UNIVERSIDADE Ð COIMBRA

Nuno Antunes Ribeiro

Airport Capacity Management

Towards a Slot Allocation Modelling Approach Compliant with IATA Rules

PhD Thesis in Doctoral Program in Transport Systems supervised by Professor António Pais Antunes, Professor Alexandre Jacquillat and Doctor João Pita, presented to the Department of Civil Engineering of the Faculty of Sciences and Technology of the University of Coimbra.

March 2019

UNIVERSIDADE D COIMBRA

Nuno Antunes Ribeiro

AIRPORT CAPACITY MANAGEMENT

TOWARDS A SLOT ALLOCATION MODELLING APPROACH COMPLIANT WITH IATA RULES

PhD Thesis in Doctoral Program in Transport Systems supervised by Professor António Pais Antunes, Professor Alexandre Jacquillat and Doctor João Pita, presented to the Department of Civil Engineering of the Faculty of Sciences and Technology of the University of Coimbra

March 2019

Financial Support

This research work was conducted under the MIT-Portugal Program and financed by "Fundação para a Ciência e a Tecnologia" (FCT, Portugal) through the Ph.D. grant with reference PD/BD/113734/2015. Part of this work was carried out in the framework of the ASAP Project funded by the Compete 2020 with reference 02/SAICT/2017/29725.

Acknowledgements

First of all, I would like to express my enormous gratitude to my supervisor Professor Alexandre Jacquillat. Your confidence, optimism and encouragement have been key factors for the completion of this PhD thesis. I am really grateful for all your invaluable advice, guidance, inspiration, and friendship.

My sincere thanks also go to my two other supervisors, Prof. António Pais Antunes and Dr. João Pita. I am very thankful for all your support and encouragement during this PhD thesis. I highly benefited from all your scientific expertise and contacts. Thank you for making possible this unique collaboration.

Besides my supervisors, a special thanks to Prof. Amedeo Odoni, it has been a great honor to learn from your vast and rich experience. Your unique and valuable insights were priceless for bringing my research to a successful conclusion.

I would also like to thank the Airport Slot Coordinators of Portugal, in particular to Ms. Isabel Cysneiros, Patrícia Cavaco and Alexandra Joaquim. Thank you for your feedback and availability to provide data for this work. Without your support this research would never have been possible.

I am also grateful to Fundação para a Ciência e Tecnologia (FCT) and the MIT Portugal Program for the financial support to accomplish this PhD dissertation, which has allowed me to pursue my studies both in Portugal and in the United States (US).

A special thanks also goes to my colleagues and friends at the Department of Civil Engineering in Coimbra. You made everything much easier with all the fun we had together. Thanks also to all the friends I met while in the US. You had a vital role in my adaptation to the US, and highly contribute for one of the richest experiences I ever had in my life. Finally, I would particularly like to acknowledge my best friends Joana and Leandro. Despite all the physical distance, you have always been around when I most needed. I know I can always count on you.

Last but not the least, special thanks belong to my family – in particular to my Parents, my Sister, my Grandparents and my cousin João Pedro - without your support, encouragement, love and care I would not be where I am today. This success is not only mine, but also yours. Thank You!

Abstract

Air traffic demand has grown to exceed available capacity during extended parts of each day at many of the busiest airports in the world. Absent opportunities for capacity expansion, this may require the use of demand management measures to restore the balance between scheduled traffic and available capacity. The main demand management mechanism in use today is the administrative slot allocation process operated by the International Air Transport Association (IATA), which is in place at the great majority of the busiest airports outside the United States. At these airports, airlines need to be assigned *slots* by a slot coordinator to schedule their flights. Slot allocation is driven by a set of rules and priorities specified in the IATA Worldwide Slot Guidelines (WSG). These rules introduce coupling constraints across the allocation of slots at multiple times of the day and multiple days of the year, resulting in a highly complex combinatorial problem that carries enormous weight for airlines, airports and passengers. In recent years, integer programming models have been proposed to support slot allocation by minimizing deviations from the airlines' requests. However, due to the problem's complexity, these models have been only successfully implemented at small size airports (up to 50,000 flights per year).

In this thesis, we develop an original modelling approach aimed to advance existing slot allocation tools and procedures at the largest airports in the world. For that purpose, we formulate a novel integer programing model of slot allocation fully compliant with the WSG rules. The model – named Priority-based Slot Allocation Model (PSAM) – develops an original and efficient mathematical formulation that enables its implementation using exact optimization methods at airports at least with twice the size as previously considered. In order to solve the slot allocation problem at the busiest airports in the world, we also develop an algorithmic approach based on large-scale neighborhood search heuristics. The proposed algorithm combines a constructive heuristic to provide an initial feasible solution in short computational times, and an improvement heuristic that iteratively re-optimizes slot allocation by subdividing the slot requests into smaller subsets. Experimental results show that the algorithm proposed can provide optimal, or near-optimal solutions, in a few hours of computation in instances where direct implementation of PSAM with commercial solvers does not terminate after several days of computation.

The modelling approach proposed in this dissertation was implemented at three Portuguese airports, a small one (Madeira), a mid-size one (Porto) and a large one (Lisbon), using highly detailed data on airline slot requests and airport capacity constraints. Results suggest that its implementation in support of slot allocation at major slot-coordinated airports worldwide can result in flight schedules that match better airlines' requests and passenger demand. Equally important, the modelling approach developed in this dissertation can also be used to quantify the sensitivity of slot allocation decisions to the various priorities and requirements specified in the WSG. This allowed us to evaluate the impact of potential changes in the current slot allocation rules and procedures. Results obtained from many sensitivity analyses performed using PSAM show that adding even limited flexibility to the WSG can, on its own, bring considerable benefits in the short term to the slot allocation process.

Keywords: Air Transport Policy, Demand Management, IATA Slot Allocation Process, Integer Programming, Large-Scale Neighborhood Search

Resumo

A procura pelo tráfego aéreo tem aumentado nos aeroportos mais movimentados do mundo, ao ponto de superar a capacidade neles disponível durante longos períodos do dia. Na ausência de possibilidades de expansão, é essencial recorrer a medidas de gestão de procura para restabelecer o equilíbrio entre o número de voos calendarizados e a capacidade disponível no aeroporto. A principal medida de gestão de procura utilizada consiste no processo de atribuição de slots da Associação Internacional de Transportes Aéreos (IATA). De acordo com este processo, qualquer companhia aérea que pretenda operar um voo num aeroporto coordenado terá de obter antecipadamente uma permissão para a hora em que deseja que a aterragem ou descolagem do voo tenha lugar. O processo de atribuição de slots é regido por um conjunto de regras e prioridades que estão definidas nas *IATA Worldwide Slot Guidelines* (WSG). Estas regras introduzem um conjunto de restrições que tornam o problema de atribuição de slots bastante complexo. Nos últimos anos, vários modelos de programação inteira foram desenvolvidos com o intuito de auxiliar os coordenadores de slots a otimizar as suas decisões. No entanto, devido à complexidade do problema, estes modelos apenas podem ser eficazmente implementados em aeroportos de pequenas dimensões (até 50,000 voos por ano).

Nesta tese é desenvolvida uma abordagem de modelação destinada a melhorar os procedimentos utilizados na atribuição de slots em aeroportos de grandes dimensões. Com esse propósito, é formulado um modelo de programação inteira de atribuição de slots totalmente compatível com as regras especificadas nas WSG. O modelo, denominado *Priority-based Slot Allocation Model* (PSAM), utiliza uma formulação matemática eficiente que permite a respetiva implementação através de métodos exatos de otimização em aeroportos com o dobro (ou mesmo mais) das dimensões previamente consideradas na literatura. No sentido de resolver o problema da atribuição de slots em aeroportos de ainda maiores dimensões é também desenvolvido um algoritmo aproximado que se baseia em heurísticas de *large-scale neighborhood search*. O algoritmo proposto combina uma heurística construtiva, utilizada para gerar solução iniciais admissíveis, e uma heurística de melhoramento, utilizada para melhorar essas soluções iterativamente. Os resultados experimentais da aplicação destas heurísticas mostram que o algoritmo proposto fornece, em poucas horas de computação, soluções ótimas ou muito próximas das ótimas quando a implementação direta de PSAM utilizando software comercial de otimização não fornece a solução ótima após vários dias de computação.

A abordagem de modelação proposta nesta dissertação foi implementada em três aeroportos portugueses, nomeadamente um aeroporto de pequena dimensão (Madeira), um de média dimensão (Porto) e um de grande dimensão (Lisboa). Os estudos de caso analisados foram sustentados por dados detalhados referentes aos pedidos de slots feitos pelas companhias aéreas para esses aeroportos. Os resultados obtidos sugerem que a abordagem de modelação desenvolvida nesta tese pode auxiliar o coordenador de slots a tomar melhores decisões, nomeadamente encontrando soluções que se aproximam mais dos interesses das companhias aéreas e dos passageiros. Igualmente importante, a abordagem de modelação proposta nesta dissertação pode ser utilizada com o intuito de avaliar o impacto de pequenas alterações às regras existentes de atribuição de slots especificadas nas WSG. Resultados obtidos através de várias análises de sensibilidade realizadas usando o PSAM mostram que mesmo pequenas alterações podem, por si só, trazer no curto prazo consideráveis benefícios ao processo de atribuição de slots.

Keywords: Políticas de Transporte Aéreo, Gestão de Procura, Processo de Atribuição de Slots, Modelos de Programação Inteira, *Large-Scale Neighborhood Search*

Table of Contents

1 Introduction 1

1.1 Contributions 3

1.2 Dissemination 4

1.3 Outline 7

2 Research Background on Airport Capacity Management 9

2.1 Airport Capacity 9

2.1.1 Airside Capacity 11

2.1.2 Runway Capacity Models 16

2.1.3 Apron Capacity Models 19

2.1.4 Landside Capacity 19

2.2 Airport Congestion 20

2.2.1 Flight Delays 21

2.2.2 Modelling Flight Delays 23

2.3 Congestion Mitigation 23

2.3.1 Airport Expansion and New ATM Technologies 24

2.3.2 Demand Management 25

2.3.3 Air Traffic Flow Management 31

2.4 Slot Allocation Models 32

2.4.1 Zografos, Salouras and Madas Model 33

2.4.2 Pyrgiotis and Odoni Model 39

2.4.3 Jacquillat and Odoni Model 44

3 An Optimization Approach for Airport Slot Allocation under IATA Guidelines 47

3.1 Introduction 47

3.1.1 IATA Slot Allocation Process 48

3.1.2 Prior Work and Contributions 51

3.2 Case Study Data 53

3.2.1 Format of Slot Requests 54

3.2.2 Summary of the Data 56

3.3 The Priority-based Slot Allocation Model (PSAM) 58

3.3.1 Inputs....58
3.3.2 Decision Variables....60
3.3.3 Objective....61
3.3.4 Constraints....62
3.3.5 IATA Priority Constraints....65
3.3.6 Model Size....69
3.4 Model Enhancement....69
3.4.1 Formulation Strengthening....69
3.4.2 Computational Performance....73
3.5 Model Results....75
3.5.1 Sensitivity Analysis to Slot Allocation Constraints and Objectives of PSAM....75
3.5.2 Results at the Madeira Airport and the Porto Airport....81
3.6 Conclusion....87

4 A Large-scale Neighborhood Search Approach to Airport Slot Allocation....89
4.1 Introduction....89
4.1.1 Background on the Slot Allocation Process....90
4.1.2 Literature Review on Airport Slot Allocation....92
4.1.3 Literature Review on Large-scale Neighborhood Search (LNS)....94
4.1.4 Contributions....95
4.2 Slot Allocation Model....96
4.2.1 Priority-based Slot Allocation Model (PSAM): Baseline Formulation....97
4.2.2 Terminal and Apron Capacity Constraints....101
4.2.3 Experimental Setup and Direct Implementation Results....103
4.3 Algorithm Development....105
4.3.1 Constructive Heuristic....107
4.3.2 Improvement Heuristic....110
4.3.3 Consideration of IATA Priorities....117
4.4 Experimental Results....118
4.4.1 Constructive Heuristic Results....119
4.4.2 Improvement Heuristic Results....121
4.4.3 Comparison of Proposed Algorithm with Direct CPLEX Implementation....122
4.5 Sensitivity Analysis to the Algorithm Parameters....123

4.5.1 Impact of the Number of Groups in the Constructive Heuristic (G) ... 124
4.5.2 Impact of the Size of Time Window (s) and Rolling Periods (r) ... 124
4.5.3 Impact of Optimization Runtime (to) ... 127
4.5.4 Impact of Calibration Parameters ($\rho, \beta, \delta, \lambda$) ... 128
4.5.5 Discussion ... 131
4.6 Conclusion ... 132

5 Improving Slot Allocation at Level 3 Airports ... 135
5.1 Introduction ... 135
5.2 The Slot Allocation Process ... 138
5.2.1 Capacity Constraints ... 139
5.2.2 Airline Slot Requests and the Rules of Slot Allocation ... 141
5.2.3 Initial Slot Allocation Rules and Request Priorities ... 142
5.2.4 Case Study Data ... 144
5.3 Determining the Initial Slot Allocation ... 145
5.3.1 The Optimization Problem ... 145
5.3.2 An Illustrative Example ... 148
5.4 The Power of Optimization Models ... 151
5.4.1 Improving Initial Slot Allocations ... 151
5.4.2 The Value of Flexibility ... 156
5.4.3 The Impact of the Declared Capacity on Slot Allocation ... 167
5.4.4 The Issue of Fairness ... 169
5.5 Further Areas for Improvement ... 170
5.5.1 Designation as Level 3 ... 170
5.5.2 A New Class of "Level 4" Airports? ... 172
5.5.3 Network Considerations ... 173
5.6 The Larger Context ... 175
5.7 Conclusion ... 179

6 Conclusion ... 181
6.1 Summary of research ... 182
6.2 Future research plans ... 185

Index of Figures

Figure 2.1 – John F. Kennedy Airport Layout 12
Figure 2.2 – (a) Horizontal projection of flight path; (b) Trajectories of arriving and departing aircraft (Newell, 1979) 13
Figure 2.3 – Capacity Coverage Curve, Boston Logan Airport (Simpson, 1980) 15
Figure 2.4 – (a) Typical capacity envelope for a single runway (b) Good weather and poor weather capacity envelopes (de Neufville and Odoni, 2013). 18
Figure 2.5 – Nonlinear response of “stochastic delays” 22
Figure 2.6 – Internal and External Costs (Neufville and Odoni 2013) 29
Figure 2.7 – Requested slot times 37
Figure 2.8 – Allocated slot times – Zografos et al (2012) model 38
Figure 2.9 – Allocated slot times considering frequency – Zografos et al. (2012) model 39
Figure 2.10 – Allocated slot times – Pyrgiotis and Odoni (2016) model 43
Figure 2.11 – Allocated slot times, restricting flights at time period 3 – Pyrgiotis and Odoni (2016) model 44
Figure 3.1 - Demand for slots at Madeira and Porto airports for the busiest day of the Summer of 2014 57
Figure 3.2 - Evolution of the Pareto-optimal frontiers for the Madeira Airport considering different connectivity parameters 78
Figure 3.3 – Pareto-optimal frontier at the Madeira airport with different slot allocation constraints 80
Figure 3.4 – Number of slots allocated on the busiest day at Madeira and Porto airports 82
Figure 3.5 – Average daily displacement at Madeira and Porto airports per month and day of week 85
Figure 3.6 – Histogram of number of slots displaced per minutes of displacement in Madeira and Porto airports 87
Figure 4.1 – Schematic representation of the heuristic proposed 106
Figure 4.2 – Schematic representation of the improvement heuristic 111
Figure 4.3 - Average gap (%) over time for different values of s 126
Figure 4.4 - Average gap (%) over time for different values of $\boldsymbol{to}$ 128

Figure 5.1 - Outline of the slot allocation process, as performed bi-annually at Level 3 airports..138
Figure 5.2 - Coordinator's solution of the SAP for Madeira Airport and four Pareto-optimal frontiers. 149

Index of Tables

Table 2.1 – Separation Requirements in United Sates (Neufville and Odoni, 2012) 14
Table 2.2 – (a) IATA LOS description (b) IATA LOS standards (IATA, 2004) 20
Table 3.1– Madeira Airport Declared Capacities for the Summer Season of 2014........................... 54
Table 3.2 – Sample slot request codes at Madeira 56
Table 3.3 – Number of 5-minute periods where demand exceeds capacity in Madeira and Porto airports 58
Table 3.4 – Size of the Model 69
Table 3.5 –Inputs, integer solution, and linear relaxation for a simple example 70
Table 3.6 – Integer solution and linear relaxation with the modified model 73
Table 3.7 – Improvements in the model's performance with constraints (3.28) and (3.29)............. 74
Table 3.8 - Pareto-optimal solutions for the Madeira airport without and with interdependencies .. 77
Table 3.9 - Pareto-optimal solutions for the Madeira Airport with different connectivity parameters 78
Table 3.10 – Sensitivity analysis to the PSAM objectives at the Porto and Madeira airports........... 81
Table 3.11 – Coordinator and model solutions for slot allocation at Madeira and Porto airports 83
Table 4.1 – Declared capacities at the Lisbon airport... 104
Table 4.2 – PSAM-RTA results using direct CPLEX implementation 105
Table 4.3 – Inputs of the improvement heuristic and baseline values 115
Table 4.4 – Evolution of the improvement heuristic 116
Table 4.5 - Constructive heuristic results for Other slots 120
Table 4.6 – Improvement heuristic results.. 122
Table 4.7 – Solutions found with the proposed algorithm and direct CPLEX implementation 123
Table 4.8 – Sensitivity analysis to parameter G.. 124
Table 4.9 – Sensitivity analysis to parameter s 125
Table 4.10 – Sensitivity analysis to the ratio r/s 127
Table 4.11 – Sensitivity analysis to parameter to.. 127
Table 4.12 – Sensitivity analysis with respect to parameter ρ 129
Table 4.13 - Sensitivity analysis with respect to parameter δ 130
Table 4.14 - Sensitivity analysis with respect to parameter λ... 131
Table 5.1 - Declared capacities for Lisbon Airport. 140

Table 5.2 - An example of slot requests. 141
Table 5.3 - Tradeoffs among different objectives at Porto. 152
Table 5.4 - Lisbon solutions of PSAM for Summer seasons of 2014 and 2015. 154
Table 5.5 - Computational statistics for a GAMS/CPLEX implementation of PSAM. 155
Table 5.6 - Sensitivity of displacement to flexibility in connection times at Porto. 158
Table 5.7 - PSAM results at Porto when minimizing slot displacement vs. passenger displacement. 159
Table 5.8 - Total displacement results for Porto for different values of the weight α and β. 161
Table 5.9 - Sensitivity of the displacement at Porto to changes in the schedule regularity constraints. 165
Table 5.10 - Total displacement results (in minutes) for different declared capacities 167

1 Introduction

Given the fast air traffic growth, demand often exceeds available capacity at the busiest airports worldwide. This may lead to significant airport congestion, creating serious problems of delays and consequently high costs for airports, airlines and passengers. In 2017, about 20% of the flights arrived delayed more than 15 minutes to the airports in the United States (US) and in Europe (FAA, 2018; Eurocontrol, 2019). Ball et al. (2010) estimated the costs resulting from these delays for the US in a total of $32 billion. According to the most recent forecasts (IATA, 2016a), it is expected that air traffic demand will double in the next 20 years. Taking these prospects into account, it is crucial that airports find solutions aiming to mitigate the problem of airport congestion. One possible solution is by building new airports or developing new air traffic management technologies. However, these interventions are generally investment-intensive, very time consuming and sometimes even infeasible in densest urban areas. A second solution is through demand management measures, which basically involves the rescheduling of flights from the busiest hours to less demanded hours, aiming to use more efficiently the current capacities of the airports.

Demand management mechanisms fall into three main categories: *laissez faire*, market-based and non-monetary mechanisms (see Czerny et al. 2008 and Gillen et al. 2016 for reviews). *Laissez faire* mechanisms are essentially in place at US airports. In these airports, it is assumed that delay costs will be internalized by the airlines, therefore flight scheduling is not subjected to any demand management constraints. As compared to similar airports located elsewhere, it leads to higher scheduling levels but also higher levels of congestion. Market-based mechanisms aim to incentivize the airlines to schedule fewer flights at peak hours. The two prominent types of mechanisms are congestion pricing (Carlin and Park 1970; Daniel 1995; Brueckner 2002) and slot auctions (Rassenti et al. 1982, Ball et al. 2006). However, they have not been implemented in practice to date, due to their monetary transfers and potential barriers to entry and competition. Finally, non-monetary mechanisms consist of administrative mechanisms based on schedule adjustments performed by airport representatives or independent entities to reduce the number of flights scheduled at peak hours by distributing them more evenly over the day.

The IATA slot allocation process is the foremost demand management mechanism in use today. According to this process, any airline intending to operate a flight to and from a coordinated airport needs to receive access to the airport in the form of a slot. Each airport provides a value of its declared capacity, which determines the number of slots available per unit of time. The slot allocation process is applied bi-annually for the “Summer” and “Winter” seasons. For each season, the airlines submit their slot requests to a slot coordinator that allocates available slots in the most neutral, transparent and non-discriminatory way, and according to a set of rules and priorities defined in the IATA’s World Slot Guidelines (WSG) (IATA, 2017).

According to these guidelines, slot coordinators must allocate, in the first place, flights that belong to series of flights. These are regular flights that take place at least five times over a season, on the same day of the week and at the same time of the day. Among series of flights, slot coordinators need to consider three other priorities. First, they must allocate flights holding historic rights (or “grandfather rights”). An airline earns historic rights if a series of flights is operated at least 80% of the time in the previous equivalent season. Second, they must allocate historic flights that for some reason requested a change, such as a re-timing or the use of a larger aircraft. Finally, slot coordinators must reserve at least 50% of the remaining available slots to new entrants. An airline is considered a new entrant if it holds less than five slots in a certain day of the season after the requested slots are allocated. Following the allocation to new entrants, any remaining slots are then allocated to flight requests that do not belong to any of the three priority classes. The WSG include many more specifications in addition to these priority rules.

In the Summer season of 2017, 177 airports in the world were designated as slot coordinated. Despite their relatively small number (only about 4.5% of the roughly 4,000 airports in the world with scheduled airline service), these airports play a truly critical role in global air transport. In 2016, they served approximately 3.15 billion airport passengers – or about 43% of the worldwide total of 7.4 billion and about 55% of the roughly 5.75 billion passengers outside the United States. Besides, they practically include all the major connecting hubs outside the US. For all these reasons, the process and rules under which access to slot coordinated airports is determined carry enormous economic and regulatory implications for the global air transport sector.

Nowadays, slot coordinators are assisted in their activity by specialized software (e.g., PDC SCORE). This software handles automatically the flight requests made by airlines providing instantaneous information about the priority class of each flight, as well as about the availability of slots. However, the slot requests are typically treated in an *ad hoc* basis, which provides limited visibility on the whole set of slot requests and their interactions. Recently, some optimization models have emerged in the literature with the purpose of supporting slot coordinators to better allocate slots (Zografos et al., 2012, Pyrgiotis and Odoni, 2016 and Jacquillat and Odoni, 2015a). These models have shown that there is great room for improvements on the current slot allocation process. Despite the positive results, these models are not yet totally in accordance to the WSG as well as to some important constraints faced by the airports. Moreover, existing models of slot allocation can only be efficiently implemented at small-size airports, which significantly limits their applicability at slot-coordinated airports.

Accordingly, the main goal of this PhD dissertation is to develop a modeling approach aiming to assist slot coordinators in the decisions they make during the slot allocation process. This approach must be compliant with IATA Worldwide Slot Guidelines and take into account all the airport operational and regulatory constraints in order to be totally efficient when applied to the real-world context. This modeling approach is expected to enhance the effectiveness of slot allocation based on existing IATA rules. At the same time, we also aim to provide recommendations concerning potential changes in the current IATA slot allocation process.

1.1 Contributions

In this dissertation we develop an original optimization tool compliant with existing rules of the IATA's Worldwide Slot Guidelines, aimed to enhance slot allocation in practice at the busiest airports in the world. Its application in support of slot allocation at major slot-coordinated airports worldwide can result in flight schedules that match airlines' requests and passenger demand more effectively than existing approaches.

The main contributions of this dissertation can be summarized as follows:

- *Formulation of an integer programming model of slot allocation that captures all IATA Worldwide Slot Guidelines:* The optimization model – named Priority-based Slot Allocation Model (PSAM) – proposes an original and efficient mathematical formulation to optimize slot allocation decisions. This formulation permits the implementation of the model using exact optimization methods in reasonable computational times at mid-size airports. Specifically, PSAM was implemented at Porto airport, which operates nearly 100,000 aircraft movements per annum. This volume of traffic is twice as large as that at the busiest airports previously considered in the literature (Zografos et al., 2012). Comparisons with real-world data suggested that the PSAM can improve significantly slot allocation outcomes, and thus mitigate the costs of schedule coordination to the airlines and other airport stakeholders.

- *Development of an original algorithm to solve the slot allocation problem at the largest schedule-coordinated airports:* The PSAM can only be successfully implemented with exact methods at small- and mid-size airports – the slot allocation problem remains too complex to be solved exactly at the larges schedule-coordinated airports. Therefore, I have developed an original heuristic algorithm based on large-scale neighborhood search to solve PSAM. The proposed algorithm combines a *constructive heuristic*, which provides an initial feasible solution in short computational times, and an *improvement heuristic*, which iteratively re-optimizes slot allocation by subdividing the slot requests into smaller subsets. The algorithm was implemented at Lisbon's Airport (LIS), one of the top-20 busiest airports in Europe. In test instances where commercial optimization solvers cannot find the optimal solution after 7 days of computation, our algorithm can consistently generate solutions within 0.1% of the optimum in a few hours, thus improving the solutions from commercial solvers both in terms of quality and computational times
- *Assessment of the IATA Worldwide Slot Guidelines and proposals for improvement.* PSAM can be used to optimize slot allocation outcomes for any given set of rules and procedures, but also to evaluate, through *what-if* sensitivity analyses, the impact of changes in the slot allocation policies on airport scheduling and operating performance. The results obtained show that adding even limited flexibility to the WSG can, on its own, bring considerable benefits (i.e. better matching airlines' scheduling requests) in the short term to the slot allocation process.

Over the different stages of this dissertation, we have worked with the collaboration of the slot coordinators from Portugal (ANA) and Brazil (ANAC), which provided us with data and feedback on the methodologies being used.

1.2 Dissemination

Most of the research developed throughout this PhD dissertation has been presented and discussed in several international and national conferences of transportation and/or optimization between 2016 and 2018. In total this work was presented at 15 conferences, having received two international awards: (i) an honorable mention from the INFORMS Aviation Applications Section for the best student presentation competition in the INFORMS Annual Meeting in 2017; (ii) the Anna Valicek Silver Medal from the Airline Group of the International Federation of Operational Research Societies (AGIFORS) in the AGIFORS Annual Symposium in 2018, recognizing "original and innovative research in the application of operations research to airline and/or airline related business problems". See below the list of conferences where this work was presented.

- Ribeiro, N. A., Jacquillat, A., Antunes, A. P., Odoni, A. R., and Pita, J. P. "A Large Neighborhood Search Approach to Airport Slot Allocation", INFORMS Annual Meeting, Phoenix, Arizona (USA), November 2018;
- Ribeiro, N. A., Jacquillat, A., Antunes, A. P., Odoni, A. R., and Pita, J. P. "A Large Neighborhood Search Approach to Airport Slot Allocation" CITTA 11th Annual Conference, Coimbra, Portugal, October 2018
- Ribeiro, N.A., Jacquillat, A. and Antunes, A.P. "Important Issues on Airport Slot Allocation", ASAP research project kickoff meeting, Lisbon, Portugal, October 2018
- Ribeiro, N.A., Jacquillat, A., Antunes, A.P., Odoni, and A.R., Pita, J.P. "An Optimization Approach for Airport Slot Allocation Under IATA Guidelines", AGIFORS 58th Annual Symposium, Airline Group of the International Federation of Operational Research, Tokyo, Japan, October 2018.
- Ribeiro, N. A., Jacquillat, A., Antunes, A. P. and Odoni. "Optimizing Slot Allocation at Level 3 airports", OR 60 Annual Conference, The Operational Research Society, Lancaster, UK, September 2018
- Ribeiro, N. A., Jacquillat, A., Antunes, A. P. and Odoni. "A large neighbourhood search heuristic for solving the airport slot allocation problem", APDIO XIX Congress, Portuguese Association of Operations Research, Aveiro, Portugal, September 2018
- Ribeiro, N. A., Jacquillat, A., Antunes, A. P., Odoni, A. R., and Pita, J. P. "A Heuristic Algorithm for Solving the IATA Slot Allocation Problem at Large-Size Airports", 15th Annual Transports Study Group Conference, Fátima, Portugal, February 2018.
- Odoni, A.R., Ribeiro, N.A., Jacquillat, A. Pita, J.P., A.R., and Antunes, A.P. "On the Allocation of Airport Slots", Martin Kunz Lecture, European Aviation Conference, Dublin (Ireland), November 2017.
- Ribeiro, N.A., Jacquillat, A. Pita, J.P., Odoni, A.R., and Antunes, A.P. "An Optimization Approach for Airport Slot Allocation Under IATA Guidelines", INFORMS Annual Meeting, Houston, Texas (USA), October 2017.
- Ribeiro, N.A., Jacquillat, A. Pita, J.P., Odoni, A.R., and Antunes, A.P. "An Optimization Approach for Airport Slot Allocation Under IATA Guidelines", 21st ATRS (Air Transport Research Society) World Conference, Antwerp (Belgium), July 2017.

- Ribeiro, N.A., Jacquillat, A., Antunes, A.P., Odoni, and A.R., Pita, J.P. “Airport capacity management under IATA guidelines: modeling approach and real-world application”, Mathematics of Complex Systems: from precision medicine to smart cities - CoLaB Workshop, Coimbra (Portugal), December 2016.
- Ribeiro, N.A., Jacquillat, A. Pita, J.P., Odoni, and A.R., Antunes, A.P., “A Model for Airport Schedule Coordination based on the IATA Guidelines”, INFORMS Annual Meeting, Nashville, Tennessee (USA), November 2016.
- Ribeiro, N.A., Jacquillat, A. Pita, J.P., Odoni, A.R., and Antunes, A.P. “An Airport Slot Allocation Model Compliant with IATA Guidelines”, 5th Symposium arranged by European Association for Research in Transportation (hEART), Delft (Netherlands), September 2016.
- Ribeiro, N.A., Jacquillat, A. Pita, J.P., Odoni, A.R., and Antunes, A.P. “An Airport Slot Allocation Model Fully Compliant with IATA Rules”, Poster presented at the 6th MIT Portugal Program Conference, Braga, Portugal, June 2016.
- Ribeiro, N.A., and Antunes A.P. “Towards a Slot Allocation Model Fully Compliant with IATA rules”, 13th Annual Transports Study Group Conference, Figueira da Foz, Portugal, January 2016.

The work developed in this thesis was also submitted to top peer-reviewed journals of transportation, namely Transportation Research Part B – Methodological (already published), Transportation Research Part A: Policy and Practice (under review) and Transportation Science (under review). The three papers are presented in this thesis as Chapters 3, 4 and 5.

- Ribeiro, N. A., Jacquillat, A., Antunes, A. P., Odoni, A. R., and Pita, J. P. “An optimization approach for airport slot allocation under IATA guidelines”. Transportation Research Part B: Methodological, 112, 132-156, *2018*.
- Ribeiro, N. A., Jacquillat, A., Antunes, A. P., and Odoni, A. R. “Improving Slot Allocation at Level 3 Airports”, Transportation Research Part A: Policy and Practice, *under review*.
- Ribeiro, N. A., Jacquillat and A., Antunes, A. P. “A Large-scale Neighborhood Search Approach to Airport Slot Allocation”, Transportation Science, *under review*.

Finally, note that a part of this thesis, and corresponding thesis proposal, served as basis to an application to a FCT scientific project. The project was approved and will be taking place over the next three years (ref. 02/SAICT/2017/29725). The kick-off meeting of this research project took place on October 2018 in Lisbon, and had the participation of high level managers from the main industry stakeholders (IATA, ACI, WWACG, Eurocontrol, etc.) and leading international researchers.

1.3 Outline

This thesis is divided into 6 chapters. Chapter 1 introduces the thesis dissertation. Chapter 2 provides a research background on airport capacity management to provide the reader with adequate background for the subsequent chapters. Chapter 3 proposes a novel multi-objective integer programming model to optimize slot allocation decisions, while complying with the complex set of priorities and requirements specified by the WSG. Chapter 4 develops an original algorithm based on large-scale neighborhood search to solve the slot allocation problem at the largest slot-coordinated airports. Chapter 5 discusses important issues on airport slot allocation and evaluates the benefits that can be attained from possible small changes in the current guidelines and procedures. Chapter 6 concludes this dissertation.

Note that the chapters 3 to 5, are all written in the format of scientific papers. As described in Section 1.3 these papers were published or submitted to peer-reviewed scientific journals. Since these papers cover the same research topic there might be some repetition of concepts and information from chapter to chapter. We decided to keep this information, since in many cases it might be helpful to recall the reader of certain details to support their understanding on specific methodologies and assumptions adopted throughout the thesis. Additionally, this dissertation structure allows the reader to read each chapter independently.

2 Research Background on Airport Capacity Management

2.1 Airport Capacity

Airports are divided into two subsystems, specifically the landside and airside areas. The airside area comprises the runway system, the taxiway system and the apron complex. The landside area entails the ground access system and the passenger (or cargo) terminal buildings. Every day these subsystems need to accommodate different types of entities, such as aircraft that are landing and taking off, passengers that are arriving and departing, and cargo that needs to be sent to their final destination. The ability to handle these different types of entities defines the overall airport capacity (Janic, 2000).

Generally, the runway system is the bottleneck infrastructure of an airport system, mainly because it is extremely difficult and time-consuming to increase substantially their available capacity at major airports. In fact, new runways require acquisition of large amounts of additional land and have significant environmental and economic impacts, forcing long and difficult review and approval processes. By contrast, the capacity of other airport elements (such as terminals, road access, apron, etc.) may be increased more easily in order to at least equal the capacity of the runway system. Thus, airport capacity is typically defined as the expected number of aircraft movements that can be operated per unit of time at an airport under continuous demand. However, as already stated, other measures of capacity can be also important to characterize the operating capabilities of sub-components of airport systems, such as the passengers processing capacity at the terminal areas, the number of aircrafts than can be parked in the apron areas, and so on (Wells and Young, 2004).

There are two basic capacity concepts: throughput and practical capacity. While practical capacity considers a certain acceptable level of service when computing capacity (often measured as delay), throughput capacity does not take this into account, representing the full capacity of a facility. These two concepts are accepted and used in most relevant literature (de Neufville and Odoni 2013; Janic 2009; Ashford and Wright 1992; Hockaday and Kanafani, 1974). In addition to these concepts, capacity may also be defined as static or dynamic. The former represents the maximum number of entities that can be simultaneously accommodated at a facility, taking into account a set of given conditions. Typically, it represents the potential storage of a facility. The later represents the maximum service rate, which is expressed by the number of entities that can be served in a given unit of time under conditions of constant demand for service (de Neufville and Odoni 2013; Janic, 2000).

Taking into account the several definitions of airport capacity presented, some measures are described below (de Neufville and Odoni 2013):

- *Maximum throughput capacity or saturation capacity (MTC)* indicates the average number of movements that can be performed on the runway system in 1 hour in the presence of continuous demand, while adhering to all separation requirements imposed by the Air Traffic Management (ATM) system;
- *Practical Hour Capacity (PHCAP)* is defined as the expected number of movements that can be performed on the runway system in 1 hour, with an average delay of 4 minutes per movement. Usually PHCAP represents approximately 80 to 90 percent of the MTC.
- *Sustained Capacity* indicates the number of movements per hour that can be reasonably sustained over a period of several hours. Usually, MTC cannot be sustained for more than one or two consecutive hours. Hence, sustained capacity is considered a more realistic target, since airports are often subjected to continuous heavy demand for several hours or even entire days. The sustained capacity is set to approximately 90 percent of MTC for runway configurations with high values of MTC and almost 100 percent for configurations with low values of MTC:
- *Declared Capacity* is defined as the number of movements per hour that an airport can accommodate at a reasonable level of service. Delay is typically used as the principal indicator of level of service. There is no generally accepted definition of declared capacity and no standard methodology for setting it. In most cases declared capacity is set close to approximately 85% to 90% of the MTC. However, there are some instances where the passenger terminals and the apron capacity are also used in the definition of the declared capacity. Note that, the declared capacity is commonly considered the capacity indicator used in the slot allocation process (the main research focus of this dissertation).

The measures presented here are often estimated considering a mix of 50% arrivals and 50% departures, which sometimes may not be realistic. Thus, when needed, these capacities can be estimated considering other mixes of movements. A way to represent the airport capacity for several mixes of movements is using a capacity envelope. We present this concept in Section 2.1.2.

Estimating airport capacity accurately is crucial to airport planning and management. On one hand, if capacity is overestimated, demand may exceed the real capacity of the airport, which will lead to over-scheduling, resulting in queues, delays and a low service level (See Section 2.2). On the other hand, underestimating capacity may lead to refusing unnecessarily flights requested by the airlines, or barring entry to new competitors, which will affect the revenues of the airport and its competitiveness (See Section 2.3).

2.1.1 Airside Capacity

The airside system is planned, designed, and managed to accommodate the volume and type of aircraft that use the airport. The most important facilities that are located on the airside area are the runway system, the taxiway system and the apron complex. The runway system is where aircraft land and take off. It is often considered the most critical facility of an airport, since the number of runways, their layout and length will determine the kinds of aircraft that can use the airport as well as the number of aircraft that can be accommodated in any given period. The taxiway system connects runways with aprons, terminals and other facilities. Busy airports typically construct high speed or rapid-exit taxiways to allow aircraft to leave the runway quickly. This allows other aircraft to land or take off in a shorter space of time. The apron complex is the area where aircraft are parked, loaded or unloaded, refueled or boarded. Occasionally, the apron capacity may be a constraining factor on the overall airside capacity.

There are many factors affecting the airside capacity, and in particular the runway capacity. In the literature there is an extensive classification of these factors. The most relevant factors are described below (de Neufville and Odoni, 2013; Janic, 2009; Ashford and Wright 1992; Newell, 1979).

a) Number and geometric layout of runways

The capacity of an airport depends in large measure on the number of runways available and their interactions. The greater the number of runways, the higher the capacity and flexibility of a runway system. The runways may be designed in parallel or with different directions. When they are parallel, three possible layouts may occur depending on their separation: “close parallel” runways, “independent” runways or “medium-spaced” runways. In the case of “close parallel runways”, flight operations, such as arrivals and departures, cannot take place at the same time on each runway, they must be made one at a time. In the case of “independent runways”, there are no restrictions and the arrivals and departures may take place at the same time in each runway. Finally, when the runways are “medium-spaced”, arrivals at the same time on both runways are not permitted, however independent departures or independent segregated parallel operations (arrival and departures) are allowed. Layouts with different runway directions ensure greater flexibility, since they provide an adequate coverage for different wind directions (crosswinds).

Airports with several runways often cannot have all their runways active simultaneously. Depending on several factors, - level of demand, weather conditions, mix of movements and noise restrictions - the set of runways selected at any given time is called a configuration. An airport may have several configurations. As an example, John F. Kennedy (JFK) airport in United States has four runways and eight possible configurations with two or three active runways (Jacquillat et al., 2016). In Figure 2.1 we present the layout of the JFK airport.

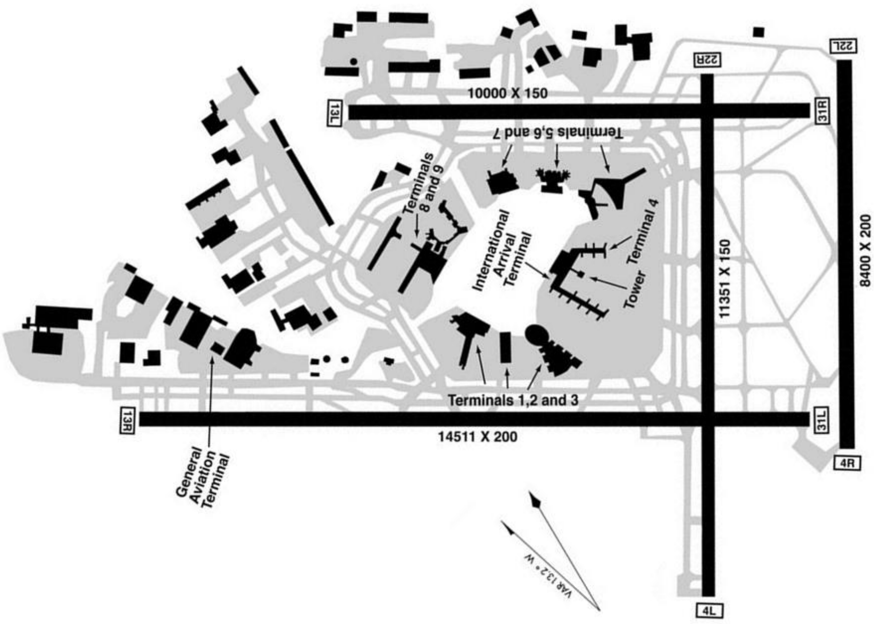


Figure 2.1 – John F. Kennedy Airport Layout

Another important feature of a runway is its length, since it defines the kind of aircraft that can use the airport.

b) Aircraft Mix and Performance

The capacity of a runway can vary greatly with the types of aircraft using it. The characteristics of the aircraft – size, aerodynamics, propulsion and braking performance – affect the capacity of the runways. Indeed, larger aircraft will require longer runways and consume more time on landing and taking off. Another problem related with aircraft performance is wake vortices. Airborne aircraft generate a wake vortex that can persist for 2 minutes or even longer under some weather conditions. The strength of the vortex increases with the weight of the aircraft, which affects the separation requirements set out by the Air Traffic Management (ATM) system. Pilot training and experience is another important factor affecting runway capacity. If air traffic controllers notice that a pilot is inexperienced or has difficulty understanding instructions, they slow down operations in order to allow additional margins of safety, thus reducing airport capacity.

c) Air Traffic Management (ATM) System Performance

As already noted, capacity is highly variable, since it depends on several factors. The ATM system is supported by personnel and software that make it possible for airports to adapt their capacity efficiently to such factors variations throughout the day. Hence, the elements of the ATM system are crucial in the assessment of the airport capacity.

First, a high-quality ATM system should have well-trained and motivated air traffic controllers. Furthermore, they should be supported by good decision support systems. Improvements in aircraft surveillance, navigation, and communication equipment will certainly increase the ability of pilots and air traffic controllers to maintain high capacity levels during all unexpected situations.

One challenge that air traffic controllers need to face every day is related with the mix of aircraft movements. In fact, the assignment of arrivals and departures to runways is often a difficult task. To simplify the task, air traffic controllers often choose to assign landings and takeoffs to separate runways. However, this is not necessarily an optimal or even feasible solution, since departures tend to require less time on the runway and terminal airspace than arrivals (see Figure 2.2). Hence, using separate runways for landings and takeoffs may create a serious imbalance, creating large delays for arrivals in comparison with departures.

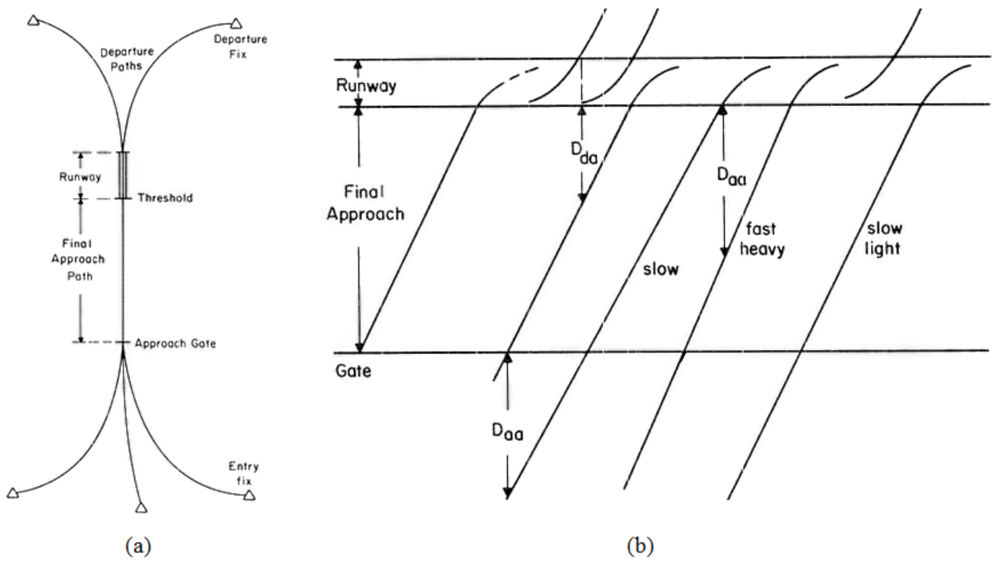


Figure 2.2 – (a) Horizontal projection of flight path; (b) Trajectories of arriving and departing aircraft (Newell, 1979)

In figure 2.2 (b), a possible sequence of aircraft movements is illustrated. As can be seen, sometimes it may be useful to insert departures between arrivals taking advantage of free time that may exist between two consecutive arrivals. However, in order to simplify operations, air traffic controllers usually try to assign aircraft to runways considering waves of arrivals followed by waves of departures. It is the responsibility of the air traffic controllers to manage both types of movements aiming to ensure a good balance between landings and departures.

Another important issue concerning the ATM system is related to the separation requirements between aircraft. These separation requirements prevent aircraft from coming closer to each other than prescribed, ensuring that only a single aircraft is occupying the runway at any time. The separation rules are defined by the ATM system operator, and depend on several factors, such as the mix of movements (Arrival – Arrival, Departure – Departure, Arrival – Departure, Departure – Arrival), weather, aircraft speed and type of aircraft (problem of the wake vortex). A table with the separation requirements in USA is presented below taking into account the mix of movements and type of aircraft (Heavy, Large or Small). It is clear that the more restrictive the separation rules, the lower will be the capacity of a runway.

Table 2.1 – Separation Requirements in United Sates (Neufville and Odoni, 2012)

Arrival followed by arrival				**Departure followed by departure**			
Trailing Aircraft (nautical miles)				Trailing Aircraft (seconds)			
	H	L	S		H	L	S
H	4	5	5 or 6	H	90	120	120
L	2.5 or 3	2.5 or 3	3 or 4	L	60	60	60
S	2.5 or 3	2.5 or 3	2.5 or 3	S	45	45	45
The trailing aircraft cannot touch down before the runway is clear							
Arrival followed by departure - Clearance for takeoff run of the trailing departure is granted after the preceding landing is clear of the runway							
Departure followed by arrival - The trailing arrival on final approach must be at least 2 nmi from runway when departing aircraft begins its takeoff run, and cannot touch down until departing aircraft is clear of the runway							

d) Weather

The weather is clearly one of the main factors affecting airport capacity and the one that is the least predictable. Heavy fog, snow, strong winds or icy runway surfaces reduce the ability of an airport to accommodate aircraft and may even close the airport completely.

Depending on the prevailing weather conditions, airports may operate in visual meteorological conditions (VMC) or in instrumental meteorological conditions (IMC). This affects the separations requirements defined by the ATM System. In the United States (US) this is particularly relevant, since pilots are requested to use visual flight rules (VFR) in case of good weather conditions. Pilots under VFR are free to maintain visually a safe separation from preceding aircraft during the final approach to the runway. This practice results in higher capacities per runway than can be achieved with strict adherence to instrument flight rules (IFR).

In the rest of the world, VFR are not often applied at airports and consequently runways cannot achieve capacities as high as in USA. Nevertheless, some busy European airports are already allowing VFR operations under certain conditions, which has led to an increase in their capacities.

Another important factor is the direction and strength of the prevailing winds. Typically, airport designers use wind statistics to determine the orientation of the runways that should be provided to achieve adequate crosswind coverage. Crosswinds have a great impact on the selection of the runway configuration to operate on each day. Sometimes it may be impossible to use configurations with high capacity because of its presence.

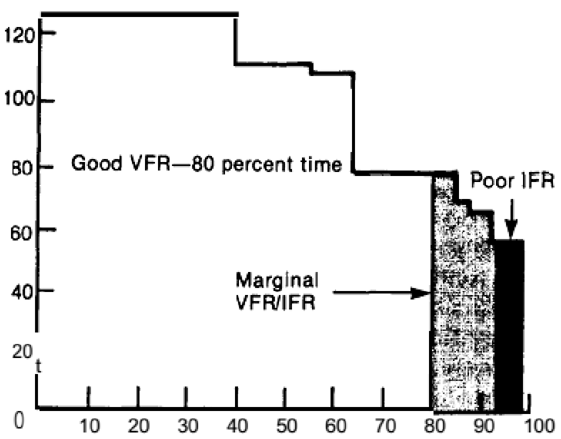


Figure 2.3 – Capacity Coverage Curve, Boston Logan Airport (Simpson, 1980)

A very convenient way to summarize the range of airport capacities by frequency is through the use of a capacity coverage curve (CCC). This diagram assumes that the runway selected at any given time is the one providing the highest capacity under the prevailing weather conditions. Figure 2.3 shows the CCC for Boston Logan Airport in USA. It can be seen that the highest airport capacity is about 120 movements per hour (considering a mix of 50% arrivals, 50% departures). This capacity is available 40% of the year. During the remaining 60% of the year, 40% is still in VMC, which means the changes in runway configuration are likely related to crosswinds, and 20% in IMC, which means pilots are requested to use IFR. There is a small percentage of the year when poor visibility, ceiling and snow completely close the airport.

A flat CCC means more predictable runway capacities, which allows a more effective utilization of airport resources and facilities.

e) Environmental Considerations

Another important factor constraining airport capacity is environmental considerations and, in particular, noise impacts. In fact, aircraft noise has made airports unpopular with their neighbors and has forced airports to operate with constraints imposed by noise mitigation procedures. The primary constraint is the limitation of flights during the night. Moreover, there may also be noise constraints during the day that may limit the number of flights that can be scheduled per hour. Finally, if weather permits, air traffic controllers may select runway configurations in a way that distributes noise more evenly in the neighborhood of the airport.

One method to reduce noise is introducing quieter aircraft. Many airlines are already re-equipping old aircraft with quieter engines.

f) Taxiway and Apron Capacity

Usually, the taxiway system and the apron complex are not the most restrictive components of the airside system. However, it is crucial to ensure that these components are well designed and managed. The taxiway system may have some "hot points" where problems may occur. Taxiway intersections, short taxiway segments between two intersections, points where taxiing aircraft must cross an active runway, and locations where high-speed runway exits merge with taxiways can all be potential local "hot points". These points can be identified easily by airport operators, which have the task to try to anticipate and prevent expected delay problems. Occasionally, the apron complex is a constraining factor on the overall airport capacity. It is up to the airport to manage properly aircraft stands and parking operations.

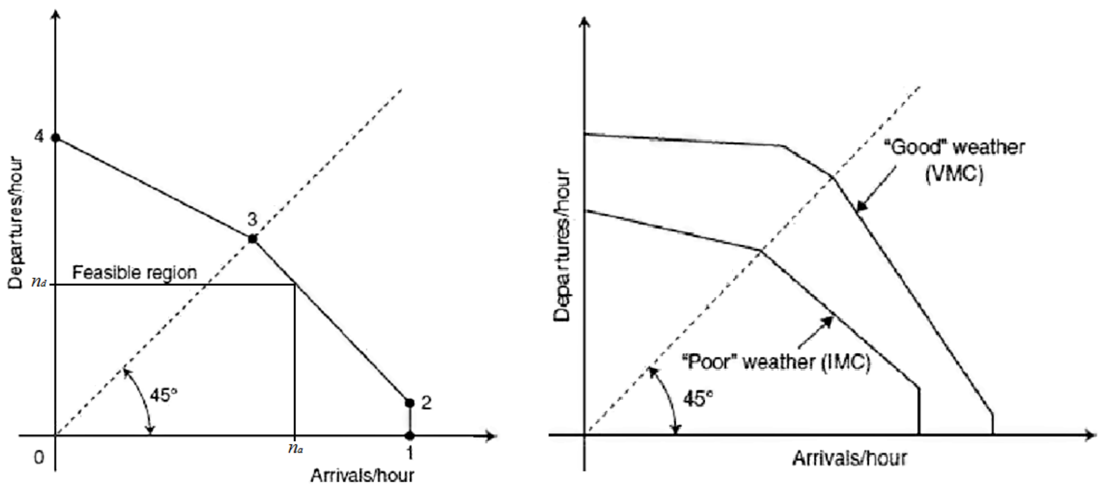
2.1.2 Runway Capacity Models

Accurate runway capacity estimation is crucial for the proper functioning of an airport system. To this end, there is a comprehensive set of models that estimate runway capacity. The earliest model of runway capacity was developed by Blumstein in 1959 (Blumstein, 1959) and is still today the basis for other runway capacity models. This model estimates the capacity of a single runway considering that just arrivals can take place. Blumstein's model assumes that aircraft use a common path in their final approach to the runway, as illustrated in Figure 2.2 (a). They must maintain a safe longitudinal distance from each other as specified by the ATM separation requirements and must be safely out of the runway before the next landing (see Table 2.1).

Several extensions and improvements to Blumstein's model have been made. The first were proposed by Harris (1972) and Odoni (1972). They adapted the model to include departures and mixed operations and introduced stochasticity. For this purpose, they considered the approach speed to the runway and the occupancy time as random variables with associated probability distributions. Furthermore, the distance between successive aircraft on final approach also became a random variable whose probability distribution depends on the ATM separation requirements and other characteristics. After that, Hockaday and Kanafani (1974) incorporated into the model the case of multiple runways of various configurations and included the effect of wake turbulence created by large aircraft. They also studied the possibility of stretching the separation between arrivals so that a departure can take off during the time between two arrivals, as explained in Figure 2.2.

During the 1970's an airfield capacity model was developed by a consortium composed by Peat, Warwick, Mitchell and Company (PMM&Co) and McDonnell Douglas Automation (MCAUTO). Further, the software was modified by the Federal Aviation Administration (FAA), the national aviation authority of the United States, and became known as the FAA Airfield Capacity Model (Swedish, 1981). Today, after some improvements, the model has the ability to estimate the hourly capacity of 15 common runways configurations ranging from a single active runway to four active runways (Odoni et al., 1997).

Meanwhile, other more detailed models addressing all these possibilities were developed. Some of these models are the LMI model first presented by Lee at al. (1997) and more recently the MACAD model, an integrated airside model, which provides estimates of the capacity and associated delays for every airfield element (Andreatta et al, 1999; Stamatopoulos et al, 2004). The main output of these models is the estimation of the runway capacity envelope, first presented by Gilbo (1993) and recently extended by Simaiakis (2012) (see Figure 2.4).



(a) (b)

Figure 2.4 – (a) Typical capacity envelope for a single runway (b) Good weather and poor weather capacity envelopes (de Neufville and Odoni, 2013).

According to Gilbo (1993), arrival and departure capacities may be represented by a capacity curve. The boundary of the curve indicates the set of capacities that can be sustained by the runway for each mix of arrivals and departures and is called the runway capacity envelope. Figure 2.4 (a) presents a typical capacity envelope of a single runway and can easily show the relationships between arrivals and departures. For example, if there are n_a landings on the runway, the number of departures can only be n_d or less, which means that any point outside the envelope is considered infeasible.

The estimation of the runway capacity envelope can be made empirically or theoretically. The first method uses real observed data on the number of arrivals and departures. However, the data should reflect the airport performance at or near its capacity limit. The second method uses models, such as the Blumstein’s model and its extensions, to estimate runway capacity for several possible mixes of arrivals and departures.

Note that capacity envelopes may be drawn for each runway or runway configuration. It is also common to show the capacity envelope that applies to each weather conditions (VMC or IMC). In poor weather conditions the curve will be more restrictive, as can be seen in Figure 2.4 (b). Usually runway capacity envelopes are computed for 1 hour, however they may also be computed for other time intervals, such as 15 minutes.

2.1.3 Apron Capacity Models

Some apron capacity models can be found in the literature based on the same approach applied by Blumstein for computing runway capacity. They calculate dynamic apron capacity based on three parameters: apron layout (number of stands); use strategy (aircraft type or user) and scheduled occupancy time (time that an aircraft is scheduled to spend at the stand). Two different models are typically observed in literature. One assumes that all aircraft can use all the stands available. The other assumes restrictions on stand use by aircraft type. The application of these models is presented and exemplified in several relevant literature (Horonjeff et al., 2010; Ashford et al., 2011; Mirkovic 2011; Neufville and Odoni 2013; Mirkovic and Tosic, 2014).

Recently, Mirkovic and Tosic (2014) improved these models, considering not only size restriction but also user restrictions. The users can be different airlines (e.g. based or non based airlines) or different flights with respect to their origin/destination (e.g. domestic and international flights). They also introduced the concept of apron capacity envelope in which, as in the case of the runway capacity envelope, the apron capacity is illustrated by a curve. That curve represents the apron capacity for different combinations of type of users.

2.1.4 Landside Capacity

The airport landside system is responsible for ensuring a smooth flow of enplaning and deplaning passengers (or freight) within the airport. It might be divided into two subsystems: the airport ground access system, which is composed by a set of links and nodes that enable the airport to be connected to the outside world (roads, railways etc.); and the terminal building, which comprises a set of services aiming to prepare passengers to pass from the ground to the air transport mode (Janic, 2000).

The landside elements may be subdivided into three classes: Processing Facilities, which are responsible for processing passengers and their baggage; Holding Facilities, which represent locations where passengers wait, for instance for check-in opening or flight boarding; and Flow Facilities, which are used by passengers to move among the landside elements or to the terminal (TRB, 1987).

The aggregate landside capacity is determined from the individual capacities of each one of these landside elements and it depends on the level of service (LOS) required by airport managers. LOS is a qualitative measure, which represents the passenger’s perception of the quality and service conditions of one or more airport facilities. However, it is often measured in terms of more quantitative metrics such as waiting time, processing time, walking time, crowding, etc.

The establishment of LOS targets is crucial in landside planning, since they will have implications for airport costs and its "external image". If, on one hand, a high LOS is specified, the available capacity may not be sufficient to satisfy it; on the other hand, a low LOS may lead to loss of passengers and business opportunities (Brunetta et al. 1999 and Ashford et all, 2011).

In order to take these considerations into account a good capacity estimation and selection of the LOS is crucial to landside planning. One of the methods most often used worldwide is the IATA method. With this method, airport managers select the LOS on the basis of tables such as those in Table 2.2. Table 2.2a provides a description of the LOS classifications and Table 2.2b the LOS standards associated with each LOS.

Table 2.2 – (a) IATA LOS description (b) IATA LOS standards (IATA, 2004)

LOS	Description: Flow	Description: Delay	Description: Comfort
A. Excellent	Free	None	Excellent
B. High	Stable	Very Few	High
C. Good	Stable	Acceptable	Good
D. Adequate	Unstable	Acceptable for short time	Adequate
E. Inadequate	Unstable	Unacceptable	Inadequate
F. Unacceptable	Total system breakdown	Unacceptable	

(a)

Sub-system	LOS standards (square meters per occupants)					
	A	..B. .	..C. .	..D. .	E ..	..F..
Check-in queue area	1.8	1.6	1.4	1.2	1.0	*Total*
Wait/circulate	2.7	2.3	1.9	1.5	1.0	*System*
Hold room	1.4	1.2	1.0	0.8	0.6	*Breakdown*
Bag claim area	2.0	1.8	1.6	1.4	1.2	
Government inspection	1.4	1.2	1.0	0.8	0.6	

(b)

Although being widely used by airports, the IATA method just provides gross estimates of capacity that are mostly static and deterministic. Over the years, several tools have been developed, most of them belonging to the class of *queuing theory* models. They were mostly applied to the modelling of processing facilities. Examples of applications are found in Rallis (1958, 1963 and 1967), Lee (1966), Barbo (1967), Horojneff (1967) Newell (1971) and Tosic et al. (1983), which applied this concept to problems of check-in procedures, security screening and baggage claim.

More recently, Brunetta et al. (1999) developed an aggregate model to estimate landside capacity, called Simple Landside Aggregate Model (SLAM). The model is strategic and its objective is not to provide a thorough analysis of each landside facility, but to help in the estimation of the overall capacity and level of service of the several landside components.

2.2 Airport Congestion

Congestion occurs whenever the demand for a certain service exceeds a given available capacity. In the case of the airports, congestion typically results from an imbalance between the number of scheduled flights and the maximum throughput capacity of the airport. Such imbalances lead to the degradation of the airport service conditions, which consequently contributes to the occurrence of flight delays.

Flight delays are currently a major threat for the future of air transportation worldwide. The fast air traffic growth combined with the difficulty of the airports to increase their capacity has led airports worldwide to face severe flight delays over the last decades. In 2017, about 20% of the flights arrived delayed more than 15 minutes to the airports of United States (US) and Europe (FAA, 2018; Eurocontrol, 2010). Ball et. al. (2010) estimated the underlying costs resulted from these delays for US in a total of $32 billion.

The expected air traffic growth and the high costs caused by flight delays make increasingly important to find solutions to cope with the problem of airport congestion. Possible mitigation solutions are presented in Section 2.3. In this section we analyze the complex dynamics and reasons that leads to the occurrence of flight delays.

2.2.1 Flight Delays

Flight delays may result from several reasons. First, flight delays will certainly occur at times when the expected flight demand exceeds the maximum throughput capacity of the airport. This type of delays is generally a consequence of an excessive schedule of flights and they increase linearly with the time length of the overload period. For that reason, they are called “overload delays”.

Secondly, flight delays may also be present during periods when expected flight demand is lower than the airport capacity. This type of delays is caused by a set of factors that makes the times at which flights effectively take place and the times that airports need to process these flights, to be different than expected. This variability on flight demand and airport service times results from several sources of uncertainty, which makes this type of delays to be designated “stochastic delays”.

In the case of demand, this uncertainty stems from two main aspects. In the first place, the time instants at which demands actually occur are randomized as a result of inevitable deviations from the original schedule of flights, instigated by several reasons such as mechanical problems with an aircraft, slow processing of passengers in terminals, delays at other airports, etc. Secondly, the time instants at which flight demands are scheduled to take place are generally not evenly spaced over the day but often “bunched together” around certain times, such as “on-the-hour” or “on-the-half-hour”, which leads to high concentration of flights during small periods of time and consequently to congestion during these periods. (Neufville and Odoni, 2013).

Concerning the airport service times, the number of flights that the airports can process over the day depends on several factors already presented in Section 2.1, such as the weather conditions, the performance of the ATM system, the type of aircraft intending to use the airport, the aircraft mix, etc. The uncertainty of these factors justifies the variations of the airport service times over the day.

These small variations in the demand and service rates may lead to the occurrence of severe flight delays. According to *queuing theory* (See Larson and Odoni, 1981), the steady-state level of "stochastic delays" increases nonlinearly with the utilization ratio ρ of the airport, which is the ratio between the expected demand rate λ and the expected service rate μ, in a proportion equal to $1/(1-\rho)$. This nonlinearity is illustrated in Figure 2.5.

The main conclusion that can be taken from Figure 2.5 is that, for an airport with a utilization ratio ρ close to 1, a small variation in flight demand or in airport capacity can have significant impacts on flight delays. This means that small decreases (increases) in flight demand, or increases (decreases) in airport capacity, may lead to very significant reductions (increases) in flight delays.

