40	10:40		10:39
DEP.	AZ803	J	10:40	10:42		10:41
ARR.	ARR100	L			11:00	
DEP.	AZ900	L	11:00	11:01		11:00
ARR.	ARR200	L			11:10	
DEP.	AZ901	M	11:10	11:11		11:10
ARR.	ARR300	L			11:20	
DEP.	AZ902	H	11:20	11:21		11:20
ARR.	ARR400	L			11:30	
DEP.	AZ903	J	11:30	11:31		11:30
ARR.	ARR101	M			11:40	
DEP.	AB100	L	11:40	11:42		11:41
ARR.	ARR201	M			11:50	
DEP.	AB101	M	11:50	11:51		11:50
ARR.	ARR301	M			12:00	
DEP.	AB102	H	12:00	12:01		12:00

Table 4.3: Output given by DMAN

As we can see from the solution computed by DMAN all constraints are satisfied, in particular:

- if AZ700 and AZ800 have the same priority the DMAN would plan for take off AZ800, before AZ700. But since AZ800 has a high priority it is scheduled as the first aircraft to take off, despite belonging to a higher category (J - Jumbo);
- all wake vortex separation and CTOT slot are satisfied;
- the total delays is equal to 18 minutes (sum of all delays) and it is the lowest possible value;
- the time arrival of all landing flight remains unchanged;

  Furthermore, this solution reflects the expected solution provided by SESM's researchers.

4.3 Scenario #2

In this scenario will test the constraints related to the assigned slot. In this case there are more departures who compete for the same slot. The algorithm, if there is not the possibility of finding an admissible schedule, it generates an alert to the flight controller. The alert message can specify the departure ID that not satisfy the constraints. In the Table 4.4 there are 19 flights with the same CTOT assigned.

Call Sign	**Category**	**ADEP**	**ETOT**	**CTOT**	**Prt**
A1	L	LIMC	08:55	09:00	N
A2	L	LIMC	08:56	09:00	N
A3	L	LIMC	08:57	09:00	N
A4	L	LIMC	08:58	09:00	N
A5	L	LIMC	08:59	09:00	N
A6	L	LIMC	09:00	09:00	N
A7	L	LIMC	09:01	09:00	N
A8	L	LIMC	09:02	09:00	N
A9	L	LIMC	09:03	09:00	N
A10	L	LIMC	09:04	09:00	N
A11	L	LIMC	09:05	-	N
A12	L	LIMC	09:06	09:00	N
A13	L	LIMC	09:07	09:00	N
A14	L	LIMC	09:08	09:00	N
A15	L	LIMC	09:09	09:00	N
A16	L	LIMC	09:09	09:00	N
A17	L	LIMC	09:05	09:00	N
A18	L	LIMC	09:10	09:00	N
A19	M	LIMC	09:10	09:00	N

Table 4.4: Flight Strips for departure aircraft

Here the DMAN generates a warning message and it not processes a departure planning. Indeed, all flights in Table 4.4 have the same CTOT assigned, but only 17 flights can be properly scheduled.

4.4 Scenario #3

In this scenario there are two runways active (35L and 35R), that accept take-off and landing operations, and the capacity of the runways is limited to ten operation. The runway 35L not accept aircraft of category High, while the runway 35L has no limits on wake vortex category. Therefore, with this scenario we test that the DMAN in the allocation of the runways takes into account:

- the consistence of aircraft type with the runway category;
- state and runway capacity;
- regulation of the traffic for daily use of the runways;
- CTOT - Calculated Take-Off Time.

Flight Strip for departure planes are shown in Table 4.4.

Flight Strip for landing aircraft are (assigned to the runways **35R**):

For abbreviation we omit some flight strips' fields in the evaluated solution shown in the following table:

Call Sign	Category	ETOT	CTOT	ADEP	PRT
D20	L	09:37	09:35	LIMC	H
D21	M	10:05	10:05	LIMC	N
D22	L	10:10	10:10	LIMC	N
D23	L	10:15	10:15	LIMC	N
D24	M	10:20	10:20	LIMC	N
D26	L	10:30	10:30	LIMC	H
D27	H	10:35	10:35	LIMC	N
D28	M	10:40	10:40	LIMC	N
D29	M	10:45	10:45	LIMC	N
D30	L	10:45	10:45	LIMC	N
D31	M	10:50	10:50	LIMC	N
D32	M	10:55	10:55	LIMC	N

