A market mechanism for multiple air traffic resources

Irene Brugnara[a], Lorenzo Castelli[a,*], Raffaele Pesenti[b]

[a] *Università degli Studi di Trieste, Trieste, Italy*
[b] *Università Ca' Foscari di Venezia, Venezia, Italy*

Highlights

- Our model optimises the cost of delay in multiple ATFM regulations while preserving the confidentiality of costs for airlines
- Our market mechanism is individual rational, weakly budget balanced, and allocative efficient
- We apply Lagrangean relaxation and the subgradient method to an integer programming problem
- All numerical experiments are performed on real traffic data

*Corresponding author
Email address: castelli@units.it (Lorenzo Castelli)

Title Page (with Author Details)

A market mechanism for multiple air traffic resources

Irene Brugnara[a], Lorenzo Castelli[a,*], Raffaele Pesenti[b]

[a] *Università degli Studi di Trieste, Trieste, Italy*
[b] *Università Ca' Foscari di Venezia, Venezia, Italy*

Abstract

We introduce a model that extends the concept of air traffic flow management slot to the concept of time window, allowing to effectively deal with a network of interacting regulations. The model aims at minimising the total cost of delay of a time window allocation to flights and is based on an integer programming problem. It consists in a market-based mechanism between flights and a central authority to trade time windows, which fulfils the properties of individual rationality (every participating airline has a non-negative profit from the mechanism) and weak budget-balance (the mechanism requires no external subsidisation). Equity is assumed to be respected because the First Planned First Served allocation as an endowment guaranteed to all flights and allocated for free. The proposed market mechanism can be implemented in a distributed manner preventing the disclosure of confidential information by airlines, and is based on the Lagrangian relaxation of the integer optimisation problem, solved through the subgradient algorithm. We present some computational experiments conducted to test the model on some real instances of air traffic data.

Keywords: Air Traffic Management (ATM), Air Traffic Flow Management (ATFM), ATFM regulation, subgradient algorithm, market mechanism, pricing equilibrium.

*Corresponding author
Email address: castelli@units.it (Lorenzo Castelli)

Manuscript (without Author Details) Click here to view linked References

A market mechanism for multiple air traffic resources

Abstract

We introduce a model that extends the concept of air traffic flow management slot to the concept of time window, allowing to effectively deal with a network of interacting regulations. The model aims at minimising the total cost of delay of a time window allocation to flights and is based on an integer programming problem. It consists in a market-based mechanism between flights and a central authority to trade time windows, which fulfils the properties of individual rationality (every participating airline has a non-negative profit from the mechanism) and weak budget-balance (the mechanism requires no external subsidisation). Equity is assumed to be respected because the First Planned First Served allocation is an endowment guaranteed to all flights and allocated for free. The proposed market mechanism can be implemented in a distributed manner preventing the disclosure of confidential information by airlines, and is based on the Lagrangian relaxation of the integer optimisation problem, solved through the subgradient algorithm. We present some computational experiments conducted to test the model on some real instances of air traffic data.

Keywords: Air Traffic Management (ATM), Air Traffic Flow Management (ATFM), ATFM regulation, subgradient algorithm, market mechanism, pricing equilibrium.

1. Introduction

Air traffic resources consisting in airports and airspace volumes have a limited capacity in terms of number of aircraft that can enter the resource in a given period of time. The factor determining capacity is the amount of traffic that can be safely handled by air traffic controllers. In the current European Air Traffic Flow Management (ATFM) system, when an imbalance between traffic demand and available capacity is foreseen in an airport or airspace volume, the Network Manager (NM) can impose an ATFM regulation, which limits the rate of aircraft that can enter the regulated resource in a given period of time. This ATFM measure is achieved by delaying the departure of flights from their origin airport. ATFM delays are imposed to flights through an ATFM slot, a 15-minute tolerance time interval that flights have to comply with for departure. In the current system, ATFM slots are allocated to flights according to a First Planned First Served (FPFS) principle.

Besides a better management of ATC sectors through configurations or splitting of sectors, alternatives to ATFM regulations to resolve congestion are for example re-routing of traffic flows. These choices, however, "have a negative impact on the environment due to longer routes and/or sub-optimal altitude profiles" (Dalmau, 2022). ATFM regulations also have a negative economic impact because ATFM delays represent a cost for airlines: pre-pandemic estimations report 1.93B€ in 2018 and 1.76 B€ in 2019 (Performance Review Commission, 2019). Recently, new quantitative and qualitative indicators to assess the expected impact of the costs ATFM regulations would impose on airspace users have been defined (Delgado et al., 2021). Therefore economic benefits are possible by considering an allocation of ATFM slots that takes into account the different impact that delays have on costs for airlines. Added to these savings are environmental benefits since a possible reduction in costs due to ATFM delays makes the rerouting option less attractive.

The most immediate way to change the FPFS allocation is to let the NM centrally determine an allocation that minimises the overall costs of the delay. However, this solution is not feasible mainly

for two reasons: on the one hand, a single airline could eventually have a higher cost of delay with this new allocation than with FPFS (for the benefit of the system optimum a user is particularly penalised) and therefore it is not clear why it should agree to deviate from it. On the other hand, airlines need to communicate their delay costs to the NM and this is sensitive information that is released very reluctantly. It is therefore necessary to identify mechanisms which in some way involve airlines or which at least do not oblige them to disclose confidential information.

Since for each regulation the number of slots is fixed (see Section 3.1), modifying the FPFS allocation means in practice allowing an exchange of slots (which we also refer to as slot *trade* or *swap*). In the United States context, Vossen and Ball (2006a,b) and Sherali et al. (2011) propose mathematical programming models for the exchange between airlines (inter-airlines) of slots if some flights are cancelled or further delayed compared to the initial allocation. Delays are assigned within a *Ground delay program* (Liu et al., 2019) where the ration by schedule (RBS) procedure is used to allocate slots (RBS follows the same principles of the FPFS). These models require trading to be regulated by a "mediator" (the FAA in their case), but they do not involve monetary aspects. However, the possibility of designing "market-based mechanisms in which airlines would be able to buy and sell slots" is mentioned since "benefits could be substantial" (Vossen and Ball, 2006b). This suggestion was taken in by Castelli et al. (2011b) who developed a market mechanism that enables airlines to pay for delay reduction or receive compensation for delay increase by adapting the Vossen and Ball (2006a) model to the European context. This mechanism fulfils some desirable properties for a market including individual rationality (i.e., each participant has a non-negative payoff from entering the market) and budget balance (i.e., the overall amount of prices paid and received by participants sums up to zero), and it can be implemented through two alternative distributed approaches that do not require airlines to disclose confidential information. Later, Granberg and Polishchuk (2012) show how to design mechanisms that can be used for the allocation of many different types of ATM resources, including ATFM slots. Even though these mechanisms are socially optimal (i.e., resources are distributed in the way that best serves the users community as a whole), truthful (i.e., each individual user has incentive to play fairly), and, under certain assumptions, individually rational, they are not budget balanced because the resource owner gains profit from the users' payments. Other truthful market mechanisms for ATFM slot allocation are proposed by Rosenthal and Eisenstein (2016) and by Mehta and Vazirani (2020). However, these mechanisms are centralised and therefore - even if no airline has an incentive to misreport the delay costs of its flights - airlines are forced to reveal information that they prefer to keep to themselves. To simultaneously respect this request for confidentiality and still involve airlines in the ATFM slot allocation process, for some years now EUROCONTROL[1] has been developing UDPP, the User Driven Prioritisation Process (Pilon et al., 2016, 2019; Ruiz et al., 2019a). This mechanism requires that each AU associates a priority value to each of its flights, on the basis of which the FPFS allocation can then be modified (intra-airline). No inter-airline exchanges are allowed, but in certain circumstances an AU can also get a slot that did not previously belong to it and thus obtain a substantial reduction in the cost of the associated delay. ~~Each AU is also free not to participate in the UDPP; in this case its FPFS allocation is not changed. UDPP has received a wide consensus among airlines and has reached a high level of maturity after several human in the loop validation exercises. Besides a study addressing an outdated version of UDPP, the current version is purely empirical since it is not supported, as far as we know, by any methodological framework that can quantify or at least estimate a-priori the benefits that this mechanism could bring in terms of cost reduction with respect to FPFS.~~ Each AU can choose not to participate in the UDPP, and if they do, their FPFS allocation remains unchanged. UDPP has gained widespread acceptance among airlines and has become highly refined after numerous validation exercises involving *human in the loop* simulations

[1]EUROCONTROL, the European Organisation for the Safety of Air Navigation, is the international organisation that develops and maintains an efficient air traffic management across Europe.