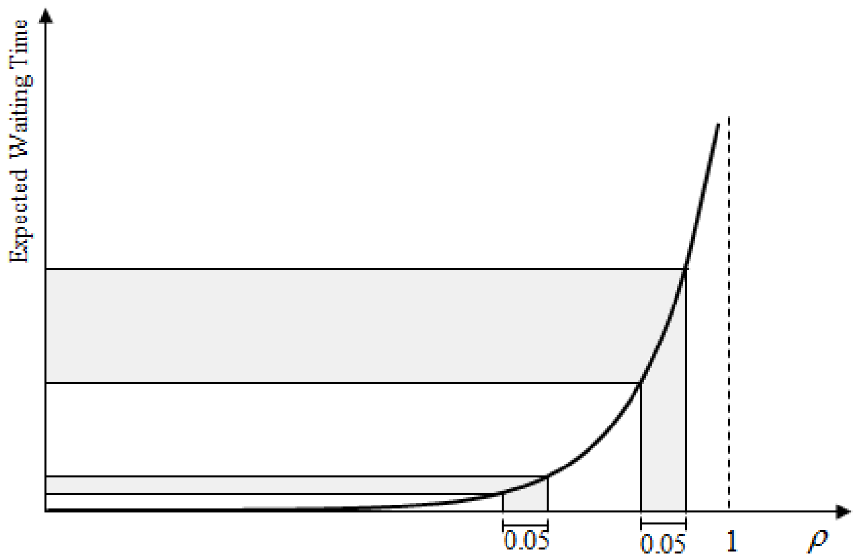


Figure 2.5 – Nonlinear response of "stochastic delays"

These observations have motivated the appearance of several initiatives aiming at either managing flight demand or at increasing airport capacity through improvements of the ATM system (see Chapter 2.3).

2.2.2 Modelling Flight Delays

In general, we can divide flight delay models into three main categories, specifically macroscopic, mesoscopic and microscopic models.

First, microscopic models consider each aircraft individually, trying to reproduce as precisely as possible airport operations, including the specificities of each airport layout and its operating rules. Microscopic models are typically simulations and they are particularly useful for analyzing how different airport procedures and tactical methods can reduce airport congestion. Because of its level of detail and data requirements, these types of models are not the most appropriate for strategic planning decisions. A review of these models can be found in Odoni et al. (1997). The best known are the SIMMOD and the TAAM models.

Mesoscopic models consider flights at an aggregate manner. They are typically applied to the design of procedures for optimizing surface operations, such as taxiway and apron operations (Simaiakis et al., 2014 and Khadilkar and Balakrishnan, 2014). Their heavy reliance on detailed operational data (e.g., demand for landings, runway configuration in use, etc.) may limit their applicability to modeling flight delays for strategic planning purposes, especially when such data are not available.

Finally, macroscopic models aggregate operations at the airport level, providing computationally efficient estimates of the relationships between flight schedules, airport capacity and flight delays. These types of models are much less computationally intensive and can be used to explore quickly a wide range of possible scenarios. The existing macroscopic models of congestion are mainly based on econometric models (Kwan and Hansen, 2010, Morrison and Winston, 2008 and Xu, 2007), deterministic queuing models (Hansen, 2002) and stochastic queuing models (Kivestu, 1976, Gupta, 2010, Pyrgiotis and Odoni, 2016).

2.3 Congestion Mitigation

In this section we describe possible solutions to mitigate airport congestion. We classify them into three different categories. First, in the long term, congestion may be alleviated through the expansion or construction of airports, and the development of new air traffic management (ATM) technologies. Secondly, in the short and medium term, airports can mitigate congestion through demand management measures. Finally, at the day of operations, air traffic flow management procedures may be the only alternative available to minimize the impacts of congestion.

2.3.1 Airport Expansion and New ATM Technologies

As stated in Chapter 2.2, airport congestion has been increasing over the last decades. For this reason, it is crucial to find solutions aiming to accommodate the expected flight demand in the long term.

First, airports may expand their available capacity by building new landside and airside facilities, such as new runways, new apron areas or new terminals, depending on the capacity bottlenecks of the airports. The main limitation of these types of interventions are usually related with the lack of empty space close to the airports to expand those facilities, and noise restrictions, which typically represent an obstacle for airports located close to dense urban areas.

When these types of interventions are not possible or desired, building a new airport to complement, or even replace, existing ones may be the only alternative available. Building a new airport always represents a huge undertaking for local communities, since it requires substantial amounts of capital that are not always available. Moreover, these types of interventions are typically subjected to numerous constrains and might not even be feasible in many cases due to the geographic, environmental, socio-economic and political issues associated with such large projects (Hamzawi, 1992).

In order to support aviation authorities to make decisions about the type of interventions to perform, a set of decision support tools have emerged in literature. They can basically fit into two categories: airport expansion economics and airport site selection. The former was surveyed some years ago by Cohen and Coughlin (2003), consisting on general theoretical principles to be taken into account when making decisions on the expansion of individual airports. Recently, Zou and Hansen (2012) extended these analyses to two airports. The airport site selection problems mainly consist on the comparison of alternative locations for building or expanding a single airport in a given region. Two type of techniques are typically used for this purpose, cost benefit analysis (Cohen, 1997 and Jorge and De Rus, 2004) and multi-criteria analysis (Paelinck, 1977; Min, 1994; Min, et al., 1997 and Vreeker et al., 2002). At the network level, Saatcioglu (1982) developed three optimization models. The first model determines the minimum number of airports required for a given population of passengers. The second model determines the airport locations and capacities that minimize total airport construction costs and bus transportation costs for a given demand. The third model is an extension of the second one that also considers that demand can be assigned to different types of aircraft and buses. Santos and Antunes (2015) developed an alternative optimization model aiming to support aviation authorities to select the best expansion interventions to implement in an airport network for a given budget.

Still in respect to long term alternatives, it is also crucial that researchers and aviation industry continue developing air traffic management technologies. They typically consist on improvements in aircraft surveillance, navigation and communication systems. These types of technologies intend to support air traffic controllers to make decisions more easily during the day of operations, leading to a more efficient use of the airport capacity.

2.3.2 Demand Management

Although airport construction and expansion can be considered the most obvious way to increase airport capacity, such solutions typically require a long time before becoming effective and constitute very costly and controversial processes. Thus, in the medium and short terms, demand management measures may be the only alternative solution available for keeping delays within reasonable bounds.

Demand management refers to any set of administrative or economic measures and regulations aimed at constraining flight demand at busy airports. Demand management strategies typically involve two ways of handling demand. First, by limiting in some way the demand for access to congested airports, for instance by declaring a limit capacity that prevents flight demand to exceed a desired level. Second, by modifying the spatial and temporal distribution of this demand to bring it closer to available capacity, which can be done either by performing schedule adjustments or through market mechanisms.

The motivation for demand management comes directly from the fundamental observation in Section 2.2 – when demand approaches capacity, the relationship between delay and demand becomes nonlinear, with small increases in flight demand at peak hours resulting in proportionally much larger increases in delays.

The available approaches to airport demand management are typically divided in three categories, specifically administrative, economic and hybrid approaches. Below we discuss each one of them. Note that in US, flight scheduling is not subjected to any demand management mechanism. Instead it is assumed that delay costs are internalized by the airlines. However, as compared to similar airports located elsewhere this approach leads to higher levels of congestion (Pyrgiotis and Odoni, 2016 and Jacquillat and Odoni, 2015a) *a) Administrative Approaches*

By definition, administrative approaches do not involve economic incentives to influence the choices of the airlines concerning the time when they will operate at an airport. Typically, they consist on voluntary schedule adjustments performed by airport representatives, or other entity, in order to reduce the number of flights scheduled at peak hours by distributing them more evenly over the day.

The IATA slot allocation process has been the dominant administrative demand management mechanism practiced by the majority of the busiest airports outside US. The IATA slot allocation process takes place several months before the day of operations, after the airlines have sent their requested schedules of flights to the airports. According to this process, any aircraft that intends to operate a flight to and from a congested airport needs to have a slot allocated. The slot represents a permission given to an airline allowing the use of the airport during the slot time. The number of slots available per unit of time is constrained by a declared capacity elected by airport authorities (See Section 2.1). The slots available are allocated by a slot coordinator taking into account a set of rules and priorities defined in the IATA World Slot Guidelines (WSG) (IATA, 2017). Below we summarize some of these rules. A more detailed description of the IATA slot allocation process is provided across the chapters of this dissertation.

According to the WSG, first priority is given to series of flights. A series of flights is defined as a sequence of at least five flights requested for the same time of the day on the same day of the week, distributed regularly in the same season, and allocated in that way or, if that is not possible, allocated at approximately the same time. This definition intends to capture the airline's regular flights by giving them priority.

Among series of flights, slot coordinators must account for three other priorities: (i) first, slot coordinators must give priority to historic flights. According to the WSG, an airline can earn historic rights (or grandfather rights) if a series of flights is operated at least 80% of the time in the previous equivalent season (winter or summer). This rule allows airlines to keep their regular flights without being threatened by changes every season; (ii) then, change-to-historic flights have priority over new requests. These are flights for which an airline already holds an historic flight, but because of a variety of operational reasons the airline requests some changes, such as a re-timing or a change to a larger aircraft; (iii) after historic and changed historic flights have been allocated, the coordinator creates a slot pool, which includes all the slots rejected and not yet used, as well as any newly created slots due to an increase in airport capacity. From this slot pool, 50% of the slots are allocated to new entrant airlines. An airline is considered a new entrant if it holds less than five slots in a certain day of the season after the requested slots are allocated; (iv) finally, the remaining slots are allocated to flight requests that do not own any of these three priorities.

There is a vast literature focusing on the efficiency of the IATA slot allocation process. Czerny et. al. (2008) provides an extensive coverage on the subject, presenting also some alternatives to overcome the limitations of this approach, some of them presented in Section 2.3.2 *b)*.

Ulrich (2008) denotes a set of advantages of using these types of administrative rules. First, these guidelines are already extensively implemented worldwide and generally accepted by airlines and airport representatives. They establish a set of standard rules to be used at congested airports, increasing the efficiency of the slot allocation process. Moreover, they also seem to have enough flexibility to consider special situations, which allows the incorporation of local regulations. Ulrich also states that this process is inexpensive, giving the example of the German airports where the direct costs linked to the slot allocation process are lower than 2.50€ for each coordinated operation.

Despite the several advantages of this process, the slot allocation experience has demonstrated a set of inefficiencies (NERA, 2004; Harsha, 2009; and European Commission, 2011). One of the main concerns is the barrier to new entrants. In fact, empty slots tend only to be available at unattractive times, which is a barrier to competition, as new services may not be launched. Furthermore, airlines are reluctant to give up their historic slots even if they make financial losses on their services, since in the long term those slots may become profitable. In the meantime, the services are operated inefficiently, and in some cases a proportion of the flights are cancelled in order to reduce the airline’s costs. This phenomenon is known as “slot misuse”.

b) Economic Approaches

By definition, economic approaches involve market mechanisms aiming to influence the choice of the airlines concerning airport access. Typically, they consist in a system of access fees based on congestion pricing, which takes into account the pattern of delay at an airport over time. The pricing scheme varies with the time of the day, as well as possibly by season and even by day of the week, with higher fees during peak demand periods and lower fees during off-peak periods. The fundamentals of congestion pricing can be in general summarized as follows.

Consider a facility that experiences congestion, in this case an airport. Every user that obtains access to that facility during periods when delays exist generates a congestion cost that consist of two components: i) an "internal delay cost", which corresponds to the cost of the delay to this particular user; and ii) an "external delay cost", which represents the cost of the additional delay to all other prospective users caused by this particular user.

Currently, aircraft only pay for access to airports a landing fee proportional to their weight. These fees do not vary with flight demand, and therefore they do not help airports reducing congestion. Moreover, the only cost in addition to the landing fee that the aircraft will perceive is the internal delay cost. This means that the users with highest tolerance to delays will persist the longest in using the airport as delays grow. In contrast, flights with large number of passengers, tight connections, and short turnaround times are the ones that will be most sensitive to the aggravation of congestion.

The main idea of congestion pricing is then to impose a cost on each user equal to the external cost associated with the access of this user to the facility. Basically, users will pay an additional fee in order to compensate "society" for the external costs they impose on the other users. This is typically referred by economists as the "internalization of external costs". The congestion fee works here as a device for optimizing the use of the facility. The optimal use is achieved when the congestion price equals the external cost associated to each user.

The previous result is illustrated in Figure 2.6. The curve *D* represents the demand curve for an airport facing capacity constraints. Curve *I* shows how the internal delay cost increases with the demand. Curve *T* shows how the internal and external costs together increase with the demand. The difference between *I* and *T* gives the external cost at each level of demand. When there is no congestion pricing the equilibrium point is at 1, where demand *D* intersects the internal cost *I*. In case of congestion pricing, the optimal congestion fee is located at point 2, when demand *D* intercepts the internal and external costs.

There is an extensive literature advocating the use of congestion pricing to the case of airports (Levine, 1969; Vickrey, 1969; Carlin and Park, 1970; Morrison, 1983, Daniel, 1995; to name a few). Despite the several advantages identified by these authors, there are also some concerns associated to this type of approach (Brander and Cook, 1986; Zografos and Madas, 2003; and Neufville and Odoni, 2013).

In the first place, the determination of the exact level of congestion fee to apply in practice is typically very difficult to determine exactly. This might need several years of trial and error to set its value. In fact, if the fee is set too low, the airport will face congestion, if the fee is set too high, the airport will lose some potential flights. The idea is to find a fee that ensures an equilibrium between demand and congestion.

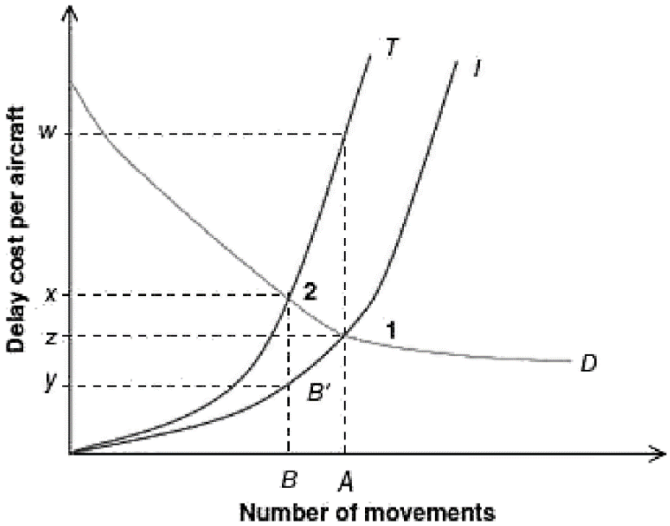


Figure 2.6 – Internal and External Costs (Neufville and Odoni 2013)

Another important point is that, at airports operating as hubs dominated by a single airline, congestion pricing may not be effective, since the dominant airline will absorb nearly all the external costs generated by any aircraft movement. Then, we can say that congestion pricing works better at airports with no dominant airlines and large number of competing airlines.

Finally, some argue that congestion pricing is discriminatory to small airlines, since these airlines do not have the ability to pay for the highest demand slots, leading to an oligopoly of the major airlines.

c) Hybrid Approaches

Hybrid approaches combine administrative and economic mechanisms to influence airlines choices. On one hand, they take into account a capacity that is declared by the airport in order to limit the available number of slots per time period, as the administrative approaches. On the other hand, they rely on market mechanisms in order to allocate the slots among the airport users, as the economic approaches. Typically, hybrid approaches involve three type of methods, specifically hybrid congestion pricing, slot auctions and secondary trading.

In the case of hybrid congestion pricing, the procedure is very similar to the one presented before in (b). The main difference is that in this case there is a limited number of slots per time period. This means that, when the number of slots requested by airlines for a certain time period is higher than the limit capacity of the airport, the allocation of slots is performed using administrative rules, such as the IATA rules. This type of mechanism is already implemented in a few number of airports, mainly in UK (Manchester and some London airports, specifically Heathrow, Gatwick and Stansted) (Neufville and Odoni, 2013).

In the case of slot auctions, airlines are invited to bid for all or some specified percentage of the available slots and each slot is awarded to the highest bidders. The main advantage when compared with congestion pricing is that it avoids the problem of setting the exact level of congestion fee to implement, since in this case the real value of each slot is revealed by the bids made by the airlines. Despite this advantage, there are some concerns inherent to the slot auction process (Ball et al., 2006; Harsha, 2009 ; Neufville and Odoni, 2013).

First, the true value of the slots will not be clear to the airlines until all these slots are allocated. This is justified by the strong interdependence between flights, both at the local level and across airports. For instance, if an airline acquires a departure slot at an airport but fails to acquire the desired arrival slot at the airport of destination, then the slot acquired do not have any value for the airline. The same happens if an airline cannot acquire a slot that guarantees the desired turnaround time between two flights.

Secondly, it is crucial to specify precisely the rights and obligations associated with a slot, such as length of time for which a slot is valid, and in which conditions a slot can be cancelled or withdrawal by an authority.

Finally, it is also important to define who will receive the auctions profits. Some argue that the airport should receive them in order to make the necessary improvements to the airport. Others believe that the funds raised by auctions should be distributed to airlines operating at off-peak hours providing them an incentive to offer service at such times.

Slot auctions have been supported by several researchers that have been trying to find solutions aiming to overcome the limitations pointed out above (Grether et al. 1981; Ball et al., 2006; Cramton et al. 2007 and Harsha, 2009; to name a few). Nevertheless, in contrast to slot pricing, slot auctions were never implemented by any airport worldwide.

A system of secondary trading can be a solution to overcome the first problem mentioned for the slot auctions. In this case, airlines can exchange or sell slots that they do not intend to use after the auction process. However, some argue that this type of strategy can benefit major airlines as they are able to acquire more slots, which gives them more flexibility to optimize their schedules and then sell the remaining slots to the small airlines.

Secondary trading is also often used in complement to the administrative slot allocation process. Accordingly, after slots have been allocated, they can be exchanged or sold between airlines in order to improve airlines' schedules. In practice, only in the USA and in the UK secondary trading was performed with monetary payments.

2.3.3 Air Traffic Flow Management

As stated in Chapter 2.2, congestion may occur during the day of operations due to unexpected situations, such as weather disturbances, problems with an aircraft and other disruptions. These situations are highly unpredictable and may cause significant capacity-demand imbalances. In order to manage these situations, air traffic controllers can employ air traffic flow management (ATFM) procedures to minimize congestion impacts.

The main objective of ATFM is to ensure that aircraft can flow through the airspace safely and efficiently. For that purpose, air traffic controllers must prevent the occurrence of expected overloads that might affect airspace safety, and minimize the economic impacts and other penalties imposed by flight delays. This is typically accomplished by adjusting the flow of aircraft dynamically so that demand matches available capacity at airports and airspace sectors.

Three fundamental strategies are used by air traffic controllers in order to avoid or minimize the impacts of delays. First, controllers can activate ground holding programs whenever congestion is expected at the airports of destination or other airspace sectors. These types of programs consist on intentionally delaying the departure of a flight in order to minimize airborne delays, thus consequently providing savings on fuel consumption. Second, controllers may reroute some flights in order to avoid congestion at certain airspace sectors. This operation is known as rerouting. Finally, controllers may adjust the space between aircraft and their speed in order to control the rate at which traffic crosses some specified special airspace boundaries. This operation is known as metering.

Throughout the years several models have emerged in literature aiming to support air traffic controllers to take this type of decisions on a daily basis. They are typically divided in four groups: Single Airport Ground-Holding Models (SAGHM), which only make ground-holding decisions for a single airport; Multi-Airport Ground Holding Models (MAGHM), which make ground-holding decisions considering a network of airports; Air Traffic Flow Management Models (TFMM), which additionally determine the optimal speed of aircraft while airborne for a network of airports; and Air

Traffic Flow Management Rerouting Models (TFMRM), which also include rerouting decisions. Reviews of these models can be found in Odoni (1994) and Agustín et al. (2010).

Another important mission of air traffic controller is the selection and sequencing of airport configurations, and the balancing of arrival and departure service rates on the runways over time. These types of decisions are typically made in the beginning of each day, taking into account the schedule of flights and expected weather conditions.

As described in Chapter 2.1, an airport may have several runway configurations, which may be represented by a capacity envelope (see Figure 2.4). These envelopes give information of the airport capacity for each combination of arrivals and departures depending on the expected weather conditions.

Based on these capacity envelopes, a set of optimization models have emerged in literature aiming to support air traffic controllers to decide the runway configurations to use throughout the day and the optimal balancing of arrivals and departures (Li and Clarke, 2010; Weld et al., 2010; Bertsimas et al., 2011; and Jacquillat et al., 2016).

2.4 Slot Allocation Models

Nowadays, slot coordinators are assisted in their activity by specialized software (e.g., PDC SCORE). These software handles automatically the flight requests made by airlines providing instantaneous information about the priority class of each flight, as well as about the availability of slots. The system is capable to show exactly which flights are in violation with the declared capacities of the airports and indicate the closest solutions available. However, the slot requests are typically treated by slot coordinators in an ad hoc basis, which provides limited visibility into the full set of requests and interdependencies between decisions. In recent years, some optimization models have emerged in the literature, showing that optimization can improve the current performance of the slot allocation procedures. In this section we present the main existing slot allocation models in the literature. Their main goal is to support slot coordinators to better accommodate airlines' preferences at coordinated airports. For that purpose, they minimize the differences between the requested and allocated slot times, taking into account scheduling rules, coordination procedures, and operational constraints.

Three core models of slot allocation can be found in the literature, specifically the Zografos et al. (2012) model, the Pyrgiotis and Odoni (2016) model and the Jacquillat and Odoni (2015a) model. They have different characteristics, but they can complement each other. The first one is focused on the priorities set by IATA (although it does not account for all of them), being in accordance with the concept of series of slots and thus optimizing the slot allocation process for the entire season. The second one examines the implication of the slot allocation process when considering a constrained airport network, however does not consider the IATA rules, and it is only applied to single days. The last one is an extension of the Pyrgiotis and Odoni (2016), which additionally optimizes capacity utilization policies, aiming to have a more efficient use of airport capacity, without relying on an arbitrary, administrative notion of declared capacity. In this section we explore these three models in detail, presenting their formulations and applying them to simple examples. The analysis of the dynamics of these models, provide valuable insights for the development of the optimization tools planned for this dissertation.

2.4.1 Zografos, Salouras and Madas Model

The Zografos et al. (2012) model is a single-airport optimization model, which minimizes the total absolute displacement between the airlines' requested and allocated slot times, while complying with scheduling rules, coordination procedures, and operational constrains. It takes into account the slot priorities set by IATA and solves the allocation problem for the entire scheduling season, fully complying with the definition of series of slots. The main limitation of this model is that can only be applied at small-size airports (up to 50,000 flights per year). The author applied its model at three Greek airports: (i) Chania and Rhodes airports, which operate 20,000 and 37,000 flights annually, were solved optimally (i.e. with a 0% of optimally gap) in 12 and 50 seconds respectively; (ii) Heraklion airport, which operates about 50,000, was solved in about 5 minutes, but with an optimally gap of 1.58%.

a) Model Presentation

Before presenting the model formulation, the notation used to represent the sets, parameters and decision variables is introduced.

Sets:

$\boldsymbol{T} = \{1,...,T\}$: set of time intervals defined per day, indexed by *t*;

$\boldsymbol{D} = \{1,...,D\}$: set of days, indexed by *d*;

$\boldsymbol{F} = \{1,...,F\}$: set of series of flights, indexed by *i* or *j*;

$\boldsymbol{P} \subset \boldsymbol{F} \times \boldsymbol{F}$: set of fligh pairs (i, j) such that there is a connection between *i* and *j*;

$\boldsymbol{C} = \{1,...,C\}$: set of airport capacity constraints, indexed by *c*;

$\boldsymbol{T}_c = \{t \in T \mid t < T - D_c + 1\}$: set of coordination intervals over which the constraint *c* is checked;

$\boldsymbol{T}_c^s = \{t \in T \mid s \leq t < s + D_c\}$: set of consecutive coordination time intervals over which the constraint *c* is checked.

Parameters:

f_{it} = cost of allocating flight *i* to interval *t*;

$$B_{id} = \begin{cases} 1, & \text{if flight } i \text{ is schedulled to operate on day } d \\ 0, & \text{otherwise} \end{cases};$$

C_{tdc} = capacity of constraint *c* for day *d* and time interval *t*;

D_c = duration (number of intervals) of each capacity constraint *c*;

$T_{ij}^{\min}$ = minimum connection time between flight *i* and *j*;

$$F_{ic} = \begin{cases} 1, & \text{if constraint } c \text{ is applied to flight } i \\ 0, & \text{otherwise} \end{cases};$$

Decision Variables:

$$Y_{it} = \begin{cases} 1, & \text{if flight } i \text{ is rescheduled to take place during period } t \\ 0, & \text{otherwise} \end{cases};$$

The parameter f_{it} may be calculated as $|t - t_i|$, where t_i is the time interval originally requested for movement *i*. The parameter $T_{ij}^{\min}$ represents the minimum connection time between movements *i* and *j*, which may be an aircraft connection or a passenger connection. The capacity of constraint $c \in \boldsymbol{C}$ is denoted by parameter C_{tdc}. Each capacity constraint *c* is applied during a certain period of time D_c. This may be illustrated when airports have two capacity constraints for different periods, for instance one for 15 minutes and other for 30 minutes, which gives $D_c = 1$ and $D_c = 2$ if we assume that each period comprises 5 minutes. The set T_c represents the number of coordination intervals that is applied to each constraint *c*. If $D_c = 1$ this means the constraint *c* is applied during every time interval, consequently there are *N* coordination intervals where *c* must be checked. If $D_c = 2$, this means the constraint *c* is applied to every two consecutive intervals, and so, there are *N*-1 coordination intervals where *c* must be checked, one for each pair of time periods. The same thinking is used considering other values of D_c. The set T_c^s defines the set of consecutive coordination time intervals over which the constraint *c* is checked. If $D_c = 1$, the set of intervals is given by $s \le t < s+1$ for all $s \in \boldsymbol{T}_c$. F_{ic} specifies whether constraint *c* is applied to arrivals, departures or both.

Using the notation above, the model can be formulated as follows:

Objective Function:

$$\text{minimize} \sum_{i \in \boldsymbol{F}} \sum_{t \in \boldsymbol{T}} f_{it} Y_{it} \qquad (2.1)$$

subject to:

$$\sum_{t \in \boldsymbol{T}} Y_{it} = 1 \qquad \forall i \in \boldsymbol{F} \qquad (2.2)$$

$$\sum_{i \in F} \sum_{t \in T_c^s} B_{id} F_{ic} Y_{it} \le C_{sdc} \qquad \forall c \in \boldsymbol{C}, d \in \boldsymbol{D}, s \in \boldsymbol{T_c} \qquad (2.3)$$

$$\sum_{t \in [0,k]} Y_{jt} + \sum_{t \in \left[k - T_{ij}^{\min}, T\right]} Y_{it} \le 1 \qquad \forall \{i, j\} \in \boldsymbol{P}, k \in [T_{ij}^{\min}, T-1] \qquad (2.4)$$

$$Y_{it} \in \{0,1\} \qquad \forall i \in \boldsymbol{F}, t \in \boldsymbol{T} \qquad (2.5)$$

The objective function (2.1) minimizes the total absolute difference between the requested and allocated slot time. Another possible alternative presented by Zografos et al. (2012) is to multiply the cost coefficients by the total number of days that each movement will operate B_{id}, weighting movements by their frequency.

Constraints (2.2) ensure that no flight movement is eliminated. Constraint (2.3) state that the total number of movements cannot exceed the slot limits defined for each capacity constraint. Constraints (2.4) force connection time to be larger than the minimum turnaround time imposed. Basically, they state that for any $k \geq T_{ij}^{\min}$, if the departure movement has been allocated at any of the intervals $[0,k]$, it is not possible to allocate the corresponding arrival after interval $k - T_{ij}^{\min}$. Finally, constraint (2.5) defines the domain of the decision variables.

In order to take into account the priorities defined by IATA, the model is applied hierarchically for each priority class.

b) A Hypothetical Example

The results that can be obtained through the application of the model dealt with in the previous section are exemplified here for a hypothetical example. This example does not intend to represent a real situation but only to reflect the dynamics of the model.

Essentially, the example considers three days of operations, which are divided into five periods, and ten flight movements that need to be allocated. Not all flights are made every day and some of them are subjected to connection flights. Moreover, no priority classes are considered here in order to make it easier to understand the results. This situation was randomly generated.

Figure 2.7 presents the requested slot times for each day of operations. At red are presented the time periods at which the flight movements are requested by airlines. For instance, flight 1 is requested to take place at period 2 and constitutes a series of flights that take place at day 1, 2 and 3.

Note that during periods 2, 3 and 4 there is a high concentration of flights requested and during period 1 and 5, no slots are requested. Because of capacity constraints some requested slots will need to be reallocated. This reallocation process will take into account the definition of series of slots, hence any movement reallocated during a certain day will need to be also changed in the other days in which it takes place.

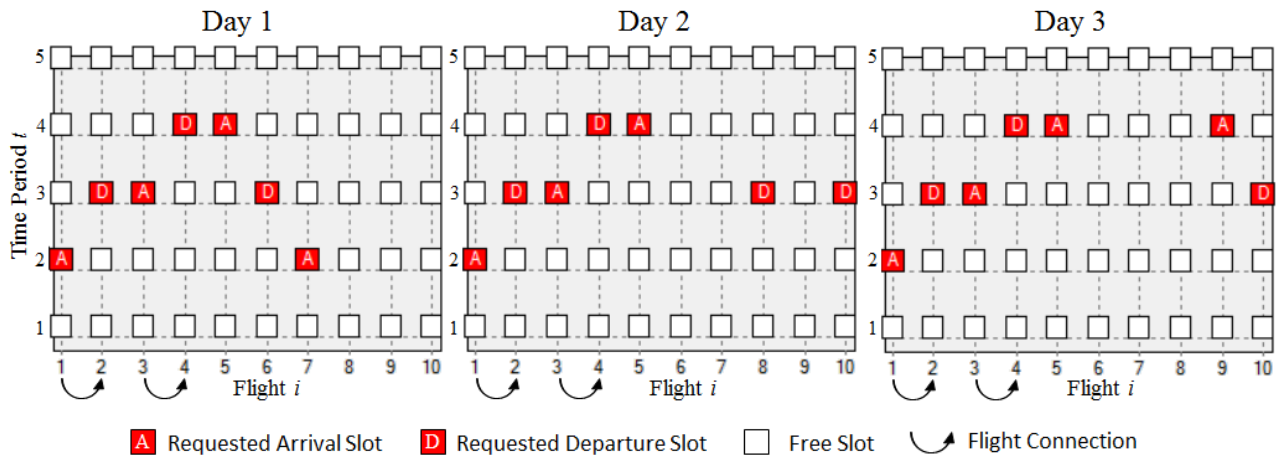


Figure 2.7 – Requested slot times

The example considers three capacity constraints $\boldsymbol{C} = \{1,...,3\}$. The capacity constraints 1 and 2 are applied in every time period (i.e., $D_1 = 1$ and $D_2 = 1$) and the capacity constraint 3 is applied every two consecutive time periods ($D_3 = 2$). Moreover, the capacity constraint 1 is applied to arriving flights $F_{i1} = [1,0,1,0,1,0,1,0,1,0]$, the capacity constraint 2 is applied to departing flights $F_{i2} = [0,1,0,1,0,1,0,1,0,1]$ and the capacity constraint 3 is applied to all flights $F_{i3} = 1$. Lastly, the capacity of constraint 1, 2 and 3 is equal to $C_{td1} = 1$, $C_{td2} = 2$ and $C_{td3} = 4$ respectively. In summary, only one arrival and two departures can take place every time period, and only four movements can take place every two consecutive periods.

The example also considers two connected flights, flight 1 and flight 2 as well as flight 3 and flight 4. The connection time between flights must be larger than the minimum turnaround time imposed, $T_{12}^{\min} = 1$ and $T_{34}^{\min} = 1$.

In Figure 2.8 we present the solution obtained by solving the model with these parameters. The green squares represent the slots allocated. When there is a red square means that the slot time requested was reallocated according to the arrow indicated, in the other cases the flight was reallocated at the same time as requested by airlines.

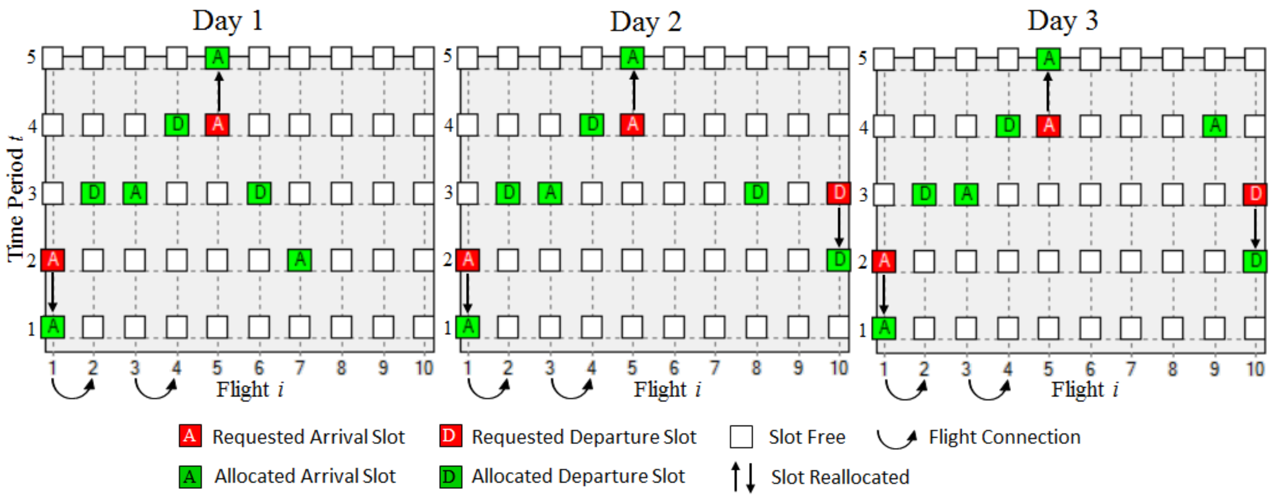


Figure 2.8 – Allocated slot times – Zografos et al (2012) model

Observing the results obtained for the first day, the changes produced in the schedules are easily explained. In time period 2, there are 2 arrivals requested (see Figure 2.7) and a maximum arrival capacity of 1 flight per time period, thus one of the flights requested for time period 2 was reallocated by the model, in this case the flight 1. Another adjustment took place in time period 3 because there are 5 flights scheduled during periods 2 and 3, a number that exceeds the limit specified by constraint 3, which states that only 4 flights can take place per two consecutive periods. Once again one of the flights was reallocated, in this case flight 5.

The results for day 2 and day 3 can be obtained following the same reasoning, however it is worth noting that if the model was only applied to day 3, the flight 1 would not be changed, since all capacity constraints would be respected. However, in order to consider the definition of series of slots, the flight 1 needs to be reallocated in order to keep the same time as day 1.

About the flight connections, the time between flight 3 and flight 4 was not changed, however the time between flight 1 and 2 increased. Since the model does not take into account any maximum turnaround time between connection flights, the solution is feasible.

The objective function value is given by expression (2.1) and is equal to 3. If movements are weighted by their frequency in the objective function, the value of the total displacement would be equal to 8. However, if we solve the model weighting flights by their frequency the optimal solution would not be the one presented in Figure 2.8 but the one illustrated in Figure 2.9.

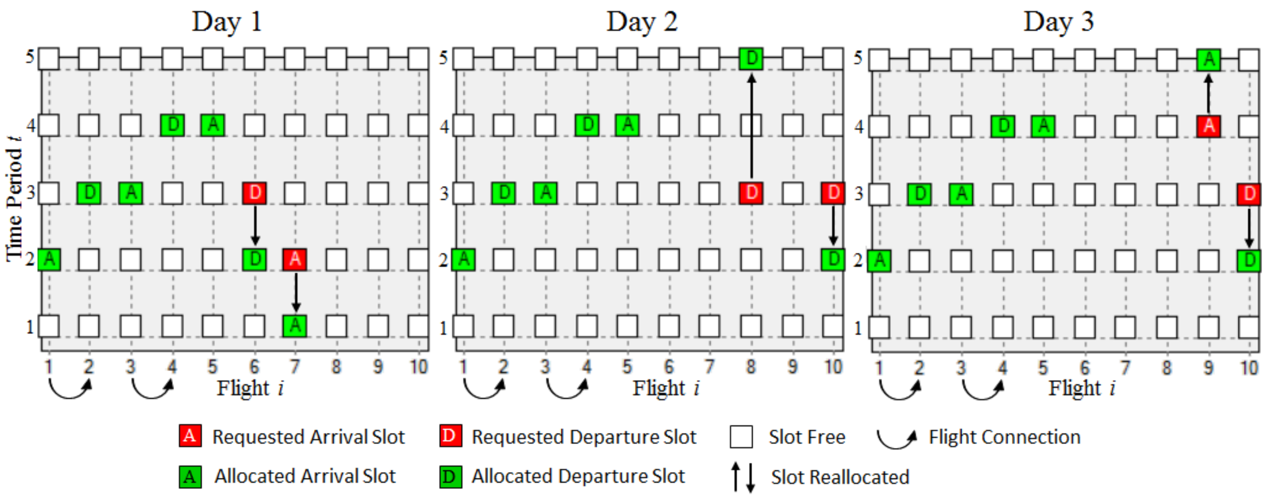


Figure 2.9 – Allocated slot times considering frequency – Zografos et al. (2012) model

In this solution the value of the first objective is 6, higher than the one obtained by the first solution, however the value of the objective considering the frequency is equal to 7, lower than the first solution that is equal to 8.

One consideration about these two outcomes is that the second solution would probably be preferred in practice over first solution, since it is closer to the schedules requested by airlines. However, in a larger example, the application of the model considering frequencies on the objective function may lead to large displacements of flights with lower frequency. Hence, the application of the second objective should be solved considering a limit for the maximal displacement.

2.4.2 Pyrgiotis and Odoni Model

The Pyrgiotis and Odoni model is an airport network optimization model, which reschedules airline flight requests taking into account scheduling limits specified by airports. For that, a new schedule is generated, first minimizing the maximum schedule displacement experienced by any single flight, and then minimizing the aggregate schedule displacement across all flights. No flights are eliminated and the model respects all aircraft itineraries and passenger connections. This model is called the Demand Smoothing (DS) optimization model.

Unlike the Zografos model, the DS model optimizes flight schedules for any single day of operations, not being in accordance with the rules set by IATA, and in particular with the concept of series of slots. By contrast, the DS optimization model may be applied to a constrained airport network, optimizing not only the effects that scheduling limits may have on local delays but also on propagated delays. Nevertheless, the model was never applied to a real network of airports, but just to single airports, considering connections that do not take place at the airport under analysis, but have an impact on its operations.

Pyrgiotis and Odoni (2016), applied the DS model at Newark Liberty International Airport (EWR) showing that with a maximum displacement of only 30 minutes and considering that limits are set as low as the airport capacity under IFR, there is a feasible schedule that can accommodate all of the traffic of the busiest day in 2007. They also found that the local delay savings that would result from introducing such scheduling limits at EWR can be of the order of 20% for arrivals and 50% for departures on busy days, and a reduction of 20% in delays propagated from EWR to the rest of the US network of airports would also be expected.

a) Model Presentation

The notation used to represent the sets, parameters and decision variables is presented below.

Set:

$\boldsymbol{T} = \{1,...,T\}$: set of periods, indexed by *t*;

$\boldsymbol{F} = \{1,...,F\}$: set of flights, indexed by *i* or *j*;

$\boldsymbol{P} \subset \boldsymbol{F} \times \boldsymbol{F}$: set of flight pairs (i, j) such that there is a connection between *i* and *j*;

$\boldsymbol{A} = \{1,...,K+1\}$: set of airports, indexed by *k*. Where *K* is the total number of airports with slot constraints, and the $(K+1)^{th}$ airport corresponds to a virtual airport that accounts for all the other airports .

Parameters:

$$A_{itk}^{arr} \,/\, A_{itk}^{dep} \,/ = \begin{cases} 1, & \text{if flight } i \text{ is scheduled to arrive / depart from airport } k \text{ during period } t \\ 0, & \text{otherwise} \end{cases};$$

C_{tk}^{dep} = maximum departure slots at *k* in period *t*;

C_{tk}^{arr} = maximum arrival slots at *k* in period *t*;

C_{tk}^{T} = total number of slots at *k* in period *t*;

$T_{ij}^{\min}$ = minimum connection time between flight *i* and *j*;

$$P_{ij} = \begin{cases} 1, & \text{if there is at least 1 passenger connecting from flight } i \text{ to } j \\ 0, & \text{otherwise} \end{cases};$$

$T_k^{\min}$ = minimum time required by any connecting passenger to transfer between two flights at a certain airport k.

Decision Variables:

X_i = number of periods that flight i is displaced;

$$Y_{itk}^{arr} / Y_{itk}^{dep} = \begin{cases} 1, & \text{if flight } i \text{ is rescheduled to arrive / depart from airport } k \text{ during period } t \\ 0, & \text{otherwise} \end{cases};$$

Using the notation above, the model can be formulated as follows:

Objective Function:

$$\text{Minimize } \lambda \max_{i \in \boldsymbol{F}} |X_i| + \sum_{i \in \boldsymbol{F}} X_i \qquad (2.6)$$

Subject to:

$$\sum_{t \in \boldsymbol{T}} \sum_{k \in \boldsymbol{A}} Y_{itk}^{dep} t = \sum_{t \in \boldsymbol{T}} \sum_{k \in \boldsymbol{A}} A_{itk}^{dep} t + X_i \qquad \forall i \in \boldsymbol{F} \qquad (2.7)$$

$$\sum_{t \in \boldsymbol{T}} \sum_{k \in \boldsymbol{A}} Y_{itk}^{arr} t = \sum_{t \in \boldsymbol{T}} \sum_{k \in \boldsymbol{A}} A_{itk}^{arr} t + X_i \qquad \forall i \in \boldsymbol{F} \qquad (2.8)$$

$$\sum_{t \in \boldsymbol{T}} \sum_{k \in \boldsymbol{A}} Y_{itk}^{dep} = 1 \qquad \forall i \in \boldsymbol{F} \qquad (2.9)$$

$$\sum_{t \in \boldsymbol{T}} \sum_{k \in \boldsymbol{A}} Y_{itk}^{arr} = 1 \qquad \forall i \in \boldsymbol{F} \qquad (2.10)$$

$$\sum_{i \in \boldsymbol{F}} Y_{itk}^{dep} \le C_{tk}^{dep} \qquad \forall t \in \boldsymbol{T}, k \in \{1, ..., K\} \qquad (2.11)$$

$$\sum_{i \in \boldsymbol{F}} Y_{itk}^{arr} \le C_{tk}^{arr} \qquad \forall t \in \boldsymbol{T}, k \in \{1, ..., K\} \qquad (2.12)$$

$$\sum_{i \in \boldsymbol{F}} Y_{itk}^{dep} + Y_{itk}^{arr} \le C_{tk}^{T} \qquad \forall t \in \boldsymbol{T}, k \in \{1, ..., K\} \qquad (2.13)$$

$$\sum_{t \in \boldsymbol{T}} \sum_{k \in \boldsymbol{A}} Y_{jtk}^{dep} t - \sum_{t \in \boldsymbol{T}} \sum_{k \in \boldsymbol{A}} Y_{itk}^{arr} t \le \sum_{t \in \boldsymbol{T}} \sum_{k \in \boldsymbol{A}} A_{jtk}^{dep} t - \sum_{t \in \boldsymbol{T}} \sum_{k \in \boldsymbol{A}} A_{itk}^{arr} t \qquad \forall i, j \in \boldsymbol{P} \qquad (2.14)$$

$$\sum_{t \in \boldsymbol{T}} \sum_{k \in \boldsymbol{A}} Y_{jtk}^{dep} t - \sum_{t \in \boldsymbol{T}} \sum_{k \in \boldsymbol{A}} Y_{itk}^{arr} t \ge T_{ij}^{\min} \qquad \forall i, j \in \boldsymbol{P} \qquad (2.15)$$

$$\sum_{t\in \boldsymbol{T}}\sum_{k\in \boldsymbol{A}} Y_{jtk}^{dep} t - \sum_{t\in \boldsymbol{T}}\sum_{k\in \boldsymbol{A}} Y_{itk}^{arr} P_{ij} t \geq P_{ij} T_k^{\min} \qquad \forall i,j \in \boldsymbol{P}, k \in \{1,...,K\} \qquad (2.16)$$

$$Y_{itk}^{dep}, Y_{itk}^{arr} \in \{0,1\} \qquad \forall i \in \boldsymbol{F}, t \in \boldsymbol{T}, k \in \boldsymbol{A} \qquad (2.17)$$

$$X_i \in I \qquad \forall i \in \boldsymbol{F} \qquad (2.18)$$

The objective function (2.6) consists of two parts. First, it minimizes the maximal displacement that any given flight will sustain. Then, among all feasible schedules that can be obtained under the first objective, the model selects the one that minimizes the total displacement. These two objectives may be considered as a single equation (2.6) where $\lambda >> 1$. This choice is motivated by equity concerns, as it ensures that no flight will incur a disproportionately large displacement.

Constraints (2.7) and (2.8) ensure that the departure time and arrival time of any given flight is always displaced by the same amount X_i . Constraints (2.9) and (2.10) ensures that no flight is eliminated. Contraints (2.11), (2.12) and (2.13) state that the total number of flights per time interval cannot exceed the arrival, departure and total slot limits set out by each slot-constrained airport. In order not to be redundant, $C_{tk}^{T} \leq C_{tk}^{arr} + C_{tk}^{dep}$. Constraints (2.14) and (2.15) forces connection time between flight *i* and *j* to be at least the same as than originally scheduled and greater than a minimum connection time ($T_{ij}^{\min}$).Constraint (2.16), ensures that all passenger connections can be achieved with at least the minimum time required ($T_k^{\min}$). Finally, expressions (2.17) and (2.18) define the domain of the decision variables.

b) A Hypothetical Example

The example applied to Pyrgiotis and Odoni (2016) model is similar to the one presented with Zografos et al. (2012) model, the main difference lies in the fact that this model can only be applied to single days. In this example only the first day was considered (see Figure 2.7).

An important point is that the Pyrgiotis and Odoni (2016) model can be applied to a constrained airport network. However, in order to simplify the solution interpretation, the example only considers a single airport (Airport 1) and an airport that represents all the others. Moreover, it was assumed that all flight movements take one hour to go from Airport 1 to the other airports, and vice-versa.

The parameters were set out in order to simulate an example similar to the one presented with the Zografos et al. (2012) model. However, there are two differences between both examples due to different types of parameters and constraints.

First of all, the capacity constraint of two consecutive periods cannot be implemented here, consequently it is possible to have more than four flights in two consecutive periods. Secondly, in Pyrgiotis and Odoni (2016) model, connection times need to be at least equal to the requested connection time, meaning that the time between connection flights cannot increase, as happened in Zografos et al. (2012) model where the time between two connected flights can be higher than initially requested.

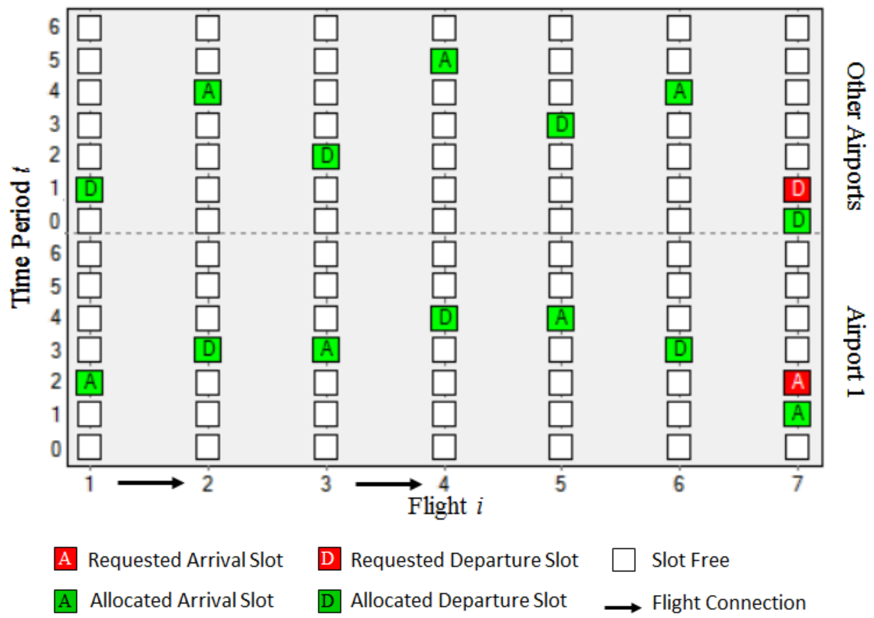


Figure 2.10 – Allocated slot times – Pyrgiotis and Odoni (2016) model

Taking this considerations into account, the parameters values were set as follows: $C_{t1}^{arr} = 1$; $C_{t1}^{dep} = 2$; $T_{12}^{\min} = 1$; $T_{34}^{\min} = 1$. Figure 2.10 presents the solution obtained by solving the Pyrgiotis and Odoni (2016) model.

An important point about Figure 2.10 is that it has seven time periods instead of the five considered in Figure 2.7. This is explained because an arrival flight in the first period of the Airport 1 will force a departure flight from other airports in the previous period. The same occurs when a departure takes place in the fifth period of airport 1. Consequently, this example considers seven periods, where the periods 1, 2, 3, 4 and 5 are the same as the ones presented in Figure 2.7. In order to constrain flights to be reallocated to time periods 0 and 6, in which Airport 1 is not operational, two capacity parameters were added to the model: $C_{0,0}^{T} = 0$; $C_{7,1}^{T} = 1$

Observing the result obtained in Figure 2.10, it can be seen that only one change was performed to the flight requests, which happens in period 3 where two arrivals are requested, while the arrival capacity limit is just 1 flight per time period. The flight 7 is the flight selected to be reallocated, because a change in flight 1 would require a change in flight 2 motivated by connection issues. Note that this solution allows to have five flights during two consecutive periods, and so there is no need to postpone any flight in period 5, as happened in Figure 2.7.

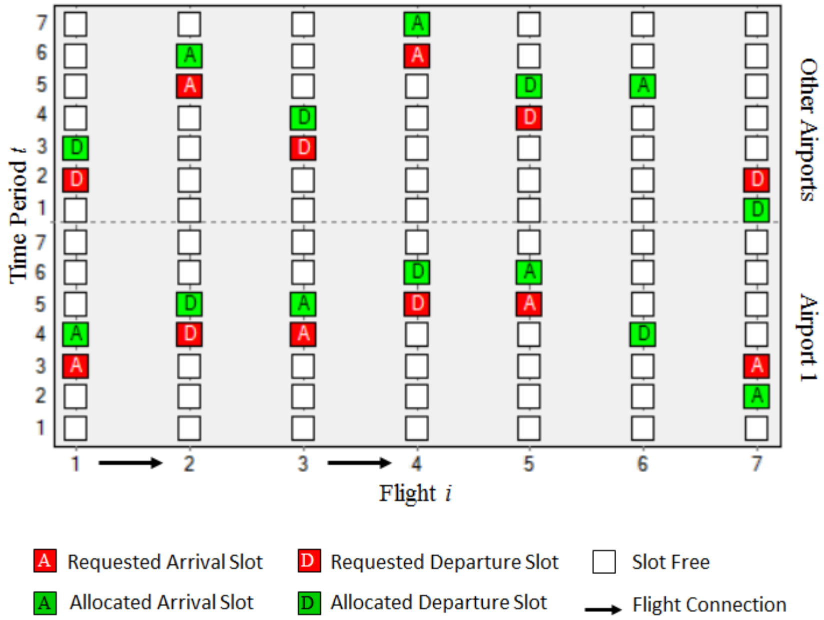


Figure 2.11 – Allocated slot times, restricting flights at time period 3 – Pyrgiotis and Odoni (2016) model

In order to understand better the dynamics of the Pyrgiotis and Odoni (2016) model, a case where no flights are allowed during period 3 was simulated $C_{3,1}^{T} = 0$. This example can show very well what happens if there is a need to change a connection flight (see Figure 2.11).

In this case Flight 1 is changed to period 4, which forces Flight 2 to be postponed as well as Flight 3 because of capacity constraints. Consequently, the change produced in Flight 3 forces its connection flight to be also postponed as well as Flight 5 due to capacity constraints. In total there is 6 changes produced in this solution.

2.4.3 Jacquillat and Odoni Model

The Integrated Capacity Utilization and Scheduling Model (ICUSM) presented by Jacquillat and Odoni (2015a) is an integrated model of airport congestion mitigation, which jointly optimizes slot allocation and capacity utilization decisions, subject to scheduling, capacity and delay reductions constraints.

The ICUSM is composed of three different models: an integer programming model of scheduling interventions, which modifies schedules requested by airlines in a similar way to the one suggested by Pyrgiotis and Odoni (2013); a dynamic programming model of capacity utilization, which optimizes the sequential control of runway configurations and the arrival and departure service rates in order to minimize congestion costs; and a stochastic queuing model of airport congestion, which quantifies flight delays as a function of flight schedules and arrival and departure service rates.

The main novelty of the ICUSM is the integration of capacity utilization decisions in the optimization of the slot allocation process. It assumes that any change in flight schedules may lead to changes in how an airport should be tactically operated. This assumption is motivated by the interdependencies between flight scheduling and airport operations. In fact, for any set of prevailing conditions, the optimal capacity utilization policy to be selected (i.e. runway configuration and arrival and departure service rates) will depend on the number of arrivals and departures scheduled for that period as well as on the demand in the preceding and following periods. This model attempts to capture these interrelationships without relying on an arbitrary, fixed, and administrative notion of declared capacity.

Jacquillat and Odoni (2015a) applied the model at John F. Kennedy International (JFK) Airport, showing that very substantial delay reductions can be achieved, without shifting the scheduled time of any flight by more than 15 or 30 minutes. Moreover, they proved that the integrated approach used by ICUSM may provide significant benefits when compared to a typical sequential approach where scheduling and operational decisions are made successively.

The models are individually presented in appendix as well as the algorithm used to solve the integrated model.

3 An Optimization Approach for Airport Slot Allocation under IATA Guidelines

3.1 Introduction

Air traffic growth coupled with limitations in available infrastructure and air traffic management operations have created severe imbalances between demand and capacity at the world's busiest airports. Limited capacity at busy airports can result in congestion and schedule unreliability. In 2015, 19% and 18% of commercial flights experienced an arrival delay of 15 minutes or more in the United States and in Europe, respectively (FAA, 2016), with the trend pointing upward in both cases. Moreover, these constraints can impose long-term economic impacts due to lost demand, higher airfares, and limitations in airlines' route development.

In the absence of supply-side interventions aimed to increase system capacity through infrastructure expansion and/or operational improvements, airport congestion mitigation may require the use of demand management mechanisms. Demand management consists of interventions that limit the number of flights scheduled at busy airports at peak hours. These interventions fall conceptually into two categories: (i) economic approaches, which involve market-based mechanisms such as congestion pricing and slot auctions, and (ii) administrative approaches, which involve non-monetary adjustments to airport flight schedules imposed by a designated schedule coordination entity. The former has received significant attention in the economics and operations research literature (see, e.g., Ball et al., 2006, and Gillen et al., 2016, for reviews). On the economics side, much research has aimed to design optimal congestion pricing schemes (Brueckner, 2002; Pels and Verhoef, 2004; Czerny and Zhang, 2011; Czerny and Zhang, 2014) and to compare price-based vs. quantity-based auction mechanisms (Brueckner, 2009; Czerny, 2010; Basso and Zhang, 2010; Verhoef, 2010). On the operations research side, Ball et al. (2006) and Harsha (2009) developed optimization models to support auctioning of airport slots. In practice, however, existing demand management practices are almost exclusively based on administrative approaches.

The foremost demand management mechanism currently in use is the schedule coordination process developed by the International Air Transport Association (IATA). With minor variations depending on geographic location and local or regional regulations (e.g., in Europe), this process, with essentially identical guidelines and priority rules, is currently applied at 175 "schedule coordinated" ("Level 3") airports worldwide, including the great majority of the busiest ones outside the United

States (IATA, 2017). In Europe, for instance, the process is mandatory for coordinated airports and driven by the EU regulation (EC, 1993). Despite there are some differences, between the IATA guidelines and the EU regulation, in general the rules and priorities are very similar.

This chapter proposes a novel model, the Priority-based Slot Allocation Model (PSAM), to optimize slot allocation decisions based on slot availability and airline slot requests. The model minimizes the costs of schedule coordination to the airlines and other airport stakeholders, as measured by the *displacement* from airline requests, while accounting for the many priorities and requirements included in the IATA guidelines. It develops an efficient computational approach that makes it possible to apply the model at even medium-size airports, with up to 100,000 aircraft movements per year, for an entire season of operations. The chapter then presents detailed applications at the Cristiano Ronaldo International Airport of Madeira and the Francisco Sá Carneiro Airport of Porto, both located in Portugal, using fine-grain data on airline slot requests. The computational results suggest that such applications may offer important benefits by accepting all slot requests, while significantly reducing the largest flight displacement, the total schedule displacement, and the number of flights displaced that are necessary to accommodate all requests. Before summarizing the chapter's contributions in more detail in Section 3.1.2, we provide additional information on current schedule coordination processes and procedures.

3.1.1 IATA Slot Allocation Process

This section provides some background on the slot allocation process endorsed by IATA, including: (i) an overview of its different stages and the scope of this chapter; (ii) some important definitions and concepts; (iii) its priorities and requirements; and (iv) the main sources of complexity of the problem considered.

The IATA schedule coordination process is carried out bi-annually to provide airlines with access to schedule coordinated airports. This access is granted in the form of a landing or takeoff "slot", defined as the permission to use the full range of an airport's infrastructure to perform aircraft arrivals or departures on a specific day and at a specific time. For each season ("Summer" or "Winter"), the IATA slot allocation process involves five main steps:

(i) *Setting of Declared Capacity*: Each airport provides the values of its "declared capacity", which specifies the number of slots made available in each time interval of a day. Declared capacities are commonly specified as hourly limits on the number of flight movements (landings and takeoffs) that may be scheduled, but may also be specified at a finer level of granularity for (i) different elements of the airport (e.g., runway capacity, apron capacity and terminal capacity), (ii) different types of movements (e.g., arrivals, departures and total), and (iii) different “block” durations (e.g., capacities per hour, per 15-minute period, per 5-minute period, as well as per day, per week, per month, or per year), etc. The declared capacities of each schedule coordinated airport are announced about one year before the start of each season.

(ii) *Slot Requests*: The airlines submit their desired schedule of flights at each airport to the schedule coordinator for the upcoming season. Flight scheduling requests are submitted in one of two forms. If a flight is to take place at least five times over a season on the same day of the week and at the same time of the day, the corresponding request must be submitted in the form of a “series of slots” (e.g. a flight that takes place every Monday in July and August at 10:15). If the flight does not satisfy these criteria, the request is provided as an “individual slot”. Requests for series of slots are submitted approximately five months in advance of each season. Individual slots may be requested up to the actual day of operations and may be awarded depending on availability of slots at the requested time.

(iii) *Initial Slot Allocation*: At each airport, the schedule coordinator is tasked to perform the initial slot allocation in an “unbiased, transparent and non-discriminatory” way. No contact is allowed between the slot coordinator and the airlines. The allocation of slots is performed solely on the basis of the priorities and requirements specified by the IATA guidelines. The coordinator provides the resulting initial schedule to the airlines about four months before the start of each season. Only *series of slots* are allocated at this stage.

(iv) *Schedule Coordination Conference*: Potential adjustments to the initial slot allocation are made in the semi-annual IATA Slot Conferences (SC), which are attended by airline representatives, slot coordinators, airport representatives and other interested parties. These adjustments primarily involve the resolution of conflicts stemming from the timing of slots allocated across multiple airports, and, if relevant, disputes among airlines competing for these slots.

(v) *Slot Return*: The airlines may “return” slots to the coordinator until two months before the start of each season, if they decide that they will not use these slots. They can also request and perform other schedule adjustments, subject to approval by the schedule coordinator, until the day of operation. The objective is to correct any inefficiencies (from the airline's standpoint) resulting from the schedule coordination process.

This work developed in this chapter is focused on the initial slot allocation (Step 3), which is the most critical step in the entire process. Consistently with the scope of the initial slot allocation, we only consider the series of slots, and not the individual slots, which are only allocated after the SC. The allocation problem takes as inputs the airport's declared capacities (Step 1) and airline requests for series of slots for the upcoming season (Step 2). Based on these inputs, the schedule coordinator allocates slots to the airlines, subject to slot availability and the priorities and requirements specified by the IATA guidelines. First, the schedule needs to exhibit some regularity: (i) it is *required* that all flights belonging to the same series of slots (i.e., slots for the same flight on the same day of the week, at least five times over the season) be given a slot at the same time of the day, and (ii) it is *recommended* that different series of slots belonging to the same slot request code (i.e., identical series of slots for different days of the week submitted together – Section 3.2.1) be given slots at the same time of the day across multiple days of the week. Second, the turnaround times between flights need to be maintained between pairs of arriving and departing flights (or, at least, adjusted with minimal changes) to maintain the connectivity of airlines' networks of flights. Third, slot allocation must follow a set of priorities specified as "primary criteria" for allocation, as well as, when necessary, some "additional criteria".

The primary criteria for allocation define priorities across four groups of slots, and allocate the series of slots sequentially across these groups. Highest priority is given to historic slots, or grandfathered slots, i.e., series of slots already held by the airline in the previous equivalent season (Winter or Summer) and operated at least 80% of the time (known as the "use-it-or-lose-it" rule). Second priority is given to "change-to-historic" slots, i.e., flights for which an airline holds a historic slot, but requests a change (e.g., in timing or in aircraft type). Some change-to-historic requests allow the flight to be scheduled at any time between the requested slot time and the historic slot time, while some others allow the flight to be scheduled only at the requested time or, if the requested time is unavailable, at the historic time. Third priority is given to "new entrant" airlines, which, according to the guidelines, must receive up to 50% of the remaining slots (if demand is sufficient). The definition of new entrant is based on market penetration (e.g., an airline that holds fewer than five slots in a certain day of the season qualifies as new entrant for that day) and, potentially, other policy considerations (e.g., flights requested for on underserved routes). Finally, any remaining slots are allocated to the "other" requests, i.e., requests that do not qualify under the first three priority classes.

In addition to these primary criteria, the IATA guidelines also provide a set of secondary criteria to differentiate slots belonging to the same priority class. Foremost, slot requests that extend existing year-round operations are given priority over new slot requests. Moreover, slot allocation decisions can also consider other factors, such as the type of route (existing route vs. new route), the type of service (scheduled, charter and cargo), the size of the aircraft (narrow-body vs. wide-body), and the type of market (domestic, regional and long haul). These criteria are mostly used for tie-breaking purposes, i.e., to determine *which* flights to schedule if several solutions achieve the same outcome per the primary criteria. For this reason, they are beyond the scope of this chapter: we focus on determining the optimal outcome (or outcomes) of the initial slot allocation based on the primary criteria.

Note that the series of slots introduce interdependencies between the different days of operations. The problem cannot be decomposed into a series of independent problems that involve making slot allocations separately for each individual day. This might result in flights belonging to the same series of slots (or to separate series of slots belonging to the same slot request code – see Section 3.2.1) being scheduled at different times on different days. Instead, slot allocation has to be performed *for the entire season all at once*. From a modeling standpoint, this creates coupling constraints across the slot requests from one day to another. From a computational standpoint, this increases greatly the size, and, in turn, the complexity of the underlying models.