TA	Call Sign	Category	ADEP	ADES
10:29	A4	H	EDDM	LIMC
10:34	A5	J	EGKK	LIMC
10:40	A6	L	LIRA	LIMC
10:42	A7	L	LIRF	LIMC
10:44	A8	L	LSZH	LIMC
10:45	A9	L	EKCH	LIMC
10:47	A10	L	LIRN	LIMC
10:49	A11	L	LIRN	LIMC
10:50	A12	L	LIRN	LIMC
10:20	A13	L	LIRN	LIMC

35L			35R		
Operation	**Call Sign**	**TTOT**	**Operation**	**Call Sign**	**TTOT**
DEP.	D22	10:10	DEP.	D21	10:05
DEP.	D23	10:15	ARR.	A4	10:29
DEP.	D24	10:20	ARR.	A5	10:33
DEP.	D26	10:30	DEP.	D27	10:35
DEP.	D19	10:29	ARR.	A6	10:40
DEP.	D28	10:40	ARR.	A7	10:42
DEP.	D30	10:45	ARR.	A8	10:44
DEP.	D29	10:46	ARR.	A9	10:45
DEP.	D31	10:50	ARR.	A10	10:47
DEP.	D32	10:55	ARR.	A11	10:49

As we can see from the solution, also in this scenario, all constraints are satisfied. In particular:

- all arrivals are placed on the runway 35R;
- the limit on capacity is complied;
- the category runway compatibility (see aircraft D27) is complied.

4.5 Scenario #4

Tn this last test we consider a more complex scenario. The following test is split in several steps and takes into account:

- the aircraft compatibility with the category (Annex B) runway;
- state and runway capacity:
- traffic regulations for the daily use of the runways.

4.5.1 Setup of scenario

The initial conditions for the launch of the test are the following:

- runway 35L and 35R in BOTH (accept landing and departure) state;
- runway capacity limited to 40 movements/hour;
- category of runway 35R is set to 1:
- category of runway 35L is set to 1;
- daily regulation
  - 9:10 - 10:30: 35L only arrivals, 35R only departures;
  - 11:00 - 12:45: 35L only departure, 35R only arrivals;
  - 14:00 - 14:40: 35L closed, 35R both;
  - 17:00 - 17:20: 35R closed, 35L both;
  - 18:00 - 18:35: 35L only departure, 35R both;
  - 19:00 - 20:00: 35R only arrivals 35L both;
- current time: 08:00.

It is assumed that we have an average taxi-time equal to 5 min for all flights. In the Table 4.5.1 are shown the aircraft ready for departing.

Call Sign	**Category**	**ETOT**	**PRT**	**ATYP**	
D1	L	08:20	N	BE36	
D2	L	08:20	N	U21	
D3	L	08:28	N	LJ28	
D4	M	08:28	N	YS11	
D5	H	08:40	N	A30B	
D6	M	08:40	N	DC3	
D7	H	08:53	H	A310	
D8	H	08:53	N	B703	

The tests require different steps to simulate a real-world scenario, and it consists of 6 steps:

STEP 1: insert into the database of DMAN the list of flight strips and calculate the optimal sequence (current time 8:00);

STEP 2: at current time 8:00am the category of runway 35L is set at 2;

STEP 3: at current time 8:00am the category of runway 35R is set at 3;

STEP 4: at current time 8:00am the category of runway 35L is set at 4;

STEP 5: at current time 8:00am the category of runway 35R is set at 2;

STEP 6: at current time 8:45am the category of runway 35L is set at 2; give to DMAN flight strips shown in Table 4.5;

STEP 7: wait current time 11:45am;

STEP 8: wait current time 14:30am;

STEP 9: wait current time 17:30am.

Call Sign	**Category**	**ETOT**	**PRT**	**ATYP**	
D9	L	09:00	N	LJ28	
D10	H	09:20	N	A310	
D11	M	09:50	N	DC3	
D12	H	10:05	H	A30B	
D13	H	10:23	N	B703	
D14	L	10:45	N	U21	
D35	H	10:45	N	B703	
D15	H	11:07	H	A310	
D16	L	01:22	N	LJ28	
D17	L	11:53	N	BE36	
D18	M	12:25	N	YS11	
D19	M	12:44	N	DC3	
D20	L	13:07	N	U21	
D21	H	13:42	H	A30B	
D22	H	14:18	N	B703	
D23	L	14:47	N	BE36	
D24	M	15:13	N	YS11	
D25	L	15:59	N	U21	
D26	M	16:28	N	DC3	
D27	H	16:57	1	A30B	
D28	M	17:28	N	YS11	
D35	M	17:28	N	DC3	
D29	L	18:01	N	BE36	
D30	H	18:34	N	A30B	
D31	L	19:05	N	LJ28	
D32	L	19:27	N	U21	
D33	H	19:53	H	A30B	
D34	M	20:07	N	YS11	

Table 4.5: Flight Strips for departure aircraft

In Annex B you can find a mapping of aircraft type (used as example in this) and runway category.