(SESAR, 2019). However, it is important to note that the assessment of the performance of the current version of UDPP is based solely on empirical data. Indeed, we have not found any established framework that can quantitatively or predictively assess the cost reduction benefits of this mechanism compared to FPFS. The only study we found regarding this mechanism was related to an outdated version (Zhang et al., 2021).

All the models that have been presented in this short review foresee the allocation of slots to a single capacitated resource. In the not infrequent case, however, that one or more flights are subject to several regulations (our analysis of traffic data shows that in the week 1-7 July 2019 47% of regulated flights are subject to more than one regulation, see Section 5.2), Barnhart et al. (2012) demonstrate with a simple example that performing FPFS for each resource independently can produce inconsistencies, i.e., a regulation may impose on a flight an ATFM slot that is not compatible with the slot imposed on the same flight by another regulation. It is therefore necessary to propose alternative algorithms. In Europe, for instance, when a flight is subject to multiple regulations, the delay of the Most Penalising Regulation (MPR), meaning the one causing the highest delay, takes precedence and is forced into all other regulations. In the U.S. other heuristics (named *precedence RBS* and *exemption RBS*) are applied (Barnhart et al., 2012).

Our contribution fits into this context as it outlines a market-based approach to minimise the cost of ATFM delays in the presence of multiple regulations, guaranteeing to airlines the confidentiality of their costs. We introduce a time interval, which we refer to as *time window*, associated to each regulation a flight is subject to. A flight is expected to enter every regulated resource it has to traverse within its corresponding time window. A set (or bundle) of time windows is initially allocated to each flight, and the time windows of this bundle are possibly traded to form another bundle that decreases the flight delay cost. This exchange can be guided by a central authority, assuming it has access to the cost information of the flights, or carried out in a distributed manner without the need for each flight to communicate the costs of the delay. Our numerical computations based on real data instances of a test day of the European airspace show that the market-based allocation may reduce the delay costs from 47% up to 89% with respect to the initial allocation, while preserving cost confidentiality.

The idea of characterising the trajectory of a flight by a set of time windows is not entirely new in the ATM context, either at the execution (while en-route) (Berechet et al., 2009; Han et al., 2010; Margellos and Lygeros, 2013; Rodríguez-Sanz et al., 2019, 2020) or tactical planning (on the day of operations) phases of a flight (Castelli et al., 2011a). More recently, the definition of a flight trajectory as a sequence of time windows allowed Bolić et al. (2021a,b) to quantify the flexibility that can be granted to flights at the strategic level (up to 6 months ahead) taking into account changing airspace configurations and capacity. Flights complying with time windows guarantee that they will not impact negatively any other flight.

None of these studies considers the time windows that characterise the flight trajectory as objects that can be traded in order to improve a system objective, such as the overall cost of the ATFM delay of flights subject to multiple regulations. A first hint of how to solve this latter problem was provided by Castelli et al. (2011c), which is now significantly refined and extended in this paper in several aspects, including the formulation of the distributed market mechanism, and its detailed implementation and resolution. Furthermore, we provide a realistic characterisation of the European airspace in terms of traffic, airspace configuration and cost data, as described in Section 5. Finally, all computational experiments are run on real data.

The remainder of this paper unfolds as follows. Section 2 introduces the main features of Air Traffic Management, and describes the algorithm currently in place to allocate ATFM slots. The approach followed when airlines make available to an authority the costs incurred when their flights are delayed is presented in Section 3, which mathematically defines (a) the centralised allocation of time window

bundles that minimises the overall cost of delay and (b) the market mechanism that allows to trade the bundles of an initial allocation and reach the minimum ATFM delay cost allocation. The model relies on an integer linear programming formulation. If, on the other hand, the costs of the delay cannot be revealed, the allocation and trading of the time window bundles can be carried out in a distributed manner. Section 4 develops the distributed model by applying the Lagrangian relaxation technique and the subgradient method. Section 5 describes how the dataset for the numerical experimentation was prepared, and Section 6 reports computational results on some subsets of the dataset. Section 7 contains the conclusions and some ideas for further development of this work.

2. Some key concepts of Air Traffic Management

ATM is an extremely complex system, which has to cope with very different situations in daily operations (Niarchakou and Sfyroeras, 2021; Cook, 2016). The evaluation of the proposed market mechanism can therefore only be based on a simplified representation of ATM, especially with regard to slot allocation procedures (Section 2.1).

The whole airspace is divided in *ATC sectors* (or simply, *sectors*), which define the area of responsibility of air traffic controllers. Adjacent sectors can be merged into *collapsed* sectors from time to time, in order to enable more effective use of resources. Sectors which are not collapsed are called *elementary* sectors.

~~The capacity of airspace resources is the ability of the ATM system to provide air navigation services to a certain volume of air traffic by meeting safety standards. In particular,~~ In Europe the capacity of an ATC sector is defined as the maximum number of aircraft that can enter the given sector during a specified period of time (usually one hour), while permitting an acceptable level of air traffic controller workload.

If traffic demand in an airspace or airport is forecast to exceed capacity, the NM decides whether to apply flow restrictions. Flow restrictions, called ATFM regulations, ~~impose delays to aircraft~~ restrict the departure times of flights, assigning them controlled take-off times (CTOTs) -which may cause delays on some flights-, so that traffic is smoothed and avoid overload of the regulated sector or airport.

ATFM regulations are based on the principle that delays are both safer and less costly to be absorbed on the ground rather than in the air. Therefore, any delay forecast in a capacity-constrained resource along a flight's route is anticipated at the departure airport before take-off, a practice known as *ground-holding* (Odoni, 1987). The flight receives an *ATFM slot* (also called *departure slot*), a 15-minute time range during which the aircraft must take off.

2.1. The allocation of ATFM slots

In this section, we present the main principles of the slot allocation procedures. ATFM slots are managed by the Computer Assisted Slot Allocation (CASA) system, which is a largely automatic and centralised tool run by EUROCONTROL. CASA initially calculates an Estimated Take-Off Time (ETOT) for each flight. This enables each flight to be given an Estimated Time Over (ETO), which is the point of entry at each sector through which the route is planned.

A regulation is characterised by a period of activation (start and end time), the allowed entering flow rate (in flights per hour), and some other specifications. The regulation is divided by CASA into a number of slots of equal width depending on the rate. Each regulation is thus associated to a Slot Allocation List which is initially empty. Normally the capacity of each slot is equal to one flight, although in special circumstances it may be higher (Ruiz et al., 2019b).

The policy under which flights are assigned slots is First Planned First Served (FPFS). CASA sorts the flights entering the regulation according to their ETO over the restricted location, and assigns a slot to each flight in this sequential order, as close as possible to the ETO.

This order is also maintained when a flight is subject to multiple regulations by giving precedence to the regulation causing the highest delay, i.e., the Most Penalising Regulation (MPR). The flight receives an ATFM slot according to the FPFS order applied to the MPR. ~~The delay thus obtained is then forced into all other regulations.~~ The delay resulting from the MPR is used to find a slot in the other regulations crossed by the flight such that it is the closest to the new ETO of the flight in those regulated sectors. The flight is then "forced" into the slots predetermined by the MPR without following the FPFS logic in those regulations.

The FPFS policy on which the CASA system is based is considered fair and equitable by all parties involved (Lulli and Odoni, 2007). However, from an efficiency point of view, it is not optimal as it does not consider the impact of ATFM delays to airlines in terms of delay costs. If one considers the economical impact of ATFM delays on the flight profitability, there is typically a trade-off between fairness and efficiency (Barnhart et al., 2012).

~~Moreover, CASA works in a passive mode from the point of view of airspace users, since ATFM slots are imposed by a central authority.~~Moreover, in the current CASA logic (MPR-FPFS), the airspace users remain as passive actors, since they receive ATFM slots that are calculated and assigned to them based on pre-existing flight plan information and with limited options to influence the slot sequences based on the impact of the resulting delays to their operations and daily schedules. ~~In contrast, an important tool to implement~~ To fix that, an important concept that could be implemented to optimise the cost of the ATFM measures is Collaborative Decision Making (CDM), a process in which decisions are agreed by all stakeholders who actively take part in the decision-making (Niarchakou and Sfyroeras, 2021). In particular, since the cost of delay for a flight is possibly known only by the airline operating it, the allocation of ATFM slots would benefit from an enhanced application of CDM. In this context, our proposal is to introduce a mechanism for exchanging slots. Since it would be impossible to replicate the operational reality with absolute precision, we have adopted certain simplifications. For example, the capacity of each TW is always equal to 1 and never higher as is sometimes the case, or the procedure for FPFS allocation of bundles can be called a CASA-like algorithm as it certainly follows the MPR rationale, but is not identical to the CASA algorithm (see Appendix C and also section 3). Thus, the objects that are exchanged, although very similar, do not inherit exactly all the dynamics and characteristics of the actual slots. In order not to misuse the term 'slots', we refer to these objects as *time windows* (TWs) in the rest of the paper.