3.1.2 Prior Work and Contributions

Current slot allocation procedures are assisted by specialized software (e.g., PDC SCORE). Slot requests are typically treated sequentially in an *ad hoc* basis, which provides only limited visibility on the whole set of slot requests and their interactions. In recent years, optimization models have emerged in the literature to support the schedule coordination process. Experience with these models (limited to date) has suggested that there is potential for improving significantly slot allocation decisions. An extensive review of the current slot allocation models is provided in Zografos et al. (2017), which divides slot allocation models into two different categories: single-airport slot allocation models (see below) and network-wide slot allocation models (Castelli et al. 2011a; Castelli et al. 2011b; Corolli et al. 2014). This chapter presents a new single-airport model, a focus consistent with current practice for the initial slot allocation (Step 3 of the process). The single-airport focus makes it possible, from a computational standpoint, to consider series of slots, all at once, across the *entire season*, which is critical to ensuring compliance with the requirements of the IATA guidelines.

Single-airport slot allocation models have been the subject of significant recent research. First, some research has focused on developing optimization models to determine the appropriate level of the declared capacity to minimize delays, maximize airline profitability and maximize passenger welfare (Swaroop et al., 2012; Vaze and Barnhart, 2012). Then, Jacquillat and Odoni (2015a) and Pyrgiotis and Odoni (2016) developed optimization models to inform and assess scheduling adjustments at US airports by quantifying their effects on airline schedules of flights and resulting airport on-time performance. However, the US focus of this research did not motivate consideration of series of slots and of some of the IATA guidelines. Finally, Zografos et al. (2012) developed an optimization model of slot allocation that captured, for the first time, some of the "primary criteria" of the IATA guidelines by considering priorities for historic and new entrant slots (but with no separate treatment of change-to-historic slots). Moreover, that paper was the first to consider explicitly the notion of a series of slots, thus increasing the time scale of decision-making to the entire season. The aforementioned models share the objective of minimizing total displacement from the slot times requested by the airlines, measured as the absolute total difference between the allocated and requested slot times. This was recently extended to incorporate fairness objectives between the airlines, both at schedule-coordinated airports and at US airports (Zografos and Jiang, 2016; Jacquillat and Vaze, 2018). From a computational standpoint, these models have been applied for a single day of operations at some of the busiest airports (Pyrgiotis and Odoni, 2016; Jacquillat and Odoni 2015a) or for an entire season at only moderately busy airports. To the best of our knowledge, the largest airport where slot allocation decisions have been addressed using exact optimization methods is the Heraklion International Airport in Greece, which operates fewer than 50,000 aircraft movements per annum (Zografos et al., 2012).

In this chapter we extend previous work in three major ways. First, from a modeling standpoint, the Priority-based Slot Allocation Model (PSAM) is the first model that optimizes slot allocation decisions at schedule-coordinated airports, while fully complying with the priorities across all the slot classes specified in IATA's primary criteria. In addition, it adds two new slot allocation objectives, namely minimizing the number of slots rejected and minimizing the number of slots displaced, and it explicitly captures the trade-offs between all the objectives underlying slot allocation decisions. Second, from a computational standpoint, the PSAM provides an efficient model formulation and solution approach that ensures, for the first time, the tractability of an exact optimization approach for slot allocation problems for an entire season at mid-size airports. This has enabled the implementation of the model at the airport of Porto, Portugal, which operates roughly 100,000 aircraft movements per annum. This volume of traffic is over twice as large as the one at the busiest airports previously considered in the literature (Zografos et al., 2012). Third, from a practical standpoint, we perform comparisons with real-world slot allocation outcomes at the airports of Madeira and Porto, Portugal, by leveraging fine-grain data on airline slot requests and the resulting airport slot allocation decisions. Results suggest that the model improves the decisions made by slot coordinators by reducing the largest displacement experienced by any flight by 10 to 25 minutes, the total schedule displacement by 4% to 27%, and the number of flights displaced by 1% to 7%, depending on scheduling and capacity patterns. Extensive computational tests also provide detailed characterizations of the optimal slot allocation decisions, and show that the optimization model can provide benefits across *all* priority classes of the IATA guidelines. In summary, this chapter provides a model-based tool that can yield significant improvements in slot allocation processes at congested airports by supporting and optimizing schedule coordination decisions based on quantitative objectives.

The remainder of this chapter is organized as follows. Section 3.2 describes and synthesizes the slot allocation data from the airports of Madeira and Porto. Section 3.3 formulates the model, including the technical aspects of capturing the IATA guidelines and priorities in optimizing the allocation of slots. Section 3.4 strengthens the formulation and quantifies the resulting improvements in computational performance. Section 3.5 presents the computational results and their implications for schedule coordination practice. Section 3.6 summarizes our work and indicates directions for future research.

3.2 Case Study Data

The case studies reported in this chapter are based on slot request and slot allocation data for the Summer Season of 2014 (from March 30, 2014 to October 25, 2014) at the airports of Madeira and Porto, Portugal. Slot allocation in Portugal is performed by ANA Aeroportos de Portugal. Both airports have runway declared capacities for each 15-minute period and 60-minute period on a 5-minute rolling horizon basis, reported in Table 3.1. In Madeira, no more than 14 movements, 7 arrivals and 7 departures can be scheduled between 8:00 and 9:00, between 8:05 and 9:05, between 8:10 and 9:10, etc., and no more than 6 movements, 4 arrivals and 4 departures can be scheduled between 8:00 and 8:15, between 8:05 and 8:20, between 8:10 and 8:25, etc. In Porto, no more than 20 movements can be scheduled per hour, and no more than 7 movements can be scheduled per 15-minute period (note that there is no separate limit on the number of arrivals and of departures in Porto).

Table 3.1– Madeira Airport Declared Capacities for the Summer Season of 2014

Declared Capacity Indicators	Madeira Airport	Porto Airport
Total flight movements / 60 min	14	20
Limit on number of arrivals / 60 min	7	20
Limit on number of departures/ 60 min	7	20
Total flight movements / 15 min	6	7
Limit on number of arrivals / 15 min	4	7
Limit on number of departures/ 15 min	4	7

In addition to the runway declared capacities shown in Table 3.1, the airports of Madeira and Porto are also subject to terminal and apron capacity constraints and to noise restrictions. These constraints were not considered in this work, as they are not limiting at these two airports. In fact, in the solutions provided by the slot coordinators, no slots were displaced due to these capacity constraints.

3.2.1 Format of Slot Requests

The slot requests from the airlines follow the standard code provided by Chapter 6 of the Standard Schedules Information Manual (IATA, 2014). Table 3.2 shows a sample of these slot request codes in Madeira, as provided by the airlines to the slot coordinators. This includes, for each slot request: (i) the priority class (historic, change-to-historic, new entrant or other); (ii) the arrival/departure flight ID; (iii) the start and end date of operations; (iv) the days of the week the slots will be operated; (v) the type of aircraft and expected number of seats; (vi) the requested arrival and departure times; (vii) (iv) the days of the week the slots will be operated; (v) the type of aircraft and expected number of seats; (vi) the requested arrival and departure times, (vii) the origin and final stop ("destination") of the aircraft's overall itinerary (viii) the last airport that the aircraft will visit before landing at the subject airport (in this case Madeira), as well as the next airport the aircraft will visit after departing from the subject airport, and (ix) the type of flight (e.g., J for scheduled passenger flight, C for chartered passenger flight, etc.). For instance, the third request shown in Table 3.2 corresponds to an aircraft itinerary that starts in Paris Orly (ORY), flies to Madeira from Porto (OPO), then flies from Madeira to OPO, and, eventually, ends at ORY.

For the purpose of the model developed in this chapter, the relevant information corresponds to Points (i), (iii), (iv) and (vi) above that specify the days and times of the slots requested and the priority of each slot request. The remaining information may be used in other stages of the slot allocation process.

In the remainder of this section we discuss further the five codes shown in Table 3.2. First, note that they belong to different priority classes. Specifically, Request Code 1 corresponds to a historic slot

(Code F), Request Codes 2 and 3 correspond to change-to-historic slots (Codes CR and CL), Request Code 4 to a new entrant (Code B), and Request Code 5 to a slot that does not belong to any of the aforementioned priority classes. The difference between the two types of change-to-historic requests is that, when an airline submits a CR code, it is willing to accept any slot between the requested and historic slot times whenever the requested slot is not available while, when an airline submits a CL code, it is only willing to accept the historic slot time when the requested slot is not available. Note that additional codes may be used by the airlines (e.g., to specify the slots that operate on a year round basis), but this chapter focuses on these five main types of requests.

Arrivals and departures may be requested in the same slot request (e.g., Request Codes 1, 2 and 3), or solely arrivals (Request Code 4) or solely departures (Request Code 5). In most instances, airlines include both an arrival and a departure in the same slot request code to ensure that the slot coordinator can maintain appropriate connection times and control the number of aircraft in the apron at any time. However, some slot requests are made specifically for each type of movement (if this is allowed by the coordinator). Such requests typically come from the bigger airlines, which derive operating flexibility from the large number of aircraft they may be operating at the subject airport.

Airlines may request more than one series of slots (for different days of the week) in the same slot request code. For instance, Request Code 1 includes only one series of slots, to be operated on Sundays (indicated by the “1000000” code). In contrast, Request Code 2 includes seven series of slots, one per day of the week (indicated by the “1234567” code). Request Code 3 includes four series of slots, to be operated on Sundays, Tuesdays, Thursdays and Saturdays (“1030507”). Note that these series of slots differ only with respect to the day of the week – all their other parameters are identical. As mentioned in the introduction, the IATA guidelines *recommend* that series of slots requested in the same slot request code be allocated to the same time on the different days of the week.

Overall, each slot request code may include a large number of slots. For instance, Request Code 1 applies to the entire season (the 30 weeks between March 30 and October 25), and consists of one series of arrival and departure slots. Therefore, this request involves a total of 30x2x1=60 slots. Similarly, Request Code 2 corresponds to seven series of slots of arrivals and departures over 10 weeks (between March 30 and June 1), and therefore consists of 10x2x7=140 slots. The structure of slot requests thus creates important combinatorial complexities in slot allocation, which motivate the optimization approach proposed in this chapter.

Table 3.2 – Sample slot request codes at Madeira

Req. Code	Priority	Arr. ID	Dep. ID	Start Date	End Date	Days of week	Seats	Aircraft
1	F	XY001	XY002	30MAR	25OCT	1000000	180	320
2	CR	XY003	XY004	30MAR	01JUN	1234567	180	320
3	CL	XY005	XY006	01APR	21OCT	1030507	180	320
4	B	XY007	---	01JUL	23SEP	0200500	170	320
5	N	---	XY008	30MAR	25OCT	0004000	130	319

Req. Code	Origin	Previous Stop	Arr. Time	Dep. Time	Next Stop	Destination	Arr. Type	Dep. Type
1	LIS	LIS	0800	0830	LIS	LIS	J	J
2	OPO	LIS	1000	1100	LIS	OPO	J	J
3	ORY	OPO	1535	1610	OPO	ORY	J	J
4	OPO	OPO	1830	---	---	---	C	---
5	---	---	---	1100	PDL	PDL	---	J

3.2.2 Summary of the Data

A total of 13,196 slots were requested by the airlines at Madeira for the Summer of 2014, distributed across 836 series of slots and 332 slot request codes. About 50% of the slots were requested as historic slots, 35% as change-to-historic slots, 1.5% as new entrant slots and 13.5% as other slots. At Porto, the number of slots requested was equal to 40,597, distributed across 1,920 series of slots and 882 slot request codes. About 64% of the slots were requested as historic slots, 21% as change-to-historic slots, 1.6% as new entrant slots and 13.4% as other slots. Figure 3.1 shows the demand for slots and the slot limits during the busiest day of the Summer of 2014 (August 18 in Madeira and August 1 in Porto). Figures 3.1a and 3.1b (resp., Figures 3.1c and 3.1d) show the number of slots requested in the most recent previous 60 minutes and 15 minutes, respectively, for every 5-minute period of the day in Madeira (resp. Porto). Note that several periods of the day are subject to imbalances between demand and capacity, as the number of slots requested exceeds the declared capacity at the airport. For these specific days, such imbalances occur during 17 5-minute periods in Madeira and 14 in Porto. This is explained by the fact that the morning peak period is slightly longer in Madeira than in Porto, and that the Madeira airport also imposes separate limits on arrivals and on departures.

Imbalances between demand and capacity are also found during other days of the season. Table 3.3 shows the number of 5-minute periods with imbalances between demand and capacity during the entire season, for the Madeira and Porto airports. Specifically, it reports the number of days with imbalances by day of the week and by month of the season. Note that, at both airports, most imbalances occur in July, August and September, i.e., during the peak of the Summer season. Turning to the days of the week, the capacity restrictions are binding only on Mondays and Thursdays in

Madeira, while imbalances are found on any day of the week in Porto (the busiest days being Thursdays, Fridays, and Sundays). Note that, even though demand for slots may fall below declared capacities on the least busy days of the season, these days need to be considered in the slot allocation decisions nonetheless because of the interdependencies between slots over the days of the week. These interdependencies across the entire season underscore the complexity of the problem of finding slot allocation solutions that will comply consistently with the values of the airport's declared capacities, as well as with the IATA priorities and requirements regarding slot series and slot requests. This again motivates the development and use of large-scale optimization techniques to help coordinators make slot allocation decisions more efficiently and faster.

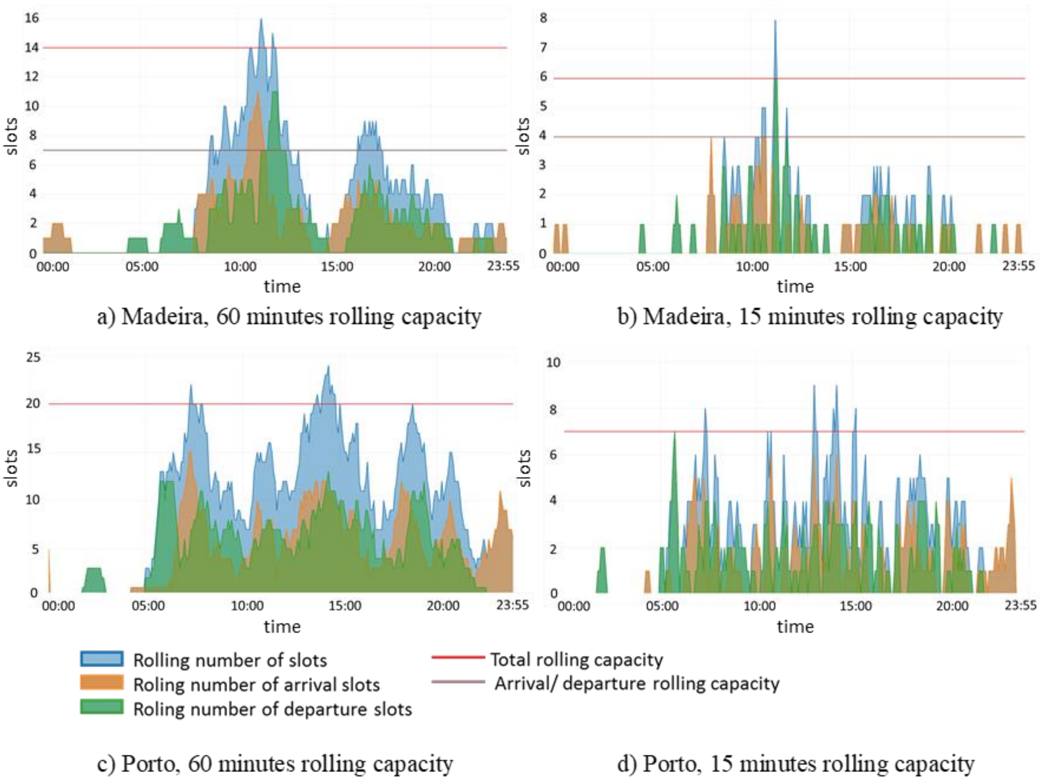


Figure 3.1 - Demand for slots at Madeira and Porto airports for the busiest day of the Summer of 2014

At the two airports considered, the imbalances between demand and declared capacities, although significant, can be addressed by rescheduling slot requests to different times of the day, without rejecting any slot request. Nonetheless, the model presented in the next section considers the possibility of rejecting slots. This is motivated by two considerations. First, it provides a more general framework that can also be applied at airports where total demand is so high that some slot requests may have to be rejected to satisfy the declared capacity constraints. Second, even at airports where total demand falls below total declared capacity, it may be necessary to consider slot rejections when the IATA slot priorities are considered. For example, some new entrant requests may be rejected if they exceed 50% of the remaining capacity after slots have been allocated to historic and change-to-historic series.

Table 3.3 – Number of 5-minute periods where demand exceeds capacity in Madeira and Porto airports

Madeira Airport				Porto Airport			
Weekday	Number of Imbalances	Month	Number of Imbalances	Weekday	Number of Imbalances	Month	Number of Imbalances
Mon	279	Mar/Apr	66	Mon	124	Mar/Apr	82
Tue	0	May	94	Tue	90	May	74
Wed	0	Jun	78	Wed	120	Jun	71
Thu	450	Jul	130	Thu	175	Jul	232
Fri	0	Aug	136	Fri	212	Aug	323
Sat	0	Sep	136	Sat	68	Sep	118
Sun	0	Oct	89	Sun	182	Oct	71

3.3 The Priority-based Slot Allocation Model (PSAM)

In this section, we present PSAM. This optimization model takes as inputs the values of airport declared capacities (see Table 3.1) and the slot requests of the airlines, as described in Section 3.2.1. It then produces a schedule that minimizes the displacement from the slot requests, subject to the constraints resulting from the priorities and requirements specified by the IATA guidelines and from the capacities declared by the airports. We present sequentially its inputs, decision variables, baseline formulation, and adjustments to account explicitly for the IATA guidelines.

3.3.1 Inputs

a) Sets

$\boldsymbol{T} = \{1,...,T\}$: set of time periods, indexed by *t*

$\boldsymbol{D} = \{1,...,D\}$: set of days, indexed by *d*

$\boldsymbol{S} = \{1,...,S\}$: set of slot requests codes, indexed by *i* or *j*

$\boldsymbol{S}_{arr} \subset \boldsymbol{S}$: set of arrivals

$\boldsymbol{S}_{dep} \subset \boldsymbol{S}$: set of departures

$\boldsymbol{P} \subset \boldsymbol{S} \times \boldsymbol{S}$: set of slot request pairs $(i, j) \in S \times S$ such that there is a connection between *i* and *j*

$\boldsymbol{C} = \{1,...,C\}$: set of capacity time scales, indexed by *c*

The set ***T*** consists of the number of periods of the day plus a "sink" period at the end of the time horizon (period *T*) used for slots rejected. Note that the set ***S*** processes the series of slots provided in the same request code together. As described in the introduction and in Section 3.2.1, the IATA guidelines *require* that the flights requested in the same series of slots (i.e., for a given day of the week) be allocated at the same time of day, and *recommend* that the series of slots requested in the same request code (i.e., same series of slots for different days of the week) be allocated at the same time of the day. By processing together all the slot requests in a slot request code, the PSAM actually also *requires* the latter. Moreover, for the slot requests that include both an arrival and a departure (see Table 3.2), we include two different requests in the set ***S***, and track the types of movements and connections with the subsets $\boldsymbol{S_{arr}}$ and $\boldsymbol{S_{dep}}$ and the set of flight pairs ***P***, respectively. Finally, the set ***C*** includes all the different time scales that are subject to declared capacity constraints (e.g., in the case of Madeira or Porto shown in Table 3.1, it includes a 60-minute time scale and a 15-minute time scale).

b) Parameters:

$$A_{it} = \begin{cases} 1, & \text{if slot } i \text{ is requested to operate no earlier than period } t \\ 0, & \text{otherwise} \end{cases}$$

$$B_{id} = \begin{cases} 1, & \text{if slot } i \text{ is requested to operate on day } d \\ 0, & \text{otherwise} \end{cases}$$

C_{tdc}^{dep} = departure capacity at period t, day d and time scale c

C_{tdc}^{arr} = arrival capacity at period t, day d and time scale c

C_{tdc}^{T} = total capacity at period t, day d and time scale c

L_c = length of time scale c

$T^{\max}$ = maximum allowable increase in the connection time of two slots in comparison to the requested connection time

$T^{\min}$ = maximum allowable decrease in the connection time of two slots in comparison to the requested connection time

Note that the connection parameters $T^{\max}$ and $T^{\min}$ are not provided in the data, but are considered in the model to either force connection times to be maintained to their requested values, or to explore the trade-off between changes in connection times and resulting schedule displacement (see Section 3.5.1.*b* for more details). We also assume that the final "sink" period (i.e., period *T*) has infinite departure, arrival and total capacities (since this period is only used for flights rejected and is not capacity-constrained).

3.3.2 Decision Variables

During the slot allocation process, each slot request may be subject to four possible outcomes: (i) a slot request may be allocated at the requested time; (ii) a slot request may be allocated at a later time; (iii) a slot request may be allocated at an earlier time; (iv) a slot request may be rejected. The decision variables capture these four outcomes. First, the decision variables Y_{it} indicate the allocated time of each slot requested. Then, the decision variables X_i^+ and X_i^- define the displacement of each slot request, and the decision variables W_i^+ and W_i^- indicate if a slot is displaced or not Last, the decision variables that indicate whether a slot is rejected or not are denoted as Z_i . The logical relationships between variables will be defined as part of the model's constraints in Section 3.3.4.

$$Y_{it} = \begin{cases} 1, & \text{if slot } i \text{ is rescheduled to arrive / depart no earlier than period } t \\ 0, & \text{otherwise} \end{cases}$$

X_i^+ = displacement of slot *i* if rescheduled to a later time period

X_i^- = displacement of slot *i* if rescheduled to an earlier time period

$$W_i^+ = \begin{cases} 1, & \text{if slot } i \text{ is displaced to a later time} \\ 0, & \text{otherwise} \end{cases}$$

$$W_i^- = \begin{cases} 1, & \text{if slot } i \text{ is displaced to a earlier time} \\ 0, & \text{otherwise} \end{cases}$$

$$Z_i = \begin{cases} 1, & \text{if slot } i \text{ is rejected} \\ 0, & \text{otherwise} \end{cases}$$

Note that each row of the Y_{it} variables is of the form (1,..,1,0,..,0), instead of the (0,..,0,1,0,..,0) format used in Zografos et al. (2012) and Pyrgiotis and Odoni (2016). This follows the formulation in Jacquillat and Odoni (2015a), which is inspired by some efficient air traffic flow management optimization models (Bertsimas and Patterson, 1998). The other decision variables must satisfy one of the following combinations of values: (i) $X_i^+ = X_i^- = W_i^+ = W_i^- = Z_i = 0$ if a slot request *i* is scheduled at the requested time; (ii) $X_i^+ > 0$; $X_i^- = 0$; $W_i^+ = 1$; $W_i^- = 0$; $Z_i = 0$ if a slot request *i* is displaced to a later time; (iii) $X_i^+ = 0$; $X_i^- > 0$; $W_i^+ = 0$; $W_i^- = 1$; $Z_i = 0$ if slot request *i* is displaced to an earlier time; (iv) $X_i^+ = X_i^- = W_i^+ = W_i^- = 0$; $Z_i = 1$ if slot request *i* is rejected.

3.3.3 Objective

We consider the following objective function, where w_1, w_2 and w_3 represent weighting parameters:

$$min\ w_1 \sum_{i \in \boldsymbol{S}} \sum_{d \in \boldsymbol{D}} B_{id} Z_i + w_2 \max_{i \in \boldsymbol{S}} \left(X_i^+, X_i^- \right) + w_3 \sum_{i \in \boldsymbol{S}} \sum_{d \in \boldsymbol{D}} B_{id} \left(X_i^+ + X_i^- \right) + \sum_{i \in \boldsymbol{S}} \sum_{d \in \boldsymbol{D}} B_{id} \left(W_i^+ + W_i^- \right) \tag{3.1}$$

This objective function includes four terms. The first corresponds to the total number of slots rejected. The second indicates the maximum displacement imposed on any slot. The third quantifies the total displacement across all the flights throughout the season. The last term captures the total number of slots displaced. The parameters w_1, w_2 and w_3 are used to set the relative weight of each of these four objectives.

In most of this work, we consider weights such that $w_1 >> w_2 >> w_3 >> 1$. In other words, the first goal is to ensure that all the slot requests will be scheduled and none will be rejected. After that, the main objective is to allocate these slots as close as possible from the requested times. This is captured by our objectives of minimizing the maximum displacement, and then the total displacement. The order of these two objectives is mainly motivated by equity reasons, as it ensures that no slots will incur a disproportionately large displacement. Finally, we add to this model the novel objective of minimizing the number of slots displaced, to reduce the complexity of the process and ease the subsequent negotiations during the slot conference. This order among the four objectives is motivated by current practice from the slot coordinators and the interests of the airlines, and is consistent with the existing literature (Zografos et al., 2012; Jacquillat and Odoni, 2015a; Pyrgiotis and Odoni, 2016).

However, our modeling framework is flexible and can capture other priorities among the different objectives of PSAM. Eliciting the entire efficient frontier across four objectives is computationally very intensive. We elicit in Section 3.5.1 the efficient frontier between the two main objectives of PSAM (i.e., the maximum displacement and the total displacement) through the ε-constraint method. Additionally, we analyze in Section 3.5.1.e) the impact of different priorities, and show that PSAM can provide solutions that reflect alternative trade-offs among the four objectives considered here.

3.3.4 Constraints

The constraints to include in the model are as follows:

$$Y_{i1} = 1 \qquad \forall i \in \boldsymbol{S} \qquad (3.2)$$

$$Y_{it} \geq Y_{i,t+1} \qquad \forall i \in \boldsymbol{S}, t \in \boldsymbol{T} \qquad (3.3)$$

$$Y_{i,T} = Z_i \qquad \forall i \in \boldsymbol{S} \qquad (3.4)$$

$$\sum_{t \in \boldsymbol{T}} \left(Y_{it} - A_{it} \right) = X_i^+ - X_i^- + \sum_{t \in \boldsymbol{T}} \left(1 - A_{it} \right) Z_i \qquad \forall i \in \boldsymbol{S} \qquad (3.5)$$

$$W_i^+ \geq Y_{it} - A_{it} - Z_i \qquad \forall i \in \boldsymbol{S}, t \in \boldsymbol{T} \qquad (3.6)$$

$$W_i^- \geq -Y_{it} + A_{it} \qquad \forall i \in \boldsymbol{S}, t \in \boldsymbol{T} \qquad (3.7)$$

$$\sum_{i \in \boldsymbol{S}_{arr}} \sum_{t=s}^{s+L_c} \left(Y_{it} - Y_{i,t+1} \right) B_{id} \leq C_{sdc}^{arr} \qquad \forall s \in \boldsymbol{T} \mid s < T - L_c + 1, d \in \boldsymbol{D}, c \in \boldsymbol{C} \qquad (3.8)$$

$$\sum_{i \in \boldsymbol{S}_{dep}} \sum_{t=s}^{s+L_c} \left(Y_{it} - Y_{i,t+1} \right) B_{id} \leq C_{sdc}^{dep} \qquad \forall s \in \boldsymbol{T} \mid s < T - L_c + 1, d \in \boldsymbol{D}, c \in \boldsymbol{C} \qquad (3.9)$$

$$\sum_{i \in \boldsymbol{S}} \sum_{t=s}^{s+L_c} \left(Y_{it} - Y_{i,t+1} \right) B_{id} \leq C_{sdc}^{T} \qquad \forall s \in \boldsymbol{T} \mid s < T - L_c + 1, d \in \boldsymbol{D}, c \in \boldsymbol{C} \qquad (3.10)$$

$$\sum_{t \in \boldsymbol{T}} \left(Y_{jt} - Y_{it} \right) - \sum_{t \in \boldsymbol{T}} \left(A_{jt} - A_{it} \right) \geq T^{min} - T \left(Z_i + Z_j \right) \qquad \forall i, j \in \boldsymbol{P} \qquad (3.11)$$

$$\sum_{t \in \boldsymbol{T}} \left(Y_{jt} - Y_{it} \right) - \sum_{t \in \boldsymbol{T}} \left(A_{jt} - A_{it} \right) \leq T^{max} + T \left(Z_i + Z_j \right) \qquad \forall i, j \in \boldsymbol{P} \qquad (3.12)$$

$$X_i^+, X_i^- \in \mathbb{N}_0 \qquad (3.13)$$

$$Y_{it}, W_i^+, W_i^-, Z_i \in \{1, 0\} \tag{3.14}$$

Constraints (3.2) ensure that all the slots requested are allocated to some period. Constraints (3.3) ensure that the variables Y are non-increasing in t, which is consistent with their definition. Constraints (3.4) to (3.7) define the logical relationships between the variables $X_i^+, X_i^-, Y_{it}, W_i^+, W_i^-, Z_i$ (see details below and in Proposition 1). Constraints (3.8), (3.9) and (3.10) ensure that the airport capacities for arrivals, departures and total number of movements are never exceeded over any day d. The formulation of these three constraints is similar to the one presented in Zografos et al. (2012) and enables the consideration of capacities for different time scales c. Constraints (3.11) and (3.12) ensure that the time between two connected flights does not increase/decrease by more than the allowable limits. The term $T(Z_i + Z_j)$ ensures that the slots rejected are not constrained by the connectivity parameters T^{max} and T^{min}: Since T is a large number, which corresponds to the total number of periods in a day (i.e. the maximum number of periods by which a time connection may increase or decrease), the constraints are necessarily not violated for rejected slots (i.e., when Z_i=1 or Z_j=1). Finally, constraints (3.13) and (3.14) specify the domains of the decision variables.

We now describe how the logical relationships between the different variables are captured through constraints (3.4) to (3.7). At a high level, constraints (3.4) define whether a slot is rejected or not, which happens when the slot is displaced to the last time period T. Constraints (3.5) define the displacement of each slot as the difference between the requested time (the parameters A_{it}) and the allocated time (the decisions Y_{it}). The term $\sum_{t \in \boldsymbol{T}} (1 - A_{it}) Z_i$ forces the displacement of a rejected slot to be equal to zero to avoid double-counting the penalty associated with flight rejections. Constraints (3.6) and (3.7) define the binary variables W_i, which indicate whether a slot is displaced or not, by forcing each to be equal to 1 if there is any discrepancy between slot i's scheduled and requested times, and if slot request i is not rejected. More specifically, we show in Proposition 1 that the optimal solution can only be of four types, and characterize these four cases.

Proposition 1: Let us consider an optimal solution to the model. Then, for each flight request i in $\boldsymbol{S}$, one of the following four properties is satisfied:

(i) $Z_i = X_i^+ = X_i^- = W_i^+ = W_i^- = 0$, i.e., flight request i is allocated to the requested time.

(ii) $Z_i = 0$ and $X_i^+ > 0,\ X_i^- = 0$ and $W_i^+ = 1,\ W_i^- = 0$, i.e., flight request i is rescheduled to a later time.

(iii) $Z_i = 0$ and $X_i^+ = 0$, $X_i^- > 0$ and $W_i^+ = 0$, $W_i^- = 1$, i.e., flight request *i* is rescheduled to an earlier time.

(iv) $Z_i = 1$ and $X_i^+ = X_i^- = W_i^+ = W_i^- = 0$, i.e., flight request *i* is rejected.

Proof: Let us consider an optimal solution $Z_i / X_i^+ / X_i^- / W_i^+ / W_i^- / Y_{it}$. Let us also consider a given slot request $i \in \boldsymbol{S}$.

We first show that either $X_i^+ = 0$ or $X_i^- = 0$. By contradiction, we assume that $X_i^+ > 0$ and $X_i^- > 0$. Then, without loss of generality, we assume that $X_i^+ \geq X_i^-$. We define a new solution X_i^{+*} / X_i^{-*} as follows: $X_i^{+*} = X_i^+ - X_i^-$, $X_i^{-*} = 0$, and $X_j^{+*} = X_j^+$, $X_j^{-*} = X_j^-$ for all $j \neq i$. This solution is a feasible solution, as it satisfies Constraints (3.5) (because $X_i^+ - X_i^- = X_i^{+*} - X_i^{-*}$), and all other constraints are unchanged. Moreover, we have $\max_{i\in \boldsymbol{S}} \left(X_i^{+*}, X_i^{-*} \right) \leq \max_{i\in \boldsymbol{S}} \left(X_i^+, X_i^- \right)$, and $\sum_{i\in \boldsymbol{S}} \left(X_i^{+*} + X_i^{-*} \right) < \sum_{i\in \boldsymbol{S}} \left(X_i^+ + X_i^- \right)$. This contradicts the fact that X_i^+ / X_i^- is an optimal solution.

We now investigate the case where $Z_i = 0$. Per the result above, this case is separated into three subcases: (a) $X_i^+ = X_i^- = 0$; (b) $X_i^+ > 0$ and $X_i^- = 0$; and (c) $X_i^+ = 0$ and $X_i^- > 0$. We investigate these three cases sequentially, and show that they are equivalent to properties (i), (ii), and (iii), respectively.

First, let us consider the case where $Z_i = 0$ and $X_i^+ = X_i^- = 0$ (case (a)). We have $\sum_{t\in \boldsymbol{T}} \left(Y_{it} - A_{it} \right) = 0$ (constraint (3.5)) and, since Y_{it} and A_{it} are both of the form (1,1,…,1,0,…,0), this implies that $Y_{it} = A_{it}$ for all $t \in \boldsymbol{T}$. From constraints (3.6) and (3.7), we obtain $W_i^+ \geq 0$ and $W_i^- = 0$ i.e., $W_i^+ = W_i^- = 0$ because the solution is optimal. This proves (i).

Second, let us consider the case where $Z_i = 0$ and $X_i^+ > 0$, $X_i^- = 0$ (case (b)). We have $\sum_{t\in \boldsymbol{T}} \left(Y_{it} - A_{it} \right) = X_i^+ > 0$ (constraint (3.5)), so $\sum_{t\in \boldsymbol{T}} Y_{it} > \sum_{t\in \boldsymbol{T}} A_{it}$. Since Y_{it} and A_{it} are both of the form (1,1,…,1,0,…,0), this implies that $Y_{it} \geq A_{it}$ for all $t \in \boldsymbol{T}$ and there exists at least one period $s \in \boldsymbol{T}$ such that $Y_{is} > A_{is}$. From constraints (3.6) and (3.7), we obtain $W_i^+ \geq 1$ and $W_i^- \geq 0$ i.e., $W_i^+ = 1$ and $W_i^- = 0$ because the solution is optimal. This proves (ii). We proceed similarly in the case where $Z_i = 0$ and $X_i^+ = 0$, $X_i^- > 0$ (case (c)) and prove (iii).

Finally, we investigate the case where $Z_i = 1$. From constraints (3.3) and (3.4), we have $Y_{it} = 1$ for all *t*. From constraints (3.5), we have $\sum_{t \in \boldsymbol{T}} (Y_{it} - 1) = X_i^+ - X_i^-$, i.e., $X_i^+ - X_i^- = 0$. Since the solution is optimal, this implies that $X_i^+ = X_i^- = 0$ (this can be easily checked by contradiction as done in the first part of this proof). Since $Z_i = 1$, we have $Y_{it} - A_{it} - Z_i \leq 0$ for all $t \in \boldsymbol{T}$, so constraints (3.6) become $W_i^+ \geq 0$ and, since the solution is optimal, $W_i^+ = 0$. Moreover, since $Y_{it} = 1$ for all $t \in \boldsymbol{T}$, $-Y_{it} + A_{it} \leq 0$ for all $t \in \boldsymbol{T}$, so constraints (3.7) become $W_i^- \geq 0$ and, since the solution is optimal, $W_i^- = 0$. This proves (iv) and concludes the proof.

We now turn to the additional constraints that arise from the consideration of the successive IATA priority classes.

3.3.5 IATA Priority Constraints

The IATA guidelines require consideration of the different priorities assigned to the various types of slot requests. This is achieved through a sequential approach that first allocates historic series of slots, followed by the "change to historic" series of slots, followed by new entrant slots, and, finally, by the remaining slots.

From a modeling standpoint, this is formulated through a lexicographic approach that solves each of these four sub problems in sequence. Accordingly, we divide the model into four sub-models, one for each slot priority. For each sub-model, we store the optimal value of the objective function (*OV*) for the slot priority considered (Equation (3.1)). The reason why we fix this optimal value rather than the decision variables is that there may be more than one optimal solution for the priority class considered, so fixing the slot allocation decisions might be constraining the resulting slot allocation for the following priority classes more than necessary. Specifically, we partition the set of slot requests $\boldsymbol{S}$ into subsets $\boldsymbol{S_H}$, $\boldsymbol{S_{CH}}$, $\boldsymbol{S_{NE}}$, $\boldsymbol{S_{OS}}$, which include the historic slots, "change-to-historic" slots, new entrants, and other slots, respectively. We first solve the model for historic slots (see Section 3.3.5.a), and store the optimal value of the objective function, denoted by OV_H. Note that OV_H is typically equal to 0, as historic slots are typically not displaced (see below). Then, we add Constraints (3.15) to the sub-models of "change-to-historic", new entrants and other slots, to ensure that the allocation of slots remains optimal for the historic slots. We then turn to the sub-model for change-to-historic slots (Section 3.3.5.b), store the optimal value of the objective function OV_{CH}, add Constraints (3.16) to the sub-models of new entrants and other slots, solve the sub-model of new entrants, store the optimal value of the objective function OV_{NE}, add Constraints (3.17) to the sub-model of other slots, and solve the sub-model of other slots.

$$w_1 \sum_{i \in \boldsymbol{S}_H} \sum_{d \in \boldsymbol{D}} B_{id} Z_i + w_2 \max_{i \in \boldsymbol{S}_H} \left(X_i^+, X_i^- \right) + w_3 \sum_{i \in \boldsymbol{S}_H} \sum_{d \in \boldsymbol{D}} B_{id} \left(X_i^+ + X_i^- \right) + \sum_{i \in \boldsymbol{S}_H} \sum_{d \in \boldsymbol{D}} B_{id} \left(W_i^+ + W_i^- \right) = OV_H \tag{3.15}$$

$$w_1 \sum_{i \in \boldsymbol{S_{CL}} \cup \boldsymbol{S_{CR}}} \sum_{d \in \boldsymbol{D}} B_{id} Z_i + w_2 \max_{i \in \boldsymbol{S_{CL}} \cup \boldsymbol{S_{CR}}} \left(X_i^+, X_i^- \right) + w_3 \sum_{i \in \boldsymbol{S_{CL}} \cup \boldsymbol{S_{CR}}} \sum_{d \in \boldsymbol{D}} B_{id} \left(X_i^+ + X_i^- \right) + \sum_{i \in \boldsymbol{S_{CL}} \cup \boldsymbol{S_{CR}}} \sum_{d \in \boldsymbol{D}} B_{id} \left(W_i^+ + W_i^- \right) = OV_{CH} \tag{3.16}$$

$$w_1 \sum_{i \in \boldsymbol{S_{NE}}} \sum_{d \in \boldsymbol{D}} B_{id} Z_i + w_2 \max_{i \in \boldsymbol{S_{NE}}} \left(X_i^+, X_i^- \right) + w_3 \sum_{i \in \boldsymbol{S_{NE}}} \sum_{d \in \boldsymbol{D}} B_{id} \left(X_i^+ + X_i^- \right) + \sum_{i \in \boldsymbol{S_{NE}}} \sum_{d \in \boldsymbol{D}} B_{id} \left(W_i^+ + W_i^- \right) = OV_{NE} \tag{3.17}$$

From a computational standpoint, this lexicographic approach improves the tractability of the model by decomposing it into four smaller problems. On the negative side, it does not search for alternative solutions that could potentially meet airlines' requests to a greater extent through even modest adjustments in the IATA requirements. This can be addressed in future research by relaxing some of the constraints derived from the IATA guidelines and quantifying the resulting impacts on slot allocation efficiency.

Some additional constraints are now needed to capture the rules mandated by the IATA guidelines for each of the priority classes. These are formulated below

a) Historic Slots

Since the historic slots have absolute priority, they can be simply allocated their requested times, without running an optimization model. We will therefore have $X_i^+ = X_i^- = W_i^+ = W_i^- = Z_i = 0$ for all slot *requests i* in $\boldsymbol{S_H}$. As a result, the optimal value of the objective function OV_H will be equal to 0. Note that this assumes that there is sufficient capacity to accommodate all the historic slots requests, which will be the case in practice, as long as the declared capacity does not decrease from year to year.

b) Change-to-historic Slots

As described in Section 3.2.1, changes to historic slots may be requested in two different ways: if the requested slots are not available, then "CR" code requests can be scheduled at any time between the requested and historic slot times, while "CL" code requests can only be scheduled at the requested or the historic slot times. We denote the subsets of $\boldsymbol{S_{CH}}$ that include all "CR" code requests and all "CL" code requests by $\boldsymbol{S_{CR}}$ and $\boldsymbol{S_{CL}}$, respectively. We also introduce a new model parameter H_{it}, which indicates whether the historic slot time of *i* was no earlier than period *t* (this parameter has the same

(1,…,1,0,…,0) format as the parameter *A* and the decision variable *Y*). Constraints (3.18) and (3.19) ensure that CR slots are allocated between the historic and the requested slot time. Constraints (3.20) and (3.21) ensure that CL slots are allocated either at the requested slot time or at the historic slot time.

$$X_i^+ \leq \sum_{t \in \boldsymbol{T}} \left(H_{it} - A_{it} \right) W_i^+ \qquad \forall i \in \boldsymbol{S}_{CR} \qquad (3.18)$$

$$X_i^- \leq \sum_{t \in \boldsymbol{T}} \left(A_{it} - H_{it} \right) W_i^- \qquad \forall i \in \boldsymbol{S}_{CR} \qquad (3.19)$$

$$X_i^+ = \sum_{t \in \boldsymbol{T}} \left(H_{it} - A_{it} \right) W_i^+ \qquad \forall i \in \boldsymbol{S}_{CL} \qquad (3.20)$$

$$X_i^- = \sum_{t \in \boldsymbol{T}} \left(A_{it} - H_{it} \right) W_i^- \qquad \forall i \in \boldsymbol{S}_{CL} \qquad (3.21)$$

In addition, we must make sure that the connection times between two change-to-historic slots lie between the requested connection times and the historic connection times. We introduce a set $\boldsymbol{P_C}$ in $\boldsymbol{P}$ that includes all the pairs of flights in $\boldsymbol{S_{CH}}$. We define, for each pair (i, j), the requested connection time and the historic connection time, denoted by ΔA_{ij} and ΔH_{ij}, respectively. Mathematically, this is expressed as $\Delta A_{ij} = \sum_{t \in \boldsymbol{T}} A_{jt} - A_{it}$ and $\Delta H_{ij} = \sum_{t \in \boldsymbol{T}} H_{jt} - H_{it}$. We then ensure appropriate connection times with Constraints (3.22) and (3.23):

$$\sum_{t \in \boldsymbol{T}} \left(Y_{jt} - Y_{it} \right) \geq \min\left(\Delta A_{ij}, \Delta H_{ij} \right) - T\left(Z_i + Z_j \right) \qquad \forall i \in \boldsymbol{P_c} \qquad (3.22)$$

$$\sum_{t \in \boldsymbol{T}} \left(Y_{jt} - Y_{it} \right) \geq \min\left(\Delta A_{ij}, \Delta H_{ij} \right) + T\left(Z_i + Z_j \right) \qquad \forall i \in \boldsymbol{P_c} \qquad (3.23)$$

With these new constraints, the model is solved with respect to the objective function (3.1) to minimize displacement across all the change-to-historic slots. Note that the constraints presented in this section will be included not only in the sub-model of change-to-historic slots, but also in the sub-models of new entrants and other slots, in order to ensure that the allocation of change-to-historic slots is consistent with the rules of IATA.

c) New Entrant Slots

According to the IATA guidelines, after allocating historic slots and change-to-historic slots, 50% of the remaining slots must be allocated to new entrants, unless the number of requests from new entrants falls below that percentage. To capture this requirement, we denote the remaining arrival capacity, departure capacity and total capacity for new entrants by $C_c^{arr,NE}$, $C_c^{dep,NE}$ and $C_c^{T,NE}$ respectively. Expressions (3.24) define $C_c^{T,NE}$ as 50% of the remaining slots after the allocation of slots to historic and change-to-historic requests. This value is computed through preprocessing before solving the sub-model for new entrants. Analogous expressions are also used to define $C_c^{arr,NE}$ and $C_c^{dep,NE}$.

$$C_c^{T,NE} = \frac{1}{2} \sum_{d \in \boldsymbol{D}} \sum_{s=1}^{T-L_c+1} \left(C_{sdc} - \sum_{i \in \boldsymbol{S}_H \cup \boldsymbol{S}_{CR} \cup \boldsymbol{S}_{CL}} \sum_{t=s}^{s+L_c} \left(Y_{it} - Y_{i,t+1} \right) B_{id} \right) \qquad \forall c \in \boldsymbol{C} \qquad (3.24)$$

In order to ensure that the total capacity for new entrants, $C_c^{T,NE}$, is not exceeded, we add Constraints (3.25) to the model. Analogous constraints are specified for $C_c^{arr,NE}$ and $C_c^{dep,NE}$.

$$\sum_{d \in \boldsymbol{D}} \sum_{s=1}^{T-L_c+1} \sum_{i \in \boldsymbol{S}_{NE}} \sum_{t=s}^{s+L_c} \left(Y_{it} - Y_{i,t+1} \right) B_{id} \leq C_c^{T,NE} \qquad \forall c \in \boldsymbol{C} \qquad (3.25)$$

In cases where more slots than available are requested by new entrants, some of the requests will be rejected. The slots rejected at this stage will be reconsidered at the next stage of the allocation process with the same priority as all the other remaining slots.

Note that, in practice, the number of slots rejected at this stage will almost always be equal to zero, as airports typically have periods of the day with very low demand. As a result, the total number of slots remaining after the changes-to-historic allocation will typically be a large number, likely to exceed the number of slots requested by new entrants. This is why some slot coordinators tend to simplify the new entrant rule by simply prioritizing all new entrant slot requests over the “remaining” slot requests. In such cases, we can apply directly the model presented in Sections 3.3.1 to 3.3.4 (without Constraints (3.25)) to minimize displacement of the new entrant slots. One can then check the resulting solution, and make adjustments, if necessary.

d) Remaining Slots

Remaining slots are allocated according to the model formulation presented in Sections 3.3.1 to 3.3.5. In this stage, we allocate slots with no priority, including the new entrant slots rejected in the previous stage.

3.3.6 Model Size

Table 3.4 shows the number of binary and integer variables and the number of constraints in the model presented in this section, as well as the resulting size of the model for the cases analyzed at Madeira and Porto. Note that the size of the sets $\boldsymbol{T}$ and $\boldsymbol{D}$ is identical from airport to airport. Typically, each slot period corresponds to 5 minutes, so each day is divided into T=288 periods. The length of the season is defined by IATA. For instance, the Summer Season of 2014 lasted from March 30 to October 25, which corresponds to D=210 days. In contrast, the set $\boldsymbol{S}$ varies from airport to airport. For that season, Madeira received 332 slot request codes, 275 of which consisted of flight pairs of movements. Therefore, $\boldsymbol{S}$ and $\boldsymbol{P}$ comprise $275 \times 2 + (332 - 275) = 607$ elements and 275 elements, respectively. In the case of Porto, the airport received 882 slot requests codes, 312 of which consisted of flight pairs of movements. Therefore, $\boldsymbol{S}$ and $\boldsymbol{P}$ comprise 1194 elements and 312 elements, respectively. The size of the airport therefore has a significant impact on the size of the model.

Table 3.4 – Size of the Model

Model Indicators	Size of the Model	Madeira Case Study	Porto Case Study
Number of binary variables	$ST+3S$	176 346	347454
Number of integer variables	2S	1 212	2388
Number of constraints	$2\boldsymbol{ST}+3\boldsymbol{S}+\boldsymbol{P}+3\sum_{c\in\boldsymbol{C}}\left(\boldsymbol{T}-L_c+1\right)\boldsymbol{D}$	705 839	1 046 358

3.4 Model Enhancement

In this section, we strengthen the formulation of the PSAM by introducing new constraints that reduce the gap between the integer formulation of the model and its linear relaxation, thus significantly improving its computational performance. This enables, in turn, the consideration of exact optimization methods to solve real-world instances at mid-size airports, such as Porto, in reasonable computational times.

3.4.1 Formulation Strengthening

As described in Section 3.4.2, the formulation introduced in Section 3.3 can lead to computational intractability even in cases involving modest-size airports. We therefore strengthen the formulation of the model presented in Section 3.3 to find better linear relaxations and, consequently, faster solution times. To this end, we introduce new constraints to remove portions of the feasible region that contain fractional solutions without eliminating any feasible integer solutions, thus ensuring that the optimal integer solution remains unchanged.

In fact, the linear relaxation of the model proposed in Section 3.3 yields an optimal solution equal to zero in all cases considered, as long as all demands can be accommodated through temporal shifts and there is no need to reject slot requests (i.e., total demand for slots falls below total capacity during each day). This property stems from Constraints (3.5), which make it possible to satisfy the capacity and connectivity constraints with displacement variables equal to zero $(X_i^+ = X_i^- = 0)$. To illustrate this point, Table 3.5 provides a small example with a single day divided into 5 periods, 3 slot requests (Slots 1 and 2 are requested in Period 2, and Slot 3 in Period 3), and a capacity of 1 slot per period. The declared capacity constraints are only violated in Period 2. An optimal solution consists of displacing Slot 1 from Period 2 to Period 1, and allocating Slot 2 to Period 2 and Slot 3 to Period 3, as requested. The optimal value of the total displacement is equal to 1.

Table 3.5 –Inputs, integer solution, and linear relaxation for a simple example

Period	Slot Requests					Solution of the Integer Model					Solution of the Linear Model				
	Slot 1	Slot 2	Slot 3	Slots Requested	Capacity	Slot 1	Slot 2	Slot 3	Slots Allocated	Capacity	Slot 1	Slot 2	Slot 3	Slots Allocated	Capacity
1	1	1	1	0	1	1	1	1	1	1	1	1	1	1	1
2	1	1	1	2	1	0	1	1	1	1	0.5	0.5	1	0.5	1
3	0	0	1	1	1	0	0	1	1	1	0.5	0.5	0.5	1	1
4	0	0	0	0	1	0	0	0	0	1	0	0	0.5	0.5	1
5	0	0	0	0	∞	0	0	0	0	∞	0	0	0	0	∞
						Displacement = 1					Displacement = 0				

However, the linear relaxation of the model yields a fractional solution, where half of Slots 1 and 2 are displaced to Period 3, and half of Slot 3 is displaced to Period 4. We then obtain from Constraints (5) a displacement of zero, shown in Equations (3.26) and (3.27). Note that this is a very simple example, more cases of fractional solutions are found when we consider larger problems.

$$X_1^+ - X_1^- = X_2^+ - X_2^- = (1-1)+(1-0.5)+(0-0.5)+(0-0)=0 \tag{3.26}$$

$$X_3^+ - X_3^- = (1-1)+(1-1)+(1-0.5)+(0-0.5) = 0 \tag{3.27}$$

In order to eliminate such fractional solutions, we replace constraints (3.5) with constraints (3.28) and (3.29) defined below. The purpose of these constraints is to force one of the displacement variables (X_i^+ and X_i^-) to be different from zero whenever the difference between Y_{it} and A_{it} is different from zero. We refer to the model developed in Section 3.3 (with Constraints (3.5)) as the "original model" and to the model obtained by replacing Constraints (3.5) with Constraints (3.28) and (3.29) as the "modified model".

$$\sum_{t \in \boldsymbol{T}} \left(1 - A_{it}\right) Y_{it} = X_i^+ + \sum_{t \in \boldsymbol{T}} \left(1 - A_{it}\right) Z_i \qquad \forall i \in \boldsymbol{S} \qquad (3.28)$$

$$\sum_{t \in \boldsymbol{T}} A_{it} \left(1 - Y_{it}\right) = X_i^- \qquad \forall i \in \boldsymbol{S} \qquad (3.29)$$

Note that the difference between Equations (3.28) and (3.29) yields exactly Equation (3.5), so any feasible solution of the modified model is also a feasible solution of the original model. We then prove in Proposition 2 that both models yield the same optimal integer solution.

Proposition 2: Any optimal solution of the original model (with Constraints (3.5)) is an optimal solution of the modified model (with Constraints (3.28) and (3.29)).

Proof: Let us consider an optimal solution of the original model, and show that it satisfies Constraints (3.28) and (3.29). We consider a given slot request *i*. From Proposition 1, we know that it satisfies one of four cases: (i) $Z_i = X_i^+ = X_i^- = 0$; (ii) $Z_i = 0$, $X_i^+ > 0$, $X_i^- = 0$; (iii) $Z_i = 0$, $X_i^+ = 0$, $X_i^- > 0$; or (iv) $Z_i = 1$, $X_i^+ = X_i^- = 0$.

Let us first consider case (i). As in the proof of Proposition 1, we have $\sum_{t \in \boldsymbol{T}} \left(Y_{it} - A_{it}\right) = 0$ (Constraints (3.5)) and, since Y_{it} and A_{it} are both of the form (1,1,…,1,0,…,0), this implies that $Y_{it} = A_{it}$ for all $t \in \boldsymbol{T}$. Since Y_{it} and A_{it} are both binary, we then have $Y_{it}\left(1 - A_{it}\right) = A_{it}\left(1 - Y_{it}\right) = 0$ for all *t*, so $\sum_{t \in \boldsymbol{T}} Y_{it}\left(1 - A_{it}\right) = 0$ and $\sum_{t \in \boldsymbol{T}} A_{it}\left(1 - Y_{it}\right) = 0$. This proves that Constraints (3.28) and (3.29) are satisfied.

We now consider case (ii). As in the proof of Proposition 1, we show that $Y_{is} > A_{is}$, and therefore $Y_{it} = 1$ for all periods *t* such that $A_{it} = 1$. We then have $A_{it}Y_{it} = A_{it}$ for all $t \in \boldsymbol{T}$ (because if $A_{it} = 0$, then $A_{it}Y_{it} = 0$, and if $A_{it} = 1$, then $Y_{it} = 1$ and $A_{it}Y_{it} = 1$). This gives the following equality: $\sum_{t \in \boldsymbol{T}} \left(1 - A_{it}\right) Y_{it} = \sum_{t \in \boldsymbol{T}} Y_{it} - \sum_{t \in \boldsymbol{T}} A_{it}Y_{it} = \sum_{t \in \boldsymbol{T}} Y_{it} - \sum_{t \in \boldsymbol{T}} A_{it} = X_i^+$. Therefore, $\sum_{t \in \boldsymbol{T}} \left(1 - A_{it}\right) Y_{it} = X_i^+ + \sum_{t \in \boldsymbol{T}} \left(1 - A_{it}\right) Z_i$ because $Z_i = 0$. This proves that constraint (3.28) is satisfied. Similarly, $\sum_{t \in \boldsymbol{T}} A_{it}\left(1 - Y_{it}\right) = \sum_{t \in \boldsymbol{T}} A_{it} - \sum_{t \in \boldsymbol{T}} A_{it}Y_{it} = 0$, proving that constraint (3.29) is satisfied. We can proceed similarly for case (iii) $X_i^+ = 0$ and $X_i^- > 0$ (i.e., where the flight is rescheduled to an earlier time).

Finally, we consider case (iv) where slot *i* is rejected. From constraints (3.3) and (3.4), we have $Y_{it} = 1$ for all $t \in \boldsymbol{T}$. Therefore, $\sum_{t \in \boldsymbol{T}} (1 - A_{it}) Y_{it} = \sum_{t \in \boldsymbol{T}} (1 - A_{it})$, so constraint (3.28) is satisfied. Similarly, $\sum_{t \in \boldsymbol{T}} A_{it} (1 - Y_{it}) = 0$, so constraint (3.29) is also satisfied. This concludes the proof.

Note that constraints (3.28) and (3.29) render infeasible the previous fractional solution that led to an optimal displacement of zero. For the example shown in Table 3.5, equations (3.30) to (3.33) provide the new values of X_i^+ and X_i^-, based on constraints (3.28) and (3.29), for the solution of the linear relaxation of the original model. The total displacement is now equal to 3, while the optimal integer solution is still equal to 1. Therefore, the new solution is not an optimal solution of the linear relaxation of the modified model. In turn, constraints (3.28) and (3.29) strengthened the integer programming formulation by tightening the feasible region of its linear relaxation.

$$X_1^+ = X_2^+ = (1-1) \times 1 + (1-1) \times 0.5 + (1-0) \times 0.5 + (0-0) \times 0 = 0.5 \tag{3.30}$$

$$X_1^- = X_2^- = 1 \times (1-1) + 1 \times (1-0.5) + 0 \times (1-0.5) + 0 \times (0-0) = 0.5 \tag{3.31}$$

$$X_3^+ = (1-1) \times 1 + (1-1) \times 1 + (1-1) \times 0.5 + (1-0) \times 0.5 = 0.5 \tag{3.32}$$

$$X_3^- = 1 \times (1-1) + 1 \times (1-1) + 1 \times (1-0.5) + 0 \times (1-0.5) = 0.5 \tag{3.33}$$

Table 3.6 shows the integer and linear solutions for the modified model. Even though the linear relaxation yields a fractional solution, the optimal value of the objective function of the linear relaxation is now equal to 1, and is, in fact, identical to the optimal value for the integer programming model. For large instances, the optimal values of the objective function may not be identical in all cases, but the modified model presented in this section results in a much smaller gap between the integer programming model and its linear relaxation. While we are still not able to guarantee integer solutions, the modified model leads to much faster computational times than the original model, as shown in the next section.

Table 3.6 – Integer solution and linear relaxation with the modified model

Period	Slot Requests					Solution of the Integer Model					Solution of the Linear Model				
	Slot 1	Slot 2	Slot 3	Slots Requested	Capacity	Slot 1	Slot 2	Slot 3	Slots Allocated	Capacity	Slot 1	Slot 2	Slot 3	Slots Allocated	Capacity
1	1	1	1	0	1	1	1	1	1	1	1	1	1	1	1
2	1	1	1	2	1	**0**	1	1	1	1	**0.5**	**0.5**	1	1	1
3	0	0	1	1	1	0	0	1	1	1	**0**	**0**	**1**	1	1
4	0	0	0	0	1	0	0	0	0	1	0	0	**0**	0	1
5	0	0	0	0	∞	0	0	0	0	∞	0	0	**0**	0	∞
						Displacement = 1					Displacement = 1				

Finally, we also added two valid inequalities to the model proposed (Constraints (3.34) and (3.35) below), which specify that any slot i is not displaced $\left(W_i^+ = W_i^- = 0\right)$ if the displacement variables are equal to zero $\left(X_i^+ = X_i^- = 0\right)$. We can prove formally that these constraints are satisfied by the optimal solution of the problem considered and restrict the feasible region of its linear relaxation, therefore improving the computational performance of the model. We omit this proof to avoid repeating the same procedure as above.

$$W_i^+ \leq X_i^+ \qquad \forall i \in \boldsymbol{S} \qquad (3.34)$$

$$W_i^- \leq X_i^- \qquad \forall i \in \boldsymbol{S} \qquad (3.35)$$

Note that we can now relax the integrality constraint for variables X_i^+ and X_i^-, because it will be automatically satisfied based on constraints (3.28) and (3.29). This reduces the number of integer variables and therefore further improves the computational performance of the model.

3.4.2 Computational Performance

We applied the model to the Madeira and Porto airports using CPLEX 12.5, implemented using GAMS as the modeling language. We looked for exact solutions (i.e., with a 0% optimality gap). The model was run with an i7 processor @ 3.6 GHz, 8 Gb RAM computer under a Windows 10 64 bit operating system.

Table 3.7 compares the computational performance of the original model (with constraints (3.5)) and the modified model (with constraints (3.28) and (3.29)), using data from Madeira for the first three weeks of the season. For this experiment, the model was solved with *all* the slot requests without priority considerations, with the objective of minimizing the total displacement only. As expected, the solution of the modified model is obtained in significantly lower computational times than that of the original model. For the smallest instances, that consisted of optimizing the slot allocation for only the first day of the season, the modified model is more than 20 times faster. As the size of the instance increases, the computational times of the modified model increase moderately, while those of the original model increase extremely fast. For the three-week instance, the original model does not terminate after 1 day (with an optimality gap of over 15%), while the modified model finds the optimal solution in only 86 seconds. Thus, the original model cannot be scaled to provide even approximate solutions for an entire season in reasonable times, while the modified model enables the consideration of problems of size realistic for larger airports. Note, moreover, that the optimal value of the objective function of the linear relaxation of the original model is always zero, while the modified model yields linear relaxation values that are much closer to the integer solution value.

Table 3.7 – Improvements in the model's performance with constraints (3.28) and (3.29)

Model Indicators	1 Day	1 Week	2 Weeks	3 Weeks
Number of Periods (T)	288	288	288	288
Number of Days (D)	1	7	14	21
Number of Slot Requests (S)	50	260	274	286
Integer Optimal Solution (Total Displacement)	3	26	52	81
Linear Relaxation Solution with constraints (5)	0	0	0	0
Linear Relaxation Solution with constraints (26) and (27)	2.8	24.9	48.2	71.4
Solving Time with Constraints (5) (sec)	237	2963 (49 min)	55538 (15.5 h)	>1day
Solving Time with Constraints (26) and (27) (sec)	10	26	32	86

Overall, the modified model was solved for the entire season (without considering the IATA priorities) in about 15 minutes for Madeira and 45 minutes for Porto. When considering the IATA priorities, the PSAM (with the objective function specified in Section 3.3) is solved in only 4 minutes for Madeira (1 to 2 minutes for each priority class) and in 8 minutes for Porto (2 to 3 minutes for each priority class). As expected the computational times are lower than in the instance where the IATA priorities are ignored, as the problem is decomposed into four smaller problems. Finally, note that the computational performance may be sensitive to the values of the weights w_1, w_2 and w_3. For instance, assigning a high weight to the number of slots displaced can increase the model's solving times by a factor of 2 to 4. This may be due to the fact that giving top priority to the minimization of the number of slots displaced results in many more optimal, or close-to-optimal, solutions.

The takeaways from these computational experiments are threefold. First, the modeling and computational approach developed in this chapter provides fast solutions for a full season of operations at medium-size airports and can thus be used in support of slot allocation processes.

Second, the PSAM can provide in reasonable computational times alternative solutions reflecting different weights attributed to the different objectives of slot allocation. Therefore, it permits exploration of the set of Pareto-optimal solutions when more than one objectives are considered. Third, the solution of the model remains tractable even when the IATA priority classes are not considered. This makes it possible to perform sensitivity analyses with respect to the IATA rules, such as those presented in Section 3.5.1 below. Thus, in addition to providing a near-term decision-making support tool for slot coordinators, the PSAM can also be used as a more strategic tool in support of policy-making to evaluate the impact of existing and alternative rules on the slot allocation process.

3.5 Model Results

In this section, we present the results obtained through the PSAM for the Madeira and Porto airports. We first investigate the sensitivity of the slot allocation outcomes to the various constraints imposed by the IATA guidelines, as well as to different priorities among the PSAM objectives. We then compare the model's solutions at the two airports with the allocation that was made in practice to quantify the potential benefits associated with the implementation of the model in support of slot coordination decisions. We do not consider the possibility of slots being rejected, as a feasible solution can be found at both airports considered without rejecting any requests.