4.5.2 Test Results

STEP 1

The aircraft D3, D4, D5, D6, D7, D8 are incompatible with any of the two runways and therefore are not assigned. The aircraft D1 and D2 cannot be on the same runway.

35L		**35R**	
Call Sign	TTOT	Call Sign	TTOT
D1	8:20	D2	8:20

Table 4.6: Step 1: output of DMAN

STEP 2

The aircraft D5, D6, D7, D8 are incompatible with any of two runways and therefore are not assigned. The aircraft D1 and D2 cannot be on the same runway.

35L		**35R**	
Call Sign	TTOT	Call Sign	TTOT
D1	8:20	D2	8:20
D3	8:28		
D4	8:29		

Table 4.7: Step 2: output of DMAN

STEP 3

The aircraft D7 and D8 are incompatible with any of two runways and therefore are not assigned. The runway assigned to D1, D2, D3 and D4 may also vary without affecting the $TTOT$.

35L		**35R**	
Call Sign	TTOT	Call Sign	TTOT
D1	8:20	D2	8:20
D3	8:28	D4	8:28
		D6	8:40
		D5	8:41

Table 4.8: Step 3: output of DMAN

STEP 4

The runway assigned to flights D1, D2, D3, D4, D5 and D6 can be differ, as long as the TTOT will not change.

35L		**35R**	
Call Sign	TTOT	Call Sign	TTOT
D1	8:20	D2	8:20
D3	8:28	D4	8:28
D5	8:40	D6	8:40
D7	8:53		
D8	8:55		

Table 4.9: Step 4: output of DMAN

STEP 5

The runway assigned to flights can be differ, as long as the TTOT will not change.

35L		35R	
Call Sign	TTOT	Call Sign	TTOT
D1	8:20	D2	8:20
D3	8:28	D4	8:28
D5	8:40	D6	8:40
D7	8:53	D8	8:53
D8	8:55		

Table 4.10: Step 5: output of DMAN

STEP 6

The flight D15 is incompatible with the category runways.

35L		35R	
Call Sign	TTOT	Call Sign	TTOT
		D8	08:53
		D7	08:55
D9	09:00		
		D9	09:00
		D10	09:20
		D11	09:50
		D12	10:05
		D13	10:23
D14	10:45	D35	10:45
D16	11:22		

Table 4.11: Step 6: output of DMAN

STEP 7

35L		35R	
Call Sign	TTOT	Call Sign	TTOT
D17	11:53		
D18	12:25		
		D19	12:46
		D15	12:46
D20	13:07		
		D21	13:41
		D22	14:41

Table 4.12: Step 7: output of DMAN

STEP 8

35L		35R	
Call Sign	TTOT	Call Sign	TTOT
		D22	14:41
D23	14:47		
		D24	15:13
D25	15:59		
		D26	16:28
		D27	16:57
D28	17:28	D35	17:28

Table 4.13: Step 8: output of DMAN

STEP 9

35L		35R	
Call Sign	TTOT	Call Sign	TTOT
D29	18:01		
		D30	18:34
D31	19:05		
D32	10:27		
		D33	20:01
D37	20:07		

Table 4.14: Step 9: output of DMAN

Annex A - DMAN

Two Stage Algorithm & Multirunway Tool

The DMAN has been developed with Java language, to ensure portability across multiple operating systems, and in particular for use in environments unix-like.

An open-source library LpSolve[1] was used for the "transformation" of the mathematical model associated with the algorithm Two Stage and Multirunway in equivalent software model. This library allows to solve a problem of linear programming, integer and/nor mixed, or using the revisited simplex method, or the Branch And Bound.

The input to the DMAN is given through the cvs files that provide the set of arriving and departing flights. The operation of the tool is linked to the current time and then the departure are scheduled for time windows of about 60 minutes.