2.2. The cost of delays in ATM

On the day of operations, various factors cause flight delay, for example weather, ATFM measures, and issues attributable to aircraft operators. There are two main types of costs that airlines experience due to delays: strategic delay costs and tactical delay costs. Strategic costs are mainly due to schedule buffers, while tactical costs are those incurred on the day of operations due to actual delays.

A major component in the tactical cost of delay that impacts on airlines are the costs associated with delayed passengers, which fall into two categories: *hard* costs and *soft* costs. Hard costs are determined by passenger re-booking, compensation and care. Soft costs are due to the loss of market share that comes from passenger dissatisfaction.

Tactical costs also include maintenance and crew costs. Tactical maintenance costs are due for example to the mechanical attrition of aircraft waiting at gates, whereas tactical crew costs are based on the cost of crewing for additional minutes above those planned at the strategic phase.

For maintenance and crew costs, one minute of delay does not depend on the extent of the delay. In contrast, longer passenger delays have higher associated costs per minute than shorter ones, thus making tactical costs a super-linear function of delay length.

A comprehensive study on the costs of delays in the air traffic management system was carried out for EUROCONTROL by Cook et al. (2004), and then updated and extended in subsequent years (Cook

and Tanner, 2015; Cook et al., 2021). It estimates cost delay figures for different phases of flight (e.g. en-route and at-gate), for a range of specific aircraft types and three cost scenarios (low, base and high), separately for strategic and tactical delays. These estimations will be used as inputs in this research (see Section 5).

3. The central resource allocation problem

This section formalises the market mechanism for the TW allocation by a central authority when flights are subject to multiple regulations. We first describe in mathematical terms the time window bundle allocation that minimises the cost of the delay as faced by the central authority (Sections 3.1 and 3.2). This minimum cost allocation is then confronted with the initial allocation that is currently granted. With a slight abuse of terminology, we refer to it as *FPFS allocation* (see Section 5.2 and Appendix C for details on its implementation that mimics the Most Penalising Regulation principle). In particular, Section 3.3 introduces a mechanism for the allocation of time window bundles to flights, alternative to FPFS, which is cost-efficient. The mechanism is market-based since airlines can be seen as competitors contending for a limited resource, which is the capacity of regulated sectors or airports. In addition, it satisfies some of the following properties commonly used in mechanism design (see Krishna, 2009, for a formal treatment):

1. Individual rationality: each individual receives a non-negative utility from participating in the mechanism, so that it is preferable to participate than not participate.

2. Budget balance: the mechanism requires no financing from outside. In particular, a mechanism is strongly budget balanced if the total payment of the participants is equal to zero. The mechanism neither receives subsidisation from outside, nor generates a surplus; it just redistributes money among participants. A mechanism is weakly budget balanced if the total payment of the participants is larger or equal than zero: the mechanism can potentially generate a surplus.

3. Allocative efficiency: the mechanism maximises the sum of individual utilities.

4. Incentive compatibility: the best strategy for participants is to report their valuations truthfully. No agent can increase their utility by misreporting their true preferences.

~~However, due to the impossibility theorem by Myerson and Satterthwaite (1983) the four properties cannot be satisfied at the same time.~~ Myerson and Satterthwaite (1983) showed that in the presence of asymmetric information (i.e. the value of a given good for a given agent is only known by such agent), it is not always possible that the above four properties are guaranteed. Indeed, under some specific conditions and assumptions, Myerson and Satterthwaite proved that it is impossible to find a market mechanism that can satisfy all four properties simultaneously. The mechanism proposed in Section (3.3) relaxes the fourth property, and thus assumes that participants report their preferences honestly to the central authority. Section 4 will describe a distributed implementation of such mechanism in which participants are assumed to act honestly introducing their preferences in a privacy-preserving manner. The validity of this assumption is addressed in Appendix A.

3.1. Mathematical formulation

The proposed mathematical formulation requires the notation introduced in Table 1.

$\mathcal{F}$	set of flights
$\mathcal{R}$	set of regulations
K_r	capacity of regulation $r \in \mathcal{R}$ (in flights per hour)
$start_r$	start time of regulation $r \in \mathcal{R}$
end_r	end time of regulation $r \in \mathcal{R}$
L_r	TW allocation list of regulation $r \in \mathcal{R}$
Δ_r	TW width (in minutes) of regulation $r \in \mathcal{R}$
N_r	number of TWs of regulation $r \in \mathcal{R}$
I_j	lower bound of the j-th TW in L_r
U_j	upper bound of the j-th TW in L_r
ϵ	time discretisation interval
$\hat{L}_r$	augmented TW allocation list of regulation $r \in \mathcal{R}$
n_f	number of regulations crossed by flight $f \in \mathcal{F}$
R_f	list of the regulations crossed by flight $f \in \mathcal{F}$
r_i	i-th regulation crossed by flight $f \in \mathcal{F}$
E_f	list of the expected times of entry of flight $f \in \mathcal{F}$
e_i	expected entry time into r_i
Q_f	set of feasible bundles of flight $f \in \mathcal{F}$
q_f	an element of Q_f, i.e., $q_f \in Q_f$ is a feasible bundle for flight f
d_{q_f}	delay of bundle q_f
$C(f, q_f)$	cost of bundle q_f for flight f

Table 1: Model's notation

We consider a set of flights $\mathcal{F}$ that are scheduled within a period of time $\mathcal{T}$, and a set of regulations $\mathcal{R}$ active during $\mathcal{T}$ that limit the rate of flights entering a capacity constrained resource, either an airport or an airspace sector.

Each regulation $r \in \mathcal{R}$ has a capacity of K_r flights per hour and is active for a time period $[start_r, end_r]$. Each regulation r is associated to a *TW allocation list* L_r, a list of time windows of equal width that depends on the capacity. The TW width, expressed in minutes, is

$$\Delta_r = \frac{60}{K_r}$$

and given that the duration $end_r - start_r$ is expressed also in minutes, the number of time windows is

$$N_r = \left\lfloor \frac{(end_r - start_r)}{\Delta_r} \right\rceil$$

where $\lfloor \cdot \rceil$ denotes rounding to the nearest integer.

For $j = 1, \ldots, N_r$ the time interval associated to the j-th TW of the list $[I_j, U_j] \in L_r$ is given by

$$I_j = start_r + \left\lfloor (j-1) \cdot \Delta_r \right\rceil$$

$$U_j = \begin{cases} I_{j+1} - \epsilon & \text{for } j = 1, \ldots, N_r - 1. \\ end_r & \text{for } j = N_r. \end{cases}$$

where $\epsilon = 1$ second and the rounding is applied to an argument expressed in seconds (for example, an argument of 1 hour, 10 minutes, 3 seconds and 15 hundredths of a second is rounded to 1 hour, 10 minutes and 3 seconds).

The interval $[start_r, end_r]$ is thus partitioned into a set of disjoint segments, with a time discretization of 1 second. Since $\lfloor x \rceil < x + 1$ for any positive real number x, we have that

$$\left\lfloor \frac{(end_r - start_r)}{\Delta_r} \right\rceil - 1 < \frac{(end_r - start_r)}{\Delta_r}$$

and thus

$$(N_r - 1) \cdot \Delta_r < (end_r - start_r).$$

Since $\lfloor x \rceil < \lfloor y \rceil$ for any $0 < x < y$ and since $start_r$ and end_r are reported with a precision of one minute in our data, i.e. $(end_r - start_r) = \lfloor end_r - start_r \rceil$, it follows

$$\lfloor (N_r - 1) \cdot \Delta_r \rceil < (end_r - start_r)$$

so we conclude that $I_{N_r} < end_r$.

The time discretization is actually not necessary for the optimisation model, but it simplifies the bundle construction algorithm (appendix B).