3.5.1 Sensitivity Analysis to Slot Allocation Constraints and Objectives of PSAM

We first quantify the impact of the various requirements imposed by the IATA guidelines on the optimal displacement from the airline slot requests at Madeira airport. We consider here two measures of displacement: the maximum displacement $\max_{i \in \boldsymbol{S}} \left(X_i^+, X_i^- \right)$, and the total displacement $\sum_{i \in \boldsymbol{S}} \sum_{d \in \boldsymbol{D}} B_{id} \left(X_i^+ + X_i^- \right)$. We compute the Pareto optimal frontier for these two objectives (i.e., the set of solutions such that no other solution can improve one objective without worsening the other) by using an ε-constraint approach (Steuer, 1986; De Weck, 2004, Marler and Arora, 2004). In other words, we first compute the optimal value of the maximum displacement (positive or negative), denoted by δ^*, and the optimal value of the total displacement, denoted by Δ^*. We then minimize the total displacement while progressively increasing the value of the maximum displacement from δ^* by increments of 5 minutes (i.e., by the size of a slot time period), until the optimal value of the total displacement (i.e., Δ^*) is attained. This is summarized in the algorithm below.

Algorithm: PSAM Pareto-frontier elicitation via ε-constraint method

1: $min \max_{i \in \boldsymbol{S}} \left(X_i^+, X_i^- \right)$ (3.36)

subject to: PSAM constraints (Section 3.3)

$$\delta^* = \max_{i \in \boldsymbol{S}} \left(X_i^+, X_i^- \right) \quad (3.37)$$

Save δ^*

2: $min \sum_{i \in \boldsymbol{S}} \sum_{d \in \boldsymbol{D}} \left(X_i^+ + X_i^- \right) \times B_{id}$ (3.38)

subject to: PSAM constraints (Section 3.3)

$$\Delta^* = \sum_{i \in \boldsymbol{S}} \sum_{d \in \boldsymbol{D}} \left(X_i^+ + X_i^- \right) \times B_{id} \quad (3.39)$$

Save Δ^*

3: *j=1*

while $\Delta \leq \Delta^*$ *do*

$$min \sum_{i \in \boldsymbol{S}} \sum_{d \in \boldsymbol{D}} \left(X_i^+ + X_i^- \right) \times B_{id} \quad (3.40)$$

subjected to: PSAM constraints (Section 3.3)

$$\delta = \max_{i \in \boldsymbol{S}} \left(X_i^+, X_i^- \right) \quad (3.41)$$

$$\Delta = \sum_{i \in \boldsymbol{S}} \sum_{d \in \boldsymbol{D}} \left(X_i^+ + X_i^- \right) \times B_{id} \quad (3.42)$$

$$\delta \leq \delta^* + 5 \times j \quad (3.43)$$

Save Δ

j=j+1 (3.44)

end

a) Interdependencies between Slots over the Season

The complexity of the slot allocation process largely stems from the interdependencies between slot requests across the season. To ensure consistent treatment of all the flights in the same series of slots or in the same slot request code, the allocation of slots has to be performed all at once for the entire season, and not for each day individually. In order to quantify the impact of these interdependencies on the slot allocation decisions, we solved the model first individually for each single day, and then for the entire season. We did not consider the IATA priorities of slot classes at this stage. The Pareto-optimal solutions are shown in Table 3.8. The second column shows the total displacement obtained for the entire season by optimizing slot allocation decisions separately for each single day, while the third column provides the total displacement obtained for the entire season when PSAM is solved at once for all the days of the season, considering the interdependencies between slots on different days.

Note, first, that the optimal value of the maximum displacement is equal to 15 minutes in both cases. For this case, the total displacement for the solution that considers the entire season is 111% larger (i.e. 2.1 times more) than when solving individual days separately. Second, the solution that minimizes the total displacement yields a total displacement 37% larger (9,590 minutes vs. 6,990 minutes) and a maximum displacement 33% larger (40 minutes vs. 30 minutes) with the interdependencies than without them. Slot interdependencies thus have a strong impact on slot allocation decisions by restricting the solution set, therefore leading to significantly larger displacement values and rendering the computational requirements of the underlying models much more complex.

Table 3.8 - Pareto-optimal solutions for the Madeira airport without and with interdependencies

Max Disp (min)	Individual days	Entire season
15	7,385	15,600 (+111%)
20	7,160	11,815 (+65%)
25	7,050	10,775 (+53%)
30		9,755 (+40%)
35	6,990	9,750 (+39%)
40		9,590 (+37%)

b) Connectivity Parameters

According to the IATA guidelines, the coordinator shall maintain the connection times requested by the airlines between two connected slots (arrival-departure pair) or, if this is not feasible, shall endeavor to minimize the increase or decrease in connection times. In fact, there exists a trade-off between changes in connection times and schedule displacement. We demonstrate this trade-off in Table 3.9 and Figure 3.2 by showing the Pareto-optimal values of the maximal and total displacement for different values of the connectivity parameters $T^{\max}$ and $T^{\min}$, while maintaining $T^{\max} = T^{\min} = 0$. Values of $T^{\max} = T^{\min} = 0$ force the connection times to adhere to those requested by the airlines, while increasing them provides some additional flexibility in the slot allocation decisions. In all cases, the minimum value of the maximum displacement is equal to 15 minutes. The connectivity constraints have an important impact on the optimal total displacement, for any given value of the maximum displacement. If the maximum displacement is minimized, the total displacement can vary by as much as 23% in response to variations in the connectivity parameters. If the total displacement is minimized, the optimal value of the total displacement can vary by 15% and the optimal value of the maximum displacement can vary from 35 minutes to 45 minutes (a 30% increase).

Table 3.9 - Pareto-optimal solutions for the Madeira Airport with different connectivity parameters

Max Disp (min)	Connectivity Parameters				
	0	5	10	15	
	Total Displacement (min)				
15	15,600	13,690 (-12%)	13,210 (-15%)	12,660 (-19%)	11,945 (-23%)
20	11,815	10,520 (-11%)	9,605 (-19%)	9,375 (-21%)	9,315 (-21%)
25	10,775	9,210 (-15%)	9,105 (-16%)	9,040 (-16%)	8,765 (-19%)
30	9,755	8,845 (-9%)	8,660 (-11%)	8,600 (-12%)	8,480 (-13%)
35	9,750	8,695 (-11%)	8,565 (-12%)	8,500 (-13%)	8,310 (-15%)
40	9,590	8,595 (-10%)	8,435 (-12%)	8,365 (-13%)	8,240 (-14%)
45			8,430 (-12%)	8,305 (13%)	8,180 (-15%)

Note that the impact of the connectivity parameters on the optimal displacement is non-linear. In other words, limited flexibility in the connectivity parameters (e.g., 5 minutes) can lead to significant improvements in the total displacement, ranging from 9% to 15%. Further increases in the connectivity parameters by the same amount (e.g., 5 minutes) yield improvements in the resulting displacement of a much smaller magnitude. In fact, Figure 3.2 shows that more significant reductions in the schedule displacement can be achieved by increasing $T^{\max}$ and $T^{\min}$ from 0 to 5 minutes than from 5 minutes to infinite values. These results indicate that even small adjustments in the connection times can have a positive impact on overall displacement.

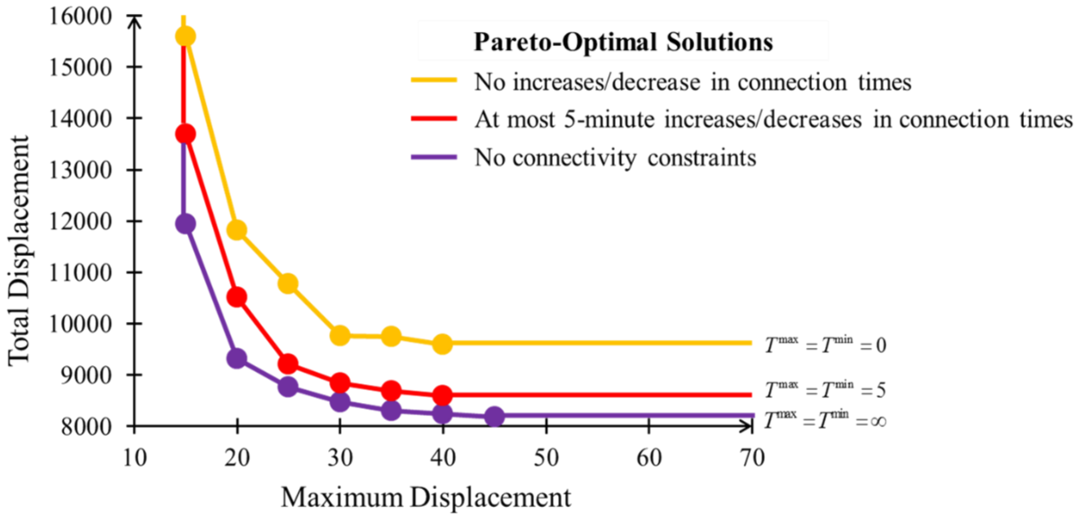


Figure 3.2 - Evolution of the Pareto-optimal frontiers for the Madeira Airport considering different connectivity parameters

c) IATA Priority Constraints

In the solutions obtained thus far, all flights were treated equally regardless of their priority class. For instance, up to 20-30% of historic slots are displaced, contradicting the grandfather rights accorded by the guidelines. When we require that historic slots cannot be displaced (constraint (3.15)), we obtain two Pareto-optimal solutions with a maximum displacement equal to 55 and 60 minutes, respectively, and a total displacement of 11,145 minutes and 10,805 minutes, respectively. In other words, the historic slot constraints result in very large increases in the maximum flight displacement (from 15 minutes to 55 minutes) and in significant increases in total schedule displacement, estimated of about 10%.

In addition to historic slots, the slot coordinator must allocate slots hierarchically across the three remaining priority classes: change-to-historic slots, new entrant slots, and other slots. For that purpose, we implement the full lexicographic model presented in Section 3.3, where each priority class is treated sequentially. In this case, we obtain a single Pareto-optimal solution, i.e., the maximum displacement and the total displacement can be jointly minimized and there is no trade-off between these two objectives. This solution has a maximum displacement equal to 70 minutes (a 12.5% improvement compared to 80 minutes in the slot coordinator's solution) and a total displacement of 11,620 minutes (a 4.3% improvement compared to 12,140 minutes in the slot coordinator's solution).

d) Summary of the Sensitivity Analysis to the IATA Constraints

Figure 3.3 shows the Pareto-optimal frontiers between the maximum displacement and the total displacement for the several instances dealt with in this section, assuming no increase/decrease in connection times, i.e., $T^{\max} = T^{\min} = 0$. Note that the slot allocation decisions are highly constrained by the IATA guidelines, as each one leads to significant increases in the maximum and/or the total displacement. First, the interdependencies between slots lead to an increase in the total displacement by an estimated 20% to 30%, as compared to the case where the slot requests are treated for each day independently. Second, the consideration of historic slots results in a smaller increase in the total displacement percent-wise, but in a very large increase in the maximum displacement, from 15 minutes to 55 minutes. Last, the IATA priorities increase the maximum displacement by another 25% and the total displacement by another 7%. This analysis highlights the impact of these priorities on slot allocation, and can then inform potential adjustments to the IATA guidelines to enhance the outcomes of schedule coordination.

Ultimately, the solution obtained with the model can improve current practice at schedule coordinated airports. First, the solution is reasonably similar to the slot coordinator's, confirming the realism of the PSAM model. But, it also results in a smaller maximum flight displacement *and* a smaller total schedule displacement, by an estimated 12.5% and 4.3%, respectively. Moreover, as discussed in the next section, this solution leaves all the connection times unchanged, unlike the one implemented in practice. Other considerations may, of course, have affected the slot coordinator's decisions, such as aircraft size or type of market served. Nonetheless, these considerations are expected to have limited impact, as they are explicitly intended for tie-breaking purposes. We further discuss the benefits of our modeling and computational approach in the next section.

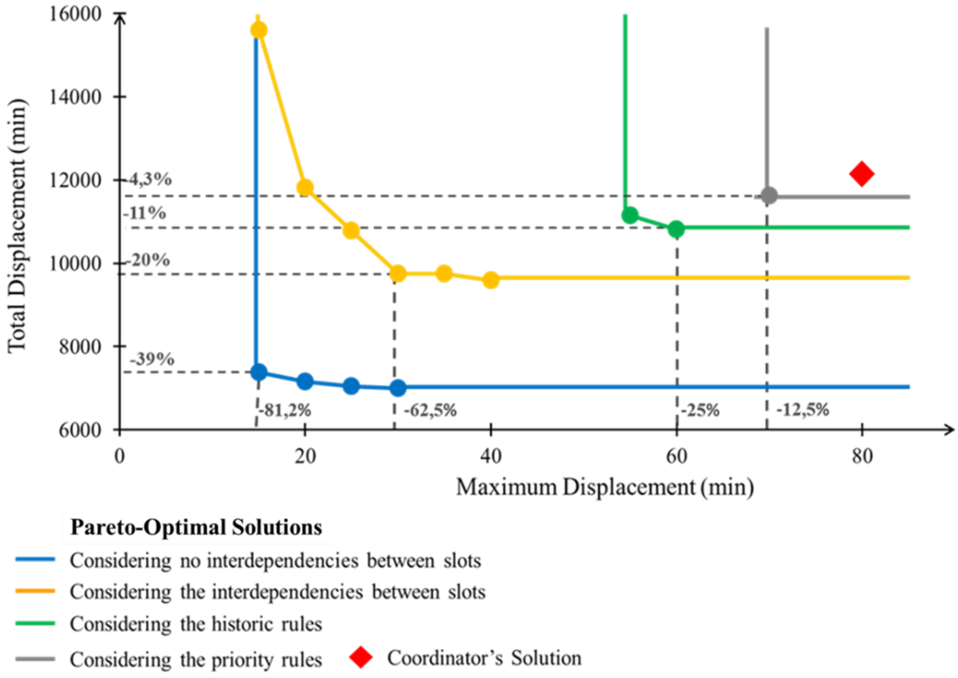


Figure 3.3 – Pareto-optimal frontier at the Madeira airport with different slot allocation constraints

e) Sensitivity Analysis to the Objectives of PSAM

As can be seen in Figure 3.3, when all the capacity constraints and priority rules are considered, the PSAM solution at the Madeira airport consists of a single point that minimizes simultaneously the maximum displacement and the total displacement. In other words, there is no trade-off between maximum and total displacement. This, of course, is not generally the case, as illustrated by Table 3.10, which summarizes the results of a sensitivity analysis with respect to the objective function of PSAM at Madeira and Porto. Note that the table also includes results for the third objective – minimizing the number of slot displacements (As already noted, no slot requests are rejected at either airport).

Table 3.10 points to the fact that, in the case of Porto (and in contrast to Madeira), there is indeed a trade-off between maximum displacement and total displacement captured by the two Pareto-optimal solutions, Sol. 1 and Sol 2, the first of which minimizes the former and the second the latter. Note that, nonetheless, both solutions improve *both* objectives, as compared to the coordinator's solution – and, in fact, also improve it with respect to the number of slots displaced.

More generally, Table 3.10 demonstrates that PSAM can support the selection of the appropriate slot allocation solution by assigning different priorities to the different objectives and quantifying the resulting trade-offs between these objectives. For example, Sol. 1 for Porto minimizes lexicographically first the maximum displacement, then the total displacement and, finally, the number of slots displaced, whereas Sol. 2 follows the order "total displacement, maximum, displacement, number of slots displaced", Sol. 3 the order "number of slots displaced, maximum displacement, total displacement" and Sol. 4 "number of slots displaced, total displacement, maximum displacement". Interestingly, all four solutions improve *all three* objectives by significant margins, as compared to the coordinator's solution. It is also noteworthy that after a certain number of slot displacements (Sol. 2), any further reductions in the number of displaced slots comes at a cost of increased total displacement.

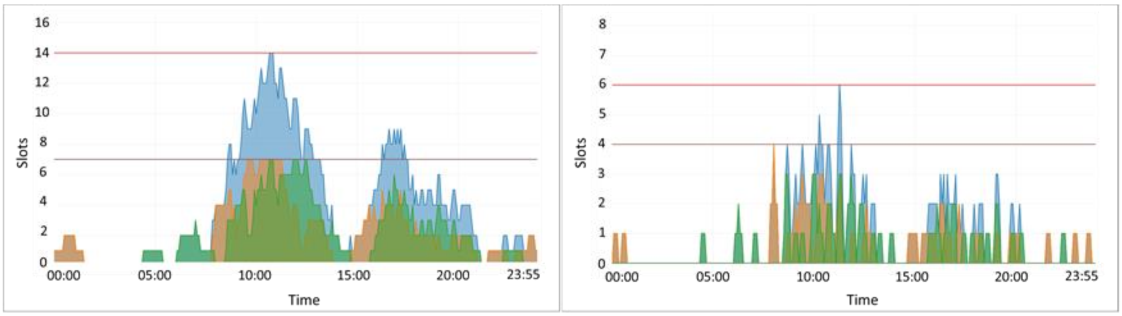
Table 3.10 – Sensitivity analysis to the PSAM objectives at the Porto and Madeira airports

Madeira Airport				Porto Airport			
Solution	Maximum Displacement (min)	Total Displacement (min)	Slots Displaced (slots)	Solution	Maximum Displacement (min)	Total Displacement (min)	Slots Displaced (slots)
Coordinator	80	12140	614	Coordinator	80	53140	2549
Sol. 1	70 **-13%**	11620 **-4%**	607 **-1%**	Sol. 1	55 **-31%**	38625 **-27%**	2379 **-7%**
				Sol. 2	60 **-25%**	37025 **-30%**	2303 **-10%**
Sol. 2	70 **-13%**	11745 **-3%**	572 **-7%**	Sol. 3	55 **-31%**	44620 **-16%**	1898 **-26%**
				Sol. 4	60 **-25%**	42930 **-19%**	1898 **-26%**

3.5.2 Results at the Madeira Airport and the Porto Airport

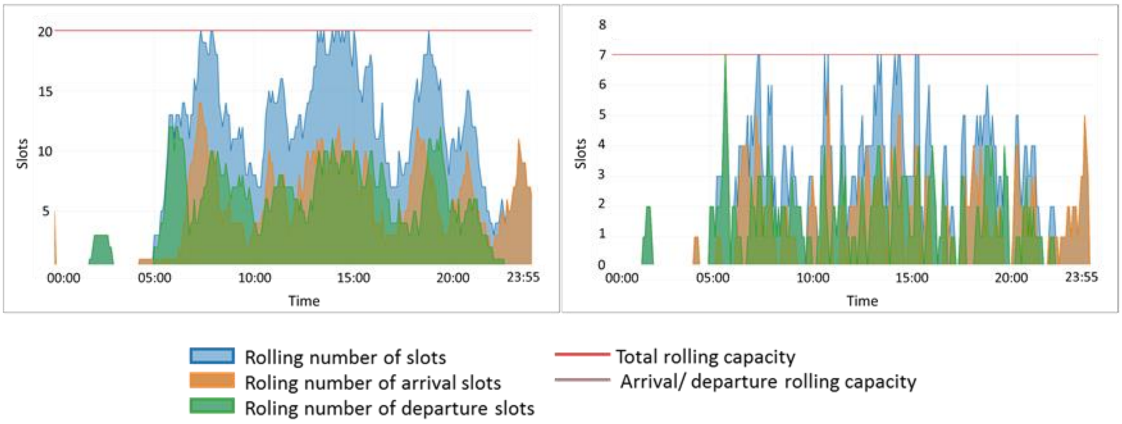
We now present in more detail the model's and the coordinator's solutions at the Madeira and Porto airports, where we maintain the order of objectives (maximum displacement, total displacement, number of slots displaced) indicated in Section 3.3, which is the most consistent with current practice and the existing literature. We discuss below the schedule of flights at each airport after the slot allocation, the displacement per priority class (historic slots, change-to-historic slots, new entrant slots, and remaining slots), the distribution of schedule displacement per day of week and per month, and the distribution of flight displacement across slot requests.

a) Flight schedule on the busiest day of the season



a) Madeira – 60-minute rolling capacity

b) Madeira – 15-minute rolling capacity



c) Porto – 60-minute rolling capacity

d) Porto – 15-minute rolling capacity

Figure 3.4 – **Number of slots allocated on the busiest day at Madeira and Porto airports**

Figure 3.4 shows the number of slots allocated per rolling period on the busiest day of the Summer of 2014 at Madeira and Porto airports. Specifically, Figures 3.4a and 3.4b (resp. Figures 3.4c and 3.4d) plot for Madeira (resp. Porto) the number of slots allocated over the preceding 60 minutes and the preceding 15 minutes, respectively, for every 5-minute period of the day. These plots are the counterparts of Figure 3.1, but indicate that the declared capacity is never exceeded over the day after slot allocation. Note, also, that the strict declared capacities lead to flat schedules at peak hours, especially at busier airports.

b) Displacement across slot priority classes

We now compare the PSAM solutions to the slot coordinator's solutions at both airports. Table 3.11 shows the maximum displacement, total displacement, number of slots displaced, and changes in connecting times in the slot coordinator's solutions ("Coord_Sol") and three model solutions: (i) the main solution that strictly complies with the IATA primary criteria and the requested connection times ("Mod_Sol"); (ii) an alternative solution that uses minor adjustments to the primary criteria to alleviate the displacement borne by new entrants ("Mod_Sol_NE"): and (iii) an alternative solution that allows for small changes in connection times to reduce the displacement ("Mod_Sol_Con"). We detail these three solutions and their rationale below. In general terms, they represent alternative options that can be used by decision-makers to select the most desirable solutions.

Table 3.11 – Coordinator and model solutions for slot allocation at Madeira and Porto airports

Madeira Airport

Solutions	Max Displacement (min)			Total Displacement (min)				Slots Displaced (slots)				Connections (min)	
	Chg. Hist	New Ent.	Other Slots	Total	Chg. Hist	New Ent.	Other Slots	Total	Chg. Hist	New Ent.	Other Slots	Max ΔCon	Sum ΔCon
Coordinator	35	0	80	12,140	4,930	0	7,210	614	354	0	260	80	670
Mod_Sol	35	0	70 -13%	11,620 -4%	4,780 -3%	0	6,840 -5%	607 -1%	343 -3%	0	264 +2%	0	0
Mod Sol Con	35	0	70 -13%	10,990 -10%	4,780 -3%	0	6,210 -14%	529 -14%	343 -3%	0	228 -12%	5 -94%	630 -6%

Porto Airport

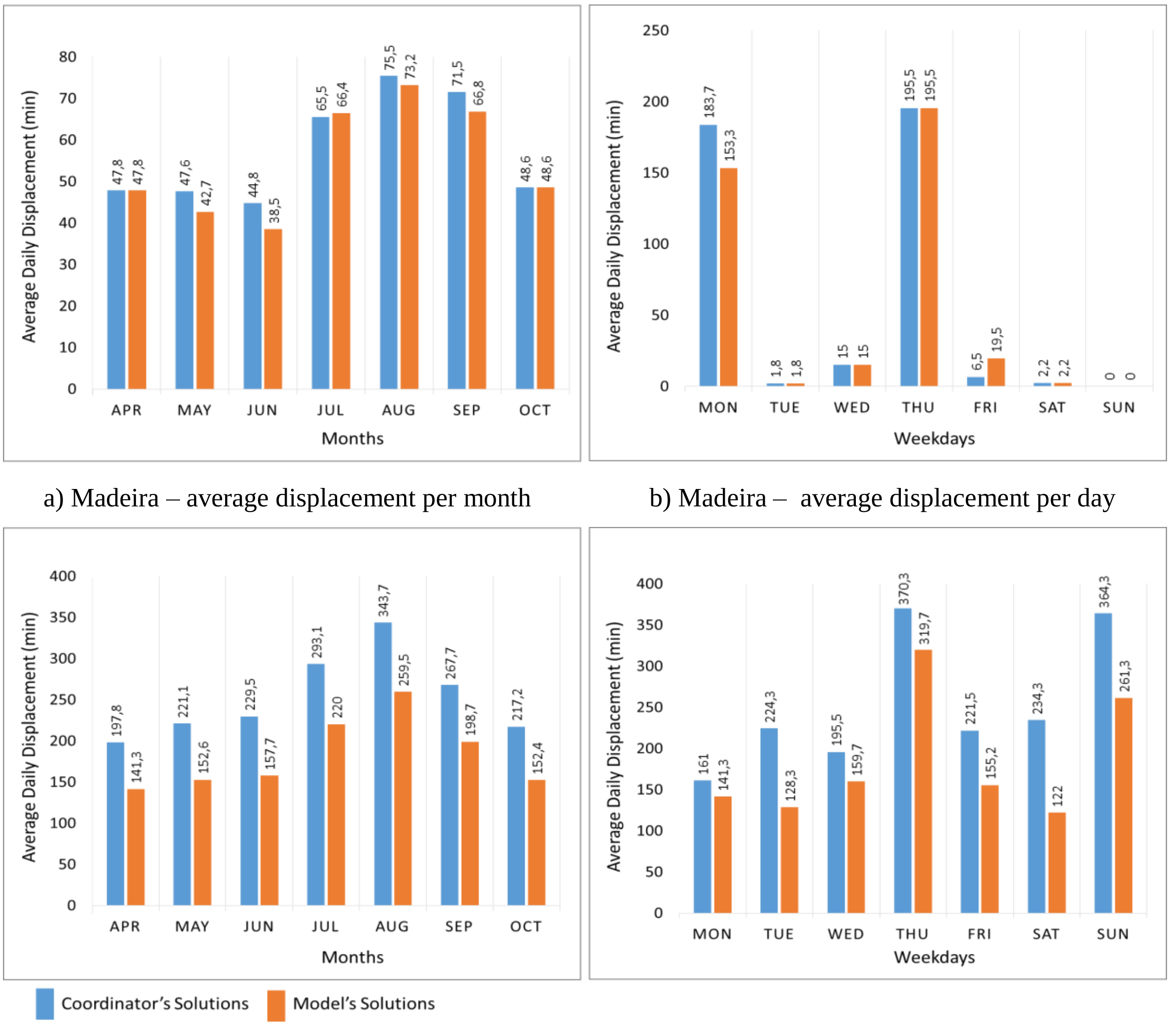
Solutions	Max Displacement (min)			Total Displacement (min)				Slots Displaced (slots)				Connections (min)	
	Chg. Hist	New Ent.	Other Slots	Total	Chg. Hist	New Ent.	Other Slots	Total	Chg. Hist	New Ent.	Other Slots	Max ΔCon	Sum ΔCon
CoordSol	45	15	80	53,140	9,600	4,515	39,025	2,549	605	403	1,541	45	1,745
Mod_Sol	25 -44%	25 +66%	55 -31%	38,625 -27%	3,560 -67%	5,220 +16%	29,845 -24%	2,379 -7%	348 -42%	396 -2%	1,635 +6%	0	0
Mod_Sol_NE	45 0%	15 0%	55 -31%	37,840 -29%	3,940 -59%	4,120 -9%	29,780 -24%	2,360 -7%	386 -36%	352 -14%	1,622 +5%	0	0
Mod_Sol_Con	25 -44%	25 +66%	55 -31%	36,105 -32%	3,560 -67%	5,220 +16%	27,325 -30%	2,379 -7%	348 -42%	396 -2%	1,648 +7%	5 -88%	2,710 +55%

First, the main solution of the model (Mod_Sol) improves the outcomes of slot allocation at both airports, as compared to the slot coordinator's. At Madeira, we observe a reduction of 4.3% in the total displacement, 12.5% in the maximum displacement, and 1.1% in the number of slots displaced. At Porto, the improvements are even more significant, with a reduction of 27% in the total displacement, 31% in the maximum displacement, and 7% in the number of slots displaced. This is not surprising because Porto is a much busier airport than Madeira and it is therefore much harder for the slot coordinators to find close-to-optimal solutions without the use of an advanced optimization model such as the one proposed in this chapter.

Moreover, the benefits of the PSAM solution (Mod_Sol) are even greater when the different priority classes are considered. Foremost, the displacement of the change-to-historic slots (the second-highest priority class) greatly decreased at both airports. This effect is noticeable in Madeira (with 3% and 3% reductions in the total displacement and in the number of slots displaced, respectively), but much stronger in Porto (with reductions as large as 44% for the maximum displacement, 67% for the total displacement, and 42% for the number of slots displaced). No request from new entrants is displaced in Madeira (as only 1.5% of all requests come from new entrants, and these are not concentrated at the busiest times), and the displacement for the lowest-priority class also declines, even though the number of such slots that are displaced increases slightly by 1.5%, or 4 slots. In Porto, the large improvements for change-to-historic slots constrains the allocation of slots to the lower priority classes by limiting the number of slots available at the busiest hours. In consequence, the results for the low-priority classes are mixed. New entrants are impacted most negatively, with an increase in maximum displacement of 10 minutes and in total displacement of 16%. In the "Mod_Sol_NE.", we show another solution in Porto that constrains the maximum displacement from new entrants. This results in slightly lower (albeit still very significant) improvements for the change-to-historic slots, but it also provides reductions in the total displacement of the new entrant slots and the remaining slots, as compared to the slot coordinator's solution. This illustrates how this model can be used to explore the tradeoffs between the displacements faced by the different priority classes, and determine the most desirable solution accordingly.

Finally, we note that the slot coordinator's solution involved some changes to airline requested connection times. Specifically, the connection times decreased by 5 minutes for 6 slot pairs and increased by 80 minutes for 8 slot pairs in Madeira. In Porto, 76 slot pairs faced changes in connection times, ranging from 5 to 45 minutes. This may be due to special considerations beyond our knowledge, or because allocating the slots with the requested connection times was infeasible given slot allocation decisions for the higher priority classes. In order to compare our solution to the schedule coordinator's, we allowed for slight increases or decreases in connection times of 5 minutes for each slot in the lowest priority class. This is shown in Table 3.11 "Mod_Sol_Con". Note that this flexibility in connecting times results in a significant decrease in total displacement (by 5.4% in Madeira and 6.5% in Porto, as compared to the main solution). This confirms the observations made in Section 3.5.1.b by showing the impact of small variations in connection times on the slot allocation process. In addition, note that the sum of changes in connection times is still lower than in the slot coordinator's solution for Madeira. This does not hold for Porto, but could be imposed as an additional constraint in the model (in which case the reduction in the total displacement from the slot coordinator's solution would be between 27% and 32%).

c) Distribution of Average Daily Displacement



a) Madeira – average displacement per month b) Madeira – average displacement per day

c) Porto – average displacement per month d) Porto – average displacement per day

Figure 3.5 – Average daily displacement at Madeira and Porto airports per month and day of week

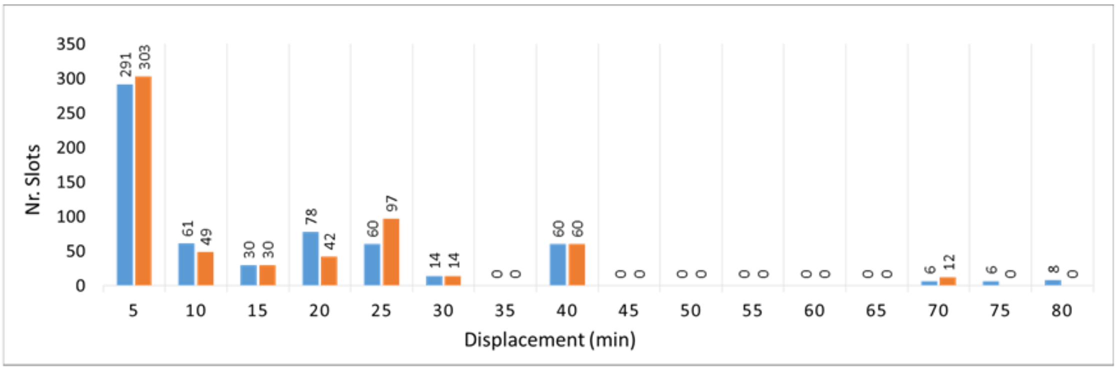
Following the results of the previous section, we calculate the average daily displacement obtained by the PSAM and the slot coordinator (i.e., $\sum_{i\in S}\sum_{d\in D} B_{id}\left(X_i^+ + X_i^-\right)/D$), as well as the average number of slots displaced per day (i.e., $\sum_{i\in S}\sum_{d\in D} B_{id}\left(W_i^+ + W_i^-\right)/D$). The average displacement per day in Madeira (resp. Porto) is equal to 55.2 minutes (resp. 183.9 minutes) per day, a reduction by 2.6 minutes (resp. 69.1 minutes) from the slot coordinator's solution. The average number of slots displaced per day is 2.89 slots (resp. 11.3 slots), a reduction of 0.03 slots (resp. 0.84 slots) per day. The *distribution* of these impacts varies across the days. For instance, out of the 210 days of the season, the displacement is improved (resp. worsened) on 20 days (resp. 13 days) in Madeira and 200 days (resp. 7 days) in Porto.

Figure 3.5 shows the average total displacement per day for each month of the season (Figure 3.5.a in Madeira, Figure 3.5.c in Porto) and for each day of the week (Figure 3.5.b in Madeira, Figure 3.5.d in Porto). Note, first, that the months with higher average displacements are those with most frequent imbalances between slot requests and declared capacities, i.e., July, August and September (see Table 3.3). Moreover, the PSAM solution improves the average displacement for almost every month over the season – with the exception of July in Madeira, when the average displacement increases by 0.9 minutes per day. Similarly, the average displacement is highest on the days of the week with the highest imbalances between slot requests and declared capacities in Madeira, i.e., Mondays and Thursdays. Note that almost no flights are displaced on the other days of the week, which is consistent with the fact that, on these days, demand for slots falls below the airport's declared capacities (Section 3.2). Nonetheless, the total displacement on these days is still positive, reflecting the interdependencies between slots over multiple days, and the fact that some slots had to be displaced on the least busy days to satisfy capacity constraints on the busiest days. In Porto, the largest displacement occurs on Thursdays, although Fridays exhibit, on average, more periods when slot demand exceeds declared capacities. This also stems from the interdependencies between slots over different days of the week. Finally, the model improves the average displacement on Mondays in Madeira (by 30 minutes), but worsens it on Fridays (by 13 minutes). In contrast, the average displacement is significantly improved on all days of the week in Porto (by up to 121 minutes for Saturdays).

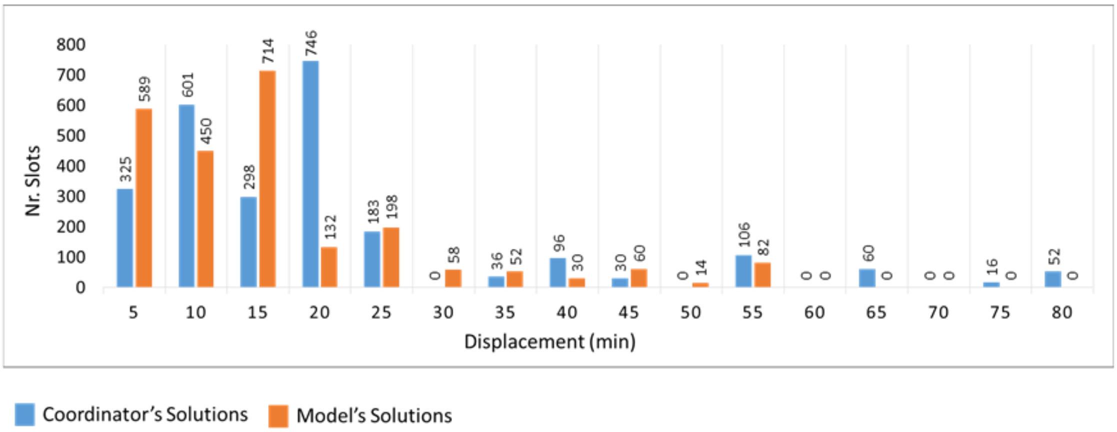
d) Distribution of displacement across slots

In addition to reducing the total displacement, the model also reduces the number of slots displaced (i.e., $\left(\sum_{i\in \boldsymbol{S}}\sum_{d\in \boldsymbol{D}} B_{id}\left(W_i^+ + W_i^-\right)\right)$) with respect to the slot coordinator's solution. In total 607 slots were displaced in Madeira and 2,379 in Porto, which corresponds to 4.7% and 5.9% of the total number of slots, respectively. The average displacement per displaced slot (i.e., $\sum_{i\in \boldsymbol{S}}\sum_{d\in \boldsymbol{D}} B_{id}\left(X_i^+ + X_i^-\right) / \sum_{i\in \boldsymbol{S}}\sum_{d\in \boldsymbol{D}} B_{id}\left(W_i^+ + W_i^-\right)$) is equal to 19.1 minutes in Madeira and 16.2 minutes in Porto – both significantly lower than in the slot coordinator's solution. However, the *distribution* of the displacement across all series of slots exhibits significant variability. Figure 3.6 shows the histogram of the number of slots displaced per displacement value in Madeira (Figure 3.6.a) and Porto (Figure 3.6.b). As seen in Table 3.10, the maximum displacement is reduced by 80 to 70 minutes in

Madeira, and by 80 to 55 minutes in Porto as compared to the slot coordinator solution. The figure shows that this reduction impacts positively a large number of flights. Indeed, the coordinator imposes a displacement that is larger than the model's maximum displacement for 14 slots (2.3% of the slots displaced) in Madeira, and 128 slots (5% of the slots displaced) in Porto. The number of slots with a displacement larger than 30 minutes is reduced from 80 to 72 in Madeira, and from 396 to 238 in Porto. Therefore, the model provides benefits not only by reducing total displacement, and/or the number of flights displaced, but also by reducing the tail of the displacement distribution, thus alleviating the costs associated with the largest displacements.



a) Madeira



b) Porto

Figure 3.6 – Histogram of number of slots displaced per minutes of displacement in Madeira and Porto airports

3.6 Conclusion

In this chapter, we have developed a novel modeling and computational approach to optimize slot allocation decisions at busy schedule-coordinated airports. We have proposed a new Priority-based Slot Allocation Model (PSAM) that minimizes the displacement from the airlines' slot requests, while fully complying with the "primary criteria" of the IATA guidelines and with airport declared capacities. We have introduced a strong formulation that provides exact solutions in reasonable computational times for mid-size airports – twice the size of those previously considered in the literature. The model has then been implemented using highly-detailed data from the airports of Madeira and Porto, Portugal. Comparisons with the slot coordinator decisions have suggested that the model captures well the main decisions and trade-offs made in practice and also improves the slot allocation outcomes by reducing the displacement experienced by the airlines by an estimated 4.5% and 27% at the two airports considered. Computational experiments also quantified the impact of the various constraints imposed by the IATA guidelines. The insights gained can be used to inform potential future adjustments to the slot allocation rules. The PSAM can thus provide significant benefits at major airports worldwide by enhancing the outcomes of slot allocation processes and making the eventual schedules of flights more consistent with the scheduling preferences of the airlines and, implicitly, with passenger demand.

The PSAM provides a methodological foundation to explore new questions in the field of airport demand and capacity management. First, extensions of the PSAM can capture additional complexities from the IATA guidelines, such as terminal, apron and noise restrictions. From a computational standpoint, the PSAM can be further strengthened, and combined with heuristic solution algorithms, to solve similar problems at even larger schedule-coordinated airports. From a practical standpoint, the model can address, in the longer term, strategic questions associated with the setting of airport declared capacities, with adjustments to the IATA guidelines, with the definition and prioritization of slot allocation objectives, and with other mechanisms for airport capacity allocation, by taking into account their individual or joint effects on airline schedules, airport operations, and passenger demand. The efficient elicitation of the full trade-off frontier between the four objectives considered in PSAM (and, possibly, others) is another important venue for future research. Ultimately, this research can support ongoing improvements in airport capacity management practices by making them more efficient, transparent, and collaborative.

4 A Large-scale Neighborhood Search Approach to Airport Slot Allocation

4.1 Introduction

Air traffic demand has grown to exceed available capacity at many airports worldwide, resulting in the routine occurrence of flight delays and high costs to airports, airlines and passengers. For instance, the nationwide impact of air traffic congestion in the United States was estimated at over $30 billion in 2007 (Ball et al., 2010). Absent opportunities to expand airport capacity, it is necessary to resort to *demand management* to prevent over-capacity scheduling. Demand management involves administrative rules or economic incentives to limit the number of flights scheduled at busy airports and at busy times by rescheduling flights over the day and, in some cases, by reducing the total number of flights. The vast majority of busy airports outside the United States are subject to *schedule coordination*, operated under the aegis of the International Air Transport Association (IATA). In 2017, schedule coordination was applied at 177 airports, serving a total of 3.15 billion passengers annually.

Any airline intending to operate a flight to and from a schedule-coordinated airport needs to receive access in the form of a slot. Airlines submit slot requests before each season to a slot coordinator, which performs slot allocation according to the Worldwide Slot Guidelines (WSG) set forth by IATA (2018). These guidelines specify rules and priorities that create coupling constraints across the allocation of slots at multiple times of day and on multiple days of year. As a result, slot allocation is a highly complex combinatorial problem, which carries enormous weight for airlines, airports, and passengers.

Optimization models have been proposed to support slot allocation decisions at schedule-coordinated airports. These models primarily aim to minimize the deviations of the schedule of flights from the airlines' requests. They have shown considerable promise to improve slot allocation outcomes. However, existing optimization approaches remain limited to small- and medium-size airports due to the combinatorial complexity of slot allocation. In contrast, the implementation of slot allocation optimization models at large-size airports remains intractable.

This chapter addresses this issue by developing an original optimization approach to solve the slot allocation problem at the largest schedule-coordinated airports. We formulate a model that captures all the rules and priorities from IATA's Worldwide Slot Guidelines, and develop a new algorithm based on large-scale neighborhood search to solve it efficiently at the busiest airports. The proposed algorithm starts by generating a feasible slot allocation solution, and then improves it iteratively by re-optimizing slot allocation decisions for a subset of slot requests. The algorithm is implemented at the Lisbon Portela Airport (LIS), one of the top-20 busiest airports in Europe. Results show that it provides optimal or near-optimal solutions in 6-10 hours of computation in settings where commercial solvers fail to identify the optimal solution after 7 days of computation. Thus, this chapter considerably enhances the capabilities of slot allocation models and algorithms.

Before presenting the contributions of this chapter in more detail in Section 4.1.4, we provide some background on airport slot allocation in Section 4.1.1 and review the related literature in Section 4.1.2. We also review large-scale neighborhood search algorithms in Section 4.1.3. The remainder of the chapter is organized as follows. Section 4.2 formulates our slot allocation model and evaluates its implementation with commercial solvers. Section 4.3 develops our large-scale neighborhood search algorithm. Section 4.4 evaluates the algorithm's performance at Lisbon airport. Section 4.5 performs a sensitivity analysis with respect to the algorithm's parameters. It demonstrates the algorithm's robustness and its benefits as compared to more straightforward applications of large-scale neighborhood search methods. Section 4.6 summarizes the findings of this chapter and outlines venues for future research.

4.1.1 Background on the Slot Allocation Process

The IATA slot allocation process is carried out bi-annually (for the summer and winter seasons), to provide airlines with landing or takeoff slots at schedule-coordinated airports. This process involves five main steps:

(i) About one year before the season, each schedule-coordinated airport provides its *declared capacity*, which specifies the number of slots to allocate at each time of a day. Capacity may be declared for different types of movements (arrivals, departures and total) and different time scales (15 minutes, 60 minutes etc.). Capacity is typically declared on a rolling-horizon basis. For instance, an airport with a declared capacity of 20 movements per hour on a 5-minute rolling basis cannot schedule more than 20 movements between 10:00 and 11:00, 10:05 and 11:05, 10:10 and 11:10, etc. Last, declared capacities may be defined for different elements of the airport; while existing studies predominantly consider runway capacities (the major bottleneck of operations), slot allocation is also subject to terminal and apron capacities.

(ii) About five months before the season, the airlines provide slot requests to slot coordinators. Each request includes: (i) a priority class (see below); (ii) the type of movement (arrival or departure); (iii) the preferred time to schedule the flight; (iv) the days of the flight's operations; (v) the turnaround time between two consecutive flights operated by the same aircraft, if applicable; and (vi) the aircraft type and corresponding number of seats.

(iii) About four months before the season, the airport's slot coordinator performs *initial slot allocation*. Slot allocation is subject to IATA's Worldwide Slot Guidelines and needs to comply with the declared capacity values specified by the airport.

(iv) Following initial slot allocation, adjustments are made during a "slot conference", attended by airline representatives, slot coordinators, airport representatives and other parties. These adjustments resolve conflicts stemming from slot allocation decisions made at multiple airports and disputes among airlines competing for the same slots.

(v) Until the day of operations, airlines can perform last-minute adjustments, as long as slots are still available and adjustments are approved by slot coordinators.

This chapter focuses on the third step, i.e., initial slot allocation. At almost all schedule-coordinated airports, all slot requests can be accommodated (of course, not necessarily at their preferred times). The slot allocation problem therefore involves finding a schedule of flights that minimizes deviations from airlines' requests, without exceeding the values of the airport's declared capacity.

Slot allocation must follow rules and priorities specified by IATA's Worldwide Slot Guidelines. First, slot allocation is subject to *connectivity* constraints to maintain minimum turnaround times between consecutive flights operated by the same aircraft. Second, *schedule regularity* constraints state that flights belonging to the same slot request must be allocated to the same time of the day on each day of the season. For instance, if a slot is requested on Mondays and Wednesdays for 15 weeks, the flights need to be scheduled at the same time of the day on the corresponding 30 days. This requirement creates coupling constraints across all days in the season, thereby considerably increasing the complexity of the slot allocation problem. Third, the guidelines also involve *grandfather rights*, which give priority to some slot requests over others based on historical precedence. Specifically, slot requests are allocated according to four priority classes: (i) "historic slots" (i.e., slots owned by the same airline in the previous equivalent season that were used at least 80% of the time); (ii) "change-to-historic slots" (i.e., historic slots for which the airline requests a change such as re-timing or the use of another aircraft); (iii) "new-entrant slots" (i.e., slots requested by airlines owning less than five slots a day); and (iv) "other slots" (i.e., slot requests that do not belong to any of the three other priority classes). The model presented in this chapter complies with all the rules and priorities from IATA's Worldwide Slot Guidelines.

In this chapter, we focus on administrative slot allocation at a single airport over a full season. This scope attempts to enhance slot allocation outcomes given the current rules and procedures currently in place. These rules and procedures are likely to remain prevalent in the future; for instance, IATA's position is summarized on its website as follows: "The WSG Strategic Review is the ongoing process of enhancing the existing WSG, not rewriting from scratch, to ensure it remains the global, single slot standard for years to come". This strengthens the need to develop optimization models and algorithms that reflect current slot allocation practice. However, the purpose of this chapter is not to assess existing rules and procedures. Other studies have argued that current slot allocation may induce inefficiencies by failing to provide access to the users who value it the most and by incentivizing inefficient utilization of slots due to the use-it-or-lose-it rule (see, e.g., Starkie, 1998; NERA, 2004; Czerny et al. 2008; Fukui, 2010; Guiomard, 2018; Valdes and Gillen, 2018). As a result, alternative market-based approaches have been promoted in the literature (see, e.g., Ball et al. (2018) on auctions and Pellegrini et al. (2012a) on secondary trading) and in governmental regulations (see, e.g., the US Federal Aviation Administration's (2008) Congestion Management Rule and the European Commission's (2011) attempt to enforce market mechanisms). However, as mentioned above, their implementation has remained limited. This suggests that, from a policy standpoint, there might exist opportunities to revisit existing slot allocation rules and procedures. But, at the same time, there also exist opportunities to enhance slot allocation outcomes under the current Worldwide Slot Guidelines. This is the focus of this Chapter.

4.1.2 Literature Review on Airport Slot Allocation

Demand management mechanisms fall into three categories: *laissez faire*, market-based and non-monetary mechanisms (see Czerny et al. 2008 and Gillen et al. 2016 for reviews). *Laissez faire* is predominantly in place at US airports, where flight scheduling is not subject to any demand management constraint. This mechanism relies on the principle that delay costs will be internalized by the airlines. As compared to similar airports located elsewhere, it leads to higher scheduling levels but also higher levels of congestion. To mitigate these delay externalities, market-based mechanisms aim to incentivize the airlines to schedule fewer flights at peak hours. The two prominent types of mechanisms are congestion pricing (Carlin and Park 1970; Daniel 1995; Brueckner 2002) and slot auctions (Rassenti et al. 1982, Ball et al. 2006). These approaches can, at least in theory, achieve economically efficient outcomes by allocating scarce airport capacity to the flights that generate the highest value. However, they have not been implemented in practice to date, due to their monetary transfers and potential barriers to entry and competition. Busy airports outside the United States are thus overwhelmingly subject to non-monetary demand management mechanisms, relying on administrative slot allocation.

Optimization models have emerged recently to support slot allocation at schedule-coordinated airports (Zografos et al. (2017) provide an extensive review). Comparisons with schedule coordinators' decisions have demonstrated that optimization techniques can improve slot allocation significantly – especially at larger airports where slot allocation is more complex. Models can be classified along two dimensions: (i) whether they optimize slot allocation at a single airport or at multiple airports in a network; and (ii) whether they optimize slot allocation for a single day or for multiple days of season.

Among single-airport single-day models, Pyrgiotis and Odoni (2016) develop an integer programming formulation to simulate the effects of schedule limits on airlines' schedules of flights. Experimental results suggest that limited demand management at busy US airports can result in significant delay reductions. Jacquillat and Odoni (2015a) extend this approach by jointly optimizing schedule coordination and operating procedures at a busy airport. Their results suggest that replacing fixed schedule limits (or declared capacities) by delay targets can result in more efficient schedule coordination outcomes. These models, however, do not consider the coupling between slot allocation decisions across multiple days in a season, and thus do not capture the schedule regularity requirements from IATA's Worldwide Slot Guidelines. In contrast, multiple-day models allocate slots for an entire season, which obviously adds considerable complexity. The first model in this category was developed by Zografos et al. (2012). This model was applied at the Heraklion Airport (HER) in Greece, which operates around 50,000 flights per annum, and was solved in 5 minutes with an optimality gap of 1.58%. Ribeiro et. al. (2018) proposed a new Priority-based Slot Allocation Model (PSAM), which captures most of the IATA Worldwide Slot Guidelines. The authors propose valid inequalities that enable to solve the PSAM to optimality in 5 minutes at the Porto airport (OPO), which operates around 85,000 flights per annum. However, the implementation of such optimization models using exact methods at the largest schedule-coordinated airports remains intractable. Note, also, that while in these papers the focus was placed on minimizing the displacement from airlines' slot requests, recent papers added alternative objectives to the optimization, such as inter-airline fairness (Zografos and Jiang, 2016; Jacquillat and Vaze, 2018) and schedule acceptability by the airlines (Zografos et al., 2017).

Finally, a recent body of work has tackled the multi-airport slot allocation problem, i.e., the simultaneous allocation of slots at several schedule-coordinated airports within a network. This aims to enhance the existing slot allocation process, where slot allocation is performed at each airport independently and conflicts are then resolved at the slot conferences. Castelli et al. (2011b) and Corolli et al. (2014), addressed this problem with a subset of the requirements from the IATA Worldwide Slot Guidelines and small networks. Pellegrini et al. (2012b) considered a larger network, and solved it using local search and variable neighborhood search heuristics. Pellegrini et al. (2017) then developed a new model called Simultaneous Optimisation of airport SloT Allocation (SOSTA), which yields exact solutions for a network of up to 60 airports, for a single day. To date, Benlic (2018) proposed the only approach, based on local search heuristics, to solve the multi-airport multiple-day slot allocation problem.

In this chapter, we focus our analysis on the single-airport slot allocation problem for the full season of operations. The consideration of the full season enables to capture the WSG rules applied in practice. The consideration of a single airport is consistent with current practice and with the IATA guidelines. Besides, most airlines are comfortable with this approach, making unlikely significant changes in the near future. In fact, IATA’s position in this respect is summarized on its website as follows: “The WSG Strategic Review is the ongoing process of enhancing the existing WSG, not rewriting from scratch, to ensure it remains the global, single slot standard for years to come – a major undertaking for 2017/18”.

4.1.3 Literature Review on Large-scale Neighborhood Search (LNS)

Methodologically, this chapter builds upon large-scale neighborhood search (LNS) algorithms to tackle the airport slot allocation problem. LNS falls into local search heuristic algorithms. In general terms, local search algorithms proceed by generating an initial feasible solution, exploring its neighborhood in search of a better solution, and repeating the process until no more improvements are possible. Well-known local search algorithms are simulated annealing (Bertsimas and Tsitsiklis, 1993), tabu search (Glover, 1989) and variable neighborhood search (Mladenović and Hansen, 1997). These algorithms all share the similar feature of exploring relatively small neighborhoods at each iteration. Thus, each iteration can be completed in very short computational times. At the same time, considering small neighborhoods bears two major risks. First, the algorithms may converge to local optima – although this can be mitigated by introducing random perturbations. Second, they might be ineffective for tightly constrained problems, where any deviation from a feasible solution involves complex interaction effects across many decision variables. This is the case in our problem, where the complex combinatorics underlying the airport capacity, aircraft connections and schedule regularity constraints make it hard to obtain solution improvements by performing "local" adjustments. It thus motivates the consideration of larger neighborhoods within the local search algorithm.

Local search algorithms with large neighborhoods fall into the field of Very-large Scale Neighborhood Search (VLSN) heuristic algorithms. Ahuja et. al. (2000) defines large-scale neighborhoods as being too large to be enumerated explicitly. A critical component of VLSN methods is the procedure applied to explore these large-scale neighborhoods. Examples include cyclic exchanges and path exchanges, which involve swapping parts of an existing solution with each other (Ahuja et. al., 2002; Pisinger and Ropke, 2010). In this chapter, we use a strategy named Large Neighborhood Search (LNS), first proposed by Shaw (1997). This strategy is based on relaxation and re-optimization, also called "destroy and repair". At each iteration, the algorithm removes parts of an existing feasible solution (the "destroy" phase) and re-optimizes the solution while fixing its other components (the "repair" phase). VLSN algorithms have been implemented in several application areas of combinatorial optimization, such as vehicle routing, scheduling and timetabling (Meyers and Orlin 2006; Kytöjoki et. al. 2007, Pisinger and Ropke 2010) and, in aviation, airline fleet assignment (Ahuja et. al., 2007) and gate assignment (Yu et al. 2017). In this chapter, we propose an original LNS algorithm tailored to the slot allocation problem by exploiting the structure of airlines' slot requests and slot allocation constraints.

4.1.4 Contributions

This chapter develops a novel optimization approach to solve the slot allocation problem at major schedule-coordinated airports. Ultimately, it provides decision-making support to enhance the existing schedule coordination process by making the schedules of flights more consistent with airlines' slot requests and passenger demand. Specifically, it makes the following contributions:

- *Formulating a model of slot allocation that captures all IATA Worldwide Slot Guidelines*. We formulate the Priority-based Slot Allocation Model with Runway, Terminal and Apron constraints (PSAM-RTA). This model builds upon the PSAM (Ribeiro et al. 2018), but incorporates terminal and apron capacity constraints. As we shall see, apron constraints introduce coupling constraints across multiple time periods of the day, and thus significantly increase the complexity of the model.

- *Developing an original algorithm based on large-scale neighborhood search to solve the slot allocation problem at the largest schedule-coordinated airports*. First, a *constructive heuristic* generates a good initial feasible solution in short computational times by treating slot requests by decreasing order of frequency (i.e., the number of days in the season on which the corresponding flight is requested). Second, an *improvement heuristic* refines the solution by iteratively decomposing the problem into smaller components and re-optimizing slot allocation decisions within a given neighborhood.
- *Demonstrating that the algorithm provides optimal or near-optimal solutions in reasonable computational times at large-scale schedule-coordinated airports.* The model is applied at the Lisbon Airport (LIS), which operates over 200,000 flight movements per annum, and is thus significantly larger than any airport to which slot allocation models have been applied thus far. In our test instances, commercial optimization solvers cannot find the optimal solution, leaving an optimality gap of 2-5% after 7 days of computation. In contrast, our algorithm consistently generates solutions within 0.1% of the optimum in a few hours, thus improving the solutions from commercial solvers both in terms of quality and computational times. Note that we know, in this case, that our heuristic approach reaches the provably optimal solution because the optimal solution coincides with that of a relaxed problem that can be solved to optimality using exact methods.
- *Providing a decision-making support tool to enhance slot allocation in practice*. This chapter's modeling and computational framework can provide high-quality slot allocation decisions at the largest schedule-coordinated airports. This can support the development of flight schedules to match airlines' requests more effectively than current procedures relying on specialized software and *ad hoc* allocation decisions.

4.2 Slot Allocation Model

This section formulates the Priority-based Slot Allocation Model with Runway, Terminal and Apron constraints (PSAM-RTA). The formulation, described in Section 4.2.1, builds upon the PSAM model from Chapter 2 (Ribeiro et. al, 2018) but provides a new set of constraints to capture capacity restrictions imposed by the airport's terminal facilities and apron areas (Section 4.2.2). We then present in Section 4.2.3 the experimental setup considered in this chapter and show that direct implementation of PSAM-RTA with commercial solvers remains intractable at the largest schedule-coordinated airports.

4.2.1 Priority-based Slot Allocation Model (PSAM): Baseline Formulation

The PSAM takes as inputs the declared capacity of the airport and the set of slot requests from all airlines. The decision variables specify the slot times assigned to all slot requests. The model minimizes the *displacement* from the airlines' requests. Specifically, it first minimizes the maximum displacement (i.e., the largest deviation, across all slot requests, from their requested times), and then minimizes the total displacement (i.e., the total deviation, across all slot requests, from their requested times). This bi-objective formulation is motivated by equity concerns to prevent any slot request from being displaced disproportionately. It is consistent with the existing literature on airport slot allocation. The constraints ensure compliance with the airport's declared runway capacities and with the rules and priorities from IATA's Worldwide Slot Guidelines. As mentioned in the introduction, these guidelines impose schedule connectivity and regularity constraints, as well as the following set of priorities among slot requests:

(i) *Historic slots:* All slots holding historic rights must be scheduled at their requested times.

(ii) *Change-to-historic slots:* All historic slots for which the airline requests a change are allocated with priority. Change-to-historic slots are divided into two categories: (i) "CR" slots can be scheduled at any time between the historic and the requested times, and (ii) "CL" slots can be scheduled only at the historic time or at the newly requested time. The historic time of a slot request corresponds to the time at which the same request was allocated in the previous year.

(iii) *New-entrant slots:* Once historic and change-to-historic slots have been allocated, 50% of the remaining capacity must be allocated to new entrants (if demand is sufficient).

(iv) *Other slots:* Once slots from the first three priority classes have been allocated, all other slots are then assigned to their respective slot times.

We formulate below the baseline PSAM, along with valid equalities that strengthen the model's linear programming relaxation.

a) Sets

$\boldsymbol{T} = \{1,...,T\}$: set of time periods, indexed by t

$\boldsymbol{D} = \{1,...,D\}$: set of days, indexed by d

$\boldsymbol{S} = \{1,...,S\}$: set of slot requests codes, indexed by *i* or *j*

$\boldsymbol{S}_{arr}\left(\text{resp. } \boldsymbol{S}_{dep}\right) \subset \boldsymbol{S}$: set of arrivals (resp. departures)

$\boldsymbol{S}_{H} \subset \boldsymbol{S}; \boldsymbol{S}_{CH} \subset \boldsymbol{S}; \boldsymbol{S}_{NE} \subset \boldsymbol{S}; \boldsymbol{S}_{OS} \subset \boldsymbol{S}$: subset of historic, change-to-historic, new-entrants and other slots

$\boldsymbol{S}_{CR}\left(\text{resp. } \boldsymbol{S}_{CL}\right) \subset \boldsymbol{S}_{CH}$: subset of "CR" (resp. "CL") requests among change-to-historic slots

$\boldsymbol{P} \subset \boldsymbol{S} \times \boldsymbol{S}$: set of slot request pairs $(i, j) \in S \times S$ such that there is a connection between *i* and *j*

$\boldsymbol{C} = \{1,...,C\}$: set of capacity time scales, indexed by *c*

b) Parameters

$$A_{it} = \begin{cases} 1, & \text{if slot } i \text{ is requested to operate no earlier than period } t \\ 0, & \text{otherwise} \end{cases}$$

$$B_{id} = \begin{cases} 1, & \text{if slot } i \text{ is requested to operate on day } d \\ 0, & \text{otherwise} \end{cases}$$

$$H_{it} = \begin{cases} 1, \text{if slot } i \text{ was operated in the previous year no earlier than period } t \in \boldsymbol{T} \\ 0, \text{otherwise} \end{cases}$$

$CR_{tdc}^{dep}\left(\text{resp. } CR_{tdc}^{arr}, CR_{tdc}^{tot}\right)$ departure (resp. arrival, total) declared runway capacity at the airport in period $t \in \boldsymbol{T}$, day $d \in \boldsymbol{D}$ and time scale $c \in \boldsymbol{C}$

L_c = length of time scale c

$T^{\max}\left(\text{resp. } T^{\min}\right)$ = maximum allowable increase (resp. decrease) in the connection time of two slots in comparison to the requested connection time

c) Decision Variables

$$Y_{it} = \begin{cases} 1, & \text{if slot } i \text{ is rescheduled to arrive / depart no earlier than period } t \\ 0, & \text{otherwise} \end{cases}$$

$X_i^{+}\left(\text{resp. } X_i^{-}\right)$ = displacement of slot *i* if rescheduled to a later (resp. earlier) time period

$$W_i^+ \left(\text{resp.}\, W_i^-\right) = \begin{cases} 1, & \text{if slot } i \text{ is displaced to a later (resp. earlier) time} \\ 0, & \text{otherwise} \end{cases}$$

d) Model Formulation

$$\textit{minimize} \quad w_0 \max_{i \in \boldsymbol{S}} \left(X_i^+, X_i^-\right) + \sum_{i \in \boldsymbol{S}} \sum_{d \in \boldsymbol{D}} \left(X_i^+ + X_i^-\right) \times B_{id} \qquad (4.1)$$

$$Y_{i1} = 1 \qquad \forall i \in \boldsymbol{S} \qquad (4.2)$$

$$Y_{it} \geq Y_{i,t+1} \qquad \forall i \in \boldsymbol{S}, t \in \boldsymbol{T} \qquad (4.3)$$

$$\sum_{t \in \boldsymbol{T}} \left(1 - A_{it}\right) Y_{it} = X_i^+ \qquad \forall i \in \boldsymbol{S} \qquad (4.4)$$

$$\sum_{t \in \boldsymbol{T}} A_{it} \left(1 - Y_{it}\right) = X_i^- \qquad \forall i \in \boldsymbol{S} \qquad (4.5)$$

$$X_i^+ = X_i^- = 0 \qquad \forall i \in \boldsymbol{S}_H \qquad (4.6)$$

$$X_i^+ \leq \left(H_{it} - A_{it}\right) \qquad \forall i \in \boldsymbol{S}_{CR}, t \in \boldsymbol{T} \qquad (4.7)$$

$$X_i^- \leq \left(A_{it} - H_{it}\right) \qquad \forall i \in \boldsymbol{S}_{CR}, t \in \boldsymbol{T} \qquad (4.8)$$

$$W_i^+ \geq Y_{it} - A_{it} \qquad \forall i \in \boldsymbol{S}_{CL}, t \in \boldsymbol{T} \qquad (4.9)$$

$$W_i^- \geq -Y_{it} + A_{it} \qquad \forall i \in \boldsymbol{S}_{CL}, t \in \boldsymbol{T} \qquad (4.10)$$

$$X_i^+ = \left(H_{it} - A_{it}\right) W_i \qquad \forall i \in \boldsymbol{S}_{CL}, t \in \boldsymbol{T} \qquad (4.11)$$

$$X_i^- = \left(A_{it} - H_{it}\right) W_i \qquad \forall i \in \boldsymbol{S}_{CL}, t \in \boldsymbol{T} \qquad (4.12)$$

$$\sum_{i \in \boldsymbol{S}_{arr}} \sum_{t=s}^{s+L_c} \left(Y_{it} - Y_{i,t+1}\right) B_{id} \leq CR_{sdc}^{arr} \qquad \forall s \in \boldsymbol{T} \mid s < T - L_c + 1, d \in \boldsymbol{D}, c \in \boldsymbol{C} \qquad (4.13)$$

$$\sum_{i \in \boldsymbol{S}_{dep}} \sum_{t=s}^{s+L_c} \left(Y_{it} - Y_{i,t+1}\right) B_{id} \leq CR_{sdc}^{dep} \qquad \forall s \in \boldsymbol{T} \mid s < T - L_c + 1, d \in \boldsymbol{D}, c \in \boldsymbol{C} \qquad (4.14)$$

$$\sum_{i \in \boldsymbol{S}} \sum_{t=s}^{s+L_c} \left(Y_{it} - Y_{i,t+1}\right) B_{id} \leq CR_{sdc}^{tot} \qquad \forall s \in \boldsymbol{T} \mid s < T - L_c + 1, d \in \boldsymbol{D}, c \in \boldsymbol{C} \qquad (4.15)$$

$$\Delta A_{ij} = \sum_{t \in \boldsymbol{T}} \left(A_{jt} - A_{it}\right) \qquad \forall (i,j) \in \boldsymbol{P} \qquad (4.16)$$

$$\sum_{t \in \boldsymbol{T}} \left(Y_{jt} - Y_{it}\right) - \Delta A_{ij} \geq T^{min} \qquad \forall (i,j) \in \boldsymbol{P} \qquad (4.17)$$

$$\sum_{t \in \boldsymbol{T}} \left(Y_{jt} - Y_{it}\right) - \Delta A_{ij} \leq T^{max} \qquad \forall (i,j) \in \boldsymbol{P} \qquad (4.18)$$

$$\Delta H_{ij} = \sum_{t \in \boldsymbol{T}} \left(H_{jt} - H_{it} \right) \qquad \forall (i,j) \in \boldsymbol{P} \cap \left(\boldsymbol{S}_{CH} \times \boldsymbol{S}_{CH} \right) \qquad (4.19)$$

$$\sum_{t \in \boldsymbol{T}} \left(Y_{jt} - Y_{it} \right) \geq \min \left(\Delta A_{ij}, \Delta H_{ij} \right) \qquad \forall (i,j) \in \boldsymbol{P} \cap \left(\boldsymbol{S}_{CH} \times \boldsymbol{S}_{CH} \right) \qquad (4.20)$$

$$\sum_{t \in \boldsymbol{T}} \left(Y_{jt} - Y_{it} \right) \leq \max \left(\Delta A_{ij}, \Delta H_{ij} \right) \qquad \forall (i,j) \in \boldsymbol{P} \cap \left(\boldsymbol{S}_{CH} \times \boldsymbol{S}_{CH} \right) \qquad (4.21)$$

$$X_i^+, X_i^- \in \mathbb{N}_0 \qquad (4.22)$$

$$Y_{it}, W_{it}^+, W_{it}^- \in \{0,1\} \qquad (4.23)$$

Equation (4.1) formulates the bi-objective problem of minimizing the maximum displacement and the total displacement. The parameter w_0 is set sufficiently large to prioritize the minimization of the maximum displacement vs. the total displacement. Constraint (4.2) ensures that all slots are assigned to a time period. Constraints (4.3) force the variables Y to be non-increasing in t, consistently with their definition. Constraints (4.4) and (4.5) define the logical relationships between the variables (i.e., each slot request can be scheduled at the requested time, displaced to a later slot, or displaced to an earlier slot). These constraints provide valid equalities that provide tight linear programming relaxations (Ribeiro et. al., 2018). Constraints (4.6) ensure that historic slots are never displaced. Constraints (4.7) and (4.8) specify that change-to-historic slots with a "CR" code are assigned to a time slot between the historical and the requested time slots. Constraints (4.9) to (4.12) specify that change-to-historic slots with a "CL" code are assigned to either the historic time slot or to the requested time slot. Constraints (4.13) to (4.15) ensure that the number of arrivals, departures and total number of movements, respectively, scheduled in any time period does not exceed the corresponding runway capacities. Constraints (4.16) to (4.18) ensure that the time between two connected flights does not increase or decrease by more than the allowable limits. Constraints (4.19) to (4.21) ensure that the connecting time between two change-to-historic slots lies between the requested connecting time and the historic connecting time, as specified by the IATA guidelines. Constraints (4.22) and (4.23) define the domain of the decision variables.