[1]LpSolve is a Mixed Integer Linear Programming (MILP) solver. http://lpsolve.sourceforge.net/

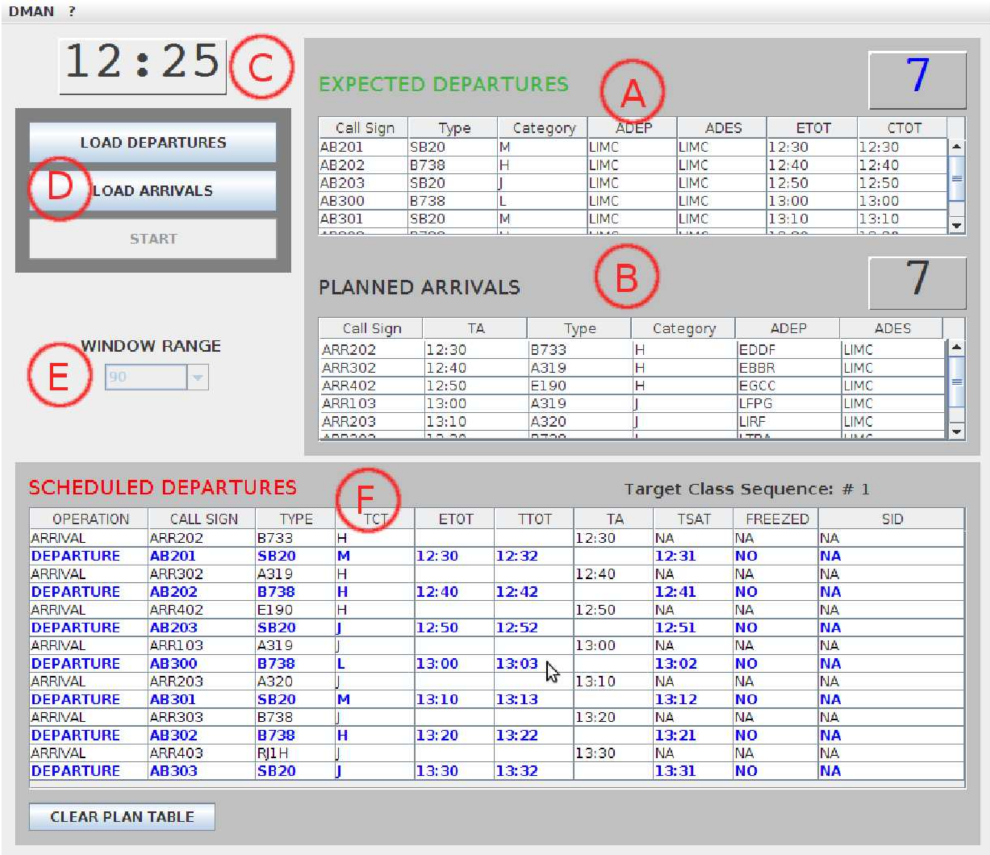

Figure 1: DMAN gui

Figure 1 shows the GUI of the DMAN, and in particular in position A the list of expected departures, in B the list of arriving flights, and in C the current time is displayed. The buttons in D allow to load, respectively, the list of departures, arrivals and start scheduling. The list box in position E allows to set the time window scheduling. Finally in the table located in position F the scheduling results are shown.

The DMAN can work together with the tool for the multirunway, that unlike the DMAN have not a GUI, but it is used through a shell on which issue commands. This tool, based on the chosen configuration for the runway, draws up a schedule of the aircraft for the active runway (not runway closed) then giving it in input to the DMAN.

Annex B - Runway Category

Each runway have a category associated (CAT. I/II/III) and only aircrafts that satisfy specific constraints can use it. These categories are established by ICAO and it are:

1. **Category I operation** – A precision instrument approach and landing with a decision height not lower than 200 feet (60 m) and with either a visibility of not less than 12 statute mile (800 m) or a runway visual range of not less than 2600 feet (800 m).

2. **Category II operation** – A precision instrument approach and landing with:

   (a) a decision height lower than 200 feet (60 m) but not lower than 100 feet (30 m);

   (b) a runway visual range not less than 1,200 feet (350 m) at RVR A; and

   (c) a runway visual range not less than 600 feet (175 m) at RVR B.

3. **Category III (A) operation** – A precision instrument approach and landing with:

   (a) a decision height lower than 100 feet (30 m), or no decision height:

   (b) and a runway visual range not less than 600 feet (175 m) at each of RVR A, RVR B and RVR C.