Hereinafter, a TW will be denoted interchangeably by an integer j representing its position in the list L_r, or an interval $[I_j, U_j]$.

Each TW has capacity of 1 flight and represents the time interval in which a flight is allowed to enter the regulated resource during its regulated period. However, a flight is also allowed to enter a regulated resource before or after its regulated period. To deal with the two situations uniformly, we augment the TW allocation list of each regulation r with two "dummy" time windows $j = 0$ and $j = N_r + 1$ defined by

$$I_0 = -\infty, \quad U_0 = start_r - \epsilon$$

$$I_{N_r+1} = end_r + \epsilon, \quad U_{N_r+1} = +\infty.$$

To simplify the model formulation, we assume these two additional time windows, placed at the beginning and at the end of L_r, have infinite capacity, so that the number of flights that they can accommodate is not limited. Let us call $\hat{L}_r$ the TW allocation list augmented in this way, $\hat{L}_r = [0, 1, 2, \ldots, N_r + 1]$.

The flight plan of each flight $f \in \mathcal{F}$ individuates a list $R_f = [r_1, ..., r_{n_f}]$ of the regulations crossed along the route of f from its departure to its destination, and a list $E_f = [e_1, ..., e_{n_f}]$ of the expected times of entry into each regulated resource. The list E_f is defined with respect to the original (pre-regulated) flight plan and its i-th element e_i is the expected time of entry into the i-th element r_i of R_f. The length of the two lists $n_f = |R_f|$ is the number of regulations affecting flight f.

One time window must be assigned to each flight $f \in \mathcal{F}$ for each regulated resource it crosses. A dummy time window is also a feasible assignment. We denote this *bundle* of time windows by $q_f = [TW_1, \ldots, TW_{n_f}]$ where $TW_i \in \hat{L}_{r_i}$ for $i = 1, ..., n_f$.

Since we assume that flights cannot be anticipated, every time window in q_f must end after the corresponding entry time of f, i.e. $e_i \leq U_{TW_i}$ for all $i = 1, \ldots, n_f$.

If $n_f > 1$, we assume that the flying time $e_{i+1} - e_i$ between consecutive resources r_i and r_{i+1}, for $i = 1, ..., n_f - 1$, is fixed. A bundle q_f is compatible with the fixed flying times if there exists a sequence of time instants $[t_1, \ldots, t_{n_f}]$ such that $I_{TW_i} \leq t_i \leq U_{TW_i}$ for all $i = 1, ..., n_f$ and $t_{i+1} - t_i = e_{i+1} - e_i$ for all $i = 1, ..., n_f - 1$. The sequence $[t_1, \ldots, t_{n_f}]$ represents the re-planned times of entry in each regulated resources and is a temporal shift of the trajectory $[e_1, \ldots, e_{n_f}]$. Among the possible sequences $[t_1, \ldots, t_{n_f}]$, for each TW_i the smallest shift occurs when $t_i = I_{TW_i}$. Therefore, the *delay* d_{q_f} experienced by flight f due to bundle q_f is due to the time window that leads to the largest among these smallest shifts. Specifically,

$$d_{q_f} = \max_{i=1,\ldots,n_f} \min_{t_i} \{(t_i - e_i)^+ : I_{TW_i} \leq t_i \leq U_{TW_i}\} = \max_{i=1,\ldots,n_f} \{(I_{TW_i} - e_i)^+\} \tag{1}$$

being $(\cdot)^+ = \max\{\cdot, 0\}$. The delay is strictly positive if $e_i < I_{TW_i}$ for at least one $TW_i \in q_f$.

A delay d_{q_f} causes to flight f a cost $C(f, q_f)$, which is a non-linear non-decreasing function of the delay, and depends on the flight f through factors such as type of aircraft and number of passengers.

We say that a bundle q_f is feasible for f if either q_f is empty, and in this case f is cancelled, or it satisfies the following requirements:

(i) it contains a time window TW_i for each regulation in R_f and $E_i \leq U_{TW_i}$ component-wise;

(ii) it is compatible with the fixed flying times;

(iii) the delay is acceptable, i.e. it satisfies the bound $d_{q_f} \leq MaxDel_f$ where $MaxDel_f$ is the delay beyond which it is more convenient to cancel f.

We denote by Q_f the set of all feasible bundles for flight f. Appendix B describes a simple algorithm for constructing the set Q_f.

3.2. The optimal allocation

The allocation of time windows to flights that minimises the total cost of delay is given by the optimal solution of the following binary optimisation problem (TW allocation problem):

$$\min \sum_{f \in \mathcal{F}} \sum_{q \in Q_f} C(f, q) x(f, q) \tag{2a}$$

$$\sum_{f \in \mathcal{F}} \sum_{q \in Q_f : q \ni k} x(f, q) \leq 1 \quad \forall r \in \mathcal{R}, k \in L_r \tag{2b}$$

$$\sum_{q \in Q_f} x(f, q) = 1 \quad \forall f \in \mathcal{F} \tag{2c}$$

$$x(f, q) \in \{0, 1\} \quad \forall f \in \mathcal{F}, q \in Q_f \tag{2d}$$

The objective function (2a) minimises the sum of all delay costs. Constraint (2b) is the capacity constraint, which guarantees that no more than one flight is assigned to any time window. Constraint (2c) is the allocation constraint, which guarantees that every flight receives one and only one bundle from its set of requests. Constraint (2d) is the integrality constraint. The variable $x(f, q)$ is equal to one when flight f is assigned to bundle q, and zero otherwise. Problem (2) is NP-hard as we can reduce to it the NP-complete Maximal Independent Set (MIS) problem (Lawler et al., 1980).

A feasible solution of problem (2) always exists, because for all $f \in \mathcal{F}$, Q_f always contains either the empty bundle corresponding to flight cancellation, or the bundle composed of all "dummy" time windows that consumes no capacity.

The allocation given by the application of the optimal solution of problem (2) will be denoted by $\mathcal{X}^* = \{q_f^*\}_{f \in \mathcal{F}}$. The allocation $\mathcal{A} = \{a_f\}_{f \in \mathcal{F}}$ given by the FPFS rule constitutes a feasible solution of problem (2).

In the particular case when $|\mathcal{R}| = 1$ and flights compete for time windows in a single capacity constrained resource r, problem (2) simplifies into the following:

$$\min \sum_{f \in \mathcal{F}} \sum_{k \in Q_f} C(f, k) x(f, k) \tag{3a}$$

$$\sum_{f \in \mathcal{F}} \sum_{k \in Q_f} x(f, k) \leq 1 \quad \forall k \in L_r \tag{3b}$$

$$\sum_{k \in Q_f} x(f, k) = 1 \quad \forall f \in \mathcal{F} \tag{3c}$$

$$x(f, k) \geq 0 \quad \forall f \in \mathcal{F}, k \in Q_f \tag{3d}$$

The integrality constraint was dropped since, in this case, the solutions of the linear relaxation are automatically integral. The reason is that problem (3) has the form of an assignment problem, and so the constraint matrix is totally unimodular.

In presence of a single regulation, the FPFS allocation $\mathcal{A}$ minimises the total delay, or in other words $\mathcal{A}$ is an optimal solution of problem (3) when $C(f,k) = d_k$ (Castelli et al., 2011b). This fact is no longer true in presence of many interacting regulations (Ruiz et al., 2019c).

For what will follow in the next section, it is convenient to reformulate problem (2) in terms of maximisation of a total value, instead of minimisation of a total cost. Let us define the *value* of a bundle q to flight f as the difference between the cost of the bundle a_f assigned to f under the FPFS, and the cost of q:

$$V(f,q) = C(f,a_f) - C(f,q). \tag{4}$$

The value is positive if q causes a delay smaller than the delay of a_f, and negative otherwise. Then, we consider the following problem:

$$Z_{IP} = \max \sum_{f\in\mathcal{F}} \sum_{q\in Q_f} V(f,q)x(f,q) \tag{5a}$$

$$\sum_{f\in\mathcal{F}} \sum_{q\in Q_f : q\ni k} x(f,q) \leq 1 \quad \forall r \in \mathcal{R}, k \in L_r \tag{5b}$$

$$\sum_{q\in Q_f} x(f,q) = 1 \quad \forall f \in \mathcal{F} \tag{5c}$$

$$x(f,q) \in \{0,1\} \quad \forall f \in \mathcal{F}, q \in Q_f \tag{5d}$$

Trivially, problem (5) is equivalent to problem (2) in the sense that their optimal solution is the same, because the objective function only differs by a constant term. In fact,

$$\begin{aligned}\sum_{f\in\mathcal{F}} \sum_{q\in Q_f} \big(C(f,a_f) - C(f,q)\big)x(f,q) &= \sum_{f\in\mathcal{F}} C(f,a_f) \sum_{q\in Q_f} x(f,q) - \sum_{f\in\mathcal{F}} \sum_{q\in Q_f} C(f,q)x(f,q) \\ &= \sum_{f\in\mathcal{F}} C(f,a_f) - \sum_{f\in\mathcal{F}} \sum_{q\in Q_f} C(f,q)x(f,q)\end{aligned}$$

where the last equality follows from constraint (5c).