Mathematically, two approaches can capture the priorities among historic, change-to-historic, new-entrant and other slot requests. The first one is a *sequential approach*, and consists of solving PSAM-RTA three times, once for each priority class. Specifically, it involves, first, allocating historic slots to their historic times, and, the, solving PSAM-RTA for the change-to-historic slots, followed by the new-entrant slots, and the other slots (this is the approach adopted in Ribeiro et al. 2018). The second is a *weight-based approach*, and consists of solving PSAM-RTA only once by weighting the displacement of each class according to its priority. Specifically, the objective function (4.1) is replaced by Equation (4.24), with w_1 and w_2 such that $w_1 >> w_2 >> 1$. Equations (4.25) to (4.27) define the objective value for change-to-historic, new-entrant and other slots, respectively. Historic slots are not considered here because their displacement is always zero per Equation (4.6).

$$min \quad w_1\delta_{CH} + w_2\delta_{NE} + \delta_{OS} \tag{4.24}$$

$$\delta_{CH} = w_0 \max_{i \in \boldsymbol{S}_{CH}} \left(X_i^+, X_i^- \right) + \sum_{i \in \boldsymbol{S}_{CH}} \sum_{d \in \boldsymbol{D}} \left(X_i^+ + X_i^- \right) \times B_{id} \tag{4.25}$$

$$\delta_{NE} = w_0 \max_{i \in \boldsymbol{S}_{NE}} \left(X_i^+, X_i^- \right) + \sum_{i \in \boldsymbol{S}_{NE}} \sum_{d \in \boldsymbol{D}} \left(X_i^+ + X_i^- \right) \times B_{id} \tag{4.26}$$

$$\delta_{OS} = w_0 \max_{i \in \boldsymbol{S}_{OS}} \left(X_i^+, X_i^- \right) + \sum_{i \in \boldsymbol{S}_{OS}} \sum_{d \in \boldsymbol{D}} \left(X_i^+ + X_i^- \right) \times B_{id} \tag{4.27}$$

In this chapter, we report results from the implementation of the model with the two approaches. We show that the most effective approach depends on the test instance, and that both approaches face similar limitations at large schedule-coordinated airports (see Section 4.2.3).

4.2.2 Terminal and Apron Capacity Constraints

The PSAM model presented above only accounts for runway capacity constraints. We have extended the model to also capture terminal and apron capacity constraints. The extended model is referred to as Priority-based Slot Allocation model with Runway, Terminal and Apron constraints (PSAM-RTA).

a) Terminal Constraints

Terminal capacities restrict the number of arriving and departing passengers at the airport. Many airports organize their terminals per type of airlines and/or flights. For instance, Lisbon airport has two terminals. One is mainly used by legacy carriers, the other is only used by low-cost carriers. Both terminals have parts reserved for Schengen and non-Schengen flights. We thus define a set of terminal capacities corresponding to different parts of the airport’s terminals facilities, and the following parameters:

$\boldsymbol{K} = \{1,...,K\}$ set of terminal capacities, indexed by k

CT_{tdck} = capacity of terminal $k \in \boldsymbol{K}$ in period $t \in \boldsymbol{T}$, day $d \in \boldsymbol{D}$ at time scale $c \in \boldsymbol{C}$

S_i = number of seats in the aircraft used for the flight associated with slot $i \in \boldsymbol{S}$

LF_i = predicted load factor of the flights associated with slot $i \in \boldsymbol{S}$

$$I_{ik} = \begin{cases} 1, \text{ if the flights associated with slot } i \text{ are operated from terminal } k \in \boldsymbol{K} \\ 0, \text{otherwise} \end{cases}$$

Note that parameters CT, S and I are directly available from the airlines' slot requests. In contrast, parameter LF is unknown at the time of slot allocation. It can either be approximated by a uniform value across all slot requests (we adopt this approach in this chapter) or estimated at a more granular level from historical records of operations.

We then add Constraint (4.28) to take terminal capacity into account. This constraint is similar to the runway capacity constraints (Equations (4.13) to (4.15)), but weights the displacement of each slot request i, equal to $\sum_{t=s}^{s+L_c}(Y_{it} - Y_{i,t+1})$, by the number of seats S_i and the expected load factor LF_i. This formulation assumes that passengers impact terminal operations within one hour of the corresponding flight's arrival or departure. This is consistent with current practice from coordinators.

$$\sum_{i \in \boldsymbol{S}} \sum_{t=s}^{s+L_c} \left(Y_{it} - Y_{i,t+1}\right) B_{id} I_{ik} S_i LF_i \leq CT_{sdck} \quad \forall s \in \boldsymbol{T} \mid s < T - L_c + 1, d \in \boldsymbol{D}, c \in \boldsymbol{C}, k \in \boldsymbol{K} \qquad (4.28)$$

b) Apron Constraints

Aprons designate the locations where aircraft are parked between operations. To formulate apron capacity constraints, we need a new decision variable that characterizes the number of aircraft on the apron at each time period. Note that, to compute this variable, we need to track aircraft operations over the full day, which introduces coupling constraints across time periods. As we shall see, this will considerably increase the complexity of the resulting slot allocation problem.

Specifically, we define the following capacity parameter (CA) and decision variables (Q and N):

CA_{td} = apron capacity in period $t \in \boldsymbol{T}$ and day $d \in \boldsymbol{D}$

Q_{td} = number of aircraft on the apron at time $t \in \boldsymbol{T}$ and day $d \in \boldsymbol{D}$

N_d = number of aircraft on the apron at the beginning of day $d \in \boldsymbol{D}$

The variables N_d are used to ensure the problem's feasibility when more departures than arrivals are requested at the airport on day d. Equation (4.29) initializes the number of aircraft on the apron at the beginning of the day (to simplify the exposition, we slightly abuse notations by using Q_{td} for $t = 0$). Equation (4.30) formulates balance constraints, specifying that the number of aircraft on the apron grows in period t by the number of scheduled arrivals minus the number of scheduled departures. Constraint (4.31) applies the apron capacity, and Constraint (4.32) ensures that the variables Q and N are non-negative. Note that the variables Q and N are integer, but we can simply define them as non-negative variables—in which case they are bound to take integer values at the optimum per Equations (4.29) and (4.30).

$$Q_{0d} = N_d \qquad \forall d \in \boldsymbol{D} \qquad (4.29)$$

$$\sum_{i \in \boldsymbol{S}_{arr}} \left(Y_{it} - Y_{i,t+1}\right) B_{id} - \sum_{i \in \boldsymbol{S}_{dep}} \left(Y_{it} - Y_{i,t+1}\right) B_{id} + Q_{t-1,\mathrm{d}} = Q_{td} \qquad \forall t \in \boldsymbol{T}, d \in \boldsymbol{D} \qquad (4.30)$$

$$Q_{td} \leq CA_{td} \qquad \forall t \in \boldsymbol{T}, d \in \boldsymbol{D} \qquad (4.31)$$

$$Q_{td} \geq 0 \; ; \; N_d \geq 0 \qquad \forall t \in \boldsymbol{T}, d \in \boldsymbol{D} \qquad (4.32)$$

4.2.3 Experimental Setup and Direct Implementation Results

The case studies reported in this chapter use data from the airports of Madeira, Porto and Lisbon for the Summer seasons of 2014 and 2015. Madeira and Porto both operate with runway capacity limits for each 15-minute period and each 60-minute period, applied on a 5-minute rolling horizon basis. In Madeira, the declared capacities are 6 movements, 4 arrivals and 4 departures per 15-minute period, and 14 movements, 7 arrivals and 7 departures per hour. In Porto, the declared capacities are 7 movements per 15-minute period and 20 movements per hour (with no arrival and departure limits). They are also subject to terminal and apron capacity constraints, but these are typically not binding. Table 4.1 shows the declared capacities in Lisbon in 2014 and 2015. The 15-minute runway capacities and the apron capacities are applied on a 5-minute rolling horizon basis. All the other capacities are applied on a 60-minute rolling horizon basis. Note, also, that the runway capacity values vary over the day in 2015.

The three airports under consideration face different levels of demand. The Madeira airport needed to accommodate around 13,000 slots in the Summer of 2014. In comparison, three times as many slots were requested at the Porto airport, and eight times as many slots were requested at the Lisbon airport. As we shall see, the resulting slot allocation problem will thus be much more complex in

Lisbon than in Madeira and Porto. Note, also, that the number of slots requests in Lisbon grew by 4% between 2014 and 2015. Moreover, the share of non-historic slots (i.e., those that are effectively subject to slot allocation) increased from 57% to 69%. As a result, slot allocation will also be significantly more complex in the Summer of 2015 than in the Summer of 2014 at the Lisbon airport.

Table 4.1 – Declared capacities at the Lisbon airport

Declared capacity categories		Lisbon 2014	Lisbon 2015								
			00:00 06:59	07:00 08:59	09:00 09:59	10:00 13:59	14:00 14:59	15:00 17:59	18:00 19:59	20:00 20:59	21:00 23:59
Flight Movements / hour		38	38	40	34	38	39	38	40	34	38
Arrivals and Departures / hour		26	26	26	23	26	26	26	26	23	26
Flight Movements / 15 min		12	12	12	10	12	12	12	12	10	12
Arrivals and Departures / 15 min		10	10	10	9	10	10	10	10	9	10
Terminal 1 Arrivals / hour	Schengen	2,500	2,500								
	Non- Schengen	1,500	2,000								
Terminal 1 Departures / hour	Schengen	2,300	2,300								
	Non- Schengen	1,500	1,500								
Terminal 2 Departures / hour	Schengen	600	600								
	Non- Schengen	300	450								
Apron / Number of aircraft		63	63								

Throughout this chapter, we solve each optimization model with the CPLEX 12.5 solver, implemented using the GAMS modeling language on a computer with an i7 processor @3.6 GHz, 8Gb RAM and the Windows 10 64-bit operating system. We refer to as direct CPLEX implementation the implementation of PSAM-RTA using only the exact methods of optimization provided by the CPLEX solver, i.e. without any other algorithm such as the one presented in this chapter. Table 4.2 reports results from direct CPLEX implementation of PSAM-RTA at six instances, using the sequential and weight-based approaches.

The first two instances use data from Madeira and Porto in 2014. At these mid-size airports, PSAM-RTA is solved to optimality in a few minutes. The third instance uses data from Lisbon in 2014 without terminal and apron capacities. As expected, the computational complexity of the slot allocation problem is significantly larger in Lisbon: PSAM-RTA is still solved to optimality, but CPLEX terminates in 5 hours with the weight-based approach and 9 hours with the sequential approach. Next, the growth in the number of slot requests in Lisbon from 2014 to 2015 results in an even more complex slot allocation problem. With 2015 data, it takes 3 days to find the optimal solution of PSAM-RTA with the weight-based approach. The sequential approach does not yield the optimal solution in 7 days, leaving an optimally gap of 0.1%. In the last two test instances, we also work with Lisbon data from 2015 but add terminal and apron capacity constraints. The terminal constraints provide cuts that restrict the model’s feasible integer region, thus reducing computational time from 3 days to 2.5 days with the weight-based approach and from over 7 days to 2.5 days with the sequential approach. In contrast, the apron capacity constraints introduce inter-period coupling and significantly increase the model's requirements. In the last instance, CPLEX cannot find the optimal PSAM-RTA solution after 7 days of computations, leaving an optimality gap of 2% with the sequential approach and 5% with the weight-based approach.

Table 4.2 – PSAM-RTA results using direct CPLEX implementation

Inst.	Airport	Season	Term.	Apron	Nr. of decision variables		Nr. of constraints	Priority approach	Total displacement (min)			Gap (%)	CPU Time
					Integer	Binary			CH	NE	OS		
1	Madeira	2014	No	No	1,212	176,346	889,462	Weight	4,780	0	6,840	0%	2 min
								Sequent.	4,780	0	6,840	0%	4 min
2	Porto	2014	No	No	2,388	347,346	1,993,956	Weight	3,560	5,220	29,845	0%	5 min
								Sequent.	3,560	5,220	29,845	0%	8 min
3	Lisbon	2014	No	No	5,058	735,939	2,567,255	Weight	27,790	2,990	352,100	0%	5 hours
								Sequent.	27,790	2,990	352,100	0%	9 hours
4	Lisbon	2015	No	No	6,898	1,003,659	3,358,397	Weight	23,385	17,550	365,285	0%	3 days
								Sequent.	23,385	17,550	365,345	0.1%	7 days
5	Lisbon	2015	Yes	No	6,898	1,003,659	3,691,247	Weight	23,385	20,450	374,005	0%	2,5 days
								Sequent.	23,385	20,450	374,005	0%	2,5 days
6	Lisbon	2015	Yes	Yes	6,898	1,003,659	3,872,687	Weight	23,385	20,450	381,620	5%	7 days
								Sequent.	23,385	20,450	375,810	2%	7 days

In conclusion, direct CPLEX implementation can solve PSAM-RTA in Madeira and Porto, which operate around 25,000 and 85,000 movements a year, respectively. However, it remains intractable for larger schedule-coordinated airports such as Lisbon, which operates over 200,000 movements a year. This is the motivation for the algorithmic developments presented in the following section.

4.3 Algorithm Development

We propose a scalable algorithm based on large-scale neighborhood search to solve PSAM-RTA. The goal of our algorithm is to derive optimal, or near-optimal, slot allocation solutions using data from Lisbon in 2015, where direct CPLEX implementation of PSAM-RTA remains intractable.

The proposed algorithm relies on the following logic. In general terms, there exists a "limit" for solving PSAM-RTA with commercial solvers. One of the main determinants of this limit is, of course, the number of slot requests. However, there is no one-to-one relationship between size and computational performance; for instance, all else being equal, the more significant the imbalances between slot demand and airport capacity, the more computation effort is required to solve PSAM-

RTA. Therefore, we subdivide the full set of slots into smaller subsets based on the size of the problem and other factors (e.g., demand-capacity imbalances).

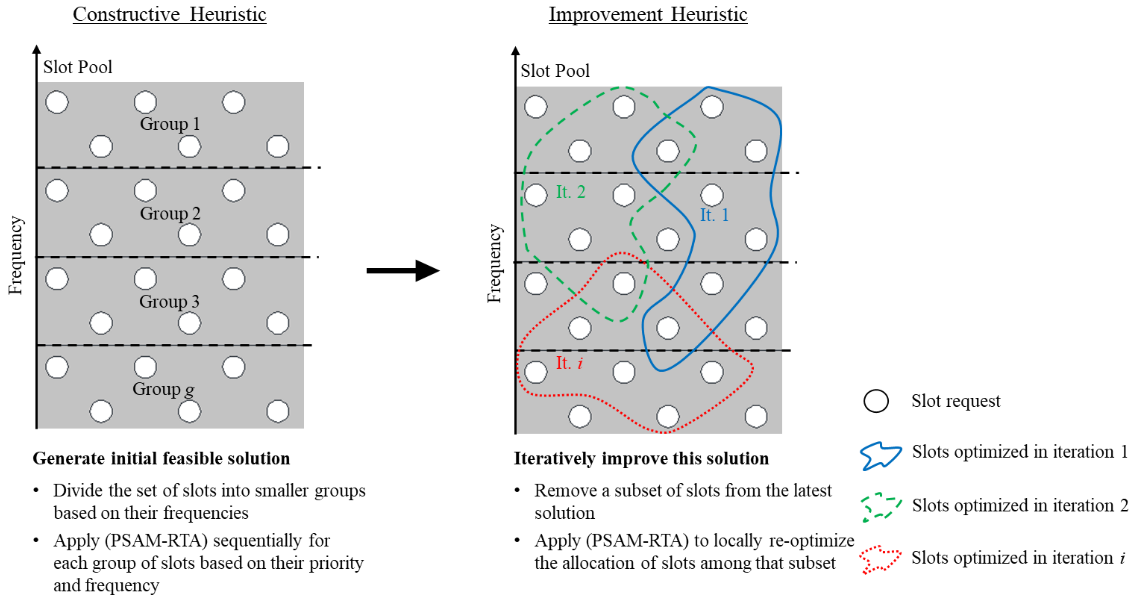


Figure 4.1 – Schematic representation of the heuristic proposed

Our algorithm involves a *constructive heuristic* and an *improvement heuristic* (shown in Figure 4.1). The constructive heuristic (Section 4.3.1) aims to find an initial feasible solution to PSAM-RTA by dividing the set of slot requests into smaller groups by decreasing order of priority (i.e. change-to-historic, new-entrants, and other slots) and frequency. Thus, for each priority class, the constructive heuristic allocates, first, the slots requested for the full season, then those requested on most weeks of the season, etc. Then, the improvement heuristic (Section 4.3.2) iteratively improves this solution using a “destroy and repair” approach. At each iteration, it removes a subset of slot requests from the assignment determined by the latest solution and solves PSAM-RTA for the remaining slot requests. The full set of slot requests is still included in the model to ensure global feasibility, rather than local feasibility. However, only a subset of all slot requests are re-allocated. In other words, the improvement heuristic explores the full solution space iteratively by decomposing the slot allocation problem into smaller sub-problems, fixing many decision variables to their previous values, and performing local optimization at each iteration. We also discuss in Section 4.3.3 additional algorithmic considerations that arise from the rules and priorities established by IATA’s Worldwide Slot Guidelines.

4.3.1 Constructive Heuristic

We develop a constructive heuristic to generate a feasible PSAM-RTA solution in short computational times. To do so, we decompose the initial problem into a set of smaller sub-problems that we solve in sequence, while maintaining global feasibility from one sub-problem to the next. The quality of this constructive heuristic depends on the decomposition approach. While we do not aim to reach optimality, we nonetheless thrive to leverage the structure of the problem to obtain a good "starting point" for the improvement heuristic.

Our constructive heuristic relies on the idea that slot requests with higher frequencies constrain PSAM-RTA solution to a greater extent than slot requests with lower frequencies. At one extreme, if all slots were requested for a single day, then slot allocation could be performed independently for each day, thereby considerably simplifying PSAM-RTA combinatorics. In contrast, slot requests with higher frequency create coupling constraints across multiple days of season. Therefore, in our constructive heuristic, we start by allocating the slot requests that span multiple days of operations, which exhibit the least flexibility. We then proceed by decreasing order of frequency. We allocate last the requests spanning fewer days of operations, which can be allocated more flexibly at various times of the day. We apply this approach to each priority class in sequence (see details in Section 4.3.3).

Equation (4.33) computes the frequency f_i of each slot request $i \in \boldsymbol{S}$, given by the number of days on which the flight under consideration will be scheduled. We then partition the full set of slot requests $\boldsymbol{S}$ into groups by decreasing order of frequency, f_i, and optimize slot allocation within each group sequentially using PSAM-RTA. We detail these two steps next.

$$f_i = \sum_{t \in \boldsymbol{T}} \sum_{d \epsilon \boldsymbol{D}} \left(Y_{it} - Y_{i,t+1}\right) B_{id}\,, \qquad \forall i \,\epsilon\, \boldsymbol{S} \tag{4.33}$$

The effectiveness of the constructive heuristic depends on the number of groups considered. At one extreme, if we consider a single group, the constructive heuristic is equivalent to solving PSAM-RTA with the full set of inputs, thus leading to prohibitively long computational times. At the other extreme, a very large number of groups would involve myopic optimization, essentially allocating slots one by one without considering the coupling constraints across slot requests and across time periods. Therefore, we hypothesize that there exists a "sweet spot" in terms of the number of groups that one should consider in the constructive heuristic. We explore this question computationally in Section 4.4.

a) Procedure to Generate the Groups of Slots

We denote by G the number of groups and by $\bar{\mu}$ the average number of slots per group, i.e.:

$$\bar{\mu} = \frac{\sum_{i \epsilon S} f_i}{G} \tag{4.34}$$

For instance, assume that the set $\boldsymbol{S}$ comprises 50 slot requests of 10 slots each and 50 slot requests of 5 slots each. If we generate 5 groups, each group will comprise 150 of the 750 slots requested. Slots are distributed sequentially based to frequency: groups 1, 2 and 3 comprise 15 requests of 10 slots; group 4 comprises 5 requests of 10 slots and 20 requests of 5 slots; and group 5 comprises 30 requests of 5 slots.

Procedure to generate the groups of slots

1: **input**: Number of groups of slots *G*; PSAM-RTA data.
2: Compute the frequency of each series of slots f_i (Equation (4.33)) and average number of slots per group $\bar{\mu}$ (Equation (4.34))
3: Define sets $\boldsymbol{S_g} = \emptyset, \forall g = 1, \dots, G$, where $\boldsymbol{S_g}$ is the set of slots included in group g
4: **Initialize:** *g*=1; $\boldsymbol{S^o} = \emptyset$ ($\boldsymbol{S^o}$ indicates the set of slots that have been assigned to a group).
5: ***while*** $\boldsymbol{S^o} \neq \boldsymbol{S}$
6: Determine the subset of remaining slot requests in $\boldsymbol{S} \setminus \boldsymbol{S^o}$ with the highest frequency, denoted by $\boldsymbol{S^F}$
7: Select randomly a slot request j within $\boldsymbol{S^F}$
8: Allocate slot request *j* to group $g \; \epsilon \; \boldsymbol{G}$ by updating set $\boldsymbol{S_g} = \boldsymbol{S_g} \cup \{j\}$ and $\boldsymbol{S^o} = \boldsymbol{S^o} \cup \{j\}$
9: ***if*** $|\boldsymbol{S_g}| > n$ ***then***
10: $g = g + 1$
11: ***end-if***
12: ***end-while***
13: ***for*** $(i, j) \in \boldsymbol{P}$
14: Let g be the group such that $i \in \boldsymbol{S_g}$ and h be the group such that $j \in \boldsymbol{S_h}$,
15: Set $\boldsymbol{S_m} = \boldsymbol{S_m} \cup \{j\}$ and $\boldsymbol{S_M} = \boldsymbol{S_M} \setminus \{j\}$, where $m = \min(g, h)$ and $M = \max(g, h)$
16: ***end-for***
17: **outputs:** Return sets $\boldsymbol{S_g}, \forall g = 1, \dots, G$

The procedure to generate the groups of slots is detailed as follows. It mainly involves sorting the set of slot requests by decreasing order of frequency and selecting $\bar{\mu}$ slot requests for each group. Nonetheless, the number of slots in each group may be slightly higher or lower than $\bar{\mu}$, for three reasons. First, Equation (4.34) may provide fractional values. Based on the procedure presented (rows 9 to 11), all groups but the last one will have $\bar{\mu}$ elements rounded up to the closest integer. Second, slots belonging to the same request must be allocated to the same group of slots to apply schedule regularity constraints. Third, we must allocate all pairs of slot requests tied by a connection – that is, any pair $(i,j) \in \boldsymbol{P}$ – to the same group to apply connectivity constraints. If two connected series fall into different groups, the procedure (rows 13 to 16) moves the one in the lower-priority group to the higher-priority one.

Note that the groups of slots could also be generated by assigning the same number of slot requests to each group (instead of the same number of slots). This would involve replacing Equation (4.34) by $\bar{\mu} = S/G$, without weighting slot requests by their frequencies. However, we found that this leads to longer computation times and lower solution quality.

b) Intra-group Optimization

Once all groups are created, we solve PSAM-RTA sequentially, from the group with the highest-frequency requests to the group with the lowest-frequency requests. Note that, once each solution is obtained (for any group), we need to update the capacity inputs to reflect that a lower value of capacity remains available for the subsequent groups. We define in Equations (4.35) to (4.38) the effective values of the runway, terminal and apron capacities when slots in a subset $\boldsymbol{S_A}$ have been allocated according to solution Y – that is, the remaining capacities for all slots in $\boldsymbol{S} \setminus \boldsymbol{S_A}$. We denote them by $\widehat{CR}_{sdc}(\boldsymbol{S_A}, Y)$, $\widehat{CT}_{sdc}(\boldsymbol{S_A}, Y)$ and $\widehat{CA}_{td}(\boldsymbol{S_A}, Y)$, respectively. We will also use these definitions later in this section to define our improvement heuristic.

$$\widehat{CR}_{sdc}(\boldsymbol{S_A}, Y), = CR_{sdc} - \sum_{i \epsilon \boldsymbol{S_A}} \sum_{t=s}^{s+L_c} (Y_{it} - Y_{i,t+1}) B_{id}\,, \qquad \forall\, d \,\epsilon\, \boldsymbol{D}, s \,\epsilon\, \boldsymbol{T} | s < T - L_c + 1, c \,\epsilon\, \boldsymbol{C} \tag{4.35}$$

$$\widehat{CT}_{sdc}(\boldsymbol{S_A}, Y) = CT_{sdc} - \sum_{i \epsilon \boldsymbol{S_A}} \sum_{t=s}^{s+L_c} (Y_{it} - Y_{i,t+1}) B_{id} I_{ik} S_i LF_i\,, \quad \forall d \,\epsilon\, \boldsymbol{D}, s \,\epsilon\, \boldsymbol{T} | s < T - L_c + 1, c \,\epsilon\, \boldsymbol{C}, k \,\epsilon\, \boldsymbol{K} \tag{4.36}$$

$$\hat{Q}_{\mathrm{td}} = \sum_{i \epsilon \boldsymbol{S}_{arr} \cap \boldsymbol{S_A}} (Y_{it} - Y_{i,t+1}) B_{id} - \sum_{i \epsilon \boldsymbol{S}_{dep} \cap \boldsymbol{S_A}} (Y_{it} - Y_{i,t+1}) B_{id} + \hat{Q}_{\mathrm{t-1,d}}\,, \qquad \forall\, d \,\epsilon\, \boldsymbol{D}, t \,\epsilon\, \boldsymbol{T} \tag{4.37}$$

$$\widehat{CA}_{td}(\boldsymbol{S_A}, Y) = CA_{td} - \hat{Q}_{\mathrm{td}} - \left| \min_{t \in \boldsymbol{T}} \hat{Q}_{\mathrm{td}} \right|, \qquad \forall\, d \,\epsilon\, \boldsymbol{D}, t \,\epsilon\, \boldsymbol{T} \tag{4.38}$$

Constructive heuristic

1: **input:** Number of groups of slots G; PSAM-RTA data.
2: Apply the procedure to generate the groups of slots: $\boldsymbol{S_g}, \forall g = 1, \dots, G$
3: Initialize subset of slots that have already been allocated, denoted by $\boldsymbol{S_A}$: $S_A = \emptyset$
4: Initialize solution of constructive heuristic, denoted by Y^C: $Y_{it}^C = 0, \forall\, i \in \boldsymbol{S}, t \in \boldsymbol{T}$
5: ***for*** $g = 1, \dots, G$
6: Define capacities $\widehat{CR}_{sdc}(\boldsymbol{S_A}, Y^C)$, $\widehat{CT}_{sdc}(\boldsymbol{S_A}, Y^C)$ and $\widehat{CA}_{td}(\boldsymbol{S_A}, Y^C)$ (Equations (4.35) to (4.38))
7: Optimize the slots belonging to group g using PSAM-RTA, using capacities $\widehat{CR}_{sdc}(\boldsymbol{S_A}, Y^C)$, $\widehat{CT}_{sdc}(\boldsymbol{S_A}, Y^C)$ and $\widehat{CA}_{td}(\boldsymbol{S_A}, Y^C)$
8: Update subset of slots that have already been allocated: $S_A = \boldsymbol{S_1} \cup \dots \cup \boldsymbol{S_g}$
9: Update constructive heuristic solution, Y_{it}^C, according to the results of PSAM-RTA
10: ***end-for***

4.3.2 Improvement Heuristic

The improvement heuristic starts from the output of the constructive heuristic, and proceeds with "destroy and repair" perturbations to improve the solution iteratively. At each iteration, PSAM-RTA is solved for a subset of slot requests, $\boldsymbol{S^I} \subset \boldsymbol{S}$, by fixing all other slot requests in $\boldsymbol{S} \setminus \boldsymbol{S^I}$ to the times allocated in the previous solution. Specifically, we include all slot requests in $\boldsymbol{S}$ in the model to ensure global feasibility, but restrict the decision variables to the subset $\boldsymbol{S^I} \subset \boldsymbol{S}$. We denote by Y^I the latest solution and compute the remaining runway, terminal and apron capacities, i.e., $\widehat{CR}_{sdc}(\boldsymbol{S} \setminus \boldsymbol{S^I}, Y^I)$, $\widehat{CT}_{sdc}(\boldsymbol{S} \setminus \boldsymbol{S^I}, Y^I)$ and $\widehat{CA}_{td}(\boldsymbol{S} \setminus \boldsymbol{S^I}, Y^I)$, respectively (Equations (4.35) to (4.38)). At each iteration, the solution will be at least as good as the previous one.

The critical idea underlying the improvement heuristic is to subdivide the set of slot requests $\boldsymbol{S}$ into smaller subsets $\boldsymbol{S^I}$ by decomposing the problem over *times of the day*. This is motivated by the fact that most of the coupling constraints (e.g., airport capacity constraints, aircraft connection constraints, schedule regularity constraints) apply across slot requests that take place at approximately the same time of the day. For instance, if one perturbs the allocation of a slot request from 10:00 to 10:05, the capacity thus made available will probably be captured by slot requests that were requested within the same time window (e.g. between 09:30 and 10:30); in contrast, the other requests (e.g., those taking place at 7 pm) will not benefit as much from this perturbation. Our improvement heuristic leverages this observation by re-optimizing, at each iteration, the allocation of slot requests within a time-based neighborhood to capture most of the aforementioned interdependencies. Figure 4.2 illustrates this procedure.

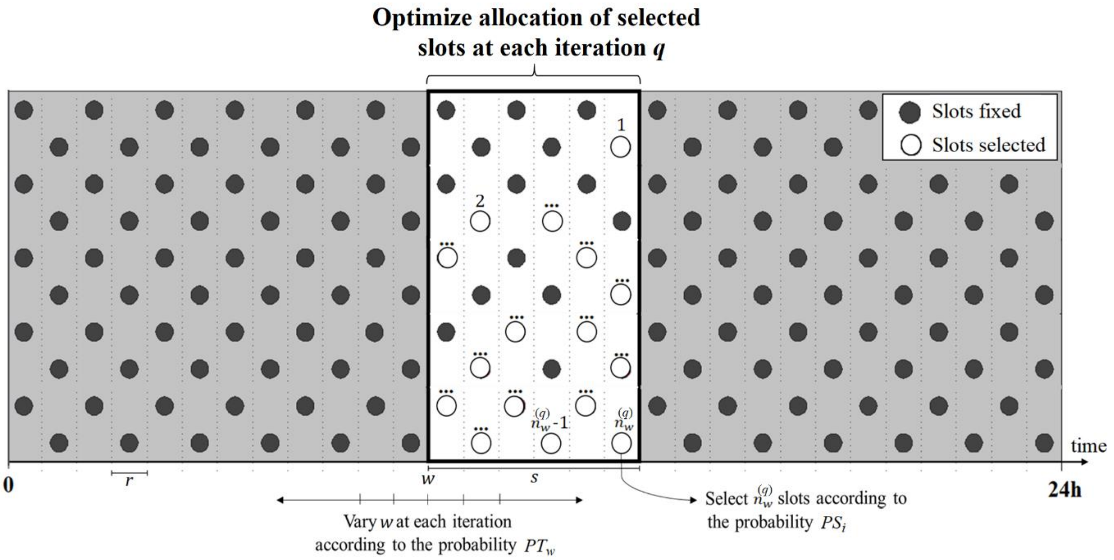


Figure 4.2 – Schematic representation of the improvement heuristic

Specifically, we define a set of time windows $\boldsymbol{W}$ by specifying two parameters. First, let s denote the size of the time windows – e.g., a time window from 9:00 to 10:00 has a size $s = 1$ hour. Second, let $r \leq s$ denote the size of rolling periods – e.g., if $r = 30$ minutes, we create time windows from 09:00 to 10:00; from 09:30 to 10:30, etc. Note that if $s = r$, the time windows are non-overlapping and the benefits obtained by exchanging slots across two consecutive time windows will not be captured. On the other hand, if $r << s$, any slot request will be included in many time windows, thus increasing the computational requirements of the algorithm.

At each iteration, we select a time window $w \in \boldsymbol{W}$. Let $\boldsymbol{T_w} \subseteq \boldsymbol{T}$ be the set of time periods in time window w, and $\boldsymbol{S_w^0} \subseteq \boldsymbol{S}$ be the subset of slots requested to depart or arrive at the airport during time window w. We then augment $\boldsymbol{S_w^0}$ by adding the slots that are connected to one of the slots requested to depart or arrive in w (to maintain aircraft connections). We denote the resulting subset of slots by $\boldsymbol{S_w} \subseteq \boldsymbol{S}$. This is written in Equations (4.39) and (4.40).

$$\boldsymbol{S_w^0} = \left\{ i \in \boldsymbol{S} \middle| \sum_{t \in \boldsymbol{T_W}} \left(A_{it} - A_{i,t+1}\right) = 1 \right\} \tag{4.39}$$

$$\boldsymbol{S_w} = \boldsymbol{S_w^0} \cup \{i \in \boldsymbol{S} | \exists j \in \boldsymbol{S_w^0}, (i,j) \in \boldsymbol{P}\} \cup \{j \in \boldsymbol{S} | \exists i \in \boldsymbol{S_w^0}, (i,j) \in \boldsymbol{P}\} \tag{4.40}$$

Through the improvement heuristic, PSAM-RTA will be solved for slot requests in $\boldsymbol{S_w}$ at each iteration. Ideally, we would solve PSAM-RTA to all slot requests in $\boldsymbol{S_w}$, while fixing all others. However, for large values of s, this might require long computational times (especially in periods with strong demand-capacity imbalances). This concern could be alleviated by decreasing the size of the time windows s. However, if s is set too low, the time windows will be too small to allow sufficient flexibility to swap slots across time periods. Therefore, we adopt another approach by specifying a maximum runtime for PSAM-RTA at each iteration, denoted by t_o. If the optimal solution for time window $w \in \boldsymbol{W}$ has not been found after t_o, then we stop the run, store the best solution found, and proceed to the next iteration.

When, for a given time window $w \in \boldsymbol{W}$, PSAM-RTA is not solved to optimality within the limit of t_o, this indicates that slot allocation in time window w is computationally intensive. Thus, whenever the algorithm returns to time window w, it will select a subset of slots in $\boldsymbol{S_w}$ for PSAM-RTA optimization (according to some probability distribution described below). In other words, we further divide time window w into smaller neighborhoods. Specifically, we reduce the number of slot requests selected by a factor ρ. For example, if time window $w \in \boldsymbol{W}$ comprises 200 slots, PSAM-RTA is not solved optimally for that time window and ρ is equal to 0.8, then the next time the algorithm visits time window w, we sample only 160 slot requests out of the 200 original ones. If PSAM-RTA is then solved optimally, we continue to select 160 slots for time window w moving forward; otherwise, we select 128 slots in the following iteration, etc. In the remainder of this section, we denote by $n_w^{(q)}$ the number of slots included in PSAM-RTA in time window w at iteration q. By design, $n_w^{(1)} = |\boldsymbol{S_w}|$.

We now describe how we select the time window $w \in \boldsymbol{W}$ in each iteration. A simple option would be to select time windows cyclically according to a deterministic sequence, until some stopping criterion is met (e.g., maximum computation time, solution quality). However, this option may not be most efficient, as it may repeatedly consider time windows where the latest solution is already close to optimality. Instead, we adopt a probabilistic approach to orient the search toward neighborhoods where solution improvements are more likely. Specifically, we select time windows according to a probability distribution, $PT_w^{(q)}$, defined as follows. First, $PT_w^{(1)}$ is initialized to prioritize time windows with larger displacements (from the constructive heuristic). This is shown in Equation (4.41), where X_i^C denotes the displacement of slot i under the solution of the constructive heuristic.

$$PT_w^{(1)} = \frac{\sum_{i \in \boldsymbol{S_w}} X_i^C}{\sum_{w' \in \boldsymbol{W}} \sum_{i \in \boldsymbol{S_{w'}}} X_i^C}, \forall w \in \boldsymbol{W} \tag{4.41}$$

At each iteration q, probabilities $PT_w^{(q)}$ are updated using a history-based approach similar to that proposed by Ropke and Pisinger (2006). In other words, we leverage the latest solution to identify time windows where most significant solution improvements are expected. Specifically, if an improvement is found at an iteration, then probabilities $PT_w^{(q)}$ are not changed. Otherwise, they are updated according to Equations (4.42) to (4.44), where β and γ_q are calibration parameters, and $R_w^{(q)}$ denotes the ratio of the number of slots selected at iteration q to the total number of slots requested in time window w.

$$PT_w^{(q)\prime} = PT_w^{(q-1)} \times e^{-\beta\left(R_w^{(q)}\right)^{\gamma_q}} \tag{4.42}$$

$$PT_{w'}^{(q)} = \frac{PT_w^{(q)\prime}}{\sum_{u\epsilon \boldsymbol{W}} PT_u^{(q)\prime}} \quad ,\forall w' \epsilon \boldsymbol{W} \tag{4.43}$$

$$R_w^{(q)} = \frac{n_w^{(q)}}{|\boldsymbol{S_W}|} \tag{4.44}$$

If no improvement is found in a given time window w after the time limit t_0, then we infer that, with high likelihood, no improvement will be found the next time we visit time window w. Therefore, we reduce the corresponding selection probability $PT_w^{(q)}$ (Equation 4.42). This is particularly true if all the slots of time window w were selected – i.e., $R_w^{(q)} = 1$. In this case, the chances of finding another improvement in this time window are very slim. We thus set the value of β sufficiently large so that the probability of re-visiting a time window that was solved to optimality with all the slots is almost zero, i.e., $e^{-\beta} \approx 0$.

Then, to simplify the algorithm's parametrization exposition, we replace γ_q by another parameter δ as follows:

$$R_{min}^{(q)} = \min_{w'\in \boldsymbol{W}} R_{w'}^{(q)} \tag{4.45}$$

$$\delta = e^{-\beta\left(R_{min}^{(q)}\right)^{\gamma_q}} \text{ – i.e., } \gamma_q = \log_{R_{min}^{(q)}}\left(-\frac{\ln \delta}{\beta}\right) \tag{4.46}$$

This change is motivated by the fact that the impact of δ (or γ_q) is larger when a lower proportion of slots are selected in any given time window, i.e., for smaller values of $R_w^{(q)}$. Therefore, we define $R_{min}^{(q)}$ as the lowest value of $R_w^{(q)}$ across all time windows. At one extreme, if $R_w^{(q)} = 1$, the parameter δ does not matter, and $PT_w^{(q)} \approx 0$ per our earlier discussion. At the other extreme, when $R_w^{(q)} = R_{min}^{(q)}$, then the factor $e^{-\beta\left(R_w^{(q)}\right)^{\gamma q}}$ is equal to δ. Thus, the smaller δ, the more significant the variations in probabilities $PT_w^{(q)}$ from one iteration to the next. The relationship between δ and γ_q is specified in Equation (4.46). Note that $R_{min}^{(q)}$ and γ_q may vary from one iteration to the next but, by design, δ is a constant parameter.

We now turn to the selection of the $n_w^{(q)}$ slots subject to re-optimization within each time window $w \in \boldsymbol{W}$. Again, we select them according to a probability distribution, denoted by $PS_i^{(q)}$. This approach prioritizes slot requests that were selected fewer times in previous iterations to ensure sufficient exploration. $PS_i^{(q)}$ is given in Equation (4.47), where λ is a calibration parameter, and $s_i^{(q)}$ indicates the number of times slot request i was selected up to iteration q. The larger the λ, the more the algorithm favors slot requests that were selected fewer times before. If $\lambda = 0$, slot requests are selected completely at random; if $\lambda = \infty$ we select slot requests exclusively among those that were explored the least numbers of times in previous iterations.

$$PS_i^{(q)} = \frac{e^{-\lambda s_i^{(q)}}}{\sum_{j \in \boldsymbol{S}} e^{-\lambda s_j^{(q)}}} \quad ,\forall i \epsilon \boldsymbol{S_W} \tag{4.47}$$

Finally, at each iteration, we know a feasible solution for the problem (obtained initially from the constructive heuristic, and then from the latest iteration of the improvement heuristic). Therefore, we use “warm start” techniques to solve PSAM-RTA with a good upper bound (Klotz and Newman, 2013). This reduces considerably the computation time required at each iteration.

Table 4.3 lists the inputs of the improvement heuristic, and their baseline values. These values are set based on initial computational experience, and will be discussed in Section 4.5.

Note that the improvement heuristic is stochastic, as the time windows and, in some cases, the subset of slots under consideration, are selected randomly at each iteration. Therefore, we implement the heuristic 10 times for each test instance. As we shall see, the overall performance of the heuristic exhibits little variability across runs in most test instances.

Table 4.3 – Inputs of the improvement heuristic and baseline values

Parameters	Symbols	Baseline values
Size of the time windows	s	60 min
Size of the rolling periods	r	30 min
Maximum computation time for optimization at each iteration	t_o	10 min
Decrease factor of number of slots selected	ρ	0.8
Calibration parameters for updating the probability distribution, $PT_w^{(q)}$	β	5
	δ	0.8
Calibration parameters for updating the probability distribution, $PS_i^{(q)}$	λ	1

Improvement Heuristic

input: PSAM-RTA data; Constructive heuristic solution (X^C, Y^C); Improvement heuristic parameters (Table 4.3)
Compute the total displacement of the current solution, $TD^C = \sum_{i \epsilon \boldsymbol{S}} \sum_{d \epsilon \boldsymbol{D}} B_{id} X_i^C$
Initialize with the constructive heuristic solution: $Y^I = Y^C$ and $TD^I = TD^C$
Generate the set of time windows $\boldsymbol{W}$, according to parameters s and r
Compute initial $PT_w^{(1)}$ for each $w \in \boldsymbol{W}$ (Equation (4.41))
Compute $PS_i^{(1)}$ for each $i \in \boldsymbol{S_W}$ (Equation (4.47))
initialize: $q = 1$
while Stopping criterion is not met (maximum computation time, or convergence for a solution)
    Select a time window, w, according to probability distribution $PT_w^{(q)}$
    Initialize: $n^s = 0$, $\boldsymbol{S^I} = \emptyset$
    while $n^s < n_w^{(q)}$
        select a slot request $i \in \boldsymbol{S_W}$ according to PS_i
        set $\boldsymbol{S^I} = \boldsymbol{S^I} \cup \{i\}, n^s = n^s + f_i$
    end-while
    Update $s_i^{(q+1)} = s_i^{(q)} + 1$, $\forall i \in \boldsymbol{S^I}$, and update $PS_i^{(q+1)}$ for each $i \in \boldsymbol{S^I}$ (Equation (4.47))
    Update capacities $\widehat{CR}_{sdc}(\boldsymbol{S} \setminus \boldsymbol{S^I}, Y^I)$, $\widehat{CT}_{sdc}(\boldsymbol{S} \setminus \boldsymbol{S^I}, Y^I)$ and $\widehat{CA}_{td}(\boldsymbol{S} \setminus \boldsymbol{S^I}, Y^I)$ (Equations (4.35) to (4.38))
    Consider initial feasible solution given by Y^I for warm starting (PSAM-RTA)
    Solve (PSAM-RTA), using capacities $\widehat{CR}_{sdc}(\boldsymbol{S} \setminus \boldsymbol{S^I}, Y^I)$, $\widehat{CT}_{sdc}(\boldsymbol{S} \setminus \boldsymbol{S^I}, Y^I)$ and $\widehat{CA}_{td}(\boldsymbol{S} \setminus \boldsymbol{S^I}, Y^I)$
    Store the solution found at time t_o: Y' and the corresponding displacement TD'
    if Solution found is not optimal for time window w ***then***
        $n_w^{(q+1)} = \rho \cdot n_w^{(q)}$
    end-if
    Compute $R_w^{(q+1)}$ for each $w \in \boldsymbol{W}$ (Equation 4.44), $R_{min}^{(q+1)}$ (Equation 4.45) and γ_{q+1} (Equation (4.46))
    if $TD' = TD^I$ ***then***
        Compute $PT_w^{(q+1)}$ for each $w \in \boldsymbol{W}$ using Equations (4.42) and (4.43)
    else
        Set $PT_w^{(q+1)} = PT_w^{(q)}$ for each $w \in \boldsymbol{W}$
    end-if
    Update the improvement heuristic solution, $Y^I = Y'$ and $TD^I = TD'$
    $q = q + 1$
end-while

a) Illustration Example

We illustrate the improvement heuristic in a hypothetical example. We consider 3 time windows of size $s = 12$ hours with a rolling period of $r = 6$ hours, i.e., windows 1, 2 and 3 span from 00:00 to 12:00, from 06:00 to 18:00, and from 12:00 to 24:00, respectively. The other parameters are set to their baseline values (see Table 4.3). From the constructive heuristic solution, let us assume that the initial probabilities of time window selection are $PT_1^{(1)} = 0.25$, $PT_2^{(1)} = 0.50$ and $PT_3^{(1)} = 0.25$. Table 4.4 shows 10 iterations of the improvement heuristic in this setting.

Table 4.4 – Evolution of the improvement heuristic

q	W	Optimal solution?	Solution improved?	Value of $R_w^{(q+1)}$			Value of $R_{min}^{(q+1)}$	Value of γ_{q+1}	Value of $PT_w^{(q+1)}$		
				1	2	3			1	2	3
0	-	-	-	1	1	1	1	-	25.0%	50.0%	25.0%
1	2	**No**	Yes	1	**0.8**	1	**0.8**	**13.93**	25.0%	50.0%	25.0%
2	2	**No**	**No**	1	**0.64**	1	**0.64**	**6.97**	**27.8%**	**44.4%**	**27.8%**
3	1	Yes	**No**	1	0.64	1	0.64	6.97	**0.3%**	**61.4%**	**38.4%**
4	3	**No**	Yes	1	0.64	**0.8**	0.64	6.97	0.3%	61.4%	38.4%
5	2	Yes	**No**	1	0.64	0.8	0.64	6.97	**0.3%**	**56.0%**	**43.7%**
6	3	Yes	Yes	1	0.64	0.8	0.64	6.97	0.3%	56.0%	43.7%
7	2	Yes	**No**	1	0.64	0.8	0.64	6.97	**0.3%**	**50.4%**	**49.2%**
8	2	Yes	**No**	1	0.64	0.8	0.64	6.97	**0.4%**	**44.9%**	**54.8%**
9	3	Yes	Yes	1	0.64	0.8	0.64	6.97	0.4%	44.9%	54.8%
10	2	Yes	Yes	1	0.64	0.8	0.64	6.97	0.4%	44.9%	54.8%

In the first iteration, window 2 is randomly selected. Initially, we have $R_2^{(1)} = 1$, so all slots of window 2 are selected. After $t_o = 10$ minutes, we make two observations. First, the solution is not optimal for the subproblem under consideration. Thus, the number of slots to select in window 2 in subsequent iterations is reduced by a factor $\rho = 0.8$. Therefore, $R_{min}^{(2)} = 0.8$ and $\gamma_2 = 13.93$ (Equation (4.46)). Second, the solution is better than the previous one. Therefore, we select the next time window with the same probabilities: $PT_1^{(2)} = 0.25$, $PT_2^{(2)} = 0.50$ and $PT_3^{(2)} = 0.25$.

In iteration 2, window 2 is selected again. Now, only 80% of its slots are selected, according to probability distribution $PS_i^{(q)}$ (Equation (4.47)). Again, the solution found after t_o is not optimal, so $R_2^{(3)} = 0.8 \times 0.8 = 0.64$. Accordingly, $R_{min}^{(3)} = 0.64$ and $\gamma_3 = 6.97$. This time, the solution is not improved after t_o. Therefore, we reduce the probability of selecting window 2 in the following iterations. Specifically, the probabilities $PT_w^{(3)}$ are computed as follows:

$$PT_2^{(3)'} = 0.5 \times e^{-5(0.64)^{6.97}} = 0.400 \tag{4.48}$$

$$PT_1^{(3)} = PT_3^{(3)} = \frac{0.25}{0.25 + 0.4 + 0.25} = 0.278 \tag{4.49}$$

$$PT_2^{(3)} = \frac{0.4}{0.25 + 0.4 + 0.25} = 0.444 \tag{4.50}$$

In the third iteration, window 1 is selected. Since this window has not been explored yet, all slots are selected, i.e. $R_1^{(3)} = 1$. After $t_o = 10$ minutes, the solution does not improve the latest one, so $PT_w^{(4)}$ are computed from Equations (4.42) and (4.43). In this case, the solution obtained is proved to be optimal (with all slots from Window 1), so the probability of selecting window 1 becomes $PT_1^{(4)} \approx 0$.

We proceed similarly for the next iterations. After 10 iterations, window 1 was selected once (with no improvement), window 2 was selected 6 times (with improvements in two iterations), and window 3 was selected 3 times (with improvements each time). After 8 iterations, the probability of selecting window 3 becomes larger than the probability of selecting window 2. This shows that the heuristic orients the search to explore time windows where solution improvements are most likely.

4.3.3 Consideration of IATA Priorities

Finally, we design adjustments to the algorithm to account for the priorities among the four slot classes. Specifically, we subdivide our problem into three sub-problems (historic slots are omitted because they are automatically allocated). We describe each of the sub-problems below.

(i) Change-to-historic slots: We implement directly the constructive heuristic and the improvement heuristic. This provides the time allocated to each change-to-historic slot $i \epsilon \boldsymbol{S_{CH}}$, i.e., Y_{it} for $i \in \boldsymbol{S_{CH}}, t \in \boldsymbol{T}$, and the total displacement of change-to-historic slots, denoted by TD^{CH}.

(ii) New-entrants slots: We first implement the constructive heuristic by fixing all the change-to-historic slots to the slot times given by Y_{it} for $i \in \boldsymbol{S_{CH}}, t \in \boldsymbol{T}$ (and by updating capacities with Equations (4.35) to (4.38)). We then implement the improvement heuristic, but instead of fixing the slot times provided by Y_{it} for $i \in \boldsymbol{S_{CH}}, t \in \boldsymbol{T}$, we limit the total displacement of the change-to-historic slots to TD^{CH}. This is formulated in Equation (4.51). This is because there might exist several solutions with a change-to-historic displacement equal to TD^{CH}, so fixing the variables might over-constrain the problem. This yields the (possibly new) allocation time of each change-to-historic slot, i.e., Y_{it} for each $i \in \boldsymbol{S_{CH}}, t \in \boldsymbol{T}$, and the allocation time of each new-entrant slot, i.e., Y_{it} for $i \in \boldsymbol{S_{NE}}, t \in \boldsymbol{T}$. This also provides the total displacement of new-entrants slots, denoted by TD^{NE}.

(iii) Other slots: We follow the same procedure as in (ii), with Constraints (4.51) and (4.52), defined below, added to PSAM-RTA to comply with the total displacement of change-to-historic and new-entrant slots, respectively. This provides the (possibly new) allocation time of each change-to-historic and new-entrant slots, i.e., Y_{it} for $i \epsilon \boldsymbol{S_{CH}} \cup \boldsymbol{S_{NE}}, t \in \boldsymbol{T}$, as well as the time allocated to the other slots, i.e., Y_{it} for $i \in \boldsymbol{S_{OS}}, t \in \boldsymbol{T}$.

$$\sum_{i \in \boldsymbol{S_{CH}}} \sum_{t \epsilon \boldsymbol{T}} |Y_{it} - A_{it}| \leq TD^{CH} \tag{4.51}$$

$$\sum_{i \in \boldsymbol{S_{NE}}} \sum_{t \epsilon \boldsymbol{T}} |Y_{it} - A_{it}| \leq TD^{NE} \tag{4.52}$$

Another important aspect resulting from IATA priorities is that the constructive heuristic may lead to infeasible solutions when allocating change-to-historic slots, since these slots can only be scheduled in a restricted set of time periods. Specifically, the allocation of slots belonging to one of the early groups generated by the procedure presented in Section 4.3.1 restricts the capacity that remains available at the historic and requested times (for "CL" slots) or between the historic and requested times (for "CR" slots), and thus constrains the allocation of the slot requests belonging to the subsequent groups. To address this issue, we add to the procedure to generate the groups of slots (Section 4.3.1) a new step before step 6 to fix (temporarily) the allocation of all the change-to-historic slots belonging to the subsequent groups to their historic time (determined through the parameters H_{it}). In other words, we update the capacities using Equations (4.35) to (4.38), by setting $Y_{it} = H_{it}$ for slots belonging to groups from $g + 1$ to G (where g indicates the group under consideration). Then, as we progress from group g to group $g + 1$ (until group G), these allocations are revisited in a way that maintains global feasibility of the allocation of change-to-historic slots.

4.4 Experimental Results

We now present the results of the algorithm using Lisbon data from 2015. In this case, PSAM-RTA can be solved for the change-to-historic slots in 4-6 minutes and for the new-entrant slots in 1-2 minutes. In contrast, its computational requirements for the other slots are much more significant. Recall, from Table 4.2, that without terminal and apron constraints, direct CPLEX implementation of PSAM-RTA finds the optimal solution in 3 days, with a total displacement for other slots of 365,285 minutes, and that, with terminal and apron constraints, direct CPLEX implementation does not find the optimal solution after 7 days. In the latter case, the best available solution has a total displacement for other slots of 375,810 minutes and a 2% optimality gap (the best available lower bound is equal to 368,295).. This section reports results of the constructive and improvement heuristics for the other slots requests, with the objective of finding faster and/or higher-quality solutions than direct CPLEX implementation of PSAM-RTA.

Note, also, that the total displacement of 374,005 minutes obtained with the terminal constraints but without the apron constraints (5th test instance in Table 4.2) provides a lower bound of the optimal displacement with the terminal and apron constraints (6th test instance in Table 4.2). This lower bound is, in fact, tighter than the one obtained with CPLEX with the terminal and apron constraints after 7 days. In the remainder of this section, we thus compare any solution obtained with the terminal and apron constraints to this value of 374,005 minutes. Accordingly, the baseline optimality gap obtained with direct CPLEX implementation (using the sequential approach) after 7 days is equal to 0.48% (i.e., the difference between 375,810 and 374,005), rather than 2% as reported in Table 4.2. As we shall see in Section 4.4.2, some of the solutions found with the terminal and apron constraints will yield a total displacement of 374,005 minutes.

As we shall see in Section 4.2, some of our solutions will yield a total displacement of 374,005 minutes. This implies that the lower bound is equal to the optimal value of the total displacement with terminal and apron constraints. Therefore, in these cases, we shall be able to claim that the proposed algorithm reaches the optimal solution of PSAM-RTA with terminal and apron constraints.

4.4.1 Constructive Heuristic Results

The results of the constructive heuristics for other slots are reported in Table 4.5. We vary here the number of groups G from 2 to 10. We also analyze the extreme solution with groups of 1 slot, which simply allocates slots one by one. In this case, we consider two strategies for "ranking" slot requests: one that orders slot requests by decreasing frequency and selects slot requests randomly within each frequency group, and another one that selects slot requests completely at random within $\boldsymbol{S}$. We refer to these two strategies as "SimRandFreqs" and "SimRand", respectively, and perform 10 Monte Carlo simulations of each. From our discussions with slot coordinators, the "SimRandFreqs" strategy comes reasonably close to the approach employed in practice in the absence of optimization models such as PSAM-RTA.

Table 4.5 - Constructive heuristic results for Other slots

Slot allocation strategy	Number of slots *n* per group	Lisbon 2015 – No terminal and apron			Lisbon 2015 – With terminal and apron		
		Displacement (min)	Gap (%)	CPU Time	Displacement (min)	Gap (%)	CPU Time
SimRand	1	477,350	30.7%	5 hours	497,424	33.0%	10 hours
SimRandFreqs	1	386,585	5.8%	5 hours	399,310	6.8%	10 hours
10 Groups	1978	381,670	4.5%	13 min	394,040	5.4%	41 min
8 Groups	2472	379,315	3.8%	10 min	390,450	4.4%	37 min
7 Groups	2825	381,740	4.5%	9.6 min	392,290	4.9%	30 min
6 Groups	3296	379,450	3.9%	8.8 min	390,335	4.4%	31 min
5 Groups	3956	380,705	4.2%	7.5 min	393,335	5.2%	25 min
4 Groups	4945	374,070	2.4%	7.5 min	387,125	3.5%	22 min
3 Groups	6593	374,650	2.6%	11 min	381,365	2.0%	25 min
2 Groups	9889	370,755	1.5%	1h 4min	378,715	1.3%	6h 15 min
Direct CPLEX	19778	365,285	0.0%	3 days	374,005*	0.0%	>7 days

*Solution obtained through test instance 5

The main takeaway from Table 4.5 is that the constructive heuristic can provide good solutions in short runtimes. Specifically, we get a solution within 2.4% of the optimum in 7.5 minutes without the terminal and apron constraints, and a solution within 2% of the optimum in 25 minutes with the terminal and apron constraints. As detailed in Section 4.4.3, this solution, obtained with the constructive heuristic in less than 30 minutes, outperforms the best solution obtained with direct CPLEX implementation after 2 days. Moreover, the constructive heuristic improves the average SimRandFreqs solution by 6-7%. This demonstrates the benefits of our group-based constructive heuristic for providing a good initial solution, as compared to slot allocation strategies currently used by slot coordinators.

Overall, the total displacement decreases as fewer groups are generated. More groups induce stronger decomposition of the problem. One exception is that the 8-group solution is slightly better than the 7-group solution. This is due to the fact that slot requests that are more coupled (e.g., those that take place at approximately the same time of the day) may be included in different groups when $G = 7$, but in the same group when $G = 8$.

Note, next, that there exists a “sweet spot” in the number of groups to minimize the computational times of the constructive heuristic. On the one hand, if G is set too low, the computation time required to solve slot allocation within each group increases non-linearly, as the number of slot requests in each group increases. For instance, moving from 3 groups to 2 groups results in an extra 6 hours. One may thus expect that the total computation time would decrease with the number of groups G.

However, if G is too high, the total computation time increases due to a "fixed cost" of running each optimization model (e.g., data processing, linear relaxation, etc.). In our example, computational times increase when more than 5 groups are considered, albeit to a smaller extent than when $G \leq 3$. Therefore, the sweet spot seems to be 3 to 5 groups in the case of Lisbon. But the appropriate number of groups may vary from one airport to another. For instance, more groups may be warranted at busier airports to further decompose the problem. The modular design of our constructive heuristic permits to calibrate the size of the optimization problems to enable its scalability at the largest schedule-coordinated airports.

In conclusion, the constructive heuristic, although simple and easily implementable, can already provide benefits as compared to current practices and to direct CPLEX implementation. Nonetheless its solution remains sub-optimal. This motivates the implementation of the improvement heuristic.

4.4.2 Improvement Heuristic Results

We implement the improvement heuristic with the baseline inputs reported in Table 4.3, starting with the solution from the constructive heuristic with 4 groups. Thus, the initial value of the total displacement is equal to 374,070 minutes without the terminal and apron constraints, and to 387,125 minutes with the terminal and apron constraints. Note that any other constructive heuristic solution can be selected as initial solution for the improvement heuristic. As we shall see in Section 4.5.1, the solution with 4 groups converges faster to the optimal solution.

Recall that the improvement heuristic is probabilistic, as the time windows and the set of slot requests to consider at each iteration are selected randomly. Thus, we perform 10 runs in each instance, and report in Table 4.6 the average values of the number of iterations, the total displacement, and the optimality gap, as well as the range of optimality gap values.