4. **Category III (A) operation** – A precision instrument approach and landing with:

   (a) a decision height lower than 50 feet (15 m), or no decision height:

(b) and a runway visual range less than 600 feet (175 m) but not less than 150 feet (50 m) at each of RVR A, RVR B and RVR C.

5. **Category III (C) operation** – A precision instrument approach and landing with no decision height and no runway visual range limitation.

The Table 2 lists the type of aircraft that can use a runway of a certain category.

Codice numerico	AEROMOBILI
1	Beaver DHC-2 Beechcraft Model E18, A36, 76, B55, 80, 90, 200 Britten Norman Islander 2A Cessa Model 150, 172, 180, 310, 404, 421, Turbo Dash 7 Short SC7-3 Twin Otter
2	Lear Jet 24F, 28, 29 NAMC YS-11 Short SD-3-30
3	Airbus A-300 B2 Antonov AN-24 Convair 240, 440, 580, 600, 640 DC-3, DC-4, DC-6 Fokker F-27, F-28 Hawker Siddley HS 125-400, 700 Lear Jet 24D, 35, 36A, 54, 55
4	Airbus A-310, 300 B4 BAC 111 Boeing 707, 720, 727, 737, 747, 757, 767 Canadair CL44D Caravelle Concorde Douglas DC-8, DC-9, DC-10 Ilyushin IL-18, 62 Lockheed L1011, L100, L188 Trident Tubolev TU-134, 154 Tubolev TU-134-154

Figure 2: Mapping of Runway Category and Aircraft

Bibliography

[1] ICAO, "International civil organization." OnLine.

[2] IATA, "International air transport association." OnLine.

[3] E. SpA, "Aip italia." OnLine.

[4] Eurocontrol, "Revising wake turbulence categories to gain capacity." OnLine.

[5] ENAV, *Servizi di sorveglianza ATS*. 2011.

[6] DLR, "German aerospace center." OnLine.

[7] NLR, "National aerospace laboratory." OnLine.

[8] I. Anagnostakis, J. P. Clarke, and D. Bhme, "Runway operations planning and control sequencing and scheduling," in *Proceedings of the 34th Hawaii International Conference on System Sciences*, January 2001.

[9] I. Anagnostakis and J. P. Clarke, "Runway operations planning: A two-stage solution methodology," in *Proceedings of the 36th Hawaii International Conference on System Sciences*, 2003.

[10] I. Anagnostakis and J. P. Clarke, *A multi-objective, decomposition, based algorithm design methodology and its application to runway operations planning.* ICAT-2004-5, 2004.

[11] J. Atkin, E. Burke, J. Greenwood, and D. Reeson, "Hybrid metaheuristics to aid runway scheduling at london heathrow airport," *Transportian Science*, vol. 41, pp. 90–106, February 2007.

[12] J. Atkin, E. Burke, J. Greenwood, and D. Reeson, "A meta-heuristics approach to aircraft departure scheduling at london heathrow airport," *Journal of Scheduling*, vol. 11, pp. 323–346, October 2008.

[13] S. R. R. De Jonge H. W. G., Tuinstra E. E., "Outbound punctuality sequencing by collaborative planning," in *based on paper presented at the USA/Europe ATM 2005*, (Baltimore), June 2005. Seminar.

[14] H. Hesselink and N. Basjes, "Planning aircraft movements on airports with constraint satisfaction," in *based on a presentation held at the IMA Third International Conference on Mathematics in Transport Planning and Control*, (Kardiff, UK), April 1998.

[15] P. van Leeuwen and H. van Hanxleden, "Scheduling aircraft using constraint relaxation," in *based on a presentation held at the Workshop for Workshop of the UK Planning and Scheduling Special Interest Group*, (Glaskgow, UK), December 2003.

[16] P. van Leeuwen, H. Hesselink, and J. Rohling, "Scheduling aircraft using constraint satisfaction," in *based on a presentation held at the Workshop for Functional and (constraint) Logic Programming*, (Grado, Italy), June 2002.

[17] H. Hesselink and N. Basjes, "Mantea departure sequencer: increasing airport capacity by planning optimal sequences," in *based on Proceedings of FAA/Eurocontrol ATM'98 Conference*, (Orlando (FL), USA), December 1998.

[18] G. Pataki, "Teaching integer programming formulations using the traveling salesman problem," *SIAM*, vol. 45, no. 1, pp. 116–123, 2003.