3.3. Pricing the exchange

The allocation $\mathcal{X}^*$ given by the optimal solution of problem (5) could be perceived as unfair by airlines, because it is not always true that $C(f,q_f^*) \leq C(f,a_f)$, or in other terms the utility $V(f,q_f^*)$ can be negative. Some flights reduce their delay with respect to the FPFS while some other increase their delay. In order to design a mechanism which is both allocative efficient and individual rational, as well as weakly budget balanced, we introduce the possibility of payments between airlines that accompany the optimal allocation and attach a price $p(q_f^*)$ to each bundle $q_f^* \in \mathcal{X}^*$. In this way, airlines who are penalised by the optimal allocation with respect to the FPFS receive a monetary compensation, whereas airlines who are better off after the implementation of the optimal allocation can possibly be charged for the delay reduction.

In order to find a set of prices $\mathcal{P}^* = \{p(q_f^*)\}_{f\in\mathcal{F}}$ that support the optimal exchange, if we consider the optimal solution found for the linear relaxation of problem (5), the values of the associated dual variables are optimal for the following problem:

$$Z_{LP} = \min \sum_{f \in \mathcal{F}} u(f) + \sum_{r \in \mathcal{R}} \sum_{k \in L_r} \pi(k) \tag{6a}$$

$$u(f) + \sum_{r \in \mathcal{R}} \sum_{k \in L_r : k \in q} \pi(k) \geq V(f,q) \quad \forall f \in \mathcal{F}, q \in Q_f \tag{6b}$$

$$\pi(k) \geq 0 \quad \forall r \in \mathcal{R}, k \in L_r \tag{6c}$$

Variables $\pi(k)$ are the dual variables associated to the capacity constraint (5b) and $u(f)$ are the dual variables associated to the assignment constraint (5c). Variables $\pi(k)$ can be interpreted as prices of time windows, and $u(f)$ can be interpreted as utilities of users. It is then natural to assume a linear pricing of bundles, so that $p(q) = \sum_{k \in q} \pi^*(k)$ where $\pi^*(k)$ are the optimal dual variables. If we also assume that the utility of a flight when assigned a bundle q_f is $-C(f, q_f) - p(q_f)$, a market mechanism that charges $p(q_f^*)$ to each flight f for the assigned bundle q_f^* would not satisfy individual rationality, because some flights would incur a greater cost with q_f^* than with a_f and additionally be forced to make a payment.

In order to fulfil individual rationality, we consider the allocation a_f as an endowment guaranteed to all flights. Then the market mechanism, mediated by the central authority, takes place in two steps:

1. First, the FPFS bundles $\mathcal{A}$ are allocated for free, as in the current system.

2. Next, the optimal allocation $\mathcal{X}^*$ is implemented and the dual prices are charged for the bundle exchange: each flight pays $p(q_f^*)$ for the assigned bundle q_f^* and receives the price $p(a_f)$ for the released time windows in a_f.

For flight f the cost after step 1 is $C(f, a_f)$, and the cost after step 2 is $C(f, q_f^*) + p(q_f^*) - p(a_f)$. Therefore the utility variation of f when taking part in the mechanism is $\Delta u(f) = C(f, a_f) - C(f, q_f^*) + p(a_f) - p(q_f^*)$. Now we show the condition under which the individual rationality condition $\Delta u(f) \geq 0$ holds.

The complementary slackness conditions between the linear relaxation of problem (5) and its dual (6) are

$$x^*(f,q) > 0 \quad \Longrightarrow \quad u^*(f) + \sum_{r \in \mathcal{R}} \sum_{k \in L_r : k \in q} \pi^*(k) = V(f,q) \quad \forall f \in \mathcal{F}, q \in Q_f \tag{7a}$$

$$\sum_{f \in \mathcal{F}} \sum_{q \in Q_f : q \ni k} x^*(f,q) < 1 \quad \Longrightarrow \quad \pi^*(k) = 0 \quad \forall r \in \mathcal{R}, k \in L_r \tag{7b}$$

If the optimal solution of the linear relaxation of (5) is integer, then the optimal allocation $\{q_f^*\}_{f \in \mathcal{F}}$ satisfies

$$u^*(f) = V(f, q_f^*) - p(q_f^*) \tag{8a}$$

$$u^*(f) \geq V(f,q) - p(q) \quad \forall q \in Q_f \tag{8b}$$

Equation (8a) follows from complementary slackness (7a) and equation (8b) follows from the feasibility (6b) of the optimal solution. Putting them together yields

$$V(f, q_f^*) - p(q_f^*) \geq V(f,q) - p(q) \quad \forall q \in Q_f \tag{9}$$

In particular, since $\mathcal{A}$ is a feasible solution, taking $q = a_f$ gives

$$\Delta u(f) = C(f, a_f) - C(f, q_f^*) + p(a_f) - p(q_f^*) \geq 0 \tag{10}$$

which is the property of individual rationality: every agent has a non-negative profit when selling its FPFS bundle a_f and buying q_f^*. Equation (9) says that not only flight f prefers bundle q_f^* over a_f, but also over every other bundle $q \in Q_f$.

Now we show that the complementary slackness conditions are sufficient to impose that the weak budget balance property holds, i.e.,

$$\sum_{f \in \mathcal{F}} \big(p(q_f^*) - p(a_f)\big) \geq 0. \tag{11}$$

Let us define $\mathcal{L} = \bigcup_{r \in \mathcal{R}} L_r$ the set of all time windows. Equation (7b) says that every time window which is unassigned under the optimal solution $\mathcal{X}^*$ has a price zero. It follows that

$$\sum_{f \in \mathcal{F}} p(q_f^*) = \sum_{k \in \mathcal{L}} \pi^*(k) \tag{12}$$

where (5b) has also been used. So the net turnout resulting from the market mechanism is

$$\sum_{f \in \mathcal{F}} \big(p(q_f^*) - p(a_f)\big) = \sum_{k \in \mathcal{L}} \pi^*(k) - \sum_{f \in \mathcal{F}} \sum_{k \in a_f} \pi^*(k) = \sum_{k \in \mathcal{L} \setminus \bigcup_{f \in \mathcal{F}} a_f} \pi^*(k) \geq 0. \tag{13}$$

Taking a closer look at how the mechanism works, consider the monetary flow associated to each time window $k \in \mathcal{L}$. There are four cases:

(i) If k is assigned both under the FPFS allocation $\mathcal{A}$ and under the optimal allocation $\mathcal{X}^*$, respectively to flight f and to flight g, then f sells time window k to g at the price $\pi^*(k)$ and the central authority is not involved in the exchange.

(ii) If k is assigned under $\mathcal{A}$ to a flight f and it is unassigned under $\mathcal{X}^*$, then f receives $\pi^*(k)$ from the central authority, but $\pi^*(k) = 0$ due to complementary slackness (7b).

(iii) If k is not assigned under $\mathcal{A}$ but it is assigned to a flight g under $\mathcal{X}^*$, then g pays $\pi^*(k) \geq 0$ to the central authority.

(iv) If k is not assigned in $\mathcal{A}$ nor in $\mathcal{X}^*$, then there is no monetary flow associated to it.

Again, from this reasoning it follows that the total revenue for the central authority is larger or equal to zero. The mechanism can produce a surplus, but not incur a deficit. In presence of a single regulation $|\mathcal{R}| = 1$, Castelli et al. (2011b) proved that $\sum_{f \in \mathcal{F}} \Big(p(q_f^*) - p(a_f)\Big) = 0$ and the mechanism is strongly budget balanced. They also showed that case (iii) cannot happen for $|\mathcal{R}| = 1$, because all time windows that are unassigned under $\mathcal{A}$ are also unassigned under $\mathcal{X}^*$.