Note that the improvement heuristic yields solutions within 0.1% of the optimum in about 3 hours without the terminal and apron constraints, and in 6 hours with the terminal and apron constraints. The exact optimal solution was found 7 out of 10 times without the terminal and apron constraints after 5 hours, and 3 out of 10 times with the terminal and apron constraints after 8 hours. [Recall that the latter solution is guaranteed to be optimal because it equals the lower bound provided with terminal constraints but without apron constraints.] This shows the strong performance of the algorithm developed here, which provides optimal or near-optimal solutions much faster than with direct CPLEX implementation.

Table 4.6 – Improvement heuristic results

CPU time	Lisbon airport – No terminal and apron				Lisbon airport – terminal and apron			
	Avg. no. iterations	Avg. displacement (min)	Avg. gap (%)	Gap Range (%)	Avg. no. iterations	Avg. displacement (min)	Avg. gap (%)	Gap Range (%)
0	0	374070	2.40	-	0	387130	3.51	-
30 min	4	370713	1.49	0.5-1.9	3	384890	2.91	1.4-3.5
1h	9	369934	1.27	0.2-1.8	6	381760	2.07	1.2-3.5
1h 30	15	367389	0.58	0.2-1.4	9	379890	1.57	0.7-3.0
2h	24	366994	0.47	0.2-1.1	12	378760	1.27	0.5-2.7
3h	42	365555	0.07	0.0-0.2	20	376300	0.61	0.1-1.8
4h	59	365409	0.03	0.0-0.2	28	375160	0.31	0.0-0.9
5h	79	365397	0.03	0.0-0.2	37	374530	0.14	0.0-0.5
6h	-	-	-	-	46	374130	0.03	0.0-0.1
8h	-	-	-	-	64	374080	0.02	0.0-0.1
10h	-	-	-	-	83	374070	0.02	0.0-0.1

In early iterations, the improvement heuristic exhibits significant variability from one run to the other. For instance, after 1 hour, the optimality gap across the 10 runs ranges from 1.2% and 3.5% (with the terminal and apron constraints). This is driven by the significant impact of the random selection of time windows in initial iterations. Specifically, the algorithm may visit the most “promising” time windows in initial iterations in some runs, but in later iterations in other runs. However, after many iterations, all relevant time windows get visited several times and the optimality gap gets close to zero in all runs. For instance, after 6 hours the optimally gap ranges between 0% and 0.1%. This suggests that the ultimate performance of the algorithm is only mildly sensitive to its randomness.

In summary, the constructive and improvement heuristics together provide optimal or near-optimal solutions to PSAM-RTA in a few hours of computation. As compared to direct CPLEX implementations, they yield higher-quality solutions in much faster computational times.

4.4.3 Comparison of Proposed Algorithm with Direct CPLEX Implementation

We conclude with a detailed comparison in Table 4.7 of the outputs of the proposed algorithm (i.e., the constructive heuristic with 4 groups for 22 minutes, then the improvement heuristic) with those resulting from direct CPLEX implementation of PSAM-RTA (using both the weight-based and sequential approaches), in the instance with the terminal and apron constraints.

Table 4.7 – Solutions found with the proposed algorithm and direct CPLEX implementation

CPU Time	Direct CPLEX Implementation Weight-based approach (%gap)			Direct CPLEX Implementation Sequential-based approach (%gap)			Proposed Algorithm (%gap)		
	CH	NE	OS	CH	NE	OS	CH	NE	OS
15 min	N/A	N/A	N/A	**0.0%**	**0.0%**	N/A	**0.0%**	**0.0%**	N/A
30 min	N/A	N/A	N/A	0.0%	0.0%	N/A	0.0%	0.0%	3.5%
1 hour	N/A	N/A	N/A	0.0%	0.0%	N/A	0.0%	0.0%	2.9%
2 hours	11.9%	-25.3%	10.0%	0.0%	0.0%	N/A	0.0%	0.0%	1.4%
3 hours	10.6%	-25.5%	6.4%	0.0%	0.0%	N/A	0.0%	0.0%	0.9%
6 hours	6.9%	9.7%	5.5%	0.0%	0.0%	N/A	0.0%	0.0%	0.1%
10 hours	4.1%	**0.0%**	14.8%	0.0%	0.0%	N/A	0.0%	0.0%	**0.0%**
1 day	**0.0%**	0.0%	6.9%	0.0%	0.0%	N/A	0.0%	0.0%	0.0%
2 days	0.0%	0.0%	4.9%	0.0%	0.0%	9.8%	0.0%	0.0%	0.0%
4 days	0.0%	0.0%	3.4%	0.0%	0.0%	3.0%	0.0%	0.0%	0.0%
7 days	0.0%	0.0%	2.0%	0.0%	0.0%	0.5%	0.0%	0.0%	0.0%

The takeaways from this table fall into three categories. First, the constructive heuristic provides feasible solutions faster than CPLEX. Indeed, the constructive heuristic yields an initial solution in 30 minutes, while CPLEX finds the first feasible solution after 2 hours with the weight-based approach and 1 day with the sequential approach. Second, the outputs of the constructive heuristic obtained in less than 30 minutes outperform the best ones obtained with CPLEX after 2 days. The corresponding optimally gaps are 3.5% vs 4.9%. Third, the improvement heuristic reduces the optimality gap from 3.51% to 0.1% in 6 hours. In comparison, CPLEX cannot even find a feasible solution that reaches the optimum for the change-to-historic slots and the new-entrant slots in 6 hours and finds a solution with a larger optimality gap (0.5% to 2%) in 7 days. This underscores the benefits of our algorithm, which provides faster *and* higher-quality solutions than direct CPLEX implementation.

4.5 Sensitivity Analysis to the Algorithm Parameters

We now analyse the sensitivity of the proposed algorithm to the number of groups included in the constructive heuristic and to the 7 parameters of the improvement heuristic. Our goal here is twofold. First, we aim to establish the robustness of our results by showing that optimal or near-optimal solutions can be obtained even if some parameters are not set to their "best" value. Second, we aim to validate the algorithm by showing that our design choices (e.g., the probability distributions used to select time windows and slot requests at each iteration) provide better solutions than easier implementations of large-scale neighbourhood search algorithms. Throughout this section, we vary one parameter at a time while fixing all others to their baseline values (see Table 4.3).

4.5.1 Impact of the Number of Groups in the Constructive Heuristic (G)

We know from Section 4.4 that the constructive heuristic yields higher-quality solutions with fewer groups and terminates in shorter runtimes with intermediate numbers of groups (e.g., 3-5 groups in our test setting). Here, we explore the outcome of the improvement heuristic when different starting points are considered, resulting from the output of the constructive heuristic with 3, 4, 5 and 10 groups. The results are shown in Table 4.8. As expected, the worse the initial solution, the longer the algorithm needs to converge toward a near-optimal solution. Nonetheless, near-optimal solutions are obtained in a few hours in all cases. For instance, even if we apply the constructive heuristic with 10 groups (a notoriously bad choice from Table 4.5), the optimality gap obtained with the improvement heuristic in 10 hours ranges from 0.1% to 0.3% (when, again, the best solution obtained with direct CPLEX implementation after 7 days lies within 0.5% of the optimum). These results show that the proposed algorithm can always provide near-optimal solutions even with a sub-optimal starting point from the constructive heuristic.

Table 4.8 – Sensitivity analysis to parameter *G*

Number of groups G	Initial solution	CPU Time								Avg. no. iterations
		1 hour		3 hours		6 hours		10 hours		
	Avg. gap (%)	Avg. gap (%)	Gap range (%)	Avg. gap (%)	Gap range (%)	Avg. gap (%)	Gap range (%)	Avg. gap (%)	Gap range (%)	
3	1.82	**1.57**	1.3-1.8	**0.56**	0.0-1.1	0.06	0.0-0.3	0.06	0.0-0.3	93
4	2.91	2.07	1.2-3.5	0.61	0.1-1.8	**0.03**	0.0-0.1	**0.02**	0.0-0.1	83
5	4.41	3.44	2.3-4.7	1.36	0.1-2.8	0.20	0.0-1.3	0.04	0.0-0.1	90
10	4.79	**4.01**	3.1-4.9	**1.67**	0.9-3.1	**0.33**	0.2-0.9	**0.19**	0.1-0.3	87

4.5.2 Impact of the Size of Time Window (s) and Rolling Periods (r)

We now explore the impact of the size of the time windows (s) and the rolling periods (r) in the improvement heuristic. Recall that the larger the value of s, the longer the time windows, and thus the longer the runtime needed for solving slot allocation to optimality at each iteration. At one extreme, if $s = 24$ hours, the improvement heuristic is equivalent to direct CPLEX implementation of PSAM-RTA, with prohibitively long runtimes. On the other hand, smaller values of s result in limited flexibility to swap slots across time periods. The parameter r captures the overlap between consecutive time windows. For small values of r (comparatively to s), time windows overlap significantly, and thus many time windows need to be visited before convergence. In contrast, when $r = s$, the time windows do not overlap, which may not permit to capture the interactions among slot requests across consecutive time windows.

Table 4.9 – Sensitivity analysis to parameter *s*

Size of the window *s* (hour)	Initial solution Avg. gap (%)	CPU time 1 hour Avg. gap (%)	1 hour Gap range (%)	3 hours Avg. gap (%)	3 hours Gap range (%)	6 hours Avg. gap (%)	6 hours Gap range (%)	10 hours Avg. gap (%)	10 hours Gap range (%)	Avg. no. iterations
0.5	3.51	3.12	2.5-3.4	**2.59**	2.3-3.0	**2.44**	2.3-2.7	**2.40**	2.3-2.7	190
1	3.51	2.48	2.0-3.0	1.95	1.7-2.5	1.65	1.4-2.5	1.59	1.3-2.5	111
2	3.51	**2.07**	1.2-3.5	**0.61**	0.1-1.8	**0.03**	0.0-0.1	**0.02**	0.0-0.1	83
3	3.51	2.62	1.6-3.5	1.04	0.8-1.4	0.48	0.0-1.1	0.32	0.0-1.0	66
4	3.51	2.75	1.5-3.5	1.44	0.6-3.0	0.55	0.1-2.3	0.47	0.0-2.3	64
6	3.51	**3.50**	3.5-3.5	2.54	1.6-3.4	1.11	0.3-2.5	0.72	0.0-1.4	56
8	3.51	2.72	2.7-2.7	2.44	2.0-2.7	1.70	0.8-2.1	1.17	0.1-1.7	50

First, we vary the value of s from 30 minutes to 8 hours, with $r = 0.5 \times s$. Table 4.9 and Figure 4.3 report the average optimality gap and its range, after 1, 2, 6 and 10 hours of runtime. As expected, the longer the time windows (i.e., the larger s), the more computationally intensive PSAM-RTA is at each iteration, and thus the fewer iterations are performed. We discuss next variations in solution quality.

On the one hand, small time windows do not yield the best performance in the algorithm. Indeed, when $s = 30$ minutes the algorithm yields its worst solution, and the range of the optimality gap does not change from 6 to 10 hours of runtime. Thus, the algorithm yields a solution that is far from the optimum and convergence remains weak after a large number of iterations. This underscores the value of our large-scale neighborhood search approach (as compared to alternative local search approaches based on smaller neighborhoods) to ensure sufficient flexibility to swap slots across time periods.

At the other extreme, when the longest time windows are used, the computational requirements of the model at each iteration and the random selection of slot requests can deteriorate the performance of the improvement heuristic. Indeed, for large values of s, the size of each optimization model increases, and it is more likely that optimal solutions cannot be found within t_0. As a result, only a fraction of all slot requests contained within some time windows will be selected – specified by the parameter ρ. This leads to large variations in the algorithm's performance. For instance, when $s = 8$ hours, the optimality gap ranges from 0.1% to 1.7% across the 10 runs (as opposed to 0-0.1% when

$s = 2$ hours). Therefore, for large values of s, the computational requirements of the model at each iteration and the random selection of slot requests can deteriorate the performance of the improvement heuristic.

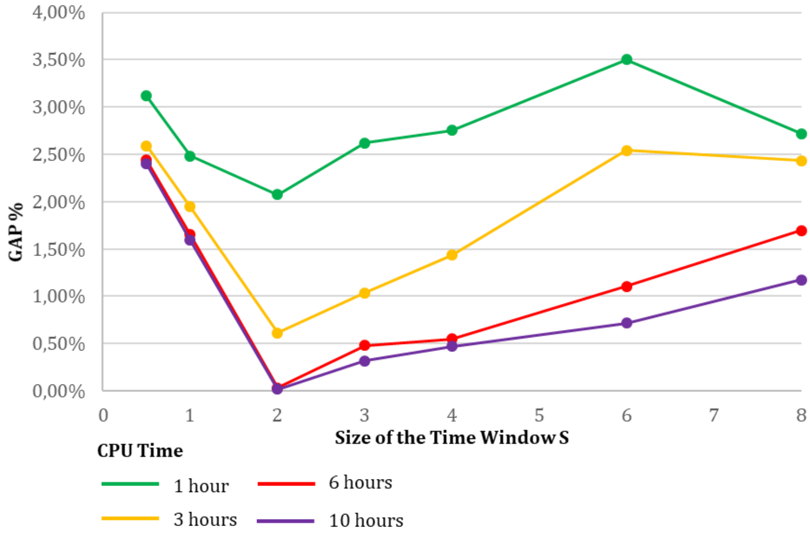


Figure 4.3 - Average gap (%) over time for different values of s

In conclusion, there exists a "sweet spot" for the size of the time windows selected at each iteration. The algorithm performs best when it captures combinatorial complexities with long enough time windows rather than relying solely on local search, while ensuring computational efficiency within each iteration with small enough neighbourhoods. In our setting, the best choice seems to be a 2-hour time window, which permits to capture connections and other interactions across slot requests without imposing high computational costs. As for the constructive heuristic, this value may vary as a function airport size, and our algorithm's modularity ensures its scalability to larger schedule-coordinated airports.

We now discuss the performance of the improvement heuristic as a function of r (relative to s). Table 4.10 shows the results with time windows of length $s = 2$ hours and ratios r/s ranging from 0.1 to 1. When r is close to s (i.e. the time windows have little overlap), the improvement heuristic does not perform best in later stages. For instance, when $r = s$, the average optimality gap is equal to 0.29% after 10 hours – the worst outcome overall. This underscores the need to capture interdependencies across time windows. At the same time, when r is too small (i.e., time windows overlap significantly), the improvement heuristic also converges slowly because too many windows need to be explored. For instance, the solution obtained with $r = 0.5 \times s$ is superior to the one obtained with $r = 0.25 \times s$ or $r = 0.1 \times s$. In addition to converging toward better solutions overall, intermediate values of r also permit to reach high-quality solutions faster, as the optimality gap is smallest with $r = 0.5 \times s$ after 1, 2, 6 and 10 hours. Again, this suggests a sweet spot in the value of r (relative to s) – in the case considered, the best results are obtained when r ranges from $0.25 \times s$ to $0.5 \times s$.

Table 4.10 – Sensitivity analysis to the ratio r/s

Ratio *r/s*	Initial solution	CPU time								Avg. no. iterations
		1 hour		3 hours		6 hours		10 hours		
	Avg. gap (%)	Avg. gap (%)	Gap range (%)	Avg. gap (%)	Gap range (%)	Avg. gap (%)	Gap range (%)	Avg. gap (%)	Gap range (%)	
0.1	3.51	2.58	1.9-3.5	1.17	0.3-2.6	0.42	0.0-0.9	0.13	0.0-0.3	62
0.25	3.51	2.50	0.9-3.4	**1.43**	0.6-2.8	**0.48**	0.0-2.1	0.09	0.0-0.3	76
0.5	3.51	**2.07**	1.2-3-5	**0.61**	0.1-2.8	**0.03**	0.0-0.1	**0.02**	0.0-0.1	83
0.75	3.51	**3.10**	2.5-3.5	1.35	0.5-2.5	0.28	0.1-0.9	0.24	0.1-0.9	81
1	3.51	2.09	1.4-3.3	0.40	0.3-0.8	0.32	0.2-0.4	**0.29**	0.2-0.4	97

4.5.3 Impact of Optimization Runtime (t_o)

We now vary the maximum optimization runtime at each iteration t_o. The larger t_o, the closer to optimal the allocation of slots in each time window, but the longer the time spent on some iterations (specifically, on iterations where slot allocation within the time windows selected is most computationally intensive),.

Table 4.11 – Sensitivity analysis to parameter t_o

Optimization runtime *to*	Initial solution	CPU Time								Avg. no. iterations
		1 hour		3 hour		6 hour		10 hour		
	Avg. gap (%)	Avg. gap (%)	Gap range (%)	Avg. gap (%)	Gap range (%)	Avg. gap (%)	Gap range (%)	Avg. gap (%)	Gap range (%)	
2	3.51	2.82	1.3-3.5	**2.01**	1.2-3.0	**1.57**	1.0-3.0	**1.42**	0.9-1.9	150
5	3.51	2.72	1.8-3.5	0.92	0.3-1.5	0.32	0.0-0.8	0.16	0.0-0.3	102
10	3.51	**2.07**	1.2-3.5	**0.61**	0.1-1.8	**0.03**	0.0-0.1	0.02	0.0-0.1	83
15	3.51	2.30	1.0-3.5	0.97	0.1-2.1	0.11	0.0-0.6	**0.01**	0.0-0.0	69
30	3.51	2.56	1.0-3.5	1.48	0.4-2.2	0.68	0.0-2.1	0.01	0.0-0.0	43
60	3.51	**3.51**	3.5-3.5	1.69	0.6-2.8	0.91	0.0-2.1	0.25	0.0-1.6	31

Table 4.11 and Figure 4.4 report results of the improvement heuristic with values of t_o ranging from 2 minutes to 1 hour. For the low values of t_o, the improvement heuristic stops before significant improvements are obtained, leading to bad convergence solutions. For instance, for $t_o = 2$ minutes the average optimality gap found after 10 hours of computation exceeds 1%. On the other hand, if t_o

is set too high, the improvement heuristic spends much time searching for local perturbations within time windows for which PSAM-RTA is computationally intensive. This ultimately slows down convergence. For instance, it takes a full 10 hours to reach a close-to-optimal solution when $t_o = 30$ minutes (as opposed to 6 hours when $t_o = 10$ minutes or $t_o = 15$ minutes).

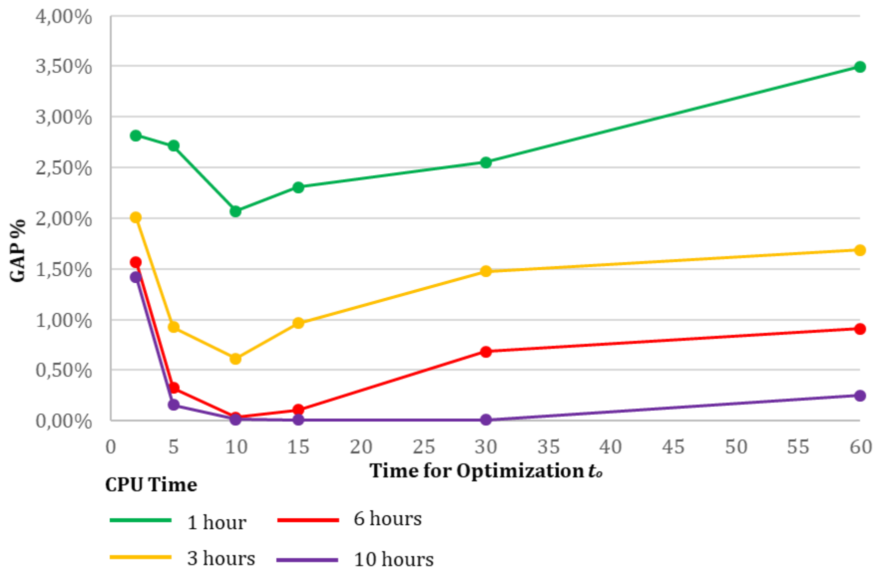


Figure 4.4 - Average gap (%) over time for different values of t_o

4.5.4 Impact of Calibration Parameters ($\rho, \beta, \delta, \lambda$)

Finally, we test the impact of ρ, β, δ and λ, on the performance of the improvement heuristic. These are all calibration parameters reflecting design choices in the development of the improvement heuristic. In this section, we establish the robustness of the results with respect to these parameters. In other words, even if these parameters are not set to their "best" value, the improvement heuristic still exhibits strong convergence properties. In addition, we show that the improvement heuristic performs better than more naive algorithmic designs that correspond to values of $\rho = 1$, $\beta = 0$, $\delta = 0$ and $\lambda = 0$. Ultimately, the results reported in this section validate the design of the improvement heuristic.

a) Parameter ρ

The parameter $\rho \in (0,1]$ specifies by how much the number of selected slot requests is reduced for time windows for which PSAM-RTA was not solved to optimality in previous iterations. A value of $\rho = 1$, captures the simple setting where all slot requests in $\boldsymbol{S_w}$ are selected in all iterations, regardless of prior computational experience.

Table 4.12 – Sensitivity analysis with respect to parameter ρ

Value of ρ	Initial solution	CPU time								Avg. no. iterations
		1 hour		3 hour		6 hour		10 hour		
	Avg. gap (%)	Avg. gap (%)	Gap range (%)	Avg. gap (%)	Gap range (%)	Avg. gap (%)	Gap range (%)	Avg. gap (%)	Gap range (%)	
0.3	3.51	2.08	1.0-3.5	0.68	0.3-1.6	**0.29**	0.0-0.4	**0.19**	0.0-0.3	116
0.5	3.51	**2.07**	0.5-3.5	0.84	0.1-1.9	0.09	0.0-0.2	0.08	0.0-0.2	104
0.7	3.51	2.26	1.3-3.5	0.73	0.1-2.3	0.07	0.0-0.2	0.03	0.0-0.2	91
0.8	3.51	2.07	1.2-3.5	**0.61**	0.1-1.8	**0.03**	0.0-0.1	**0.02**	0.0-0.1	83
0.9	3.51	2.18	1.4-3.5	1.06	0.0-2.0	0.06	0.0-0.4	0.02	0.0-0.1	71
1	3.51	**2.35**	1.4-3.5	**1.21**	0.4-1.9	**0.29**	0.1-0.4	**0.19**	0.0-0.3	59

The results are reported in Table 4.12, with values of ρ ranging from 0.3 to 1. First, note that when $\rho = 1$ the solution of the improvement heuristic is inferior to the one obtained with $\rho = 0.8$, at every stage of the algorithm. For instance, the average optimality gap after 10 hours of computation is 0.19% when $\rho = 1$, but only 0.02% when $\rho = 0.8$. This underscores the benefits of selecting a subset of slot requests in time windows for which PSAM-RTA is computationally intensive. At the opposite end, the smallest values of ρ do not provide the strongest performance in later iterations. For instance, the solution obtained with $\rho = 0.3$ is close to the best available one after 1 hour and 3 hours of computation, but it deteriorates (relative to larger values of ρ) after 6 and 10 hours of computation. This stems from the fact that, when the number of slot requests selected in each iteration is too small, the algorithm needs to visit each time window many times, thus creating a long-tail effect in the algorithm.

Overall, values of ρ between 0.7 to 0.9 ensure the best performance of the algorithm. This underscores the value of our choice of reducing the number of slot requests selected at each iteration where PSAM-RTA is computationally intensive in the selected time window.

b) Parameters β and δ

The parameters β and δ calibrate the probability distribution PD_w used to select the time window at each iteration. As discussed in Section 4.3, β is set such that $e^{-\beta} \approx 0$ to ensure that the algorithm does not return to time windows for which PSAM-RTA was solved to optimality in previous iterations. We choose a value of β=5. Next, $\delta \in [0,1]$ captures the decrease in the probability of selecting a time window where no improvement was obtained in previous iterations. If $\delta = 0$, the selection of time windows is cyclical and deterministic; if $\delta = 1$, then the algorithm will focus on the time windows where PSAM-RTA is most computational intensive.

Table 4.13 shows the results when δ varies from 0.3 to 1. It also reports an additional case (No PD_w), where the time windows are selected completely at random (i.e., $\beta = 0$ and $\delta = 0$). Note, first, that No PD_w yields the worst results after 6 and 10 hours. Its average optimality gap after 10 hours is 0.26%, which is significantly larger than the second worst one of 0.06%. This underscores the benefits of our approach for orienting the search toward time windows where improvements are most likely. Second, the average optimality gap is fairly large with the largest values of δ, especially in early stages of the heuristic. For instance, after 6 hours, the average optimality gap when $\delta = 1$ is 0.31%, (as opposed to 0.03% with $\delta = 0.8$). This suggest that when δ is set too large, the algorithm focuses almost exclusively on the time windows for which slot allocation was not solved to optimality in previous iterations, thus ignoring the other ones and ultimately slowing down convergence. In other words, "low-hanging fruits" can be obtained in time windows for which PSAM-RTA can be solved easily, and such improvements are left to later stages of the algorithm with larger values of δ. Third, small values of δ seem to be beneficial in early stages of the algorithm, as it ensures that the algorithm visits rapidly all the time windows. However, this approach is not most effective in later stages, as the algorithm then keeps searching for improvements in time windows that were already fully explored. For instance, the average optimality gap after 3 hours is lower with $\delta = 0.5$ than with $\delta = 0.8$, but the reverse is true after 6 hours.

Table 4.13 - Sensitivity analysis with respect to parameter δ

Value of δ	Initial solution	CPU Time								Avg. no. iterations
		1 hour		3 hours		6 hours		10 hours		
	Avg. gap (%)	Avg. gap (%)	Gap range (%)	Avg. gap (%)	Gap range (%)	Avg. gap (%)	Gap range (%)	Avg. gap (%)	Gap range (%)	
0.3	3.51	2.23	1.2-3.5	0.66	0.1-1.4	0.13	0.0-0.5	0.06	0.0-0.2	94
0.5	3.51	2.43	1.3-3.5	0.44	0.1-1.5	0.11	0.0-0.4	0.06	0.0.-0.3	83
0.7	3.51	**2.49**	1.6-3.5	0.58	0.0-2.0	0.09	0.0-0.4	0.06	0.0-0.3	88
0.8	3.51	**2.07**	1.2-3.5	**0.61**	0.1-2.8	**0.03**	0.0-0.1	**0.02**	0.0-0.1	83
0.9	3.51	2.38	1.3-3.5	**1.00**	0.0-2.5	0.12	0.0-0.4	0.05	0.0-0.3	77
1	3.51	2.33	1.3-3.5	0.92	0.0-1.9	0.31	0.0-0.7	0.04	0.0-0.1	84
No PD_w	3.51	2.12	0.4-3.0	0.76	0.2-1.3	**0.45**	0.1-1.3	**0.26**	0.0-0.4	116

Overall, these results validate our probabilistic approach to the selection of each time window. Moreover, the choice of the parameter δ can improve the performance of the algorithm, thus validating our history-based approach to the calibration of the underlying probability distribution. Values of δ between 0.5 and 0.8 seem to ensure the strongest performance of the algorithm.

c) Parameter λ

Last, the parameter $\lambda \geq 0$ is used to update the probability of selecting a slot request $i \in \boldsymbol{S_w}$ when a subset of slot requests in $\boldsymbol{S_w}$ are selected (as specified by the parameter ρ). The larger the λ, the higher the probability assigned to slots that were selected fewer times in previous iterations. Table 4.14 shows results of the improvement heuristic when λ varies from 0 to 5. The first observation is that the algorithm converges to near-optimal solutions regardless of the value of λ, with an average optimality gap ranging from 0.02% to 0.04% across all cases. Moreover, the worst results are obtained when $\lambda = 0$, which corresponds to the case where slot requests are selected completely at random within the time window selected in each iteration. This underscores the benefits of calibrating the probability distribution PS_i to prioritize slot requests that were selected fewer times in earlier iterations. In our case, we derive the best computational performance when λ is set between 0.5 and 1. This parameter is most useful at the busiest airports where solving PSAM-RTA is computationally intensive for more time windows.

Table 4.14 - Sensitivity analysis with respect to parameter λ

Value of λ	Initial solution	CPU time								Avg. no. iterations
		1 hour		3 hours		6 hours		10 hours		
	Avg. GAP (%)	Avg. GAP (%)	Range GAP (%)	Avg. GAP (%)	Range GAP (%)	Avg. GAP (%)	Range GAP (%)	Avg. GAP (%)	Range GAP (%)	
0	3.51	**2.25**	1.2-3.5	**0.91**	0.0-1.8	**0.16**	0.0-0.6	**0.04**	0.0-0.2	84
0.25	3.51	2.17	0.9-3.5	0.76	0.1-2.1	0.08	0.0-0.3	0.04	0.0-0.1	81
0.5	3.51	2.18	1.0-3.5	0.79	0.0-1.7	0.05	0.0-0.4	**0.02**	0.0-0.1	82
1	3.51	**2.07**	1.2-3.5	**0.61**	0.1-1.8	**0.03**	0.0-0.1	**0.02**	0.0-0.1	83
2	3.51	2.12	1.2-3.5	0.59	0.0-1.6	0.09	0.0-0.5	0.03	0.0-0.2	84
5	3.51	2.18	1.2-3.5	0.66	0.0-2.5	0.10	0.0-0.8	0.03	0.0-0.2	85

4.5.5 Discussion

The major takeaways from this section fall into three categories. First, several parameters involved in the design of the algorithm can impact the performance of the algorithm and the quality of the ultimate solution. There seems to exist a "sweet spot" for each parameter that will lead to the best algorithmic performance. Second, the proposed algorithm shows robustness, as solutions within 0.1% of the optimum are generally found in a few hours when calibration parameters evolve within a reasonable range around their "best" values. Specifically, the algorithm's parameters mainly impact its convergence speed rather than the quality of the ultimate solution. Third, results validate a number of algorithmic choices made in Section 4.3. Most notably, we found that the large-scale neighborhood search algorithm is most beneficial, as compared to alternative algorithms based on local search (captured here with shorter time windows). In addition, our probabilistic selection of time windows and slot requests at each iteration results in significant performance improvements, as compared to more straightforward applications of large-scale neighborhood search algorithms.

4.6 Conclusion

The vast majority of busy airports outside the United States are subject to schedule coordination. At these airports, flight schedules are governed by slot allocation, which follows the Worldwide Slot Guidelines from the International Air Transport Association (IATA) and carries enormous weight for airlines, airports and passengers. This chapter proposes a new modeling and computational framework to optimize slot allocation at schedule-coordinated airports. It formulates a Priority-based Slot Allocation Model with Runway, Terminal and Apron constraints (PSAM-RTA) that is fully compliant with the rules and priorities established by the Worldwide Slot Guidelines. This model can be solved with commercial solvers at medium-size airports (up to 100,000 flight movements per annum), but its implementation at the largest schedule-coordinated airports worldwide remains intractable. To address this challenge, this chapter develops an original heuristic algorithm based on large-scale neighborhood search to solve PSAM-RTA at the busiest airports – with 200,000 or more flight movements per annum.

Computational results using real-world data from the Lisbon Airport suggest that optimal or near-optimal solutions to PSAM-RTA can be obtained in reasonable runtimes. Specifically, while direct implementation of the model with commercial solvers yields a solution within 5-10% of the optimum after 2 days and within 0.5-2% of the optimum after 7 days, the proposed algorithm provides a solution within 2-5% of the optimum after 30 minutes and within 0-0.03% of the optimum after 10 hours. Extensive sensitivity analyses also showed that the algorithm performs better than more straightforward implementations of large-scale neighborhood search methods in this context, and that results are robust to a number of calibration parameters. Ultimately, this chapter augments the capabilities of slot allocation models and algorithms. Its application in support of slot allocation at major schedule-coordinated airports worldwide can result in flight schedules that match airlines' slot requests and passenger demand more effectively than existing approaches based on specialized software and *ad hoc* allocation decisions.

The positive results reported in this chapter motivate further research on airport slot allocation. First, PSAM-RTA has been applied to the Lisbon airport in this chapter but, due to a lack of data availability, it has not been applied at some of the largest schedule-coordinated airports in the world (e.g., Paris Charles de Gaulle, Amsterdam Schiphol). Testing the scalability of the proposed model and algorithm at these airports represents an important avenue for future research. Second, the constructive and improvement heuristics proposed in this chapter are approximate algorithms by nature. Important opportunities exist to augment these methods with exact algorithms. Third, PSAM-RTA has focused on the primary criteria of slot allocation thus far; in practice, slot allocation is also governed by secondary criteria, mainly employed for tie-breaking purposes. The model and algorithm proposed in this chapter could thus be enhanced to capture such additional criteria, such as minimizing passenger displacement, maximizing the number of markets served, and maximizing airport connectivity. Finally, PSAM-RTA could be extended from a single-airport setting to a network setting to solve slot allocation for the full season of operations at multiple airports simultaneously. Obviously, the resulting computational complexity would be very significant, but the modeling and computational framework developed in this chapter provides a methodological foundation to tackle this problem.

5 Improving Slot Allocation at Level 3 Airports

5.1 Introduction

According to current worldwide practice, airports offering commercial passenger service are subdivided into three "levels" (IATA, 2017). Level 1 airports are those "where the capacities of all infrastructure at the airport are generally adequate to meet the demands of users at all times". Level 2 (or "facilitated") airports are those where "there is potential for congestion during some periods of the day, week, or season which can be resolved by schedule adjustments mutually agreed between the airlines and a facilitator". And Level 3 (or "schedule coordinated") airports are those where "it is necessary for all airlines and other aircraft operators to *have a slot allocated by a coordinator in order to arrive or depart at the airport* during the periods when slot allocation occurs" [italics added]. Airports at the three levels can thus be characterized as "uncongested", "mildly congested", and "congested", respectively. The focus of this chapter is on Level 3 airports.

For the Summer season of 2017, 177 airports in the world were designated as Level 3. Of those, 37 were in the Asia/Pacific region, 103 in Europe, 10 in the Middle East and Africa, 13 in North Asia and 14 in the Americas. Despite their small number (only about 4.5% of the roughly 4,000 airports in the world with scheduled airline service), Level 3 airports play a truly critical role in global air transport. In 2016, they served approximately 3.15 billion airport passengers – or about 43% of the worldwide total of 7.4 billion and about 55% of the roughly 5.75 billion passengers outside the United States.[1] A full 70% of all airport passengers in Europe used Level 3 airports. Of the 30 busiest airports, in terms of passengers, outside the US, 29 were Level 3.[2] And, perhaps most important, practically all the major connecting hubs outside the US are Level 3 airports. Finally, the number of Level 3 airports worldwide will increase by 15%, from 177 to 203, for the Summer seasons of 2017

[1] As will be discussed later, the United States avoids designating airports as Level 3, to minimize the extent of schedule coordination. As of May 2017, the only one Level 3 airport in the US was JFK International, despite the fact that, by any standard, many airports in the US would be characterized as congested.

[2] The only exception, surprisingly, was Jakarta's Soekarno-Hatta, a notoriously congested airport, which was designated as Level 2.

and 2018 and by 10%, from 161 and 177, for the respective Winter seasons, suggesting their steadily growing influence over time.

For all these reasons, the process and rules under which access to Level 3 airports is determined carry enormous economic and regulatory implications for the global air transport sector. More generally, the issue of how to best allocate scarce capacity among airlines at congested airports has attracted much attention from academia and industry over the years,[3] beginning as far back as the late 1960s, and generated much controversy.

Slot allocation at Level 3 airports was indeed one of the topics addressed by the Economic Commission of the International Civil Aviation Organization (ICAO) during ICAO's 39th Assembly in Fall 2016. In its report at the end of the Assembly, the Commission welcomed the joint statement made by the International Air Transport Association (IATA) and the Airports Council International (ACI), "which recognized the need to optimize the use of scarce capacity, particularly at capacity constrained airports" (ICAO, 2016) and agreed to conduct a detailed review of the slot allocation process. The Commission further noted that "ACI and IATA would work with States and the industry stakeholders as partners and would report progress to the next session of the Assembly" in 2019.

This development has opened a 'window of opportunity' for improving existing slot allocation practices. With minor regional (e.g., in the European Union) or national (notably in the United States) variations, these practices follow closely and on a *global* scale the process described by IATA's Worldwide Slot Guidelines (WSG henceforth) (IATA, 2017). This process and its rules are of a purely administrative nature at this time. In a Working Paper submitted in advance of the 39th Assembly to ICAO's Economic Commission, IATA states that it "would oppose any consideration of market-based primary slot allocation mechanisms; these have been analyzed on many occasions in the past, by multiple independent academic and expert organizations, with no clear indications that such mechanisms improve the utilization of already-congested airport capacity or provide benefits to improving customer experience and choice in connectivity and fares" (IATA, 2016b). While many would disagree with this statement, it is certainly true that airlines, through their representative bodies, have strongly opposed over the years the use of congestion pricing, slot auctions, or other such market-based mechanisms for capacity allocation purposes. In view of this opposition, it is unlikely

[3] For reviews, see Chapter 12 of de Neufville and Odoni (2013) and, especially, a volume dedicated to the subject (Czerny et al, 2008).

that the use of a purely administrative approach will be abandoned in the short- and medium-terms. IATA's position in this respect is summarized in a statement on its website that refers to the ongoing ICAO-endorsed review:[4] "The WSG Strategic Review is the ongoing process of enhancing the existing WSG, not rewriting from scratch, to ensure it remains the global, single slot standard for years to come – a major undertaking for 2017/18".

The objective of this chapter is to contribute to the ongoing quest for improvements to the slot allocation process that could be realistically implemented in the short term. Given the political realities just described, the chapter's scope will necessarily be limited to ways of enhancing significantly the efficiency, effectiveness and outcomes of certain key steps in the purely administrative process prescribed by the WSG. To this purpose, Section 5.2 will first provide a summary of the process, followed by brief descriptions of the capacity constraints that airports specify, the slot requests that airlines submit, and the decision-making rules that are used to allocate slots and develop the flight schedules at each Level 3 airport. Section 5.3 will then present a description of Priority-based Slot Allocation Model (PSAM), a state-of-the-art optimization model we have developed for allocating slots to airlines, and will illustrate through an example the types of insights that the model can offer. It will be seen that optimization models not only can identify slot allocations that are more compliant with airline scheduling preferences, but may also suggest mild modifications to the existing WSG rules that have the potential for reducing significantly the negative impact of airport capacity limitations on these scheduling preferences. This motivates an investigation, made possible by the use of PSAM, of the impacts of several such modifications to the WSG rules. The results are reported in Section 5.4, the lengthiest of the chapter. Section 5.5 discusses two other areas where changes to existing practices may prove beneficial: (a) adding specificity to what qualifies an airport for designation as Level 3, as well as possibly refining the class of Level 3 airports by breaking it down into two more homogeneous subclasses; and (b) resolving potential network-level conflicts between slot allocations made separately at each individual airport. Section 5.6 provides a more general context for this chapter by discussing briefly a number of other issues, some of them of a fundamental nature, associated with the existing slot allocation process.

[4] http://www.iata.org/pressroom/facts_figures/fact_sheets/Documents/fact-sheet-airport-slots.pdf, accessed on August 16, 2017.

Finally, in Section 5.7, we summarize the main contributions of our chapter and indicate directions for future research.

5.2 The Slot Allocation Process

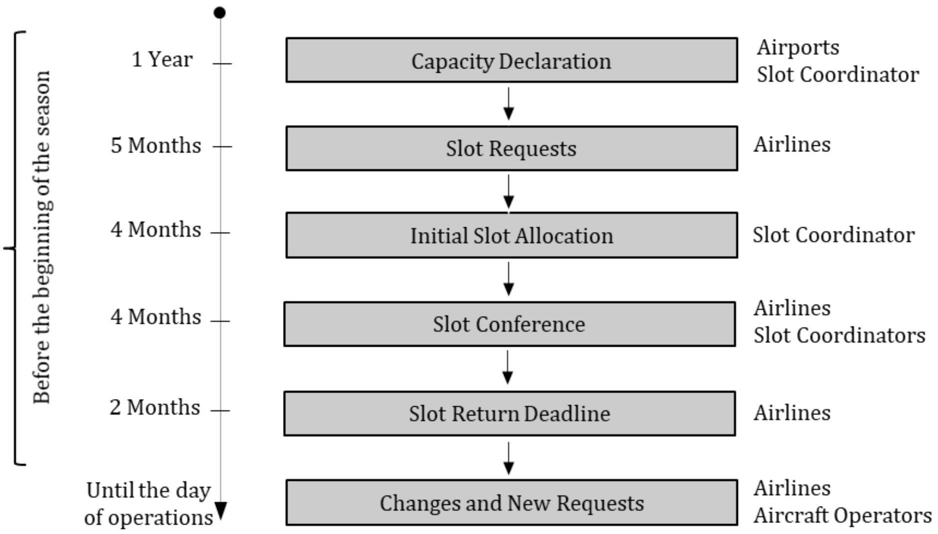


Figure 5.1 - Outline of the slot allocation process, as performed bi-annually at Level 3 airports.

Figure 5.1 summarizes the main steps in the slot allocation process[5] and indicates the associated timeline and the responsible entities (Kösters, 2007). The process is carried out bi-annually, for the "Summer" and "Winter" seasons, to provide airlines with access to Level 3 airports in the form of a landing or takeoff slot. A slot is defined as "the permission to use the full range of an airport's infrastructure to perform aircraft arrivals or departures on a specific day and at a specific time" (IATA, 2017). First, each airport provides its "declared capacity", specifying the number of arrival and departure slots made available in each time interval of a day. Second, the airlines submit their desired schedule of flights at each airport to the slot coordinator for the upcoming season. Third, the coordinator performs the initial slot allocation in an "unbiased, transparent and non-discriminatory" way and presents the results to the airlines. Fourth, adjustments are made during the Slot Conferences about four months before the start of each season, which are attended by airline representatives, slot coordinators, airport representatives and other interested parties. These adjustments involve primarily the resolution of conflicts stemming from the timing of slots allocated across multiple airports, and, if relevant, disputes among airlines competing for the same slots. Last, the airlines may "return" slots to the coordinator until two months before the start of each season, if they decide that they will not

[5] The slot allocation process is often more formally referred to as the "schedule coordination process".

use them. They can also request last-minute adjustments and carry them out, if approved by the coordinator, up to the day of operations.

This chapter focuses mainly on the third step, i.e., the initial slot allocation. This is a critical step, as it is the primary determinant of the final scheduling outcome. We first discuss the specification of declared capacities and the form of airline slot requests, which define the inputs of initial slot allocation, and then present the rules and procedures underlying initial slot allocation in more detail.

5.2.1 Capacity Constraints

The declared capacity places an upper limit on the number of slots to be allocated to the airlines and other eligible operators at an airport in each time interval of a day. Ideally, declared capacities should be equal to the throughput the airport can achieve per hour (or any other unit of time). However, this is often not feasible due to the significant variability of airport throughput, which is driven by such factors as meteorological conditions, airport operating procedures, the mix of arrivals vs. departures, the mix of aircraft types, etc. While some of these factors are predetermined by the *schedule* of flights, others pertain to airport *operating conditions* at a given time and can only be described probabilistically at the time of slot allocation. It is therefore impossible to set a schedule of flights that will match the airport's throughput capabilities exactly and with certainty. For this reason, the declared capacities are used by the airport to balance supply-side capabilities and airline demand during the schedule coordination process, with the objectives of maximizing their utilization and their responsiveness to airline requests, while maintaining an adequate level of service (e.g., acceptable delay levels).

In the simplest (and still most common) case, declared capacities take the form of a limit on the total number of aircraft movements (landings and takeoffs) that may be scheduled per hour, e.g., "up to 24 movements per hour". However, a growing number of the busiest Level 3 airports now employ ever-finer levels of granularity. The example of Lisbon Airport in 2014 and in 2015 is shown in Table 5.1. Note, first, that separate capacities are specified for the runway system, the apron and the two terminals of the airport. Second, runway capacities are specified for each of four different time intervals in 2014 (15, 30, 60 and 180 minutes) and for each of two different time intervals in 2015 (15 and 60 minutes). Third, the limits may be broken down further into limits on total number of movements, number of arrivals and number of departures, and into limits on the number of arriving, departing, Schengen,[6] and non-Schengen passengers in the case of the terminal buildings. Finally, the runway capacity limits (as at several other Level 3 airports) are treated as *5-minute rolling horizon limits.* For instance, for 2014, no more than 38 total movements, 26 arrivals and 26 departures may be scheduled between 10:00 and 11:00, between 10:05 and 11:05, between 10:10 and 11:10, etc., no more than 12, 10 and 10, respectively, between 10:00 and 10:15, between 10:05 and 10:20, etc., and similarly for the 30-minute and 180-minute limits. Finally, as in the case of Lisbon 2015, capacity limits may vary by time of day, e.g., "up to 40 movements between 8:00 and 9:00, up to 34 movements between 9:00 and 10:00, etc.". This can be used to specify lower limits during nighttime due to noise-related restrictions. Moreover, if the number of slot requests is much higher in certain periods of the day than others, the declared capacity may also be set higher during the peak demand periods to reflect airline preferences.

Table 5.1 - Declared capacities for Lisbon Airport.

Declared Capacity Indicators		Lisbon Airport 2014	Lisbon Airport 2015								
			00:00 06:59	07:00 08:59	09:00 09:59	10:00 13:59	14:00 14:59	15:00 17:59	18:00 19:59	20:00 20:59	21:00 23:59
Flight Movements / hour		38	38	40	34	38	39	38	40	34	38
Arrivals and Departures / hour		26	26	26	23	26	26	26	26	23	26
Flight Movements / 15 min		12	12	12	10	12	12	12	12	10	12
Arrivals and Departures / 15 min		10	10	10	9	10	10	10	10	9	10
Flight Movements / 30 min		21	-								
Arrivals and Departures / 30 min		19	-								
Flight Movements / 180 min		113	-								
Arrivals and Departures / 180 min		77	-								
Terminal 1 Arrivals / hour	Schengen	2500	2500								
	Non- Schengen	1500	2000								
Terminal 1 Departures / hour	Schengen	2300	2300								
	Non- Schengen	1500	1500								
Terminal 2 Departures / hour	Schengen	600	600								
	Non- Schengen	300	450								
Apron / Nr. aircraft		63	63								

Declared capacities may thus include such complications as: time intervals of different length (e.g., of 15 and 30 and 60, etc., minutes); capacities applied over rolling time windows; and constraints that apply to different elements of the airport (e.g., runways, apron, terminals) and are expressed in terms

[6] Passengers arriving from or departing for the "Schengen Area" (26 European nations) are not subject to passport and customs controls, i.e., are essentially treated as domestic passengers.

of different units (i.e., limits on the number of movements or of aircraft or of various types of passengers). Only a small number of Level 3 airports now have a set of coordination parameters as extensive and complex as Lisbon's, but the number of such airports is growing. For this reason, the optimization model to be described in Section 5.3.1 must be able to accommodate such complex constraints, if they exist, as well as consider constraints that vary by time of day.

5.2.2 Airline Slot Requests and the Rules of Slot Allocation

In the second step of the slot allocation process shown in Figure 5.1, the airlines submit their slot requests to the slot coordinators in the format specified in Chapter 6 of the IATA Standard Schedules Information Manual (IATA, 2014). Table 5.2 shows a sample of five slot requests at the airport of Madeira, Portugal. Each row of Table 5.2 contains a request by an airline for one or more *slot series* for the Summer 2014 season (March 30 – October 25, 210 days). A slot series consists of "at least 5 slots requested for the same time on the same day-of-the-week, distributed regularly in the same season" (IATA, 2017). Requests for fewer than 5 slots are not eligible for consideration during the initial slot allocation step; they may be considered after the slot return deadline (Step 5 in Figure 1) should any slots remain available at that time.

Table 5.2 - An example of slot requests.[7]

Req. no. (1)	Priority (2)	Arr. ID (3)	Dep. ID (4)	Start Date (5)	End Date (6)	Days of week (7)	Seats (8)	Aircraft (9)
1	F	XY001	XY002	30MAR	25OCT	1000000	180	320
2	CR	XY003	XY004	30MAR	31MAY	1234567	180	320
3	CL	XY005	XY006	01APR	21OCT	1030507	180	320
4	B	XY007	---	01JUL	23SEP	0200500	170	320
5	N	---	XY008	30MAR	25OCT	0004000	130	319

Req. no. (1)	Origin (10)	Previous Stop (11)	Arr. Time (12)	Dep. Time (13)	Next Stop (14)	Destination (15)	Arr. Type (16)	Dep. Type (17)
1	LIS	LIS	0800	0830	LIS	LIS	J	J
2	OPO	LIS	1000	1100	LIS	OPO	J	J
3	ORY	OPO	1535	1610	OPO	ORY	J	J
4	OPO	OPO	1830	---	---	---	C	---
5	---	---	---	1100	PDL	PDL	---	J

[7] Table 5.2 is the same as Table 3.2 presented in Chapter 3. We repeat this table to facilitate the reading of this section.

For an example, Request 1 characterizes a request for a series of flights that will take place on Sundays only (as indicated by the “1000000” entry in Column 7) during the entire season, i.e., for the 30 Sundays between March 30 (Column 5) and October 25 (Column 6). Note that this particular request is actually for a *pair* of slots on each Sunday, that is, for an arrival at 08:00 (Column 12) and a departure, by the same aircraft, at 08:30 (Column 13), i.e., for a total of 60 individual slots. The request also specifies the type of aircraft involved and the number of seats in it (Columns 9 and 8), as well as the airport from which the aircraft will begin its itinerary of the day (Column 10), the airports visited immediately prior to and after Madeira (Columns 11 and 14) and the airport where the itinerary will be completed at the end of the day (Column 15). Finally, it also indicates (Columns 16 and 17) whether the request is for a scheduled passenger flight (“J”), a chartered flight (“C”), etc.

In contrast to Request 1, Requests 2, 3 and 4 in Table 5.2 comprise more than a single series. Request 2, for example, comprises 7 different series of slots, one for each of the seven days of the week, all requested at the same time of the day – arrival at 10:00 and departure at 11:00. In addition, an arrival and a departure pair may be requested in the same slot request (as in Requests 1, 2 and 3), or a request may be solely for arrivals (Request 4) or solely for departures (Request 5). The requests that are made specifically for one type of movement typically come from bigger airlines, which derive operating flexibility from the large number of aircraft they may be operating at the airport. The optimization model of slot allocation must be able to accommodate these various specificities of slot requests.

5.2.3 Initial Slot Allocation Rules and Request Priorities

We now summarize the rules and priorities for slot allocation at Level 3 airports (Step 3). The first two of the rules below are often referred to as the *schedule regularity constraints*:

(i) All slots belonging to the same series (i.e., slots for the same flight on the same day of the week, at least five times over the season, such as Request 1 in Table 5.2) must be given the same time of the day.[8]

(ii) It is *recommended* that, unless the airline requests otherwise, identical series of slots for different days of the week which are submitted together as part of the same request (e.g., as in

8 IATA’s WSG states that, “if that is not possible, [all the slots in a series should be allocated] at approximately the same time” (IATA, 2017, §1.7.2.e). In practice, however, allocation to the *exact* same time is typically enforced. Our optimization model also treats this as a requirement.

Request 2 of Table 5.2) should be given slots at the same time of the day across multiple days of the week.

(iii) The requested turnaround times between the arrival and departure pair of slots assigned to the same aircraft (e.g., 30 minutes in the case of Request 1) must be maintained (or, at worst, adjusted with minimal changes) to avoid any increases in ground time and dilution of the connectivity of an airline's networks of flights.

(iv) Slots must be allocated in accordance with a set of priorities specified in the WSG as "primary criteria" for allocation. If necessary, "additional criteria" may also be used, usually for tie-breaking purposes, in cases when two slot series are equally eligible for assignment to a particular slot. The primary criteria are described in more detail in the remainder of this section.

Requests for slot series are classified into four main classes: requests for *historic* slots (abbreviated as 'H' henceforth), *change-to-historic* slots ('CH'), *new entrant* slots ('NE') and *'other'* slots that do not belong to any of the aforementioned three priority classes ('O'). This is shown in Column 2 of Table 5.2: Request 1 is for a H slot series (Code F), Requests 2 and 3 for CH slot series (Codes CR and CL), Request 4 for a NE series (Code B), and Request 5 for an 'O' slot series (Code N).

Top priority is accorded to requests for H slots. Under the rules of the WSG, "an airline is entitled to retain a series of slots on the basis of historic precedence" as long as it satisfies the slot usage requirement (IATA, 2017). These are called the "*grandfather rule"* and the *"use-it-or-lose-it rule*": if an airline operated any particular slot series during the previous Summer (resp., Winter) season for at least 80% of the time during which the series was authorized, then that airline is entitled to operate the same slot series during the next Summer (resp., Winter) season. For instance, the airline responsible for Request 1 in Table 5.2 would be entitled to this "historic" slot series in Summer 2014, if it had operated the same slot series on at least 24 of the 30 Sundays in Summer 2013. Second priority is accorded to CH requests. This corresponds to airlines holding H series of slots, but requesting a change in the time of the H slot series,[9] or in other attributes of the H series, such as a different aircraft type, route or type of service.

[9] The difference between 'CR' and 'CL' codes (Table 5.2) is that, when an airline submits a CR code, it is willing to accept any time between the (new) requested time and the historic time. By contrast, when an airline submits a CL code, it is only willing to accept the (new) requested time, if available, or the historic time.

After slots have been allocated to H and CH slot series, the remaining slots, if any, constitute the *slot pool* that will be allocated to NE slot series and to O slot series, in that order. According to the WSG, an airline qualifies for designation as a *new entrant (NE) airline* at an airport A, if it is "requesting a series of slots [at A] on any day when, if the airline's request were accepted, it would hold fewer than 5 slots [at A] on that day".[10] According to the WSG, 50% of the slots in the slot pool must be allocated to new entrants at the initial slot allocation period,[11] unless the total number of slots requested by new entrants is less than 50% of the slots in the pool. Any remaining slots after the allocation to new entrants are finally made available to the 'other' slot series requests. At the end of the initial slot allocation process, either all the slot requests will have been accommodated (albeit possibly not at the time requested by the airline) or some requests will have been rejected outright because total demand for slots exceeded the airport's declared capacity.

5.2.4 Case Study Data

We shall provide several examples based on data from three Level 3 airports in Portugal, those of Madeira, Porto and Lisbon. These are airports of very different size: in 2017, Lisbon served close to 27 million passengers, Porto close to 11 million and Madeira 3.2 million. We use slot request and allocation data for the Summer season of 2014 in Madeira and Porto and for the Summer seasons of 2014 and 2015 in Lisbon. Slot allocation in Portugal is performed by ANA Aeroportos de Portugal. The declared capacities for Lisbon were shown in Table 5.1. Madeira and Porto operate with runway capacity limits for each 15-minute period and 60-minute period applied on a 5-minute rolling horizon basis. In Madeira, the declared capacities are 6 movements, 4 arrivals and 4 departures per 15-minute period, and 14 movements, 7 arrivals and 7 departures per hour. In Porto, the declared capacities are 7 movements per 15-minute period and 20 movements per hour, with no separate limits on the number of arrivals and of departures. In addition, Madeira and Porto are also subject to terminal and apron capacity constraints and to noise restrictions, but these are typically not binding.

---

[10] In the European Union, the definition of a new entrant airline is somewhat less restrictive under certain circumstances (Council Regulation (EEC) No 95/93).

[11] The objective, obviously, is to facilitate access to airports by more airlines. However, the largest number of slots a new entrant can end up with in a day under the 50% provision is 4, i.e., two slot pairs – for instance, one flight in the morning and one in the evening.

5.3 Determining the Initial Slot Allocation

We now describe briefly our optimization model for slot allocation, and illustrate, through an example, its capabilities and the questions it motivates. Since this chapter focuses on the application of this model to support the Initial Slot Allocation (Step 3 of Figure 5.1) and potential enhancements to the existing IATA guidelines, we only provide here a short overview of the model. More technical details on its formulation and its solution procedure can be found in the appendix and, especially, in (Ribeiro et al., 2018).

5.3.1 The Optimization Problem

The task of the slot coordinator at a Level 3 airport during the initial slot allocation step can be viewed as an optimization problem that can be stated, in general terms, as follows: "given a set of airline requests for slots during a season of operations and a set of constraints resulting from the airport's declared capacities, propose a combination of slot assignments (i.e., a "slot allocation") that minimizes the difference between the proposed schedule of flights and the schedule that would have resulted from the airlines' requests in the absence of capacity constraints, while respecting the relative priorities of the different classes of requests". Henceforth, we will refer to this optimization problem as the Slot Allocation Problem (SAP).

In practice, coordinators do not currently employ formal optimization tools. Instead, they use a variety of approaches to perform slot allocation, often assisted by special-purpose software (e.g., PDC SCORE), which processes slot requests sequentially according to their priority class – H first, followed by CH, NE and O. Requests in each priority class are processed one-at-a-time on an *ad hoc* basis, thus not affording an opportunity to consider simultaneously the complete set of slot requests and explore the interactions among requests and the full set of combinations of potential slot assignments. The resulting slot allocation may therefore be sub-optimal.

On the other hand, the SAP is not easy to formulate as a mathematical optimization problem or to solve optimally. The first formulation that captures much of the complexity of the SAP is quite recent (Zografos et al, 2012). That same paper presents a solution of the SAP for a medium-size airport of approximately 5 million annual passengers in Greece. More recently, we have developed a new model formulation for solving the SAP called the Priority-based Slot Allocation Model (PSAM). Some of its novel features include: consideration of a complete list of capacity constraints; consideration of all types of slot requests; and the ability to solve the SAP to optimality for much larger airports than had hitherto been possible. The remainder of this subsection describes briefly the PSAM.

The PSAM takes as inputs the complete set of capacity constraints (Section 5.2.1) and the full list of airline slot requests (Section 5.2.2). The decision variables determine the slot time allocated to each request.[12] This essentially defines the *displacement* of each request, i.e., the difference between the slot time allocated to a slot series and the slot time requested by the airline. For example, if the slot times allocated to the slot series of Request 1 of Table 5.2 are 08:45 and 09:15, for the arrival and departure, respectively, of the associated aircraft, then the displacement of each slot in this slot series is equal to 45 minutes.[13] Note that the displacement can be positive or negative value depending on whether the slot series is assigned to a later time or an earlier time than requested.

Qualitatively, the PSAM is formulated as follows, where $MaxD$, $TotD$ and NoD denote the maximum displacement, the total schedule displacement and the number of slots displaced, respectively. Its full mathematical formulation is provided in the Appendix and in (Ribeiro et al., 2018).

$$\text{Minimize} \quad w_1 MaxD + w_2 TotD + NoD \tag{5.1}$$

subject to Capacity constraints
Flight connection constraints
Slot displacement constraints
Schedule regularity constraints
Technical constraints

Let us first discuss the model's objective function. The PSAM minimizes an aggregate measure of *schedule displacement*. This, however, is not unambiguous. Indeed, there may exist trade-offs between conflicting objectives such as, for instance, displacing many slots by a relatively small amount vs. displacing a smaller number of slots by a larger amount. In order to quantify, and optimize, such trade-offs, slot coordinators must minimize a measure of the overall displacement contained in

---

[12] The general version of the PSAM also includes variables to determine whether each slot request is rejected, or not to capture instances where total demand exceeds total capacity. However, this is not the case at the vast majority of Level 3 airports, including those considered in this chapter.

[13] This maintains the connection time of 30 minutes between the arrival and departure times of the aircraft.

a proposed schedule of flights. The PSAM is thus formulated as a multi-objective optimization problem that comprises three terms: (i) the maximum displacement imposed on any slot series on any day of the season, (ii) the total displacement associated with all allocated slots throughout the season, and (iii) the number of slots that were scheduled at a different time than requested.

The user-specified weighting constants (w_1 and w_2) provide flexibility in prioritizing the three displacement metrics. For example, setting $w_1 >> w_2 >> 1$ minimizes, first, the largest flight displacement, then the total displacement and then, among all the solutions that achieve these two objectives, the number of slots displaced. This corresponds to a *lexicographic* solution, where each objective is given priority over the subsequent one. It is consistent with current practices of slot coordinators, with the interests of the airlines, and with the existing literature (Zografos et al., 2012; Jacquillat and Odoni, 2015a; Pyrgiotis and Odoni, 2016). It is motivated by the underlying goal to achieve an equitable treatment of all slot requests by ensuring that no slots will incur a disproportionately large displacement, and to minimize the overall impacts of the slot allocation process, measured by the total displacement. The third objective may be helpful in cases where there still remain tied solutions. In such cases, selecting among them the solution with the smallest number of displaced slots facilitates implementation by simplifying negotiations between the coordinator and the airlines during Slot Conferences.

Note, however, that other priorities can be captured in the objective function. To this end, we will characterize the Pareto-optimal frontiers between the different objectives, that is, the set of solutions such that no other solution can improve one of these objectives without worsening the others. This can be achieved by changing the weights w_1 and w_2 in the objective function or by an ε-constraint approach. The latter involves, first, minimizing the maximum displacement, and, then, minimizing the total displacement while ensuring that the maximum displacement lies below a target that is initially set to the optimal value of the maximum displacement, and then progressively increased by increments of 5 minutes until the optimal value of the total displacement is attained. We can repeat the same procedure for the number of slots displaced, albeit at higher computational costs.

Turning to the constraints of PSAM, the capacity constraints specify the limitations imposed by the declared capacities of the runway system, apron and passenger terminals. For example, the runway capacity constraints (possibly specified on a rolling horizon basis) ensure that arrival, departure and total runway capacities are not exceeded during the course of any day in the season. Flight connection constraints ensure that the airline-requested time between the arrival of a flight and the departure of a flight performed by the same aircraft is not changed.[14] Slot displacement constraints identify the slot series that are displaced and calculate the corresponding displacement of each slot series. Schedule regularity constraint assign the same time-of-the-day to the slots belonging to the same series and to the slot series belonging to the same request, as specified or recommended in the WSG. Finally, some of the technical constraints specify the domains of the decision variables (e.g., integer or binary), while others serve to speed up greatly the solution of PSAM.

The solution procedure used by PSAM leverages the strict priorities across airline slot requests (see Section 5.2.3): requests for H slot series first, followed by CH, NE and O, in this order. The PSAM solution procedure therefore breaks down the SAP into four sub-problems, one for each of the four types of requests. A lexicographic approach is then used to solve these sub-problems sequentially, one-at-a-time, according to the priority of the requests, with H slots allocated first, and so on. The solution to the sub-problem of allocating H slots is usually trivial: each requested slot series is given its requested slots, with a resulting displacement of zero.[15] We then solve each of the next three sub-problems, with some additional constraints specific to each class. From a computational standpoint, this lexicographic approach improves the tractability of the PSAM by decomposing it into four smaller problems. On the negative side, it does not search for alternative solutions that could potentially meet the airlines' slot requests more effectively by making only modest adjustments to the priority rules specified in the WSG. We explore this issue further in Section 5.4.2.*c* by considering solutions to the PSAM that allocate all four types of requests simultaneously in a single step, instead of sequentially.

5.3.2 An Illustrative Example

We consider in this section the application of the PSAM to solve the SAP for the airport of Madeira, Portugal. The intent is to illustrate the impact of various constraints and rules associated with the existing slot allocation process and suggest promising questions to explore later on. The focus here will be on successive solutions to the SAP as more constraints and priority rules are considered.

[14] A more flexible alternative (see Section 4.2.1) is to require that the connection time will not increase or decrease by more than user-specified limits *Tmax* and *Tmin*.

[15] An exception may occur in the rare cases in which the declared capacity of a Level 3 airport is, for some reason, smaller in the next season than in the previous one.

Figure 5.2 shows the Pareto-optimal frontiers for the two objectives of minimizing the maximum displacement and minimizing the total displacement under four different sets of conditions (see details below), as well as the coordinator's solution for Summer 2014 (diamond-shaped point at the upper right side). The coordinator's solution involved maximum displacement and total displacement of 80 minutes and 12,140 minutes, respectively. This solution did modify in a few cases the airline-requested connection times between the arrival and departure pair of slots flown by the same aircraft. However, these requested connection times were treated as mandatory in the PSAM tests described below and adhered to exactly.[16] In this sense, the PSAM solved, in this case, a somewhat more constrained problem than the coordinator.

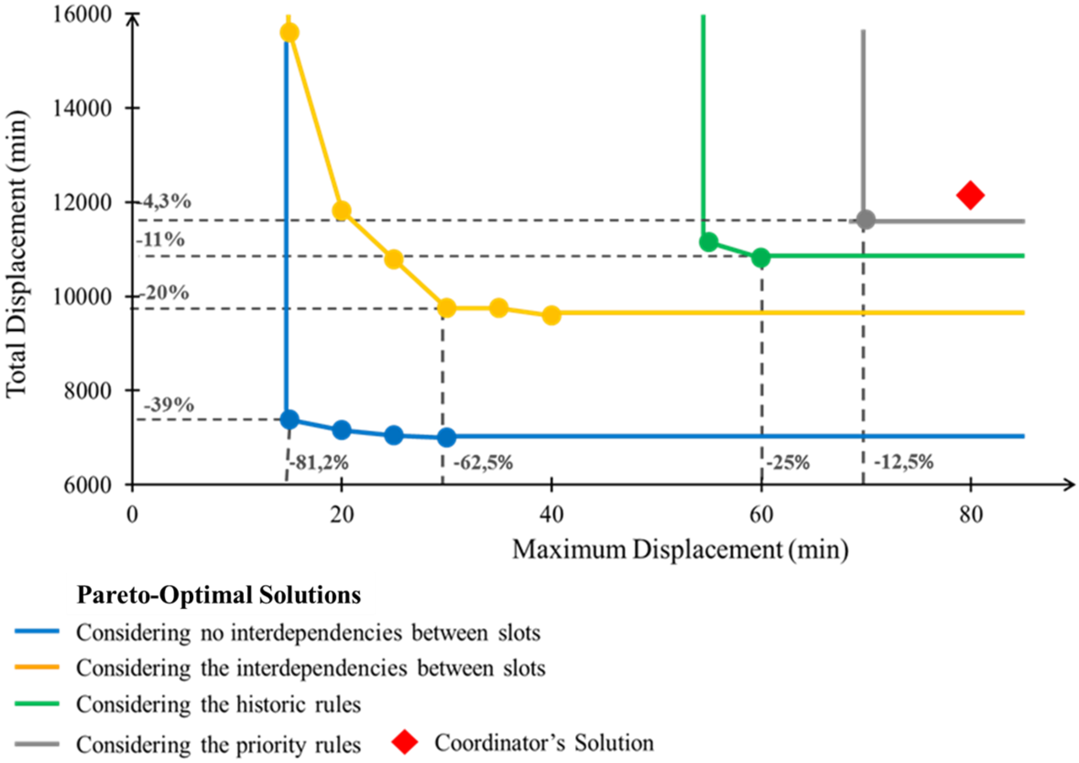


Figure 5.2 - Coordinator's solution of the SAP for Madeira Airport and four Pareto-optimal frontiers.[17]

We now discuss the Pareto frontiers obtained with PSAM and shown in Figure 5.2. First, the blue frontier corresponds to the extreme case in which there are no schedule regularity constraints and no priorities concerning the allocation of slots to different classes of requests (see Section 5.2.3). In other

[16] The impact of allowing for flexibility in turnaround times is discussed in Section 4.2.1.

[17] Figure 5.2 is the same as Figure 3.3 presented in Chapter 3. We repeat this figure to facilitate the reading of this section.

words, each day of the season is treated independently of all other days and, for each day separately, PSAM allocates slots irrespectively of whether a request comes from an H slot series, CH series, etc. The only restriction is that connection times must be maintained. As shown in Figure 5.2, the leftmost of the four points of the Pareto frontier is at (15, 7385). Thus, if days could be optimized independently and the priorities of slot requests disregarded, maximum and total displacement could be reduced by 81% (80 vs. 15 minutes) and 39% (12140 vs. 7385), respectively, compared to the coordinator's allocation.

The yellow frontier enforces schedule regularity constraints, but still disregards the priorities among the different classes of requests. Specifically, identical series of slots for different days of the week, which are submitted together as part of the same request (such as those in Request 2 of Table 5.2), must be allocated at the same time of the day across the different days of the week. The regularity constraints imply that allocations on different days become interdependent: the 210 days of the season must now be considered all at once, greatly increasing the computational complexity of the SAP. The regularity constraints will typically increase the maximum displacement and/or the total displacement needed to accommodate the slot requests. For example, the fourth point from the left of the six points that define the yellow frontier is at (30, 9,755), with a 62.5% reduction in maximum displacement and 16% reduction in total displacement, compared to the coordinator's allocation.

Up to this point, slot allocations have not considered a request's priority class. For instance, in the solutions that define the blue and yellow frontiers up to 20-30% of historic slots are displaced, in violation of the grandfather rights accorded to these slots. We now add the restriction that H slots cannot be displaced and obtain two Pareto-optimal solutions with a maximum displacement equal to 55 and 60 minutes, respectively, and a total displacement of 11,145 minutes and 10,805 minutes, respectively (shown in green in Figure 5.2). Thus, the H slot constraints result in significant increases of about 10% in total schedule displacement, compared to the yellow frontier, and more notably, in very large increases in the maximum flight displacement (from 30 minutes to 55 and 60 minutes). This is not surprising, as historic slots typically occupy the most desirable slot times and therefore tend to displace significantly slot series belonging to the three lower priority classes, especially requests for slots during times of the day when demand peaks.

Finally, we add consideration of the remaining priorities and allocate slots hierarchically to the three lower-priority classes. For this purpose, we implement the full lexicographic solution approach described in Section 5.3.1, where each priority class is treated sequentially. In this case, we obtain a single Pareto-optimal solution, shown in grey in Figure 5.2. In other words, the maximum displacement and the total displacement are jointly minimized and there is no trade-off between these two objectives. This solution has a maximum displacement of 70 minutes (a 12.5% improvement compared to 80 minutes in the slot coordinator's solution) and a total displacement of 11,620 minutes (a 4.3% improvement compared to 12,140 minutes in the slot coordinator's solution). The PSAM solution with full consideration of requirements and priorities is therefore similar to the slot coordinator's – thus confirming the realism of the model – but also results in a smaller maximum flight displacement *and* a smaller total schedule displacement, despite leaving all the connection times unchanged, unlike the solution implemented in practice. More generally, this example illustrates vividly how schedule regularity requirements and priority rules increasingly constrain the slot allocation decisions and lead to significant increases in the maximum and/or the total displacement. Note, finally, that the final PSAM solution for Madeira consists of a single point that minimizes the maximum displacement and the total displacement simultaneously. This is unusual: generally, there will be two or more Pareto-optimal solutions and decision-makers will have to consider trade-offs among measures such as maximum displacement, total displacement and number of displaced slots, as will be seen in Section 5.4.1.

5.4 The Power of Optimization Models

Optimization models, such as PSAM, offer several major benefits and opportunities. The most immediate and obvious is that they may produce improved allocations, compared to the ones that can be obtained through the heuristic approaches currently in use. They can also generate solutions that reflect different rankings of alternative objectives, such as minimizing maximum displacement, minimizing total displacement or minimizing the number of displaced slots. An equally important benefit is the opportunity the models provide for exploring ways to improve current practice in the long run. For example, they can quantify the impacts of potential changes to the rules and priorities that are currently used for slot allocation, or quantify how the total displacement is affected by small increments in declared capacity. All these points will be discussed in this section.

5.4.1 Improving Initial Slot Allocations

Table 5.3, based on the application of PSAM at Porto, provides an example of the potential of optimization models to generate improved slot allocations and produce solutions that reflect the priorities that decision makers attach to different objectives. The table shows a summary of the results of four different PSAM solutions for Porto, where different priorities are assigned to the three objectives considered in the model specified in Section 5.3.1. The trade-offs among these objectives are quantified in this way. Sol. 1 follows the order typically considered by the coordinators, i.e., “maximum displacement first, then total displacement, then number of slots displaced”, whereas Sol. 2 follows the order “total displacement first, then maximum displacement, then number of slots displaced”, Sol. 3 the order “number of slots displaced, maximum displacement, total displacement” and Sol. 4 “number of slots displaced, total displacement, maximum displacement”.

Table 5.3 - Tradeoffs among different objectives at Porto.

Solution	Maximum Displacement (min)			Total Displacement (min)				Slots Displaced (slots)			
	CH	NE	O	Total	CH	NE	O	Total	CH	NE	O
Coordinator	45	15	80	53,140	9,600	4,515	39,025	2,549	605	403	1,541
Sol. 1	25 -44%	25 +66%	55 -31%	38,625 -27%	3,560 -63%	5,220 +16%	29,845 -24%	2,379 -6.70%	348 -42%	396 -2%	1,635 +6%
Sol. 2	45 0%	25 +66%	60 -25%	37,025 -30%	2,830 -71%	5220 +16%	28,905 -26%	2,303 -9.70%	288 -52%	396 -2%	1,619 -5%
Sol. 3	25 -44%	25 +66%	55 -31%	44,620 -16%	3,590 -63%	7,360 +63%	33,670 -14%	1,898 -26%	336 -44%	370 -8%	1,192 -23%
Sol. 4	25 -44%	25 +66%	60 -25%	42,930 -19%	3,590 -63%	7,360 +63%	31,980 -18%	1,898 -26%	336 -44%	370 -8%	1,192 -23%

First, note that *all four solutions improve all three objectives* by significant margins when compared to the coordinator’s solution. For instance, Sol. 1 reduces maximum displacement by 31%, total displacement by 27% and the number of displaced slots by 7%. These are much more substantial reductions than the ones previously reported for Madeira (Figure 5.2) of 12.5%, 4.3% and 1.1%, respectively.[18] This is not surprising because Porto is a much busier airport, as it allocated 40,597 slots versus 13,196 for Madeira in Summer 2014. It is therefore much harder for the slot coordinators to find close-to-optimal solutions for Porto, without the use of an advanced optimization model, such as PSAM. Second, the trade-offs resulting from prioritizing different objectives are noteworthy. For instance, minimizing the number of displaced slots (Sol. 3 and Sol. 4) is achieved at the cost of increasing total displacement. Finally, observe that the benefits of the PSAM solution are not evenly distributed among the different priority classes. For instance, for Sol. 1 the displacement of the CH

[18] The number of displaced slots for Madeira was not reported in Section 3.2, as the discussion was limited to the maximum displacement and total displacement objectives.

slots, which are of primary interest to incumbent airlines, decreases greatly in every respect – by 44%, 67% and 42%, respectively, for each of the three measures of performance – but the opposite is the case for NE slots (+66%, +16%, -1.7%). The reason is that the large improvements in the scheduling of CH slots constrain the allocation of slots to the two lower priority classes by limiting the number of slots available at the busiest hours.

Motivated by this last observation we also analyzed an additional scenario that follows the same order of objectives as Sol. 1, but places a limit of 15 minutes on the maximum displacement that can be assigned to NE slots. In this case we reduce the improvements of the CH slots – from 45%, 67% and 42% to 45%, 59% and 36%, respectively, for each of the three measures of performance – while, at the same time, the total displacement and the number of slots displaced are now improved by 9% and 14%, respectively, compared to the coordinator's solution. This demonstrates that PSAM can also be used to explore the tradeoffs faced by the different priority classes and to determine the most desirable solution accordingly. This point will be further explored in Section 5.4.2.*c*.

The busiest airport to which we have applied PSAM to date is Lisbon, where the number of slots allocated in the Summer of 2014 and 2015 was 109,938 and 114,119, respectively – almost three times as many as at Porto. To our knowledge, Lisbon is by far the busiest airport for which an optimization model has obtained an exact solution to date.[19] Table 5.4 summarizes the model's results for Summer 2014 and Summer 2015, indicating the number of slots requested by each priority class and the corresponding amount of total displacement for each of the two seasons. Note that in this analysis we only considered capacities for the runway of 15 and 60 minutes. Comparing the results for the two years, it is interesting to observe that the number of CH slots requested in 2015 was much greater than in 2014, but the total displacement of these slots is smaller in the 2015 solution than in the one for 2014. The reason is that the number of H slots in 2015 was much smaller than in 2014, thus making more capacity available for CH slots. However, once the CH slots are assigned optimally, a smaller pool of slots remains for the NE and O requests, leading to increased total displacement for these two classes. Unfortunately, we could not compare our results with the coordinator's solution

[19] Lisbon was the 21st busiest airport in Europe in 2017 and served 186,000 movements. The top four airports (London Heathrow, Paris CDG, Amsterdam, and Frankfurt) all served around 480,000.

for the Summer 2014 and 2015 seasons because of inconsistencies in the data about the coordinator's solution.

Table 5.4 - Lisbon solutions of PSAM for Summer seasons of 2014 and 2015.

Type of Slot	Lisbon Airport 2014		Lisbon Airport 2015	
	Nr. Slots	Total Displacement (min)	Nr. Slots	Total Displacement (min)
H	47,616 (43%)	0	35,024 (31%)	0
CH	38,503 (35%)	27,790	58,126 (51%)	26,885
NE	2,362 (2%)	2,990	1,191 (1%)	18,120
O	21,457 (17%)	352,100	19,778 (17%)	365,285
Total	109,938	382,880	114,119	410,290

One of the principal conjectures that have emerged from this research is that as the number of slots allocated at an airport increases, so will the likelihood that the solution to the SAP computed by PSAM (or other optimization models) will be significantly better than the solution proposed by the coordinator with the approaches and tools currently used in practice. Stated simply, the busier the airport, the more beneficial will the use of an optimization model be. This was certainly true in the case of Porto and Madeira airports. The model's solution for Madeira, with a small number of slots, was only marginally better than the coordinator's (Section 5.3.2), whereas, in the case of Porto with twice as many slots as Madeira, it was better by a wide margin. This is to be expected as the number of possible allocations increases exponentially with the number of slots to be allocated and with the number of slots requested by the airlines. The number of possible allocations thus becomes huge for airports of the size of Porto and much more so for those of the size of Lisbon or greater. Heuristic approaches may perform reasonably well for smaller airports but, unless they are highly sophisticated, are likely to generate far less efficient solutions at the larger airports than exact optimization models.

It is important to note that the validity of our conjecture will additionally depend strongly on the "mix" of the four classes of slot requests (H, CH, NE, O), as well as on the relationship between the total number of slots requested and the number of slots available at the subject airport. Consider, for example, a Level 3 airport where, at the end of the Summer season of 2017, close to 100% of the existing slots (i.e., the entire declared capacity of the airport) are occupied by flights having historic rights and assume that the airlines that hold these rights have all met the 80% "use-it-or-lose-it" limit for the season. (This is a scenario that resembles the current situation at London Heathrow.) If all the airlines choose to keep exactly the same slots in Summer 2018 (i.e., in the extreme case where there are virtually no CH requests) and if the declared capacity of the airport remains the same for Summer 2018, then the solution to the SAP will be trivially simple (i.e., repeat, essentially, the schedule of Summer 2017) and the solution generated by an optimization model will be very close to the coordinator's. But the opposite would be true in a situation where, in Summer 2018, (i) many airlines holding historic rights choose to submit CH requests or (ii) in Summer 2017, only 85% of declared capacity was occupied (i.e., slack capacity existed at off-peak periods) and many NE and O requests were also submitted.

As already discussed, the PSAM was implemented at three Portuguese airports: Madeira, Porto and Lisbon. At Madeira and Porto, it was applied using data for the Summer Season of 2014, without considering apron and terminal constraints. In the case of Lisbon, it was implemented once for the Summer season requests of 2014, without considering apron and terminal constraints and only the runway capacities for 15 and 60 minutes, and twice for the Summer season requests of 2015, one time without apron and terminal constraints and the second time with these constraints. To solve PSAM we used CPLEX 12.5 with the GAMS programming language. The model was run with an i7 processor @3.6 GHz, 8Gb RAM computer, with a Windows 10 64-bit operating system. The complexity of the model, in terms of number of binary (0, 1) variables, integer variables, and constraints, is summarized in Table 5.5, along with computational performance, measured by the CPU time required until the optimal solution was found.

Table 5.5 - Computational statistics for a GAMS/CPLEX implementation of PSAM.

Model Indicators	Madeira Airport	Porto Airport	Lisbon Airport		
	2014[1]	2014[1]	2014[1]	2015[1]	2015[2]
Number of slots requested	13,196	40,597	109,938	114,119	
Number of binary variables	176,346	347,346	735,939	1,003,659	
Number of integer variables	1,212	2,388	5,058	6,898	
Number of constraints	889,462	1,993,956	2,567,255	3,358,397	3,872,687
CPU Time	2 minutes	8 minutes	1 hour	7 days	No Optimal Solution found after 15 days

Table 5.5 indicates that CPU time is extremely sensitive to the size of the airport, i.e., the number of slots requested. While the PSAM was solved relatively quickly for Madeira and Porto (2 and 8

minutes, respectively), Lisbon 2015 took 7 days when not considering apron and terminal constraints, and had not been completed after 15 days when these constraints were included. Thus, at the present level of development, PSAM can compute guaranteed optimal solutions at airports with traffic volumes up to Lisbon's (i.e., of the order of 200,000 movements per year). There are only roughly 15 airports in Europe at which this volume was clearly exceeded in 2017. Another aspect that impacts CPU time significantly is the distribution of slots across priority classes. The larger the share of historic slots, the easier it is to solve the PSAM, since historic slots are, first, automatically allocated to the requested time and, second, reduce the number of available times at which the requests in the other three priority classes can be accommodated. Thus, Lisbon 2015 is significantly harder to solve than Lisbon 2014 since, in addition to having more requested slots (+4%), the share of historic slots was only 31% versus 43% in 2014.

Overall, we have found that the benefits of optimization models such as PSAM increase at airports with larger number of slot requests and at airports with lower shares of historic slots. At the same time, these airports are precisely the ones that involve greater computational complexity. Ongoing research is focusing on trying to improve the computational performance of PSAM and on developing heuristic approaches that enable PSAM to obtain near-optimal solutions reasonably quickly at these airports.

5.4.2 The Value of Flexibility

One particularly important area for investigation is the impact of potential changes to some of the rules and slot priorities in the current World Slot Guidelines. We are particularly interested in quantifying the benefits that may be obtained by introducing a limited level of flexibility into these rules and priorities. Recent versions of the WSG recognize implicitly that some flexibility may yield significant benefits. The most recent version invites airlines requesting slots at Level 3 airports to indicate any flexibility they may have with respect to: (i) requested slot times and (ii) "minimum and maximum turnaround times and any other such constraints" (IATA, 2017, §9.7.3).

One of the principal advantages that optimization models offer is, in fact, the opportunity to explore the potential impacts of rule "relaxations" such as (i) and (ii). In this section, we shall explore, by using the PSAM, the sensitivity of slot allocations to small changes to the rules and priorities that are mandated by the WSG and applied in practice. This type of analysis can also inform potential future adjustments to the WSG and enhance the outcomes of the slot allocation process. The specific changes to be examined are:

(i) Flexibility in the setting of "connection times", as in (ii) above.

(ii) Consideration of aircraft size as a factor in allocating slots

(iii) Flexibility in the slot times assigned to historic slot series, as in (i) above.

(iv) "Weighted priorities" of the different classes of slot requests.

(v) Relaxation of some schedule regularity requirements in allocating slots.

a) Connection Times

The default assumption in PSAM (and in practice) is that the airline-requested connection times (or 'turnaround times') between the arrival and departure of flights performed by the same aircraft must be strictly respected. If, for example, an airline has requested an arrival-departure slot pair for a slot series at 12:10 and 12:55, respectively, then the slot pair allocated to this series must have a 45-minute time difference between the arrival slot and the departure slot. This requirement can be relaxed by specifying two *connectivity parameters*, *Tmax* and *Tmin*, which represent respectively the maximum permissible *increase* and *decrease* to the requested connection times, respectively. For instance, setting *Tmax* = 10 min and *Tmin* = 5 min means that the connection time in the above example can be equal to 40, 45, 50 or 55 minutes. Note that the default assumption corresponds to *Tmax* = *Tmin* = 0. Table 5.6 summarizes results for Porto for five scenarios in which *Tmax* = *Tmin* and both are set equal to 0, 5, 10, 15 or ∞, with the last case meaning that there is no connection time constraint, as long as this time is non-negative.

Several observations can be made. Most importantly, *even limited flexibility in connection times may generate large reductions in total displacement*. The first 5 minutes of flexibility (Scenario 2) result in an 11% overall reduction, with most of the benefits accruing to CH and NE slot requests. Even more remarkably, the total displacement of CH and NE slots is improved much more (by 30% and 46%, respectively) by varying the connectivity parameters by up to 15 minutes than by varying them from 15 minutes to infinity (an additional 9% and 4%, respectively). In fact, only the O slots may benefit significantly from any increase in the flexibility of the connectivity parameters beyond 15 minutes. This can be explained intuitively: because of their higher priority, the CH and NE slot requests will be assigned to the slots for which demand is highest, therefore "pushing" the O slots to less busy times. Thus, the O slots need additional flexibility in the connectivity parameters to obtain large benefits. Finally, it is interesting to note that the maximum displacement does not change as flexibility in connection times increases up to 15 minutes. Again, the reason is that demand for slots is concentrated around peak times and, therefore, any O requests for slots at peak times can be accommodated only after displacing them by a significant amount of time (55 minutes in this case).

Table 5.6 - Sensitivity of displacement to flexibility in connection times at Porto.

Scenario	Tmax=Tmin (min)	Maximum Displacement (min)	Total Displacement (min)			
			CH	NE	O	Total
1	0	55 0%	3,560 0%	5,220 0%	29,845 0%	38,625 0%
2	5	55 0%	3,030 -15%	4,230 -19%	27,280 -9%	34,540 -11%
3	10	55 0%	2,625 -26%	3,240 -38%	26,425 -11%	32,290 -16%
4	15	55 0%	2,475 -30%	2,830 -46%	25,225 -15%	30,530 -21%
5	∞	25 -55%	2,175 -39%	2,610 -50%	20,400 -32%	25,185 -35%

b) Size of Aircraft

Aircraft size is not one of the primary criteria considered in assigning priorities to slot requests under the existing WSG rules. The priority of a request for a slot series is not affected by whether the associated flight is performed by a narrow-body aircraft with 120 seats or a wide-body with 300. However, this may conceivably change in the future, especially at Level 3 airports that have almost reached the full limits of their capacity and cannot accommodate any additional aircraft movements (see also Section 5.5). In such cases, the only way to grow the number of passengers served is by incentivizing the use of larger aircraft by the airlines. Several Level 3 airports are now at or near this point.[20]

The PSAM can be used to explore the implications of including aircraft size as a criterion in allocating slots. For this purpose, the model can be modified, in a simple way, by modifying the objective function to consider the displacement per passenger (measured as the number of aircraft seats). Qualitatively, the new objective function is written as follows. Specifically, the second and third term are now formulated as the total number of passenger-minutes of displacement and the number of

[20] Note that, under current practice, airports charge weight-based landing fees that are higher for larger aircraft. The consideration of aircraft sizes in PSAM would therefore effect a significant shift toward incentivizing the use of larger aircraft to reflect their positive impact on passenger throughput

passengers suffering nonzero displacement, respectively (the letter S stands for aircraft seats). The full mathematical formulation is provided in the Appendix.

$$\text{Minimize} \qquad w_1 MaxD + w_2 TotD\_S + NoD\_S \tag{5.2}$$

In Table 5.7, we compare the results obtained when total displacement of slot requests is minimized with those when total passenger displacement is minimized at Porto. Aircraft have been divided into 10 classes according to number of seats. The second column shows the number of slots occupied by each class in Summer 2014, while Columns 3 and 4 show the minimum total displacement in minutes for the season under the two objective functions (i.e., without and with consideration of aircraft sizes). As expected, slot requests associated with smaller aircraft now experience more displacement, while the opposite is true for those associated with larger aircraft. For instance, the total displacement of the larger aircraft (i.e., aircraft with more than 150 seats) is reduced by 3.1%, while the total displacement of the smaller aircraft is increased by 6.5%. One might plausibly argue that this may be more consistent with the best interests of passengers and, possibly, of the airport operator and of the airlines (see also Section 5.5.2).

Table 5.7 - PSAM results at Porto when minimizing slot displacement vs. passenger displacement.

Aircraft Class (seats)	Number of Slots	Total Displacement (min)		Difference	%
		Minimize Slot Displacement	Minimize Passenger Displacement		
0-30	0	0	0	0	0%
31-60	3,847	2,700	3,120	420	16%
61-90	14	0	0	0	0%
91-120	4,128	150	150	0	0%
121-150	6,411	11,800	12,400	600	5%
151-180	8,413	9,735	9,625	-110	-1%
181-210	15,756	13,935	13,600	-335	-2%
211-240	178	0	0	0	0%
241-270	450	305	0	-305	-100%
271-300	300	0	0	0	0%

c) Weighted Priorities of the Classes of Slot Requests

We study next the third and fourth of the questions posed in the introduction to Section 5.4. This amounts to examining the impact of potential changes in the way the four classes of slot requests are prioritized. According to existing rules, the four classes are processed in a strict order of priority – H slots first, followed by CH, NE and O requests, in that order. The example of Section 5.3.2 suggested that this strict order of priorities contributes significantly to increases in displacement. We explore here the consequences of relaxing partially this strict order through a system of "weighted priorities". Particular emphasis will be given to the possibility of adding some flexibility to the assignment of historic slots by allowing some of these slots to be displaced marginally, i.e., by small amounts of time.

The weighted priorities approach requires two adjustments to the original version of PSAM. First, the objective function of PSAM is reformulated to *solve the model in a single stage*, instead of solving it lexicographically as four sequential sub-problems, as described in Section 5.3.1. Specifically, as shown in (3) below, we now state the objective function as the weighted sum of four quantities, *HVal*, *CHVal*, *NEVal* and *OVal*, which denote, respectively, the value of the objective function (5.1) for the H, CH, NE and O slots. In other words, each of the four quantities corresponds to the minimization of the displacement for each priority class. Second, the constraints on the timing of H slots is relaxed by permitting some displacement of these slots by up to a pre-specified limit (e.g., 5 minutes, 10 minutes, etc.). Constraint (4) restricts the amount of displacement suffered by any H slot, D_{H_i}, to be less than a user-specified value, $\mathrm{D}_{H}^{\mathrm{MAX}}$.

$$\textit{Minimize}\ \alpha \cdot HVal + (1-\alpha)\{\beta \cdot CHVal + (1-\beta)[\gamma \cdot NEVal + (1-\gamma) \cdot OVal]\} \qquad (5.3)$$

$$\left|D_{H_i}\right| \leq \mathrm{D}_{\mathrm{H}}^{\mathrm{MAX}} \qquad (5.4)$$

The weights α, β and γ in [0.5,1) measure the relative weight of each priority class in relation to the following class. For example, values of $\alpha = 0.5$ and $\beta = 0.99$ signify that the H and CH slots are weighted almost equally, but are given high priority over the remaining classes. In order to maintain the current priority order, each weight is at least equal to 0.5. The parameters should not be set equal to one because this would mean no cost for the displacement of slots with lower priority, and thus not result in a Pareto-optimal outcome.

To illustrate the impact of these changes, we now solve the PSAM at Porto with the weighted objective function (5.3) and the relaxed displacement constraints (5.4) for the H slot series. To simplify the analysis, we restrict our experiments to the case where the objectives *HVal*, *CHVal*, *NEVal* and *OVal* consider only the total displacement of the class considered.[21] The displacement

[21] This, of course, can be modified to capture other combinations of objectives, not just total displacement, i.e., to include consideration of maximum displacement, and/or number of slots displaced.

results are reported in Table 5.8, with α varying between 0.99 and 0.5 by increments of 0.1, and β equal to 0.99, 0.9, 0.7 or 0.5. The value of γ was always set equal to 0.99, so we did not test the trade-off between NE and O slots. We considered values of D_H^{MAX} of 5 and 15 minutes, i.e., historic slots can be displaced by a maximum of 5 or 15 minutes. We discuss these results in the remainder of this section.

Table 5.8 - Total displacement results for Porto for different values of the weight α and β.

		Total Displacement (min)											
D_H^{MAX}	α	β= 0.99 or β= 0.9				β = 0.7				β = 0.5			
		H	CH	NE	O	H	CH	NE	O	H	CH	NE	O
5 min	0.99	0	3,560 0%	5,220 0%	29,845 0%	0	4,175 +17%	2,820 -46%	26,500 -11%	0	4,825 +36%	860 -84%	27,265 -9%
	0.9	90	1,950 -45%	5,220 0%	29,775 0%	90	2,565 -28%	2,820 -46%	26,430 -11%	90	3,025 -15%	1,460 -72%	26,365 -12%
	0.8	240	1,350 -62%	5,220 0%	28,875 -3%	240	1,965 -45%	2,820 -46%	25,530 -14%	240	2,425 -32%	1,460 -72%	25,465 -15%
	0.7	290	1,210 -66%	5,220 0%	28,875 -3%	315	1,735 -51%	2,820 -46%	26,575 -11%	290	2,285 -36%	1,460 -72%	25,465 -15%
	0.6	290	1,210 -66%	5,220 0%	28,875 -3%	380	1,585 -55%	2,820 -46%	26,575 -11%	380	2,045 -43%	1,460 -72%	26,510 -11%
	0.5	290	1,210 -66%	5,220 0%	28,875 -3%	720	1,380 -61%	1,720 -67%	29,275 -2%	830	1,460 -59%	1,460 -72%	26,875 -10%
15 min	0.99	0	3,560 0%	5,220 0%	29,845 0%	0	4,175 +17%	2,820 -46%	26,500 -11%	0	4,825 +36%	860 -84%	27,265 -9%
	0.9	90	1,950 -45%	5,220 0%	29,775 0%	90	2,565 -28%	2,820 -46%	26,430 -11%	90	3,025 -15%	1,460 -72%	26,365 -12%
	0.8	240	1,350 -62%	5,220 0%	28,875 -3%	240	1,965 -45%	2,820 -46%	25,530 -14%	240	2,425 -32%	1,460 -72%	25,465 -15%
	0.7	370	980 -72%	5,220 0%	30,975 +4%	315	1,735 -51%	2,820 -46%	26,575 -11%	290	2,285 -36%	1,460 -72%	25,465 -15%
	0.6	370	980 -72%	5,220 0%	30,975 +4%	460	1,355 -62%	2,820 -46%	29,575 -1%	680	1,460 -59%	1,330 -75%	29,575 -1%
	0.5	670	410 -88%	5,220 0%	30,955 +4%	1505	435 -88%	1,590 -70%	29,555 -1%	1430	740 -79%	730 -86%	30,455 +2%

Impact of limited flexibility in historic slot times

We begin by studying the case in which (i) H slots may be displaced by up to 5 or 15 minutes, and (ii) α can vary between 0.99 and 0.5 by increments of 0.1, keeping β=0.99, i.e., CH slots still have full priority in relation to NE and O slots (left side of the table). By introducing limited flexibility to the timing of H slots, we provide an opportunity to move some slot series from the other three priority classes into more desirable times. Thus, we expect that, as the values of D_H^{MAX} and of α increase, the displacement of H slots will increase and the one of the other slots will decrease. We explore this trade-off below.

The most obvious observation suggested by the leftmost part of the Table 5.8 is that significant improvements in the displacement of CH slots may be achieved by increasing the displacement of H slots by only a small amount. For instance, increasing the displacement of H slots by 90 minutes (i.e., displacing just 18 historic slots during the entire season by only 5 minutes each) reduces CH displacement by 45%, or by 1,610 minutes (a "benefit/cost ratio" of 17.9) to 1,950 minutes. Increasing the displacement of H slots by another 150 minutes, to 240 minutes, leads to a further reduction of 600 minutes in CH displacement, or an additional 17% reduction – but with a diminished marginal benefit/cost ratio of 4.0 (=600/150).

Second, most of the benefits from increasing $\mathrm{D}_{\mathrm{H}}^{\mathrm{MAX}}$are obtained by increasing the maximum permitted displacement of historic slots from 0 to 5 minutes. Beyond this point (i.e., when increasing further ${D_H}^{MAX}$ to 15 minutes) the solution is not particularly sensitive to the additional flexibility provided by the larger $\mathrm{D}_{\mathrm{H}}^{\mathrm{MAX}}$. In fact, for values of α up to 0.8, the displacements obtained by setting $\mathrm{D}_{\mathrm{H}}^{\mathrm{MAX}}$ to 5 or 15 minutes are identical.

Third, the NE and O slot series derive only small benefits from the flexibility of the H slot times, as almost all the benefits are captured by the CH series. In fact, note that when we allow a larger displacement for H slots (small α and ${D_H}^{MAX}$=15) we actually *worse* solutions for the O slots, the ones with the lowest priority. This is because the H and CH slot requests now enjoy essentially the same level of priority and jointly occupy most of the slots at the busiest times, thus leaving fewer desirable slots for NE and O slots.

Impact of relaxing the priority of change-to-historic slots

Motivated by this last observation, we now analyze the case when the CH slot requests do not enjoy full priority over the two lower priority classes, NE and O. We therefore vary β, considering values of 0.9, 0.7 and 0.5. We expect that, for smaller values of β (i.e., for lower priority of the CH slots), the displacement of the NE and O slots will be lower. In Table 5.8, the first row shows the displacements when the priority given to CH slots is lowered without allowing displacement of historic slots (i.e., α = 0.99), and the other rows show the effects of varying α and β simultaneously.

First, note that β trades off the displacement of CH slots against the displacement of NE and O slots. For instance, increasing the CH displacement by 17% leads to a decrease of 46% and 11%, respectively, in the displacement of NE and O slots. Note that the results for the CH slots are worse than in the original solution when we consider α equal to 0.99, i.e., prioritize fully the H slot series.

However, when we allow increases to the displacement of H slots, we always obtain solutions that decrease the displacement of CH slots, no matter how much we decrease β.

Second, most of the benefits obtained by reducing the priority of CH slots are captured by the NE requests. For instance, the NE displacement can be reduced by up to 88%, but that of O slots by up to only 15%.

Finally, the effect on the displacement of O slots exhibits complex behaviors. For large values of α, the slot allocation process is highly constrained by the H and CH slot requests, so few benefits are captured by the O slot requests. On the other hand, for the smallest values of α practically all the reductions in displacement are shared by the CH and NE slots because these two classes now enjoy much more flexibility and the model is able to accommodate them better (especially given that we set γ=0.99). Thus, as far as the O slots are concerned, the benefits of the weighted priorities process considered here are largest for moderate values of α, such as α =0.8.

The main conclusions from the "weighted priorities" approach can now be summarized. First, a small increase in the displacement of H slots may lead to a large decrease in total displacement and can be beneficial to all three other classes of requests, depending on the values of α, β and γ. Second, and importantly for policy, most of the benefits obtained by displacing H slots are obtained with a small total displacement of the H slots. Large additional displacements of H slots provide only marginal additional benefits, compared to the benefits obtained from the initial small displacements. And, third, a "domino effect" can also be observed: the benefits obtained by displacing H slots are primarily captured by the CH slots and, to a much more moderate extent, by the NE slots. In turn, if priorities are weighted in a way that CH slots are prevented from capturing most of the benefits accruing from the displacement of H slots, then NE slots can experience significant decreases in displacement. An analogous observation can be made for the case of O slots.

d) Relaxing Schedule Regularity Requirements

Another area for potential changes to the current rules and practices pertains to the requirements for schedule regularity over a season, as specified in the WSG. We consider here combinations of two types of schedule regularity constraints, one with respect to time-of-the day and the other with respect to the length of the scheduling period. Concerning the first, we consider two possibilities: (i) all slots belonging to *identical series of slots for different days of the week, which are submitted together as part of the same request*, must be scheduled at the same time of the day across the different days of the week; (ii) all slots belonging to the *same slot series* must be scheduled at the same time of the day (but not necessarily all slots belonging to "identical series of slots for different days of the week, which are submitted together as part of the same request"). As noted in Section 5.2.3, current practice at most Level-3 airports is consistent with (i), the stricter of the above two possibilities,[22] so we treat (i) as the choice under the *status quo*. The default choice in the PSAM is also (i).

When it comes to the length of the scheduling period, we consider three possibilities: (a) the scheduling period spans the entire season of interest ("Summer" or "Winter"), as in current practice; (b) the scheduling period is subdivided into more homogeneous sub-periods of one or more months each; (c) the scheduling period is subdivided into sub-periods of one month each. Alternatives (b) and (c) are intended to account for the fact that at many airports, especially ones with highly seasonal demand, the airlines may have different slot requirements in different parts of a "season". For example, in the case of the Summer season airline schedules (and thus slot requirements) in April, May and October may differ significantly from those in July and August.

Table 5.9 reports the PSAM results at Porto under a set of scenarios related to the above variations for the Summer season of 2014. Scenario 1 combines (i) and (a) and Scenario 2 combines (ii) and (a). For Scenarios 3 and 4, we combine (i) and (ii), respectively, with (b), assuming that the Summer season is subdivided into three parts – April-May-June, July-August, and September-October. This particular subdivision is motivated by the fact that the number of slots requested by the airlines at Porto in July and August is about 15% higher than in the other 5 months of the Summer season, each of which has roughly the same number of requests. Scenarios 5 and 6 assume monthly scheduling periods (i.e., (c)) combined, respectively, with time-of-the-day requirements (i) and (ii). Finally, Scenario 7 represents the extreme case of no interdependences among slots. In other words, slots are now scheduled by treating each day of the season independently of all other days by minimizing the objective function (5.1) for the requests submitted for that particular day.

In the left half of Table 5.9 we see the impact of the various combinations of regularity constraints on the three objectives of PSAM – maximum displacement, total displacement, and number of slots displaced. On one hand, we observe that consideration of these constraints does not increase the

[22] The WSG actually states proposition (i), concerning identical series for different days of the week, as a recommendation, rather than a requirement (Section 2.3).

optimal solution for the maximum displacement, which is always equal to 55 minutes. On the other, they have a very significant impact on total displacement and number of slots displaced, as relaxing them could reduce the total displacement and the number of slots displaced by up to 50%.

The right half of the table presents indicators of the impact that the non-consideration of the constraints has on schedule regularity. First, it shows the percent of slot requests whose regularity constraints are violated in each scenario (e.g., in Scenario 2, only 2% of the slot requests do not have all their slot series scheduled at the same time). Second, we provide information about the amplitude of the time between slots belonging to the same slot request across the season (e.g., in Scenario 2, there are at least two slot series belonging to the same slot request that are scheduled at times differing by 30 minutes – for instance, the slots may be scheduled at 08:45 on Mondays and at 09:15 on Tuesdays). Finally, we show the average and standard deviation of this amplitude for the slot requests where a violation occurs (i.e., for Scenario 2, the average amplitude for the 2% of slot requests with a violation is 8.2 minutes and the standard deviation is 7.5 minutes).

Table 5.9 - Sensitivity of the displacement at Porto to changes in the schedule regularity constraints.

Scheduling Period		Scenario	Maximum Displacement (min)	Total Displacement (min)	Slots Displaced (slots)	Requests with Violations (%)	Maximum Amplitude (min)	Average Amplitude (min)	Standard Deviation Amplitude (min)
Entire Season	1	All series in same request	55	38,625	2,379	0.0%	0	0	0
	2	Only each individual series	55	33,885 -12%	1,897 -20%	2.0%	30	8.2	7.5
3 periods	3	All series in same request	55	27,415 -29%	2,025 -15%	10.8%	110	17.3	19.4
	4	Only each individual series	55	21,790 -44%	1,425 -40%	7.0%	110	17.6	19.1
Monthly	5	All series in same request	55	26,170 -32%	1,880 -21%	11.3%	110	18.8	18.7
	6	Only each individual series	55	20,845 -46%	1,295 -46%	7.3%	110	18.6	18.8
Daily	7	Each day independently	55	19,310 -50%	1,206 -49%	11.6%	110	19.8	19.2

Table 5.9 suggests that there may be good reasons to consider adopting changes (ii) and (b) for the schedule regularity constraints. If requirement (ii), instead of (i), were adopted with respect to the time-of-the-day regularity (Scenario 1 vs. Scenario 2), the total displacement at Porto would be reduced by 12% and the number of slots displaced by 20%. Moreover, only 2 % of the slot series in the set of “identical series of slots for different days of the week” would violate requirement (i) and

the maximum amplitude between times allocated to slots belonging to the same slot request would be only 30 minutes, with the average and standard deviation of only 8.2 and 7.5 minutes, respectively.

Table 5.9 similarly points to the potential benefits of subdividing “seasons” into shorter, more homogeneous (as far as demand is concerned) sub-periods. This can make it possible to account more effectively for the detailed characteristics of seasonality in demand. In the case of Porto, forcing slot times in April, May, June, September and October to be identical to those in July and August, as done today, means that some slot series in these five “low” months suffer significant displacement just in order to ensure regularity throughout the season. As Table 5.9 shows, Scenario 3 that subdivides the summer season into three independent sub-periods reduces total displacement by 29% and the number of slots displaced by 15%, compared to the *status quo* Scenario 1. Note that, within each sub-period, *all* schedule regularity constraints are satisfied, i.e., comply with (i), so that any violations of current regularity constraints under Scenario 3 only occur between sub-periods. If, in addition, we allow compliance with only (ii) during each of the sub-periods, total displacement and the number of slots displaced are reduced by 44% and 40%, respectively (Scenario 4). Interestingly, little further benefit is obtained by breaking down the season into monthly periods, as seen by comparing the results for Scenario 5 with those for Scenario 3 and for Scenario 4 with Scenario 6. Moreover, and importantly, this observation holds even for the extreme case of Scenario 7 in which each day is treated as a separate and independent sub-period!

Although the maximum amplitude between scheduled times for Scenarios 3 through 7 is 110 minutes[23], amplitudes of this magnitude are very rare. For example, for Scenario 3, only one slot request had a difference of 110 minutes, while all the others had amplitudes of 55 minutes or less, with more than 70% at less than 15 minutes. This suggests that the addition to PSAM of some additional constraints that limit the maximum amplitude between the scheduled times of slot series belonging to the same slot request may lead to even better results in terms of the average and standard deviation of the amplitude.

[23] This means that a slot may be scheduled at 10:50 in July and August, and at 09:00 in the other months. Note that the differences between scheduled slot times are now measured across different sub-periods of the season.

Finally, we observed that the impacts of the changes in schedule regularity constraints are shared across all the priority classes of slot requests[24] and therefore do not raise any issues regarding the distribution of potential benefits.

5.4.3 The Impact of the Declared Capacity on Slot Allocation

While the effects of flight schedules on airport on-time performance have been well documented, the impact of declared capacities on slot allocation has been the subject of more limited attention. We now use the PSAM to demonstrate through an example the strong impact of declared capacities on the displacement of slot requests. Specifically, we perform a sensitivity analysis of the total displacement at Porto as a function of its declared capacity. Recall that Porto was subject, in the Summer season of 2014, to a limit of 20 movements per rolling hour and of 7 movements per rolling 15-minute period, with no separate limits for arrivals only and departures only, and no terminal or apron capacity limits. We vary these two values (C60' and C15', respectively) and report the resulting total displacement in Table 5.10. Note that this does not consider the potential existence of latent demand, i.e., the possibility that the airlines may request more slots at peak hours, or more slots altogether, in response to increases in declared capacities.

Table 5.10 - Total displacement results (in minutes) for different declared capacities

C60'/C15'	6	**7**	8	9	10	11	12
19	Infeasible	Infeasible	Infeasible	Infeasible	Infeasible	Infeasible	Infeasible
20	Infeasible	38,625 0%	27,090 -30%	19,660 -49%	14,460 -63%	12,690 -67%	12,470 -68%
21	Infeasible	30,760 -20%	18,410 -52%	10,430 -73%	4,900 -87%	3,080 -92%	2,560 -93%
22	Infeasible	29,480 -24%	16,210 -58%	8,110 -79%	2,640 -93%	820 -98%	300 -99%
23	Infeasible	29,180 -24%	15,910 -59%	7,810 -80%	2,340 -94%	520 -99%	0 -100%
24	Infeasible	29,180 -24%	15,910 -59%	7,810 -80%	2,340 -94%	520 -99%	0 -100%
25	Infeasible	29,180 -24%	15,910 -59%	7,810 -80%	2,340 -94%	520 -99%	0 -100%

[24] Historic slots are not impacted anyway.

The main insight from Table 5.10 is that the optimal total displacement is highly sensitive to the value of the declared capacity. For instance, an increase in the total hourly capacity by one slot (from 20 to 21 movements per hour) would reduce the total displacement by 20%. A simultaneous increase in the total 15-minute capacity by one slot (from 7 to 8 movements per period), while still keeping the hourly limit to 21, would result in a displacement 52% smaller than under the current setting! Note, moreover, that any reduction in the declared capacities from the current values results in an infeasible problem. This is due to the hard constraints associated with H and CH slots. Finally, further increases in declared capacities would further reduce the total displacement, until all slot requests can be met *exactly* with a capacity of 12 movements per rolling 15-minute period and 23 movements per rolling hour. Note, however, that reductions in displacement are marginally decreasing. For instance, increasing declared capacity from 21 to 22 movements per hour, or from 8 to 9 movements per 15-minute period, would result in significantly smaller reductions in the optimal value of the total displacement than respective increases from 20 to 21 movements per hour or from 7 to 8 movements per 15-minute period.

At the same time, it is well known that air traffic delays and airport on-time performance are *also* highly sensitive to the values of declared capacities (Pyrgiotis et al., 2011; Jacquillat and Odoni, 2015b). Therefore, capacity declaration is of paramount importance in slot allocation. Declared capacities can lead to high levels of congestion and deteriorated levels of service if set too high, and to high displacement of slot requests if set too low. This suggests opportunities to define a transparent and standardized process for declaring capacities across Level 3 airports, supported by up-to-date analytical models, as well as frequent performance re-assessment by collecting appropriate data and reporting critical metrics in view of strategic goals. We summarize here the principal guidelines concerning capacity declaration that have emerged from recent research and the analysis above that could form a basis for such a process (Jacquillat and Odoni, 2015a; Zografos et al., 2017).

(i) Should be set with respect to the full spectrum of the operating conditions observed at the airport (e.g., good/poor weather conditions, different runway configurations, etc.). Focusing solely on poor weather conditions bears the risk of declaring overly conservative capacities resulting in unnecessarily low scheduling levels, while scheduling practices based on good weather alone can lead to excessive delays in poor weather.

(ii) Should be defined, ideally, at high levels of granularity. For instance, existing practices that declare separate limits for arrivals and departures, or that specify separate runway, terminal and apron capacities, tend to lead to a better matching of the scheduling levels with the airport's operating capabilities.

(iii) Should balance the supply-side operating capabilities *and* the demand-side characteristics of airline slot requests to achieve a targeted level of service, and account for the nonlinear relationships between schedule displacement and airport delays, on one hand, and declared capacities, on the other. For instance, two hypothetical airports, with the same supply-side characteristics, might end up with different declared capacities if airline demand characteristics at the two airports are very different.

(iv) Can vary from one period of the day to another to accommodate higher volumes of operations at peak hours, which can be compensated, in part, by schedule valleys at off-peak hours. This could lead to a better matching with airline demand, while maintaining high levels of service. Declared capacities that vary by time-of-day have become common practice at some of the busiest Level 3 airports (e.g., Amsterdam Schiphol, London Heathrow).

(v) Can vary from one week of the season to another to accommodate higher volumes of operations during the busiest periods of the year and/or the periods with the (historically) most favorable conditions. This is motivated by variations in airline demand and weather conditions over time (e.g., demand is 15% higher in Porto in July and August than during the rest of the "Summer" season).

5.4.4 The Issue of Fairness

Objective functions such as (5.1) or (5.3) minimize aggregate displacement costs. Airlines, however, are also greatly interested in how these costs are distributed among them. It is clearly undesirable to allocate a disproportionate amount of displacement to one airline or a small number of airlines. This issue has given rise to several recent studies on the subject of ensuring inter-airline equity (or "fairness") in assigning slots at Level 3 airports.

A "fair" allocation, in this context, can be defined as one in which the share of total displacement assigned to any airline is roughly the same as that airline's share of the total number of slots allocated at the airport[25]. This definition can be easily expressed mathematically and incorporated into

[25] This assumes that no request is rejected; if this is not the case, the definition can be modified accordingly.

optimization models, such as PSAM, in ways that force rejection of solutions in which an airline receives an unacceptably large or small share of total displacement compared to its share of slots (Zografos and Jiang, 2016; Jacquillat and Vaze, 2018). Several more refined variations of the definition of fairness have also been proposed and studied. For example, one may wish to assign more weight to the displacement of larger aircraft (thus implicitly prioritizing such aircraft) by taking into consideration the number of seats on the aircraft associated with each requested slot series (Zografos and Jiang, 2016). Or, one may require that the share of displacement assigned to any airline should be proportional to the share of slots this airline has requested during peak demand periods, thus not penalizing airlines that submit requests for slots during off-peak periods (Fairbrother and Zografos, 2018).

Several attempts have already been made to assess the impact of equity-related requirements on the efficiency of slot allocation and flight scheduling at Level 3 and other congested airports. An interesting pair of observations has emerged from this work (Jacquillat and Vaze, 2018). First, inter-airline equity can be achieved at only a small loss (or at no loss) in efficiency under a wide range of realistic conditions and hypothetical scenarios. Thus, fairness in the distribution of displacement does not necessarily mean a significant increase in total displacement. However, the reverse is also true: solving an optimization model by considering exclusively the efficiency objective, while ignoring inter-airline equity, risks the possibility of reaching highly inequitable slot allocations. In other words, the most efficient solution may be one with a very “unfair” distribution of displacement costs, whereas other solutions may exist that achieve the same (or a similar) level of efficiency, while also ensuring adequate equity. Thus, whenever issues of fairness may potentially arise, it is important to incorporate explicitly into models that support the allocation of slots, such as PSAM, the consideration of inter-airline equity.

5.5 Further Areas for Improvement

We now extend the scope of the search for improving the existing slot allocation process by examining two other important issues: the designation of airports as Level 3; and the solution of the SAP at a network level – as opposed to solving it for each airport separately.

5.5.1 Designation as Level 3

The airports currently designated as Level 3 constitute a highly non-homogeneous group. At one extreme, they include a number of airports that served more than 50 million passengers in 2017 and received more requests for slots than their declared capacities for many hours of the day and many days of their peak Summer or Winter seasons. At the opposite end, they also include many others that handle fewer than 10 or even 5 million passengers annually and with only a small number of hours in a season when the number of slot requests exceeded declared capacities.[26] These differences reflect the vagueness of the current definitions of Level 3 (and Level 2) airports in the WSG, as well as the absence of consistent national policies concerning the designation of airports as Level 3. This state of affairs leaves much room for arbitrary and divergent local practices and has led to proliferation in the number of Level 3 airports in some countries.

According to the WSG, Level 3 airports are those where “a) demand for airport infrastructure significantly exceeds the airport’s capacity during the relevant period; b) expansion of airport infrastructure to meet demand is not possible in the short term; [and] c) attempts to resolve the problem through voluntary schedule adjustments have failed or are ineffective” (IATA, 2017). But the WSG offers very limited guidance for ascertaining whether these three conditions are met. Concerning a) – the most fundamental among the three – it calls for a “demand and capacity analysis using commonly recognized methods” that “should objectively consider the ability of the airport infrastructure to accommodate demand at desired levels of service, such as queue times, levels of congestion or delay” (§6.1). It does not, however, provide any guidance for determining whether “demand significantly exceeds capacity” or any benchmarks for what constitute “desired levels of service”.

A few airports or national organizations have attempted over the years to develop such benchmarks mostly for planning purposes. However, much of the available information on these benchmarks is anecdotal and has been rarely used in determining whether an airport should be designated as Level 3. Historically, in the early 1960s, the US Federal Aviation Administration (FAA) set an average delay of 4 minutes per aircraft movement during peak hours as the threshold at which an airport would be considered to have reached its “practical hourly capacity” – essentially, the equivalent of being designated as a “congested” airport (de Neufville and Odoni, 2013). This threshold was abandoned after delays at many major airports in the US exceeded it, by a wide margin, in the late 1970s and 1980s. A number of other more recent examples of benchmarking include the following: (1) the

[26] Roughly 50% of the 103 European airports designated as Level 3 in Summer 2017 were in this category. All of these airports served fewer than 8 million passengers in 2017 with the great majority at lower than 5 million.

declared capacity of London Heathrow is believed to be determined with the aid of a detailed simulation that uses a 10-minute average delay per movement during a peak day as the target level-of-service; (2) China's Civil Aviation Agency is said to be currently using an average delay of 8 minutes per movement in a peak day as the upper limit on acceptable level of service at the country's major airports; (3) in connection with several consulting projects at major airports in North and South America, Europe and Asia/Pacific, one of the authors of this chapter has been using, with agreement from local decision-makers, a 5-minute average delay per movement during a typical day of operations and a 10-minute average delay for peak-hour movements as thresholds (when either of these levels is reached, the airport is considered to have reached its limits for acceptable level of service); and (4) the FAA has recently defined "significant congestion" as delays of 7 minutes or more per movement during more than 30 percent of the hours between 07:00 and 22:59 and "severe congestion" as delays of 15 minutes or more per movement during more than 50 percent of the hours in that time window (FAA, 2015).

Although these above thresholds vary in terms of both numerical value and level of sophistication, they all share the perspective that the maximum acceptable average level of delay in a day is in the range of 5-10 minutes per movement. This suggests that future versions of the WSG could offer some quantitative guidelines regarding "desired levels of service", a step that could contribute to more uniformity in practices around the world. A broad range of target values for delays could be stated (e.g., average of 5 – 10 minutes per movement). Decision-makers at each airport could then: (i) select a specific target value from that range based on local preferences; (ii) perform an analysis to determine whether the airport can meet that target in the absence of any slot limits; and (iii) in the event that it cannot, request designation as Level 3, so it could meet the target by declaring capacity limits and allocating slots accordingly among the airlines requesting them. Similarly, if the target level of service can be met without interfering with airline requests (or after limited interference) at an airport that has already been designated as Level 3, the airport could revert back to Level 1 (or Level 2). The underlying analyses could be supported by readily available queuing models and simulation tools. In addition to the benefits resulting from more standardized and internally consistent practices concerning designation, this process would be transparent, verifiable and objective.

5.5.2 A New Class of "Level 4" Airports?

Most Level 3 airports have sufficient capacity to accommodate virtually all slot requests, albeit often at the cost of significant slot displacement and forcing certain flights to operate at other than their most desirable times. However, a few critical airports are operating at saturation or near-saturation levels. The overwhelming majority of slots at these airports are “historic”, so they have little room for accommodating additional requests. Coordinators are thus forced to reject some new slot requests year after year. Moreover, much latent demand exists at these airports: airlines that would otherwise initiate or increase service at these airports do not bother submitting slot requests being aware of the unavailability of free slots. Should the capacity of these airports ever increase significantly, it is certain that numerous additional slot requests would be immediately submitted to claim the new slots. London Heathrow (LHR), Paris Orly (ORY) and Hong Kong International (HKG) are prominent examples of such Level 3 airports, but several others are increasingly experiencing similar conditions, in some cases due to agreed limits on annual numbers of movements in response to environmental concerns.

It may be useful to classify these extremely congested airports into a separate class of “Level 4” airports, as they face different types of decisions and underlying tradeoffs concerning slot allocation than the large majority of Level 3 airports. A new set of allocation rules may be called for at these airports, which might differ in significant ways from the set of rules that currently applies at Level 3 airports. For example, the emphasis in the new rules may be on criteria for accepting or rejecting requests for slots, rather than minimizing displacement. It is probable that this would translate into assigning more importance to what today are called “additional criteria”, for example, giving preference to slot requests for high capacity aircraft and to slot series of long duration, such as series with year-round operations. In this way, for any given maximum number of movements that an airport could accommodate per year (or per season), the airport would maximize the number of passengers it serves or some other measure of utilization. As shown in Section 5.4.2.*b*, the PSAM is capable of taking aircraft capacity (i.e., number of seats) into account as a criterion in slot allocation. The example given in Table 5.7 also demonstrated that doing so might result into a different fleet mix than under the current Level 3 slot allocation rules and priorities. Other “additional criteria” whose importance may be elevated in the case of “Level 4” airports, include consideration of the type and number of markets served and ensuring and promoting the existence of a competitive environment.

5.5.3 Network Considerations

Another important area for improvements to the state-of-the-art is the resolution of network-level issues during the slot allocation process. This is currently done in an *ad hoc* manner, mostly during the Slot Conferences, as initial slot allocations at all Level 3 airport are carried out from an entirely local, airport-level perspective. The coordinator makes the initial slot allocation decisions with little or no knowledge of the decisions made by other coordinators at other airports. It is only after all the decisions are communicated to the airlines that problems of incompatibility between allocations made at different airports become known. The likelihood of such incompatibilities for any specific flight depends on the priority class to which the associated slot request belongs. Consider, for example, a flight from a Level 3 airport, A, to another Level 3 airport, B, that a particular airline, X, wishes to schedule through two different slot series requests submitted separately at A and at B. If both the departure slot at A and the arrival slot at B are H slots, there will almost certainly be no problem of compatibility between the two slots, as (i) the schedulers at the airline will have made sure that the scheduled time between the two slots is equal to the block time necessary to fly from A to B and (b) airline X is guaranteed to receive the two slot times requested. However, any other combination of slot priority classes (e.g., the slot requested at A belongs to the CH class and the one requested at B to the O class) runs the risk that, after the initial slot allocation, there will be an inconsistency between the slot times allocated at A and at B, i.e., the time between the scheduled departure from A and the scheduled arrival at B will be different from any feasible A-to-B block time. Thus, the “decentralized” slot allocation process currently in use may yield solutions that are incompatible with one another at the network level. An additional issue (which, however, is easier to deal with in practice) may arise when an aircraft is ultimately headed to an airport C, that imposes a curfew after a certain time of the day. It is again possible that the timing of a slot assigned to that aircraft at a Level 3 airport may make it impossible to eventually reach C before the curfew begins.

Problems of this type are currently resolved by adjusting and revising the initial slot allocations at the Slot Conferences and follow-up exchanges. But these issues have also motivated interest in models for optimizing slot allocation at the network level, a topic that has attracted some attention during the current decade. Because of its great complexity, it is generally accepted that the exact solution of a network-wide PSAM-like model is intractable at this time, i.e., it is impossible to solve optimally, in reasonable computational times, the SAP for *networks of airports* of realistic size, while respecting all the slot priorities, capacity constraints and, especially, schedule regularity requirements at each and all airports. Recent research has therefore focused on heuristic approaches. To our knowledge, two efforts represent the state-of-the-art. Pellegrini et al (2017) have developed SOSTA (Simultaneous Optimization of airport SloT Allocation), a model that optimizes exactly and in reasonable computational times, the simultaneous allocation of slots for *a single day* at all airports in a large Europe-wide network, while respecting most slot allocation rules in the WSG. The slot allocation recommendations made by SOSTA for a peak day in the entire European network seem realistic and compare favorably with the allocations made in practice. The main weakness of SOSTA is that it does not consider schedule regularity requirements, i.e., by solving the slot allocation problem for a single day, rather than simultaneously for all days of a season, it does not necessarily schedule all slots of a series at the same time of the day across a season. The paper's authors have suggested a heuristic approach for partially addressing this problem by solving SOSTA sequentially for a carefully selected set of days. Benlic (2018), on the other hand, does consider schedule regularity requirements and, unlike Pellegrini et al (2017), adopts an entirely heuristic approach to solve a number of instances of network-wide slot allocation problems. These instances involve large numbers of flights, comparable to the total number of commercial flights in Europe in a season, and are solved in about 1 minute each. The instances, however, are not based on field data and are computer-generated under a number of assumptions, some of which simplify the problem greatly, including the assumption that all airports have similar or identical seasonal and daily demand distributions. A more definitive assessment of this approach must therefore await its application to more realistic problem instances.

In summary, the network-wide solution to the SAP is an important and still open problem.

5.6 The Larger Context

Any discussion of improvements to the existing slot allocation process at Level 3 airports would be incomplete without at least a brief mention of some of the major policy issues that are raised by current practices. In this section, we summarize, first, a selected subset of potential mild changes to the rules concerning historic slot rights, new entrants, and slot series, as well as to practices concerning participants to the decision-making process and their roles. We then touch very briefly on a couple of more fundamental questions that deserve, at the very least, a full debate at a time of rapid growth in demand for air travel and increasing pressures on airports.

a) Historic Slot Rights

The rules and rights associated with historic slots are probably the most controversial aspect of existing slot allocation practices. Earlier in this chapter, a number of ways were mentioned for introducing limited flexibility in the way historic slots are scheduled (e.g., by permitting small displacements of historic slot times, Section 5.4.2.*c*). The impacts of such more flexible rules were studied and their benefits assessed. At a more basic level, however, much criticism has been directed to the practice of assigning historic slots in perpetuity to airlines, with violations of "use-it-or-lose-it" rules being the only way a slot can be lost. Clearly, this severely limits change at the most popular airports, where most slots are already historic ones, and places non-incumbent airlines at a competitive disadvantage.

One of the most often mentioned potential changes to the status quo is an increase in the minimum percent of time (currently set at 80%) that a slot should be used. Presumably, such an increase (e.g., to 85% or 90%) would make it more difficult for some airlines to continue the tactic of "sitting on" their slots. Even in the (likely) case that airlines will simply increase their use of slots to meet the new threshold, the change to a higher limit will, at the very least, reduce the number of unused and wasted slots. It would also be useful to add specificity to the definitions in WSG of when a slot "is not used because of the airline's fault" and when it "is not used for reasons outside the control of the airline" (IATA, 2017).

A more drastic change might mandate the expiration of some specified fraction of historic slots each year. For example, 5% of the slots might be allowed to expire each year, meaning that the expected lifetime of historic rights to a slot would be 20 years. The expiring slots would be returned to the pool of "open" slots for re-allocation. Which specific slots would expire in any particular year might be determined through a lottery, or through an administrative procedure, possibly with inputs from the airlines (e.g., each airline would be asked to indicate the 5% of its slots it would give up), or, conceivably, through market mechanisms (e.g., airlines might trade slot elimination requirements with each other). Similarly, the newly open slots would be allocated each year through an administrative procedure or through a hybrid of administrative rules and market mechanisms. In 2008, the FAA proposed, but never implemented, an approach along these lines that included auctioning a specified fraction of the newly opening slots each year.[27]

b) New Entrants

New entrants may, in principle, act as the catalysts of increased competition at an airport, as well as contribute to the opening of new routes and markets. In this respect, the principal criticism of the existing slot allocation system is that it is too restrictive. It has often been argued that (i) more than (the current) 50% of the pool of "open" slots should be allocated among new entrants and, far more importantly, (ii) the definition of a new entrant as "an airline requesting a series of slots at an airport

[27] *The New York Times* (2008). U.S. to Auction Slots Soon at New York City Airports, Dec. 4, 2008.

on any day where, if the airline's request were accepted, it would hold fewer than 5 slots at that airport on that day" should be changed by increasing significantly the limit of 4 slots a day.[28] With regard to (i), not only the 50% limit might be increased, but consideration might also be given to assigning higher priority to some slot requests by new entrants (e.g., those proposing the use of large aircraft to serve new markets) than to *change-to-historic* requests. As for (ii), the limit of two slot pairs per day essentially relegates new entrants to the role of airlines that provide one connection in the morning and another in the evening in a single market,[29] or a single connection per day to each of two markets. This precludes the possibility that a carrier (legacy or low-cost) would move aggressively into a Level 3 airport and offer a package of flights that would enable it to compete with the main incumbents at the airport for a significant share of overall originating, terminating and connecting traffic.[30] Thus, this restriction propagates the *status quo* at so-called "fortress airports" that are typically dominated by a single carrier.

c) Slot Series

A series of slots is currently defined as "at least 5 slots requested for the same time on the same day-of-the-week, distributed regularly in the same season". Increasing this number (for example, to a minimum of 9 slots so as to offer service during a minimum of roughly one-third of a season) might prove beneficial by contributing to a more "regular" schedule of flights over a season and making more slots available for the longer series. Such a change would also simplify and facilitate the allocation process itself.

d) Participants to the Process

A striking aspect of the existing process is the limited role of airport operators in decision-making on slot allocation. The only step in which airport operators are formally involved is Step 1 of Figure 5.1, the setting of declared capacities. The only other responsibility of the airport operator mentioned explicitly in the WSG is to "provide relevant information to the coordinator" concerning some of the

[28] The EU has modified slightly this definition for special cases, such as flights between a Level 3 airport and a small regional airport.