Equations (9) and (12) together show that the allocation $\mathcal{X}^* = \{q_f^*\}_{f \in \mathcal{F}}$ and the prices $\mathcal{P}^* = \{p(q_f^*)\}_{f \in \mathcal{F}}$ form a Walrasian equilibrium and $\mathcal{P}^*$ is the set of market-clearing prices (Bikhchandani and Mamer, 1997). However, the complementary slackness conditions (7a) and (7b) can guarantee that the individual rationality and weak budget balance properties hold only in the case that the duality gap between problem (5) and its linear relaxation is zero. Otherwise, the integer optimal solution of problem (5) is not guaranteed to form a Walrasian equilibrium with the prices given by the optimal solution of (6). In particular, problem (6) may estimate utility values that are greater than the actual ones. Under these circumstances, it can be decided that initially only a subset of flights are allowed to exchange time windows so that a smaller problem for which the zero duality gap holds is considered, see Section 6.5. All flights belonging to this subset are allowed to exchange time windows and prices are negotiated between airlines. We remark that in the case of a unique regulation, the individual rationality and strong budget balance properties always hold, because problem (3) always gives integer optimal solutions, as mentioned in Section 3.2.

4. A distributed market mechanism

The market mechanism described in Section 3.3 is a centralised model for the allocation of time windows to flights and the calculation of supporting prices. It requires that the central authority, who is in charge of solving problem (5), has complete knowledge of the delay cost data $C(f,q)\ \forall f \in \mathcal{F}\ \forall q \in Q_f$, which can be possibly evaluated for each flight only by its aircraft operator. However, the cost of delay for flights represent confidential information in the commercially competitive air transport industry, and airlines could be reluctant to communicate their delay costs to the central authority.

This section develops a decentralised version of the market mechanism, which does not require the explicit disclosure of private information by airlines, and instead allows to elicit their preferences in an indirect way. This mechanism also relieves the central authority of the burden of solving the NP-hard problem (5) by relaxing it into a set of trivial problems, one for each flight $f \in \mathcal{F}$ (see problem (17) below). The solution to these latter problems is then attributed to the airlines operating the flights to prevent the explicit disclosure of private information.

Throughout the section we make use of well-known Lagrangian relaxation theory results and we refer the reader unfamiliar with them to, e.g., Fisher (2004).

4.1. The Lagrangian dual of the allocation problem

The Lagrangian relaxation of problem (5) with respect to the capacity constraint (5b) is

$$Z_{LR}(\lambda) = \max \sum_{f\in\mathcal{F}} \sum_{q\in Q_f} V(f,q)x(f,q) + \sum_{k\in\mathcal{L}} \lambda_k \left(1 - \sum_{f\in\mathcal{F}} \sum_{q\in Q_f : q\ni k} x(f,q)\right) \tag{14a}$$

$$\sum_{q\in Q_f} x(f,q) = 1, \qquad \forall f \in \mathcal{F} \tag{14b}$$

$$x(f,k) \geq 0, \qquad \forall f \in \mathcal{F}, k \in Q_f \tag{14c}$$

$$\lambda_k \geq 0, \qquad \forall k \in \mathcal{L} \tag{14d}$$

where λ_k are the Lagrangian multipliers. The second term in the objective function (14a) has the role of penalising capacity violations. The integrality constraint (5d) has been dropped because the constraint matrix of problem (14) is totally unimodular.

It is convenient to rewrite objective function (14a) in an alternative form:

$$\begin{aligned} &\sum_{f\in\mathcal{F}} \sum_{q\in Q_f} V(f,q)x(f,q) + \sum_{k\in\mathcal{L}} \lambda_k \left(1 - \sum_{f\in\mathcal{F}} \sum_{q\in Q_f : q\ni k} x(f,q)\right) = \\ &= \sum_{f\in\mathcal{F}} \sum_{q\in Q_f} V(f,q)x(f,q) - \sum_{f\in\mathcal{F}} \sum_{q\in Q_f} \sum_{k\in q} \lambda_k x(f,q) + \sum_{k\in\mathcal{L}} \lambda_k = \\ &= \sum_{f\in\mathcal{F}} \sum_{q\in Q_f} \left[V(f,q) - \sum_{k\in q} \lambda_k\right] x(f,q) + \sum_{k\in\mathcal{L}} \lambda_k \end{aligned} \tag{15}$$

Equation (15) shows that problem (14) is separable into $|\mathcal{F}|$ problems, one for each $f \in \mathcal{F}$. The Lagrangian subproblem for flight f is

$$Z_{LR}(f,\lambda) = \max \sum_{q\in Q_f} \left[V(f,q) - \sum_{k\in q} \lambda_k\right] x(f,q) \tag{16a}$$

$$\sum_{q\in Q_f} x(f,q) = 1 \tag{16b}$$

$$x(f,k) \geq 0 \qquad \forall k \in Q_f \tag{16c}$$

and $Z_{LR}(\lambda) = \sum_{f\in\mathcal{F}} Z_{LR}(f,\lambda) + \sum_{k\in\mathcal{L}} \lambda_k$. Each subproblem for $f \in \mathcal{F}$ can be locally solved by the airline operating f.

Problem (16) can be solved in linear time by inspection as

$$Z_{LR}(f,\lambda) = \max_{q\in Q_f} \left\{ V(f,q) - \sum_{k\in q} \lambda_k \right\}, \tag{17}$$

always corresponding to optimal integer values for the variables $x(f,k)$ in (16).

We can interpret $\sum_{k\in q} \lambda_k$ as the cost of a bundle q according to the prices λ. The optimal solution of problem (17) is the bundle $q \in Q_f$ that maximises the utility of f if we interpret Lagrangian multipliers as prices of time windows.

The Lagrangian dual of problem (5) is

$$Z_{LD} = \min_{\lambda\geq 0} Z_{LR}(\lambda) \tag{18}$$

Since the formulation (14) is totally unimodular, $Z_{LD} = Z_{LP}$ and the optimal Lagrange multipliers that solve problem (18) are optimal dual variables for the linear relaxation of problem (5). Hence, if the duality gap between problem (5) and its linear relaxation is null, $Z_{LD} = Z_{LP} = Z_{IP}$ and the optimal Lagrange multipliers are equilibrium prices that support the optimal exchange.

We solve problem (18) via the subgradient method (Fisher, 2004, Section 6). First of all, the central authority fixes the initial prices λ^0, for example $\lambda^0 = 0$. Then the subgradient method proceeds in an iterative way. At iteration t, each flight f determines the bundle q_f^{*t} that maximises its utility variation when exchanging the FPFS endownment a_f with another bundle $q \in Q_f$, at the current prices λ^t:

$$q_f^{*t} = \operatorname*{argmax}_{q\in Q_f} \left\{ V(f,q) - p^t(q) \right\} = \operatorname*{argmax}_{q\in Q_f} \left\{ V(f,q) - p^t(q) + p^t(a_f) \right\} \tag{19}$$

where $p^t(q) = \sum_{k\in q} \lambda_k^t$. Notice that f only needs to communicate the optimal solution x^t of (14) according to prices λ^t, i.e. its most preferred bundle q_f^{*t}, and not the optimal value $Z_{LR}(f,\lambda)$ to the central authority.

Then the central authority computes the following quantity

$$SG_k^t = 1 - \sum_{f\in\mathcal{F}} \sum_{q\in Q_f : q\ni k} x^t(f,q) \quad \forall k \in \mathcal{L}. \tag{20}$$

The vector SG^t is a subgradient of $Z_{LR}(\lambda)$ at the point λ^t. The sum in (20) represents the number of flights whose demanded bundle q_f^{*t} contains time window k. Then the prices are centrally updated according to

$$\lambda_k^{t+1} = \max(0, \lambda_k^t - \mu_t SG_k^t) \quad \forall k \in \mathcal{L} \tag{21}$$

where μ_t is the step length and will be discussed in Section 4.2.

Equation (21) has the following interpretation: for each $k \in \mathcal{L}$

(i) if $SG_k^t < 0$, the demand for time window k exceeds the capacity, so the price of k is raised;

(ii) if $SG_k^t > 0$, less capacity is used than available, so the price of k is lowered;

(iii) if $SG_k^t = 0$, there is already a balance between demand and capacity, so the price is unchanged.

The market mechanism is configured in a series of exchanges of information between the central authority and the aircraft operators. Each iteration proceeds as follows:

1. The central authority communicates prices λ^t to all aircraft operators.

2. Each aircraft operator solves problem (16) according to prices λ^t and communicates the demanded bundle q_f^{*t} to the central authority.

3. The central authority computes the imbalance between the demand q_f^{*t} and the capacity of time windows according to equation (20) and updates the prices of time windows λ^t according to (21).

In the rest of this Section, we discuss the details of the iterative market mechanism, i.e., of the subgradient method applied to problem (18). Hereinafter, we call (IP) the problem (5), (LP) its linear relaxation, $(LR(\lambda))$ the problem (14) and (LD) the problem (18).

4.1.1. Termination of the iterative market mechanism

Under appropriate choice of the stepsize the subgradient algorithm converges to the optimal solution of (LD).

If at iteration t the bundles demanded by flights happen to form an allocation x^t that respects the capacity, i.e. $SG_k^t \geq 0 \ \forall k \in \mathcal{L}$, or in other words q_f^{*t} share no time windows, then the allocation will constitute a feasible solution for (IP). Indeed, thanks to property (19), the prices λ^t are such that the exchange is individual rational, and each user also maximises the individual utility. However, we remark that there is no guarantee that the prices satisfy the weak budget balance property (11). It can occur that the solution x^t of problem $(LR(\lambda^t))$ is feasible but not optimal for (IP). In this case, the complementary slackness conditions do not hold.

If the solution is not only capacity-compliant but also satisfies complementary slackness $SG^t \cdot \lambda^t = 0$, i.e. all unassigned time windows have zero price, then $Z_{LR}(\lambda^t) = Z_{IP}$, so x^t is an optimal solution of (IP) and λ^t is an optimal solution of problem (6), they form a Walrasian equilibrium, and the subgradient algorithm stops.

When the duality gap $Z_{LD} - Z_{IP}$ is zero, an optimal solution (λ^*, x^*) always exists, i.e., (λ^*, x^*) feasible for (LD) and (IP) and such that the complementary slackness conditions hold. Unfortunately, determining the value of x^* may be not easy even when the subgradient algorithm makes the sequence λ^t converge to λ^*. In presence of multiple optimal solutions for $(LR(\lambda^*))$, each optimal solution of (IP) is among the optimal solutions of $(LR(\lambda^*))$, but the opposite is not necessarily true. Hence, the solution of $(LR(\lambda^*))$ may provide an assignment x which is not optimal for (IP).

Generally speaking, there is no way of proving that the subgradient algorithm has converged to the optimal values λ^*. To resolve this difficulty, the method is usually terminated upon reaching an arbitrary iteration limit (Fisher, 2004).

4.2. Choice of stepsize

A choice of a stepsize for the subgradient method that guarantees convergence to the minimum of the Lagrangian function is

$$\mu_t = \frac{\epsilon_t \big(Z_{LR}(\lambda^t) - Z_{LP}\big)}{\|SG^t\|^2} \tag{22}$$

where $0 < \epsilon < \epsilon_t \leq 2$. In (22) the numerator depends on the difference between the current Lagrangian objective function value and the minimum of the Lagrangian function, and the denominator is the square norm of the subgradient vector. Unfortunately, in general, the minimum of the Lagrangian function is unknown and one uses a lower bound on this minimum. In our case, both Z_{LP} and $Z_{LR}(\lambda^t)$ are unknown because the objective function coefficients of the Lagrangian function $V(f, q)$ (the costs of delay) are unknown to the central authority. The rest of this Section proposes a way to compute an estimate for the difference $Z_{LR}(\lambda^t) - Z_{LP}$ to be used in the stepsize formula.

4.2.1. Cost elicitation

First of all, we discuss how the preferences communicated by airlines through (19) during the course of the distributed market mechanism actually provide information on costs $C(f, q)$ to the central authority.

Recall that $p^t(q) = \sum_{k \in q} \lambda_k^t$ is the price of a bundle q at iteration t. Equation (19) says that for all $f \in \mathcal{F}$

$$V(f, q_f^{*t}) - V(f, q) \geq p^t(q_f^{*t}) - p^t(q) \quad \forall q \in Q_f \tag{23}$$

The right-hand side of equation (23) is a known quantity, whereas the left-hand side is unknown. Equation (23) represents a set of $|Q_f|$ inequalities each involving the values of a pair of bundles. At every iteration, we can add a new set of $|Q_f|$ inequalities for each $f \in \mathcal{F}$ and build up a system of inequalities incrementally. Notice that, however, the total number of inequalities collected by the end of the iterative market mechanism is limited, because if at some iteration t the bundle demanded by f happens to be the same bundle demanded at an earlier iteration $t' < t$, i.e. $q_f^{*t} = q_f^{*t'}$, then we can simply update the right-hand side of the old inequalities instead of adding a new set of inequalities, if the new right-hand side is larger than the old one.

We can also take advantage of the fact that the cost is a non-decreasing function of the delay to write an additional relation between values:

$$V(f, q) \geq V(f, \bar{q}) \quad \forall q \in Q_f, \bar{q} \in Q_f : d_q < d_{\bar{q}} \tag{24}$$

In addition, from Equation (4) it follows that $V(f, a_f) = C(f, a_f) - C(f, a_f)$ so

$$V(f, a_f) = 0. \tag{25}$$

The system of inequalities (23) together with (24) and (25) defines $\forall f \in \mathcal{F}$ a convex polyhedron in a $|Q_f|$-dimensional space which gets smaller during the course of iterations, and which contains a point corresponding to the real combination of values $V(f, q)$ $\forall q \in Q_f$. This polyhedron represents all the information that has been elicited about the cost of delay for flight f. Now let $q_1, q_2, \ldots, q_{|Q_f|}$ be the bundles in Q_f ordered by increasing delay. One can obtain a lower bound $LB(f, q) \leq V(f, q)$ on the value of each bundle $q \in Q_f$ by exploiting this elicited information. The tightest possible lower bound is the solution of the following linear program, whose constraints define the polyhedron described before

$$LB^t(f, q_k) = \min v(q_k) \tag{26a}$$

$$v(q_i) - v(q_j) \geq c_{ij}^t \quad \forall q_i \in Q_f, q_j \in Q_f \tag{26b}$$

$$v(q_{i-1}) \geq v(q_i) \quad \forall q_i \in Q_f \setminus \{q_1\} \tag{26c}$$

$$v(a_f) = 0 \tag{26d}$$

Constraints (26b) correspond to equations (23), constraints (26c) correspond to equations (24) and (26d) to (25). The coefficients of constraints (26b) are updated at each iteration as follows: for all $q_j \in Q_f$

$$c_{ij}^t = \begin{cases} c_{ij}^{t-1} & \text{if } q_i \neq q_f^{*t} \\ \max\left(p^t(q_i) - p^t(q_j), c_{ij}^{t-1}\right) & \text{if } q_i = q_f^{*t} \end{cases} \tag{27}$$

initialised at $t = 0$ with

$$c_{ij}^0 = -\infty \quad \forall q_i \in Q_f, q_j \in Q_f \tag{28}$$

Similarly, changing the objective function (26a), one obtains an upper bound $UB(f, q) \geq V(f, q)$ for all $q \in Q_f$

$$UB^t(f, q_k) = \max v(q_k) \tag{29a}$$
$$v(q_i) - v(q_j) \geq c^t_{ij} \quad \forall q_i \in Q_f, q_j \in Q_f \tag{29b}$$
$$v(q_{i-1}) \geq v(q_i) \quad \forall q_i \in Q_f \setminus \{q_1\} \tag{29c}$$
$$v(a_f) = 0 \tag{29d}$$

Notice that in doing this, we are effectively eliciting information on costs, but only the information which is necessary for the subgradient procedure to converge, and the bounds are not strict generally.