[29] This was the traditional, pre-liberalization role of "flag carriers" that offered service to/from their home bases.

[30] In many well-known instances at major US airports, air carriers have adopted this type of aggressive strategy, with strong positive effects on competition and on travel options at the subject airports.

"additional criteria" for slot allocation (IATA, 2017). Nor is there a role for airport operators in setting up the rules and priorities in the WSG. Preparation of the WSG is overseen by the Joint Slot Advisory Group (JSAG). Despite the claim that "the composition of JSAG reflects the global nature of international air transport" (IATA, 2017), the JSAG "is comprised of an equal number of IATA Member airlines and airport coordinators", with no representation of airport operators. A significant upgrading of the institutional role of airport operators in the process of Level 3 slot allocation would seem called for in view of (i) their deep understanding of local operating conditions and constraints and (ii) the growing role that airports play in promoting connectivity and, more generally, economic and social welfare at the local and regional levels.

A similarly limited role is also typically played by national governments. Coordinators (whether as individuals or as organizations) are typically appointed by governmental agencies or ministries, but they often operate with limited governmental input, despite the fact that decisions concerning slots, as well as those concerning the designation of airports as Level 2 or 3, are not purely "technical" matters, but have important national socio-economic implications.

e) Fundamental Questions

Finally, for the sake of completeness, it is essential to emphasize here that a number of fundamental questions about slot allocation at congested airports are still open ones. We mention only two of these here and note that they have already attracted a huge amount of attention by academics and practitioners alike.

First is the question of the property rights associated with historic slots (see, e.g., Czerny et al (2008), with numerous references to additional publications). An airport slot is defined as "a permission given by a coordinator for a planned operation to use the full range of airport infrastructure necessary to arrive or depart at a Level 3 airport on a specific date and time" (IATA, 2017). Yet, for most practical purposes, slots are treated as property of the airlines, with the right to sell, lease or exchange. The "secondary trading" of slots, a practice which is spreading – slowly but steadily – internationally, has brought to the fore these two conflicting views of property rights ("permit" vs. "property") and may force aviation policy-makers to finally resolve the issues. Haylen and Butcher (2017) present a good review.[31]

A second fundamental question concerns the use of market mechanisms – in combination with an administrative process or alone – in airport slot allocation (see Gillen et al (2016) with numerous references to additional publications). Contrary to the statement of IATA cited in this chapter's introduction, many economists and other researchers that have studied the issue have concluded that such mechanisms can, under many circumstances, improve the utilization of already-congested airport capacity or provide benefits to improving customer experience and choice in connectivity and fares.

5.7 Conclusion

The policy issues raised by slot allocation at the busiest (Level 3) airports are vast and complex. In this chapter, we have addressed several of them to various extents, with the main focus on the analysis of possible improvements to IATA's World Slot Guidelines (WSG) that could be realistically implemented in the short term. To our knowledge, the analyses we have performed on the current slot allocation process surpass, both in breadth and in depth, those previously reported in the literature and provide valuable insights for the discussions that the International Civil Aviation Organization (ICAO) has encouraged on this subject.

For carrying out the analyses, we have relied on an optimization model that is fully compliant with the WSG, i.e., that takes into account all the objectives considered in the WSG as well as the runway, apron, and terminal constraints faced by the airports. The application of the model allowed us to compare the slot allocations performed by the coordinators at a few airports in Portugal with the best possible slot allocations under a variety of objectives and constraints, and to assess the impacts of potential changes to the rules and priorities that are currently used.

The results obtained through our model suggest, first, that the solutions found by slot coordinators can be close to optimal at small airports, but could probably be improved significantly at midsize and large airports. Second, and most importantly, our results show that even limited adjustments in the WSG rules could lead to a use of airport capacity that would match better the requests of the airlines

[31] See also *The Economist*, Winning the Slottery, November 17, 2017.

(though not necessarily for all kinds of requests at the same time). This was observed for all types of changes analyzed. For instance, at the Porto airport, improvements of 66% in the displacement of change-to- historic slots were observed when the historic slots were allowed to be displaced by only 5 minutes, instead of being completely immovable as currently mandated by the WSG. Another example is the reduction in displacement when the slot allocation is performed, not for a full season as today, but for shorter and more homogeneous (as far as demand is concerned) sub-periods. In the case of the Porto airport, an improvement of 29% in total slot displacement was observed when the Summer season was divided into three sub-periods, to distinguish the two peak months (July-August) from the non-peak ones (May-June and September-October).

The changes to the WSG we have analyzed in detail could, on their own, bring considerable benefits in the short term. However, they are not the only changes to consider. We have identified and summarized in this chapter a large number of other potential changes that could make the slot allocation process at Level 3 airports more efficient, fair, transparent and inclusive. In the future, we plan to devote a substantial part of our research efforts to the study of the impacts of such changes.

6 Conclusion

Air traffic demand has grown to exceed available capacity at many airports worldwide, resulting in the routine occurrence of flight delays and high costs to airports, airlines and passengers. For instance, the nationwide impact of air traffic congestion in the United States was estimated at over $30 billion in 2007 (Ball et. al.,2010). Absent opportunities to expand airport capacity, it is necessary to resort to demand management measures for preventing over-capacity scheduling. The foremost demand management mechanism in use today is the administrative slot allocation process operated by the International Air Transport Association (IATA), which is in place at the great majority of busy airports outside the United States. In 2017, slot allocation was applied at 177 airports, serving a total of 3.15 billion passengers annually. The process is carried out according to IATA Worldwide Slot Guidelines (WSG). It involves assigning slots to thousands of slot requests while complying with a large set of criteria, rules and requirements. This results in immense complexity and in trade-offs between conflicting objectives. Current slot allocation procedures are assisted by specialized software packages, where slot requests are typically treated sequentially in an *ad hoc* basis. Recently, some optimization models of slot allocation have emerged in the literature suggesting that there is potential for improving significantly slot allocation decisions. However, existing optimization approaches do not account for all the rules specified in the WSG, and their implementation remains limited to small-size airports due to the combinatorial complexity of the problem.

In this dissertation we have developed a modelling approach for airport slot allocation compliant with the rules and priorities specified by the WSG. The modelling approach is composed by an integer programming model and heuristics capable to provide near-optimal solutions at the largest coordinated airports in the world. The tool developed can also be used in support to the assessment of the existing rules and procedures of the current slot allocation process. Overall, we believe we have created the foundations for the development of innovative technologies and policies that can effectively enhance current guidelines and procedures in a significant manner and contribute to substantial improvements in the utilization and management of airport capacity worldwide, providing substantial benefits to airports, airlines, their users, and the economies as a whole.

Before presenting our future research plans, we detail bellow the biggest achievements accomplished over the course of this dissertation.

6.1 Summary of research

The contributions of this thesis start in Chapter 3, with the formulation of our Priority-based Slot Allocation Model (PSAM), an original integer programming model that optimizes slot allocation decisions while fully complying with the priorities and requirements specified by the IATA guidelines. In relation to previous literature, we formulate for the first time the rules of the change-to-historic slots and the new entrants. We have also formulated two new objectives, specifically the minimization of flights rejected, and the minimization of number of slots displaced. The first objective can be particularly useful at airports where total demand is so high that some slot requested may have to be rejected to satisfy the declared capacity constraints. The second objective aims to reduce the complexity of the process after the initial slot allocation by making easier the subsequent negotiations during the slot conference.

To strengthen our initial formulation for PSAM, we have proposed a set of valid inequalities that provide better linear relaxations for the model, and consequently, faster solution times (e.g. instances that could not be solved optimally in less than one day, are now solved in just a few seconds). This has enabled the implementation of our modelling approach at the airports of Madeira and Porto in about 2 and 5 minutes respectively using exact methods of optimization provided by commercial solvers such as CPLEX. These airports operate roughly 25,000 and 100,000 aircraft movements per annum, which, in the case of Porto airport, represents about twice the number of annual aircraft movements of the busiest airports previously considered in the literature (Zografos et. al., 2012).

Using our modelling approach, we have performed comparisons with real-world slot allocation. Results suggest that the model can improve the efficiency of current practice by providing slot allocations that match better the slot request of airlines. Specifically, for the case of Madeira (resp. Porto) airport, our modelling approach reduces the maximum displacement experienced by the airlines by an estimated 13% (resp. 31%), the total displacement by 4.5% (resp. 27%), and the number of slots displaced by 1% (resp. 7%).

Still in Chapter 3, a sensitivity analysis to the main rules of the IATA guidelines was performed. This analysis has shown that these rules have a very significant impact in the slot allocation decisions, and therefore their consideration in any tool intending to support slot allocation is indispensable. Moreover, the results obtained from this analysis suggest that small relaxations to the IATA guidelines may provide significant improvements in the slot allocation efficiency. This observation has motivated the work developed in Chapter 5.

In Chapter 4, we extended PSAM to capture other capacity restrictions that might be limitative in some coordinated airports, such as terminal and apron constraints. The extended model is referred to as Priority-based Slot Allocation Model with Runway, Terminal and Apron constraints (PSAM-RTA). Through its implementation at Lisbon airport we have shown that while terminal constraints provide cuts that restrict the model's feasible region, thus reducing computational time of PSAM, the apron constraints, in contrast, introduce coupling constraints to PSAM, thus significantly increasing the complexity of the model.

The PSAM-RTA was applied at Lisbon airport for the summer season of 2014 and 2015. We observed that, if apron and terminal constraints are not considered, PSAM-RTA is solved optimally in 5 hours and 2.5 days respectively for the Summer seasons of 2014 and 2015 (note that the summer season of 2015 had 4% more slots requested than the summer season of 2014, increasing the complexity of solving this problem). However, if the apron and terminal constraints are considered, then PSAM-RTA cannot be solved optimally in less than 7 days (with an optimality gap of 2%). This has motivated the development of a heuristic algorithm to solve PSAM-RTA.

The proposed heuristic is based on large-scale neighborhood search methods and combines a constructive heuristic, which provides an initial feasible solution in short computational times, and an improvement heuristic, which iteratively re-optimizes slot allocation by subdividing the slot requests into smaller subsets. The heuristic was applied at Lisbon airport in the Summer season of 2015. The major takeaways from this implementation are that: (i) our constructive heuristic can provide an initial feasible solution in 30 minutes for the case with apron and terminal constraints, while CPLEX finds the first feasible solution after 2 hours; (ii) the outputs of the constructive heuristic obtained in less than 30 minutes outperform the best ones obtained with CPLEX after 2 days. The corresponding optimally gaps are 3.5% vs 4.9%; (iii) the improvement heuristic reduces the optimality gap from 3.5% to 0.1% in 6 hours.

Finally, still in Chapter 4, an extensive sensitivity analyses to the heuristic parameters showed that the heuristic algorithm proposed performs better than more straightforward implementations of large-scale neighborhood search methods in this context, and that results are robust to a number of calibration parameters.

In Chapter 5, we have conducted a comprehensive and detailed analysis of the rules and procedures of the slot allocation process. We have identified and discussed a set of important issues to be investigated with the ultimate goal of making the slot allocation process more efficient, transparent and inclusive. Some of these issues are concerned with: (i) the objectives of slot allocation; (ii) the flexibility to give to the rules and priorities of the IATA guidelines (iii) the setting of the airport declared capacities (iv) the consideration of network effects in slot allocation; (v) the designation of airports as level 3. To support us in this study, we have relied on PSAM to perform sensitivity analyses to some of the issues identified. This enabled us to quantify the benefits of potential changes to some of the rules and procedures of the current slot allocation process. As far as we are concerned, the analysis performed in this chapter greatly surpass those previously reported in the literature, and provides valuable insights for the discussions that the International Civil Aviation Organization (ICAO) has encouraged on this subject.

For issue (i), we have shown that there is a trade-off between the different objectives of PSAM. For instance, we observed for the case of Porto airport, a reduction of 4% in the total displacement and 1% in the number of slots displaced, when an increase of 9% in the maximum displacement is allowed. This motivates the consideration of weighting approaches to evaluate different objectives of the slot allocation process in future research. For issue (ii), we have performed numerous sensitivity analysis to many changes in the IATA rules. Overall, we have observed that even small relaxations to these rules can lead to slot allocation results that match better the request of the airlines. This was observed for all the analysis performed, indicating that some flexibility to the IATA rules can be beneficial for the process. For instance, at the Porto airport, improvements of 45% in the total displacement of the change-to-historic slots were observed when 0.2% of the historic slots were allowed to be displaced by only 5 minutes. Another example, was the reduction of the total displacement when the slot allocation is not performed together for the full season, as it is today, but for shorter and more homogeneous (as far as demand is concerned) sub periods. In this case, an improvement of 29% in total displacement was observed at Porto airport. For issue (iii), we have solved PSAM at Porto airport for different levels of declared capacity. The results have demonstrated the strong impact of the declared capacities on the displacement of slot requests. Specifically, we observed that a simultaneous increase of only one flight movement to the declared capacities of Porto airport can reduce the total displacement by 52%. This underscores the importance of a well-defined declared capacity.

Regarding issues (iv) and (v), no sensitivity analyses were performed, but a discussion on the existing literature, and some guidelines to explore these issues in the future were presented. We have also introduced for the first time in the literature the concept of Level 4 airports, defined as those airports that are operating at saturation, or near- saturation, and where some flights need to be rejected (i.e.

not allocated to any slot). These airports face different types of decisions and underlying trade-offs concerning slot allocation and therefore should be treated in a different way.

Overall, we believe we have identified in Chapter 5 the main issues of the slot allocation process. We have taken some general conclusions with respect to the sensitivity analyses we have performed, and we have provided some orientations on how to address these issues in the future. Ultimately, we believe the insights provided in this chapter can greatly contribute to the ongoing quests for improvements to the slot allocation process that are being conducted by IATA and ACI.

In summary, over the course of this thesis, we have developed an optimization tool to support slot allocation procedures at coordinated airports. In relation to previous literature, we have developed an optimization model for airport slot allocation fully compliant with the IATA guidelines. The model can be efficiently implemented using exact methods of optimization at airports with twice the size of those previously ever considered. We have also developed a heuristic approach to solve this problem at the largest coordinated airport. Results have proven that the heuristic proposed can provide near-optimal solutions in a few hours at instances where direct CPLEX implementation cannot find the optimal solution after several days of computation. Finally, we have shown that the modelling approach developed in this dissertation can be used to explore the impact of potential changes in the rules and procedures of the current slot allocation process, and we have shown that even small changes to the existing rules have the potential to provide significant benefits to the process.

6.2 Future research plans

The methods and procedures developed in this thesis provide a methodological foundation to explore new questions in the field of airport slot allocation. In the next years, we intend to devote part of our research to continue working on this topic. Below, we present some future research plans.

- *Develop a multi-criteria analysis tool that integrates multiple objectives of slot allocation and the WSG secondary criteria in the decision-making process*: Existing models and algorithms of slot allocation are not yet capable to capture all considerations and local characteristics arising in slot coordination practice (e.g. other objectives, public service obligations, secondary criteria, variations in the application of secondary criteria from airport to airport, etc.) and the various considerations of the airlines (e.g., home-based airlines, network airlines, low-cost airlines). Therefore, there might be the opportunity for moving forward, enhancing existing models to make them more consistent with slot coordination practice. From a practical standpoint, existing optimization models can be used to generate alternative solutions and create dashboards for coordinators to assess various solutions at the slot level. The main research directions on this topic are: (i) the design of metrics that can capture the main objectives and secondary criteria of the slot allocation process (ii) the creation of mechanisms to generate good and representative solutions for different objectives without the need of generating the entire efficient frontier for all of them; (iii) the development of a multi-criteria analysis tool to support coordinators in the selection of their preferred solutions among a large number of potential ones.

- *Advance existing slot allocation algorithms to improve their computation performance and account for additional considerations:* The heuristic algorithm presented in this dissertation is capable to provide near-optimal solutions in about six hours of computation for Lisbon airport. Although this is a very acceptable time given the strategic nature of the process, this might be limitative if we aim to analyze several alternative solutions to account for different objectives and criteria. Therefore, there might be the need to further enhance the computation performance of the existing algorithms. Another important venue for future research is the implementation of the proposed algorithms at the largest slot coordinated airports in the world to test their scalability to large-size instances (e.g. Paris Charles de Gaulle and Amsterdam Schiphol airports). This might imply the consideration of additional aspects such as that some flights might need to be rejected, i.e. not allocated to any slot time due to lack of capacity. This aspect is already captured by PSAM, but not by the proposed algorithms.

- *Design and develop a software prototype to be embedded into real-world coordination practices:* Using the optimization methods and procedures proposed in this dissertation, we aim to design a slot allocation software to be used by slot coordinators in real-world practices. The main challenges of the creation of this software are the design of the interface and mechanisms of interactions between users and tool, and the development of the devices for output visualization, such as graphic representation and evaluation metrics, to quickly provide relevant information about each solution provided. The software must be prepared to be run as many times as needed by the slot coordinators in case they want to include any additional considerations that might be relevant for them. To accelerate the re-optimization of the software we may rely on local search techniques such as those presented in Chapter 4 as well as in population search techniques.

- *Perform a qualitative study based on interviews to main airport slot allocation stakeholders:* In order to propose and evaluate adjustments on existing slot allocation processes, it is crucial to have an extensive understanding on the several perspectives of the stakeholders. For that, we may rely on surveys to gather insights from the major airport stakeholders on their needs, and to assess their opinions regarding the current IATA process. Ultimately, we aim to gain a systematic and in-depth understanding of the main stakeholders' perceptions of the IATA slot allocation process, identifying the strengths, shortcomings and opportunities for improvements. The design of the survey and the methodologies to be used to analyze the information collected, represent the major challenges to explore on this research topic.

- *Further assess the implications of mild modifications in the current rules and procedures of the slot allocation process:* The analysis performed in Chapter 5 provide us already with a good understanding on the dynamics we might observe when mild modifications to the current rules of the slot allocation process are considered. However, the main conclusions obtained are very general and based on only one airport (Porto). Future research on this topic may focus on more granular sensitivity analyses to the IATA rules, and on the application of these analyses to other airports.
- *Develop a collaborative decision-making approach for network-wide conflict resolution in airport slot allocation:* An important venue for future research is the consideration of the network effects in the slot allocation process. In fact, scheduling decisions cannot be made exclusively at a single airport because adjustments in one airport will have an impact in other airports. For instance, if a flight from airport A to airport B needs to be rescheduled by 30 minutes because of slot availability constraints at airport A, then it is necessary to ensure the availability of a slot at airport B for this flight at its new arrival time. In recent years, some optimization models have emerged in the literature to deal with the slot allocation problem at the network level (Castelli et al. 2011a; Castelli et al. 2011b; Corolli et al. 2014, Pellegrini et al. 2017). However, these models are not yet compliant with the existing rules of the WSG (specifically the rule of the series of slots), as this increases significantly the complexity of the slot allocation problem. To deal with this problem, we propose an alternative approach, which consists on the decomposition of the problem across the different stages of the process. For that, we propose a collaborative approach model for airport slot allocation composed by three optimization models: (i) the PSAM, which solves the initial slot allocation problem individually for each airport (ii) the airline's model of slot reallocation, which reoptimizes the flight schedules of each individual airline taking into account the slots received from all the airports; (iii) the airline's model of slot trading, which optimizes slot exchanges among different airlines. The integration of these models in sequence

will address the network problem in an efficient way, while at the same time ensuring consistency with existing practices.

- *Develop a systematic decision-making tool to support airport authorities in setting the declared capacities and other coordination parameters:* Results for Porto show that increasing declared capacity by even one movement per hour and/or per 15 minutes can significantly reduce schedule displacement. At the same time, flight delays are also sensitive to airport utilization levels when airports operate close to capacity. Setting the appropriate declared capacity therefore involves a trade-off between delays and schedule displacement. To date no clear and transparent method is available to support airports entities declaring their capacities and coordination parameters. Therefore, an important venue for future research consists on the development of decision support tools to assist airport authorities in setting these capacities. This can be achieved by creating mechanisms that combine slot allocation tools (as those provided in this dissertation), with queuing models.

References

Agustín, A., Alonso-Ayuso, A., Escudero, L. F., and Pizarro, C. (2010). Mathematical Optimization models for Air Traffic Flow Management: A review. Stud. Inform. Univ., 8(2), 141-184.

Ahuja, R.K., Goodstein, J., Mukherjee, A., Orlin, J.B., and Sharma, D. (2007). A very large-scale neighborhood search algorithm for the combined through-fleet-assignment model. INFORMS Journal on Computing, 19(3), 416-428.

Ahuja, R.K., Ergun, Ö., Orlin, J.B., and Punnen, A.P. (2002). A survey of very large-scale neighborhood search techniques. Discrete Applied Mathematics, 123(1-3), 75-102.

Ahuja, R.K., Orlin, J.B., & Sharma, D. (2000). Very large-scale neighborhood search. International Transactions in Operational Research, 7(4-5), 301-317.

Andreatta, G., Brunetta, L., Odoni, A. R., Righi, L., Stamatopoulos, M. A., and Zografos, K. G. (1999). A set of approximate and compatible models for airport strategic planning on airside and on landside. Air Traffic Control Quarterly, 7(4), 291-317.

Ashford, N. J., Mumayiz, S., and Wright, P. H. (2011). Airport Engineering: Planning, Design and Development of 21st Century Airports. John Wiley & Sons.

Ashford, N., and Wright, P. H. (1992). Airport Engineering, 3rd. Edition, New York

Ball, M. O., Berardino, F., and Hansen, M. (2018). The use of auctions for allocating airport access rights. Transportation Research Part A: Policy and Practice, 114, 186-202.

Ball, M., Barnhart, C., Dresner, M., Hansen, M., Neels, K., Odoni, A., Peterson, E., Sherry, L., Trani, A., and Zou, B. (2010). Total Delay Impact Study. Technical report, National Center of Excellence for Aviation Operations Research, College Park, MD.

Ball, M., Donohue, G., and Hoffman, K. (2006). Combinatorial Auctions. MIT Press (Ch. Auctions for the Safe, Efficient, and Equitable Allocation of Airspace System Resources, pp. 507–538).

Barbo W.A., (1967). The Use of Queuing Models in Design of Baggage Claim Areas at Airports. Graduate report, Institute of Transportation and Traffic Engineer, University of California, Berkeley, CA.

Basso, L. and Zhang, A. (2010). Pricing vs. slot policies when airport profits matter. Transportation. Research. Part B: Methodological, 44 (3), 381-391.

Benlic, U. (2018). Heuristic search for allocation of slots at network level. Transportation Research Part C: Emerging Technologies, 86, 488-509.

Bertsimas, D., Frankovich, M. and Odoni, A. R. (2011). Optimal Selection of Airport Runway Configurations. Operations Research 59(6) 1407–1419.

Bertsimas, D. and Patterson, S. (1998). The Air Traffic Flow Management Problem with Enroute Capacities. Operations Research, 46 (3), 406-422.

Bertsimas, D. and Tsitsiklis, J. (1993). Simulated annealing. Statistical science, 8(1), 10-15.

Blumstein, A. (1959). The landing capacity of a runway. Operations Research, 7(6), 752-763.

Brander, J. R. G. and Cook, B. A. (1986). Air transport deregulation and airport congestion: the search for efficient solutions (No. 1094).

Brueckner J. (2009). Price vs. quantity-based approaches to airport congestion management. J. Pub. Econ. 93, 681-690.

Brueckner, J. (2002). Airport congestion when carriers have market power. The American Economic Review. 92 (5), 1357-1375.

Brunetta, L., Righi, L. and Andreatta, G. (1999). An operations research model for the evaluation of an airport terminal: SLAM (simple landside aggregate model). Journal of Air Transport Management, 5(3), 161-175.

Carlin, A. and Park, R. E. (1970). Marginal cost pricing of airport runway capacity. The American Economic Review, 60(3), 310-319.

Castelli L., Pellegrini P. and Pesenti R. (2011a). Ant Colony Optimization for Allocating Airport Slots. 2nd International Conference on Models and Technologies for ITS (MT-ITS), Leuven, Belgium, June 22-24.

Castelli, L., Pellegrini, P. and Pesenti, R. (2011b). Airport slot allocation in Europe: economic efficiency and fairness. International Journal of Revenue Management, 6(1-2), 28-44.

Cohen, J. P. and Coughlin, C. C. (2003). Congestion at airports: the economics of airport expansions. Review-Federal Reserve Bank of Saint Louis, 85(3), 9-26.

Cohen, J. P. (1997). Some issues in benefit-cost analysis for airport development. Transportation Research Record: Journal of the Transportation Research Board, (1567), 1-7.

Corolli L., Lulli G. and Ntaimo L. (2014). The Time Slot Allocation Problem under Uncertain Capacity. Transportation Research Part C - Emerging Technologies, 46, 16-29.

Cramton, P., Ball, M., Ausubel, L.M., Berardino, F., Donohue, G., Hansen, M. and Hoffman, K. (2007). Market-based alternatives for managing congestion at New York's LaGuardia airport. Presented at Airneth Annual Conference, The Hague, Netherlands.

Czerny A. and Zhang, A. (2014). Airport congestion pricing when airlines price discriminate. Transportation Research Part B: Methodological, 65 (1), 77-89.

Czerny A. and Zhang, A. (2011). Airport congestion pricing and passenger types. Transportation Research Part B: Methodological, 45 (3), 595-604.

Czerny A. (2010). Airport congestion management under uncertainty. Transportation Research Part B: Methodological, 44 (3), 371-380.

Czerny, A. I., Forsyth, P., Gillen, D. and Niemeier, H-M. (2008). Airport Slots: International Experiences and Options for Reform, Ashgate Publishing, Hampshire, U.K.

Daniel, J. (1995). Congestion pricing and capacity of large hub airports: a bottleneck model with stochastic queues. Econometrica, 62 (2), 327-370.

de Neufville, R. and Odoni A. R. (2013). Airport Systems: Planning, Design and Management, 2nd Edition, McGraw-Hill Educational, New York.

De Weck O. (2004). Multiobjective optimization: History and promise. In Invited Keynote Paper, GL2-2, The Third China-Japan-Korea Joint Symposium on Optimization of Structural and Mechanical Systems, Kanazawa, Japan (Vol. 2, p. 34).

European Commision (1993). Council of the European Communities, European Council Regulation (EEC) No. 95/93 of January 1993 on Common Rules for the Allocation of Slots at Community Airports. Official Journal of the European Union, L 14, 22.01.1993, p.1. Brussels, Belgium, 1993.

European Commision (2011) “Impact Assessment of Revisions to Regulation 95/93 – Final Report, Brussels.

EUROCONTROL, (2019). Network Operations Report 2018, Main Report, Edition Number 1, Brussels, Belgium.

EUROCONTROL, (2010). Planning for Delay: influence of flight scheduling on airline punctuality. Trends in Air Traffic, Volume 7, Brussels, Belgium.

FAA (2018). FAA operations & performance data. Available at: <https://aspm.faa.gov> (last accessed 07.11.18).

FAA (2016). Comparison of Air Traffic Management-Related Operational Performance: U.S./Europe. Federal Aviation Administration, Washington, DC.

FAA (2015). FACT3: Airport Capacity Needs in the National Airspace System, Washington, DC, USA.

FAA (2008), Congestion Management Rule for LaGuardia Airport, Docket No. FAA–2006–25709; Notice No. 08–04, 73 FED. REG. 20846.

Fairbrother, J. and Zografos K. (2018). On the development of a fair and efficient slot scheduling mechanism at congested airports, Transportation Research Board Annual Conference, Washington, DC.

Fukui H., 2010. An empirical analysis of slot trading in the Unites States. Transportation Research Part B. 44, 330-357

Gilbo, E. P. (1993). Airport capacity: Representation, estimation, optimization. Control Systems Technology, IEEE Transactions on, 1(3), 144-154.

Gillen D., Jacquillat A. and Odoni R. (2016). Airport demand management: The operations research and economics perspectives and potential synergies. Transportation Research Part A: Policy and Practice, 94, 495-513.

Glover, F. (1989). Tabu search – part I. ORSA Journal on Computing, 1(3), 190-206.

Grether, D. M., Isaac, R. M. and Plott, C. R. (1981). The allocation of landing rights by unanimity among competitors. The American Economic Review,71(2), 166-171.

Guiomard, C. (2018). Airport slots: Can regulation be coordinated with competition? Evidence from Dublin airport. Transportation Research Part A: Policy and Practice, 114, 127-138.

Gupta, S. (2010). Transient Analysis of D(t)/M(t)/1 Queuing System with Applications to Computing Airport Delays. Master's thesis, Massachusetts Institute of Technology.

Hamzawi, S. G. (1992). Lack of airport capacity: Exploration of alternative solutions. Transportation Research Part A: Policy and Practice, 26(1), 47-58.

Hansen, M., Nikoleris, T., Lovell, D., Vlachou, K. and Odoni, A. R. (2009). Use of queuing models to estimate delay savings from 4D trajectory precision. In Eighth USA/Europe Air Traffic Management Research and Development Seminar.

Hansen, M. (2002). Micro-level analysis of airport delay externalities using deterministic queuing models: a case study. Journal of Air Transport Management, 8(2):73–87.

Harris, R. M. (1972) Models for runway capacity analysis, report FAA-EM-73-5, The MITRE Corporation McLean, VA

Harsha P. (2009). Mitigating Airport Congestion: Market Mechanisms and Airline Response Models. Ph.D. thesis, Massachusetts Institute of Technology.

Haylen, A. and Butcher, L. (2017). Airport Slots, Briefing Paper Number CBP 488, House of Commons Library, UK Parliament, 12 June 2017.

Hockaday, S. L. and Kanafani, A. K. (1974). Developments in airport capacity analysis. Transportation Research, 8(3), 171-180.

Horonjeff, R., McKelvey, F. X., Sproule, W. J. and Young, S. B. (2010). Planning and Design of Airports. The McGraw-Hill Companies. Inc.

IATA (2017). Worldwide Slot Guidelines, 8th Edition. Montreal, Canada.

IATA (2016a) "2036 Forecast Reveals Air Passengers Will Nearly Double to 7.8 Billion", Press Release No. 55, International Air Transport Association, Montreal, Canada.

IATA (2016b). Airport Slot Allocation, submitted to ICAO's Economic Commission, 39th Assembly, Working Paper A39-WP340, Montreal – Geneva.

IATA (2014) “Standard Schedules Information Manual (SSIM)”. International Air Transport Association, Montreal, Canada.

IATA (2004) “Airport Development Reference Manual, 9th cd., Montreal, Canada.

ICAO (2016). Report of the Economic Commission on Agenda Item 39, Report A39-WP/518, Montreal, Quebec, Canada.

Jacquillat, A. and Vaze, V. (2018). Interairline Equity in Airport Scheduling Interventions. Transportation Science, 52 (4), 941-964.

Jacquillat, A., Odoni, A. R. and Webster, M. D. (2016). Dynamic control of runway configurations and of arrival and departure service rates at JFK airport under stochastic queue conditions. Transportation Science, 51(1), 155-176.

Jacquillat A. and Odoni A. R. (2015a). An integrated scheduling and operations approach to airport congestion mitigation, Operations Research, 63 (6), 1390-1410.

Jacquillat, A. and Odoni, A. R. (2015b). Endogenous control of service rates in stochastic and dynamic queuing models of airport congestion. Transportation Research Part E: Logistics and Transportation Review, 73, 133-151.

Janic, M. (2009). Airport Analysis, Planning and Design: Demand, Capacity, and Congestion (Transportation Infrastructure-Roads, Highways, Bridges, Airports and Mass Transit Series). UK: Nova Science Publishers Inc.

Janic, M. (2000). Air Transport System Analysis and Modelling (Vol. 16). CRC Press.

Jorge, J. D. and de Rus, G. (2004). Cost–benefit analysis of investments in airport infrastructure: a practical approach. Journal of Air Transport Management, 10(5), 311-326.

Khadilkar, H. and Balakrishnan, H. (2014). Network congestion control of airport surface operations. Journal of Guidance, Control, and Dynamics, 37(3), 933-940.

Kivestu, P. (1976). Alternative Methods of Investigating the Time Dependent M/G/k Queue. Master’s thesis, Massachusetts Institute of Technology.

Klotz, E. and Newman, A. M. (2013). Practical guidelines for solving difficult mixed integer linear programs. Surveys in Operations Research and Management Science, 18(1-2), 18-32.

Kösters, D. I. D. (2007). Study on the usage of declared capacity at major German airports. Study in cooperation with Eurocontrol's Performance Review Unit (PRU), Aachen, Germany.

Kwan, I. and Hansen, M., (2010). US flight delay in the 2000's: an econometric analysis. In: 90th Transportation Research Board Annual Meeting.

Kytöjoki, J., Nuortio, T., Bräysy, O. and Gendreau, M. (2007). An efficient variable neighborhood search heuristic for very large scale vehicle routing problems. Computers & Operations Research, 34(9), 2743-2757.

Larson, R. C. and Odoni, A. R. (1981). Urban operations research (No. Monograph).

Lee, A. M. (1966). Applied Queuing Theory. MacMillian and Co. Ltd., London

Lee, D., Kostiuk, P., Hemm, R., Wingrove, W. and Shapiro, G. (1997) Estimating the effects of the terminal area productivity program, Report NS301R3, Logistics Management Institute, McLean, VA.

Levine, M.E. (1969). Landing fees and the airport congestion problem, Journal of Law and Economics 12, 79-108

Li, L. and Clarke J.P. (2010). A Stochastic Model of Runway Configuration Planning. AIAA Guidance, Navigation, and Control Conference.

Marler R. and Arora J. (2004). Survey of multi-objective optimization methods for engineering. Structural and Multidisciplinary Optimization, 26 (6), 369-395.

Meyers, C. and Orlin, J.B. (2006). Very large-scale neighborhood search techniques in timetabling problems. Practice and Theory of Automated Timetabling VI, Springer, 24-39.

Min, H., Melachrinoudis, E. and Wu, X. (1997). Dynamic expansion and location of an airport: A multiple objective approach. Transportation Research Part A: Policy and Practice, 31(5), 403-417.

Min, H. (1994). International supplier selection: a multi-attribute utility approach. International Journal of Physical Distribution & Logistics Management, 24(5), 24-33.

Mirkovic, B. and Tosic, V. (2014). Airport apron capacity: estimation, representation, and flexibility. Journal of Advanced Transportation, 48(2), 97-118.

Mirkovic, B. (2011). Airport apron capacity estimation–model enhancement. Procedia-Social and Behavioral Sciences, 20, 1108-1117.

Mladenović, N. and Hansen, P. (1997). Variable neighborhood search. Computers & Operations Research, 24(11), 1097-1100.

Morrison, S. A. and Winston, C. (2008). The effect of FAA expenditures on air travel delays. Journal of Urban Economics, 63(2), 669-678.

Morrison, S.A. (1983). Estimation of long-run prices and investment levels for airport runways, Research in Transport Economics 1, 103-130.

NERA (2004) "Study to Assess the Effects of Different Slot Allocation Schemes", Technical Report prepared for the European Commission (DG TREN), London, UK.

Newell, G. F. (1979). Airport capacity and delays. Transportation Science, 13(3), 201-241.

Newell, G.F. (1971). Application of Queuing Theory. Chapman and Hall, London.

Nikoleris, T. and Hansen, M. (2012). Queueing models for trajectory-based aircraft operations. Transportation Science, 46(4), 501-511.

Odoni, A. R., Bowman, J., Delahaye, D., Deyst, J. J., Feron, E., Hansman, R. J., ... and Simpson, R. W. (1997). Existing and required modeling capabilities for evaluating ATM systems and concepts.

Odoni, A. R. (1994). Issues in air traffic flow management. In Advanced Technologies for Air Traffic Flow Management (pp. 43-63). Springer Berlin Heidelberg.

Odoni, A. R. (1972). Efficient operation of runways. Publication of: American Society of Civil Engineers.

Paelinck, J. (1977). Qualitative multicriteria analysis: an application to airport location. Environment and Planning A, 9(8), 883-895.

Pellegrini, P., Bolić, T., Castelli, L. and Pesenti, R. (2017). SOSTA: An effective model for the Simultaneous Optimisation of airport SloT Allocation. Transportation Research Part E: Logistics and Transportation Review, 99, 34-53.

Pellegrini, P., Castelli, L., and Pesenti, R. (2012a). Secondary trading of airport slots as a combinatorial exchange. Transportation Research Part E: Logistics and Transportation Review, 48(5), 1009-1022.

Pellegrini, P., Castelli, L. and Pesenti, R. (2012b). Metaheuristic algorithms for the simultaneous slot allocation problem. IET Intelligent Transport Systems, 6(4), 453-462.

Pels E. and Verhoef E. (2004). The economics of airport congestion pricing. J. Urban Econ. 55, 257-277.

Pisinger, D. and Ropke, S. (2010). Large neighborhood search. In Handbook of Metaheuristics (pp. 399-419). Springer, Boston, MA.

Pyrgiotis N. and Odoni A.R. (2016). On the impact of scheduling limits: A case study at Newark International Airport. Transportation Science, 50 (1), 150-165.

Pyrgiotis N., Malone, K. and Odoni A.R. (2011). Modelling delay propagation within an airport network. Transportation Research Part C, 27, 6-75.

Rallis, T. (1967). Capacity of Transport Centers, Railroad Stations, Road Haulage Centers and Airports. Report 25 of the Department for Road Construction, Transportation Engineering and Town Planning, The Technical University of Denmark, Copenhagen.

Rallis, T. (1963). Terminal Transportation Engineering I, Airports. Report 27 of the Department for Road Construction, Transportation Engineering and Town Planning, The Technical University of Denmark, Copenhagen.

Rallis, T. (1958). The capacity of airports. Ingenioren International Edition, 2(89), 102

Rassenti, S., Smith, V. and Bulfin, R., (1982). A combinatorial auction mechanism for airport time slot allocation. The Bell Journal of Economics, 13 (2), 402–417

Ribeiro, N.A., Jacquillat, A., Antunes, A.P., Odoni, A. R. and Pita, J. P. (2018). An optimization approach for airport slot allocation under IATA guidelines. Transportation Research Part B: Methodological, 112, 132-156.

Ropke, S. and Pisinger, D. (2006). An adaptive large neighborhood search heuristic for the pickup and delivery problem with time windows. Transportation Science, 40(4), 455-472.

Saatcioglu, O. (1982). Mathematical programming models for airport site selection. Transportation Research Part B: Methodological, 16(6), 435-447.

Santos, M. G. and Antunes, A. P. (2015). Long-term evolution of airport networks: Optimization model and its application to the United States. Transportation Research Part E: Logistics and Transportation Review, 73, 17-46.

Shaw, P. (1997). A new local search algorithm providing high quality solutions to vehicle routing problems. APES Group, Department of Computer Science, University of Strathclyde, Glasgow, Scotland, UK.

Simaiakis, I., Khadilkar, H., Balakrishnan, H., Reynolds, T. G. and Hansman, R. J. (2014). Demonstration of reduced airport congestion through pushback rate control. Transportation Research Part A: Policy and Practice, 66, 251-267.

Simaiakis, I. (2012). Analysis, Modeling and Control of the Airport Departure Process. Ph.D. thesis, Massachusetts Institute of Technology.

Simpson, R. W. (1980) “Airside Capacity and Delay at Major U.S. Airports,” draft report prepared for the Office of Technology Assessment, U.S. Congress, Washington, D. C.

Stamatopoulos, M. A., Zografos, K. G. and Odoni, A. R. (2004). A decision support system for airport strategic planning. Transportation Research Part C: Emerging Technologies, 12(2), 91-117.

Starkie, D. (1998). Allocating airport slots: A role for the market?. Journal of Air Transport Management, 4(2), 111-116.

Steuer, R. (1986). Multiple Criteria Optimization: Theory, Computation, and Application. Wiley, New York.

Swaroop P. B., Zou M., Ball M. and Hansen M. (2012). Do More US Airports Need Slot Controls? A Welfare Based Approach to Determine Slot Levels. Transportation Research Part B: Methodological, 46(9), 1239-1259.

Swedish, W. J. (1981). Upgraded FAA Airfield Capacity Model. Volume 2. Technical Description of Revisions (No. MTR-81W16-VOL-2).

Tosic V., Babic O. and Janic M., (1983). Airport passenger terminal simulation (in Serbo-Croatian). Annales of Operations Research in Air Transportation, Faculty of Transport and Traffic Engineering, University of Belgrade, 83-100.

Transportation Research Board (TRB), (1987). Special Report 215: Measuring Airport Landside Capacity. TRB, National Research Council, Washington, D.C.

Ulrich, C. (2008) "How the Present (IATA) Slot Allocation Works". In Airport slots: international experiences and options for reform, A.I. Czerny, P. Forsyth, D. Gillen and H.M. Niemeier, Eds. Ashgate, pp. 9-20.

Valdes, V., and Gillen, D. (2018). The consumer welfare effects of slot concentration and reallocation: A study of Mexico City International Airport. Transportation Research Part A: Policy and Practice, 114, 256-269.

Vaze, V. and Barnhart C. (2012). Modeling Airline Frequency Competition for Airport Congestion Mitigation Airline frequency competition in airport congestion pricing. Transportation Science, 2266 (69-77) 46 (4), 512-535.

Verhoef, E. (2010). Congestion pricing, slot sales and slot trading in aviation. Transportation Research Part B: Methodological, 44 (3), 320-329.

Vickrey, W. S. (1969). Congestion theory and transport investment. The American Economic Review, 59(2), 251-260.

Vreeker, R., Nijkamp, P. and Ter Welle, C. (2002). A multicriteria decision support methodology for evaluating airport expansion plans. Transportation Research Part D: Transport and Environment, 7(1), 27-47.

Weld, C., Duarte, M. and Kincaid, R. (2010). A Runway Configuration Management Model with Marginally Decreasing Transition Capacities. Advances in Operations Research .

Wells, A. T. and Young, S. B. (2004). Airport Planning & Management. 5th edition.

Xu, N. (2007). Method for deriving multi-factor models for predicting airport delays. PhD thesis, George Mason University.

Yu, C., Zhang, D. and Lau, H. Y. (2017). An adaptive large neighborhood search heuristic for solving a robust gate assignment problem. Expert Systems with Applications, 84, 143-154.

Zografos G., Madas A. and Androutsopoulos N. (2017). Increasing airport capacity utilisation through optimum slot scheduling: review of current developments and identification of future needs. Journal of Scheduling, 20 (1), 3-24.

Zografos G. and Jiang Y. (2016). Modelling and Solving the Airport Slot Scheduling Problem with Efficiency, Fairness, and Accessibility Considerations. 9th Triennial Conference on Transportation Analysis (TRISTAN 2016), June 13-17, 2016, Oranjestad, Aruba.

Zografos, K. G., Salouras, Y. and Madas, M. A. (2012). Dealing with the efficient allocation of scarce resources at congested airports. Transportation Research Part C: Emerging Technologies, 21(1), 244-256.

Zografos, K. and Madas, M. (2003). Critical Assessment of Airport Demand Management Strategies in Europe and the United States: Comparative Perspective. Transportation Research Record: Journal of the Transportation Research Board, (1850), 41-48.

Zou, B. and Hansen, M. (2012). Impact of operational performance on air carrier cost structure: evidence from US airlines. Transportation Research Part E: Logistics and Transportation Review, 48(5), 1032-1048.

Appendix

In this appendix we present the Jacquillat and Odoni (2015a) model, also named Integrated Capacity Utilization and Scheduling Model (ICUSM). The ICUSM is composed of three different models: an integer programming model of scheduling interventions; a dynamic programming model of capacity utilization; and a stochastic queuing model of airport congestion. They are described below as well as the algorithm used to solve the integrated model.

a) Model of Scheduling Interventions

The model of scheduling interventions is similar to the Pyrgiotis and Odoni (2016) model, but a new formulation is introduced. Moreover, this model is applied to a single airport *K*, however all flights that are flown by an aircraft that visits *K* during the day are also included into the model. For instance, if an aircraft operates the itinerary $K \to L \to M \to K$, then the flight leg $L \to M$ is also included. This is because rescheduling a flight leg in this itinerary might require a change in the schedule time of flight leg $L \to M$.

Before presenting the model formulation, the notation used to represent the sets, parameters and decision variables is introduced.

Sets

$\boldsymbol{T} = \{1,...,N\}$: set of time periods, indexed by *t*;

$\boldsymbol{F} = \{1,...,F\}$: set of flights, indexed by *i* or *j*;

$\boldsymbol{F}_{arr}$ / $\boldsymbol{F}_{dep}$: set of flights $i \in F$ scheduled to arrive / depart at the coordinated airport;

$\boldsymbol{P} \subset F \times F$: set of flight pairs $(i, j) \in F \times F$ such that there is a connection between *i* and *j*.

Parameters

$$A_{it}^{dep} \,/\, A_{it}^{arr} = \begin{cases} 1, & \text{if flight } i \text{ is scheduled to arrive / depart no earlier than period } t \\ 0, & \text{otherwise} \end{cases} \quad ;$$

$T_{ij}^{\min}$ / $\mathrm{T}_{ij}^{\max}$ = minimum / maximum connection time between flight *i* and *j*.

Decision Variables

X_i = number of periods that flight i is displaced ;

$$Y_{it}^{arr} / Y_{it}^{dep} = \begin{cases} 1, & \text{if flight } i \text{ is rescheduled to arrive / depart no earlier than period } t \\ 0, & \text{otherwise} \end{cases} ;$$

$\lambda_t^X / \lambda_t^Y$ = number of scheduled arrivals / departures in period t ;

δ / Δ = maximal flight displacement / total scheduled displacement .

The main novelty of this model, as compared to the one presented by Pyrgiotis and Odoni (2016), lies in the form of the scheduling parameters $A_{it}^{dep} / A_{it}^{arr}$. In this case, these parameters are 1 if flight i is scheduled to depart / arrive no earlier than period t, e.g. (1,…,1,0,…,0). By contrast in Pyrgiotis and Odoni (2016) these parameters are 1 if flight i is scheduled to depart / arrive in period t, e.g. (0,…,0,1,0,…,0). The decision variables $Y_{it}^{arr} / Y_{it}^{dep}$ take the same form as $A_{it}^{dep} / A_{it}^{arr}$, since they are counterparts of these parameters. Using the notation above, the model can be formulated as follows:

Objective Function:

$$lex\min(\delta, \Delta) \tag{A.1}$$

Subject to:

$$Y_{i1}^{arr} = 1 \qquad \forall i \in \boldsymbol{F} \tag{A.2}$$

$$Y_{i1}^{dep} = 1 \qquad \forall i \in \boldsymbol{F} \tag{A.3}$$

$$\sum_{t \in \boldsymbol{T}} \left(Y_{it}^{arr} - A_{it}^{arr} \right) = X_i \qquad \forall i \in \boldsymbol{F} \tag{A.4}$$

$$\sum_{t \in \boldsymbol{T}} \left(Y_{it}^{dep} - A_{it}^{dep} \right) = X_i \qquad \forall i \in \boldsymbol{F} \tag{A.5}$$

$$\sum_{t \in \boldsymbol{T}} \left(Y_{jt}^{dep} - Y_{it}^{arr} \right) \geq T_{ij}^{\min} \qquad \forall i, j \in \boldsymbol{P} \tag{A.6}$$

$$\sum_{t \in \boldsymbol{T}} \left(Y_{jt}^{dep} - Y_{it}^{arr} \right) \leq T_{ij}^{\max} \qquad \forall i, j \in \boldsymbol{P} \tag{A.7}$$

$$Y_{it}^{arr} \geq Y_{i,t+1}^{arr} \qquad \forall i \in \boldsymbol{F}, t \in \boldsymbol{T} \tag{A.8}$$

$$Y_{it}^{dep} \geq Y_{i,t+1}^{dep} \qquad \forall i \in \boldsymbol{F}, t \in \boldsymbol{T} \tag{A.9}$$

$$\sum_{i \in \boldsymbol{F}^{arr}} \left(Y_{it}^{arr} - Y_{i,t+1}^{arr} \right) = \lambda_t^X \qquad \forall t \in \boldsymbol{T} \qquad \text{(A.10)}$$

$$\sum_{i \in \boldsymbol{F}^{dep}} \left(Y_{it}^{dep} - Y_{i,t+1}^{dep} \right) = \lambda_t^Y \qquad \forall t \in \boldsymbol{T} \qquad \text{(A.11)}$$

$$\left| X_i \right| \leq \delta \qquad \forall i \in \boldsymbol{F} \qquad \text{(A.12)}$$

$$\sum_{i \in \boldsymbol{F}} \left| X_i \right| = \Delta \qquad \forall i \in \boldsymbol{F} \qquad \text{(A.13)}$$

The objective function (A.1) is a two-stage lexicographic objective. Basically, it does the same as expression (2.6). First, the maximal displacement is minimized, and then, among all feasible schedules under this first objective, the model selects the one which minimizes the total displacement.

Constraints (A.2) and (A.3) ensure that no flight is eliminated. Constraint (A.4) and (A.5) ensure that the departure time and arrival time of any given flight is always displaced by the same amount. Constraints (A.6) and (A.7) forces the connection times to be greater than the minimum connection time and smaller than the maximum connection time imposed. Constraints (A.8) and (A.9) ensure that variables Y_{it}^{arr} and Y_{it}^{dep} are non-increasing, as considered in its definition. Constraint (A.10) and (A.11) define the aggregate number of scheduled arrivals and departures per unit of time. Finally, Constraints (A.12) and (A.13) define the maximal displacement and the total displacement of flights.

This model does not take into account any capacity limits. Instead it considers queue length reduction constraints to limit λ_t^X and λ_t^Y. These constraints are presented with the algorithm. If capacity constraints are added to this model, the model's outcome will be the same as the one obtained with the same parameters in Pyrgiotis and Odoni (2016).

b) Model of Capacity Utilization

The model of capacity utilization used in the ICUSM is a dynamic programming model, which optimizes the sequential control of capacity utilization procedures, minimizing congestion costs subject to capacity constraints. It takes as inputs the modified schedule of flights obtained through the model of scheduling interventions, and the Throughput Capacity Envelopes for each runway configuration.

This model is a simplification of the tactical capacity utilization model presented by Jacquillat and Odoni (2015b), which optimizes the capacity utilization policies over the course of each day of operations. The computational requirements of this tactical model prevent it from being used repeatedly with different flight schedules or different capacity estimates, and consequently limits its applicability to the ICUSM.

The simplified model is obtained by grouping runway configurations into clusters of similar configuration and then estimating the average Throughput Capacity Envelop for each one of the clusters. Moreover, it assumes that the schedule of use of runway configurations is exogenously determined in advance, obtained from the tactical capacity utilization model (Jacquillat and Odoni 2015b) when applied to a representative schedule of flights and from the actual patterns of runway configuration usage at the airports. These simplifications can capture well the trade-off between arrival and departure service rates and approximates the actual selection of runway configurations.

The resulting model is formulated as follows. At each period $t = 1,...,T$, the decision maker observes (i) the arrival queue length at the end of period $t-1$ (i.e. $a_{t-1} \in \{0,...,N\}$), (ii) the departure queue length at the end of period $t-1$ (i.e. $d_{t-1} \in \{0,...,N\}$), and (iii) the weather state (i.e., $w_t \in \{VMC, IMC\}$. The runway configuration cluster for each period *t*, denoted by RC_t is given. The decision-maker selects the arrival service rate for period *t*, (i.e. $\mu_t^a \in \{0,...,\Gamma_{RC_t,w_{ct}}\}$. Once the arrival rate is selected, the departure service rate μ_t^d is determined by the Throughput Capacity Envelope. Congestion costs are assumed to depend quadratically on arrival and departure queue lengths, thus the objective function is expressed as $\alpha \sum_{t=1}^{T} a_t^2 + \sum_{t=1}^{T} d_t^2$, where α intends to capture the potentially larger costs of arrival delays when compared with departure delays. The Bellman equation is formulated as follows, where $J_t\left(a_{t-1}, d_{t-1}, w_t\right)$ represents the cost-to-go of being in state $\left(a_{t-1}, d_{t-1}, w_t\right)$ at the beginning of period *t*.

$$J_t\left(a_{t-1}, d_{t-1}, w_t\right) = \min_{\mu_t^a \in \left[0, \Gamma_{RC_t, w_t}\right]} \left(E\left[a_t^2\right] + E\left[d_t^2\right] + E\left[J_{t+1}\left(a_t, d_t, w_{t+1}\right)\right]\right), \quad \forall t = 1,...,T_0 \quad \text{(A.14)}$$

c) Model of Airport Congestion

The model of airport congestion is a stochastic queuing model, which takes as inputs the demand rates determined by the model of scheduling interventions and the service rates determined by the model of capacity utilization, returning the probabilistic evolution of arrivals and departure queue lengths over the day, A_t and D_t .

As explained in Chapter 2.3, an airport may be characterized as a queuing system, where aircraft join the queue when they are ready to land or depart, and service is provided by the runway system. The arrival and departure queue may be modeled as two distinct $M(t)/E_k(t)/1$ queuing systems, where demand and arrival processes are respectively modeled as Poisson processes and Erlang processes of order *k*.

d) Iterative Solution Algorithm

The iterative solution algorithm proposed by Jacquillat and Odoni (2015a) aims to integrate the models presented in order to solve the ICUSM. Essentially, the algorithm starts creating modified schedules by solving the model of scheduling interventions. Then, with the modified schedules obtained, it approximates the optimal capacity utilization policies by solving the capacity utilization model. Finally, it simulates the resulting delays using the stochastic queuing model and compares its results with the maximum queue lengths previously established, $A_{\max}$ and $D_{\max}$. By iterating among these three steps, the optimal schedule of flights and optimal capacity utilization policies are determined by minimizing the schedule displacement according to equation (A.1)

Since, the stochastic queuing model cannot be easily integrated into the integer programming model of scheduling interventions because it would transform it in a nonlinear programming model, Jacquillat and Odoni (2015a) integrates a deterministic queue model into the model of scheduling interventions aiming to select the modified schedules that minimize the expected peak delays. It relies on the assumption that given two distinct schedules of flights, the one that leads to smallest peak deterministic delays will also lead to the smallest peak expected stochastic delays, even though stochastic queuing models usually lead to significantly smaller delays estimates than stochastic queuing models (Hansen et al. 2009, Nikoleris and Hansen 2012, Jacquillat and Odoni, 2015a).

In order to integrate the deterministic queuing model into the model of scheduling interventions, new sets, parameters, decision variables and constraints need to be included into the model. With the aim of simplify the presentation of the model, the VMC and IMC indices are omitted, however all sets and variables are defined under both VMC and IMC, which represent all VMC days and all IMC days.

Sets

$\boldsymbol{S_t}$ = set of linear segments of the Operational Throughput Envelope of the runway configuration in use during period *t*

Parameters

$\alpha_s, \beta_s, \lambda_s \geq 0$ parameters that define each segment of the Operational Throughput Envelope

Decision Variables

μ_t^a / μ_t^d = arrival / departure service rate selected during period *t*

A_t^a / D_t = arrival / departure queue length at the end of period *t*

Subject to:

$$\alpha_s \mu_t^a + \beta_s \mu_t^d \leq \gamma_s \qquad \forall t \in \boldsymbol{T}, s \in \boldsymbol{S}_t \qquad \text{(A.15)}$$

$$A_t = A_{t-1} + \lambda_t^X - \mu_t^a \qquad \forall t \in \boldsymbol{T} \qquad \text{(A.16)}$$

$$D_t = D_{t-1} + \lambda_t^X - \mu_t^a \qquad \forall t \in \boldsymbol{T} \qquad \text{(A.17)}$$

Constraints (A.15) ensure that the service rates lie within the bounds defined by the Operational Throughput Envelope. Constraints (A.16) and (A.17) define the deterministic queue dynamics. At any period *t*, the arrival / departure queue length is equal to the arrival/departure queue length in the previous period $t-1$, plus the arrival / departure flights scheduled for period *t*, minus the number of arrival / departure flights serviced in period *t.*

The objective of the first step of the algorithm is to minimize the expected queue length, and so the objective function of the model of scheduling interventions is replaced by a measure $q_{\max}$.

Objective Function:

$$\text{Minimize } q_{MAX} \qquad \text{(A.18)}$$

$$q_{MAX} := M_2 q_{MAX}^{VMC} + q_{MAX}^{IMC} \qquad \text{(A.19)}$$

$$q_{MAX}^{VMC} := M_1 \times \max\left(\frac{1}{A_{MAX}} \max_{t \in \boldsymbol{T}} A_t^{VMC}, \frac{1}{D_{MAX}} \max_{t \in \boldsymbol{T}} D_t^{VMC} \right) + \sum_{t \in \boldsymbol{T}} \left(\frac{A_t^{VMC}}{A_{\max}} + \frac{D_t^{VMC}}{D_{\max}} \right) \qquad \text{(A.20)}$$

$$q_{MAX}^{IMC} := M_1 \times \max\left(\frac{1}{A_{MAX}} \max_{t \in \boldsymbol{T}} A_t^{IMC}, \frac{1}{D_{MAX}} \max_{t \in \boldsymbol{T}} D_t^{IMC} \right) + \sum_{t \in \boldsymbol{T}} \left(\frac{A_t^{IMC}}{A_{\max}} + \frac{D_t^{IMC}}{D_{\max}} \right) \qquad \text{(A.21)}$$

The objective function (A.18) represents a lexicographic objective, captured by the very large values M_1 and M_2. First, the model minimizes a measure of peak deterministic delays that are incurred during all VMC days, and after, among all schedules obtained, the model selects the one that minimizes the total queue length (see Equation A.19). For a large schedule displacement, it might be possible to keep scheduling levels within the bounds defined by the VMC Operational Throughput Envelope, obtaining null VMC queue lengths. In that case the model minimizes a measure of deterministic delays during all IMC days (Equation A.20). The arrival / departure queue length are normalized by a factor A_{MAX} / D_{MAX}, the purpose is to capture the relative cost of increasing the expected arrival / departure queue length in relation to the target levels A_{MAX} / D_{MAX}.

With the resulting schedule of flights obtained through the model of scheduling interventions, the algorithm determines the optimal control of arrival and departure service rates and simulate delays under stochastic queue dynamics. The peak expected delays obtained in the last stage of the algorithm will define if the solution is feasible or not. In order to take into account the objective function (A.1), the algorithm is subdivided in two. The first one computes the modified schedules, minimizing the maximal displacement, and the second one selects, among all the alternatives solutions, the one that minimizes the total displacement.

Algorithm 1 Determination of the optimal maximal displacement δ^*

Initialization: $\underline{\delta} = 0$, $z_{\text{end}} = 0$

while $z_{\text{end}} = 0$ **do**

Solve the model of scheduling interventions, under *deterministic* queue dynamics $\longrightarrow \lambda_t^X / \lambda_t^Y, \forall t$
  minimize q_{MAX} (Equation 2.60)
  subject to Scheduling constraints: Equations (2.44) to (2.55)
  Deterministic queuing constraints: Equations (2.58) to (2.59)
  Maximal displacement: $\delta = \underline{\delta}$

$\Delta_0 \leftarrow \sum_{i \in \mathcal{F}} |u_i|$

Solve the model of capacity utilization (Equation(2.57)$\longrightarrow \mu_t^a(a_{t-1}, d_{t-1}, w_t) / \mu_t^d(a_{t-1}, d_{t-1}, w_t), \forall t$

Solve the model of airport congestion $\longrightarrow A_t, D_t, \forall t$

  if $A_t \le A_{\text{MAX}}, \forall t$ and $D_t \le D_{\text{MAX}}, \forall t$ **then** $z_{\text{end}} \leftarrow 1$

  else $\underline{\delta} \leftarrow \underline{\delta} + 1$

  end if

end while

$\delta^* \leftarrow \underline{\delta}$

Return δ^*, Δ_0

Algorithm 1 initializes the process considering the maximal displacement $\underline{\delta}$ equal to 0. If the queue length targets are met $A_t \le A_{\max}, \forall t \in \boldsymbol{T}$ and $D_t \le D_{\max}, \forall t \in \boldsymbol{T}$, then the optimal maximal displacement δ^* is 0, otherwise the maximal displacement is increased to $\underline{\delta} = 1$. This process is repeated until the targets are met. The total displacement that minimizes q_{max} for a maximal displacement equal to δ^* is denoted by Δ_0, which will be used in algorithm 2.

Algorithm 2 is a dichotomy algorithm. It initializes by setting the upper bound $\overline{\Delta}$ equal to 0, which corresponds to the situation where no flight is displaced, and the lower bound $\underline{\Delta}$ equal to Δ_0, which provides the smallest peak queue length possible. At each iteration, the algorithm considers a tentative value of the total displacement at the midpoint of $\overline{\Delta}$ and $\underline{\Delta}$, denoted by Δ^{try}. It calculates the modified schedule that minimizes peak deterministic delays and for that schedule simulates the stochastic delays. If the queue length targets are met, the optimal total displacement is at most equal to Δ^{try}, replacing $\overline{\Delta}$ by Δ^{try}, otherwise, the optimal total displacement is larger than Δ^{try}, and $\underline{\Delta}$ is replaced by Δ^{try}. This process is repeated until $\overline{\Delta}$ and $\underline{\Delta}$ have converged to the same values, which represents the optimal total displacement denoted by Δ^*.

Algorithm 2 Determination of the optimal total displacement Δ^*, given the optimal maximal displacement δ^*

Initialization: $\underline{\Delta} = 0,\ \overline{\Delta} = \Delta_0,\ z_{\text{end}} = 0$

while $z_{\text{end}} = 0$ **do**

$\Delta^{\text{try}} \leftarrow \frac{\overline{\Delta}+\underline{\Delta}}{2}$

Solve the model of scheduling interventions, under *deterministic* queue dynamics $\longrightarrow \lambda_t^X/\lambda_t^Y, \forall t$

minimize q_{MAX} (Equation 2.60)

subject to Scheduling constraints: Equations (2.44) to (2.55)

Deterministic queuing constraints: Equations (2.58) to (2.59)

Maximal displacement: $\delta = \delta^*$

Total displacement: $\Delta = \Delta^{\text{try}}$

Solve the model of capacity utilization (Equation (2.57))$\longrightarrow \mu_t^a(a_{t-1}, d_{t-1}, w_t)/\mu_t^d(a_{t-1}, d_{t-1}, w_t), \forall t$

Solve the model of airport congestion $\longrightarrow A_t, D_t, \forall t$

if $A_t \leq A_{\text{MAX}}, \forall t$ and $D_t \leq D_{\text{MAX}}, \forall t$ **then** $\overline{\Delta} \leftarrow \Delta^{\text{try}}$

else $\underline{\Delta} \leftarrow \Delta^{\text{try}}$

end if

if $\underline{\Delta} = \overline{\Delta}$ **then** $z_{\text{end}} \leftarrow 1$

else $z_{\text{end}} \leftarrow 0$

end if

end while

Return Δ^*

As already stated, the iterative algorithm relies on the assumption that given two distinct schedules of flights, the one that leads to smallest peak deterministic delays will also lead to the smallest peak expected stochastic delays. However, this sometimes may not be true, which may introduce an error in the final solution. Nevertheless, these errors are expected to be small and unusual and so, the modified schedule of flights obtained will be very close to the optimal solution.