If we reformulate the cost elicitation problem in terms of costs instead of values, we can write an additional relation between them: not only we know that the cost function is non-decreasing, but also that it is superlinear. This means that the *unit* cost of delay is a non-decreasing function of the delay:

$$\frac{C(f,q)}{d_q} \leq \frac{C(f,\bar{q})}{d_{\bar{q}}} \quad \forall q \in Q_f, \bar{q} \in Q_f : d_q < d_{\bar{q}}, d_q \neq 0 \tag{30}$$

This is a linear constraint, thus it can be included in our linear problem, and it makes constraint (24) redundant, since it is stricter. Relation (25) is substituted by

$$C(q) = 0 \quad \text{if } d_q = 0 \tag{31}$$

Then problem (26) becomes

$$\widetilde{LB}^t(f, q_k) = \min c(q_k) \tag{32a}$$
$$c(q_j) - c(q_i) \geq c^t_{ij} \quad \forall q_i \in Q_f, q_j \in Q_f \tag{32b}$$
$$d_{q_i} \cdot c(q_{i-1}) \leq d_{q_{i-1}} \cdot c(q_i) \quad \forall q_i \in Q_f \setminus \{q_1\} \tag{32c}$$
$$c(q_2) \geq 0 \tag{32d}$$
$$c(q_1) = 0 \tag{32e}$$

We conclude this subsection with a word of caution. Even though the distributed mechanism does not require "explicit" disclosure of delay costs, the arguments presented in this subsection show how a central authority could estimate these values based on the bundles communicated by the airlines during the algorithm iterations. The quality of the estimate depends on the number of different bundles observed: the fewer the number of bundles, the worse the estimates. In Section 6.4 we shows on an example data instance that the estimates obtained are rough in almost all cases, therefore there is no risk of cost information disclosure in practice.

4.2.2. Computation of the stepsize

Returning to the problem of choosing an appropriate stepsize for the subgradient method, a possible approach would be to compute an upper bound on $Z_{LR}(\lambda^t)$ and a lower bound on Z_{LP} based on the upper and lower bounds on values obtained in Section 4.2.1, and plug these bounds in equation (22). In particular, at each iteration t a lower bound $ZLB^t \leq Z_{LP}$ can be computed as

$$ZLB^t = \max \sum_{f \in \mathcal{F}} \sum_{q \in Q_f} LB^t(f,q) x(f,q) \tag{33a}$$
$$\sum_{f \in \mathcal{F}} \sum_{q \in Q_f : q \ni k} x(f,q) \leq 1 \quad \forall r \in \mathcal{R}, k \in L_r \tag{33b}$$
$$\sum_{q \in Q_f} x(f,q) = 1 \quad \forall f \in \mathcal{F} \tag{33c}$$
$$x(f,q) \geq 0 \quad \forall f \in \mathcal{F}, q \in Q_f \tag{33d}$$

See Appendix D.1 for a proof that problem (33) indeed provides a lower bound on Z_{LP}. An upper bound $ZUB^t(\lambda^t) \geq Z_{LR}(\lambda^t)$ can be computed as

$$ZUB^t(\lambda^t) = \sum_{f \in \mathcal{F}} UB^t(f, q_f^{*t}) + SG^t \cdot \lambda^t \tag{34}$$

since $Z_{LR}(\lambda^t) = \sum_{f \in \mathcal{F}} V(f, q_f^{*t}) + SG^t \cdot \lambda^t$. It follows that $ZUB^t(\lambda^t) - ZLB^t \geq Z_{LR}(\lambda^t) - Z_{LP}$. Then the stepsize is computed as

$$\mu_t = \frac{\epsilon_t \big(ZUB^t(\lambda^t) - ZLB^t\big)}{\|SG^t\|^2} \tag{35}$$

Problem (33) has always a finite optimal value since a_f is a feasible solution and $LB^t(f, a_f) = 0$, so $ZLB^t \geq 0$. However, problem (29) can be unbounded and so it can happen that $ZUB^t(\lambda^t) = +\infty$, thus this method is not viable. In Ranieri (2010) a method is proposed to compute a looser lower bound on $V(f, q)$ without solving the linear problem (26), but he does not provide a way to compute an upper bound on $Z_{LR}(\lambda^t)$.

Now we propose another method to compute the stepsize, which instead is always viable. The idea is to estimate the difference $Z_{LR}(\lambda^t) - Z_{LP}$ directly, instead of estimating bounds on $Z_{LR}(\lambda^t)$ and Z_{LP} separately and then taking the difference. First of all, we can write

$$Z_{LR}(\lambda^t) - Z_{LP} = \sum_{f \in \mathcal{F}} V(f, q_f^{*t}) + SG^t \cdot \lambda^t - \max_{x \in S} \sum_{f \in \mathcal{F}} \sum_{q \in Q_f} V(f, q) x(f, q) \tag{36}$$

where S is the set of solutions x satisfying constraints (5b) and (5c). Thanks to (5c) we can write

$$Z_{LR}(\lambda^t) - Z_{LP} = SG^t \cdot \lambda^t - \max_{x \in S} \sum_{f \in \mathcal{F}} \sum_{q \in Q_f} \big(V(f, q) - V(f, q_f^{*t})\big)\, x(f, q) \tag{37}$$

The maximisation problem appearing in (37) involves differences between values $V(f, q) - V(f, q_f^{*t})$. A finite upper bound on these quantities is readily available thanks to equation (23). This gives a finite lower bound on $Z_{LR}(\lambda^t) - Z_{LP}$ (due to the minus sign in front of the maximum) that can be plugged in the stepsize formula.

More precisely, once we have $UB^t_{diff}(f, q) \geq V(f, q) - V(f, q_f^{*t})$ obtained as

$$UB^t_{diff}(f, q) = p^t(q) - p^t(q_f^{*t}) \tag{38}$$

we can solve the following linear problem

$$ZUB^t_{diff} = \max \sum_{f \in \mathcal{F}} \sum_{q \in Q_f} UB^t_{diff}(f, q) x(f, q) \tag{39a}$$

$$\sum_{f \in \mathcal{F}} \sum_{q \in Q_f : q \ni k} x(f, q) \leq 1 \quad \forall r \in \mathcal{R}, k \in L_r \tag{39b}$$

$$\sum_{q \in Q_f} x(f, q) = 1 \quad \forall f \in \mathcal{F} \tag{39c}$$

$$x(f, q) \geq 0 \quad \forall f \in \mathcal{F}, q \in Q_f. \tag{39d}$$

To simplify the notation, let us define the *residual*

$$RES^t = SG^t \cdot \lambda^t - ZUB^t_{diff} \tag{40}$$

which will satisfy $RES^t \leq Z_{LR}(\lambda^t) - Z_{LP}$ (see again Appendix D.1). Finally we set

$$\mu_t = \epsilon_t \frac{RES^t}{\|SG^t\|^2}. \tag{41}$$

It can be proved (see Appendix D.2) that $RES^t \geq 0$ and so the stepsize is non-negative, whenever the duality gap $Z_{IP} - Z_{LD}$ is zero. Therefore, in that case, there is no risk that a step of subgradient iteration moves in the opposite direction of $-SG^t$.

The bounds $UB^t_{diff}(f,q)$ obtained with (23) can be improved by exploiting the ordered structure of Q_f (the monotonicity of cost functions). They are updated recursively with

$$\widehat{UB}^t_{diff}(f,q_i) = \begin{cases} p^t(q_i) - p^t(q_f^{*t}) & \text{for } i = 1 \\ \min\left(p^t(q_i) - p^t(q_f^{*t}), \widehat{UB}^t_{diff}(f,q_{i-1})\right) & \text{for } i = 2,\dots,|Q_f| \end{cases} \tag{42}$$

Then we can leverage the information obtained at previous iterations to write

$$UB^t_{diff}(f,q_i) = \min\left(\widehat{UB}^t_{diff}(f,q_i), UB^{t'}_{diff}(f,q_i)\right) \quad \forall q_i \in Q_f \text{ for } t' < t \text{ if } q_f^{*t'} \geq q_f^{*t} \tag{43}$$

Of course, the tightest possible upper bound on $V(f,q) - V(f,q_f^{*t})$ can be found by solving a linear problem analogous to (29)

$$UB^t_{diff}(f,q_k) = \max v(q_k) - v(q_f^{*t}) \tag{44a}$$
$$v(q_i) - v(q_j) \geq c^t_{ij} \quad \forall q_i \in Q_f, q_j \in Q_f \tag{44b}$$
$$v(q_{i-1}) \geq v(q_i) \quad \forall q_i \in Q_f \setminus \{q_1\} \tag{44c}$$
$$v(a_f) = 0 \tag{44d}$$

but the combination of (42) and (43) allows to drastically reduce the computational cost, and will be used to present the results in Section 6.

Similarly one could find a lower bound $LB^t_{diff}(f,q) \leq V(f,q) - V(f,q_f^{*t})$ and use it to compute an upper bound on $Z_{LR}(\lambda^t) - Z_{LP}$ to be used in the stepsize formula. However this does not always work because there could not exist a feasible solution $x \in S$ for which $\sum_{f\in\mathcal{F}} \sum_{q\in Q_f} LB^t_{diff}(f,q)x(f,q) > -\infty$.

The downside of this method is the computational cost of solving a linear problem (39) at every iteration of the subgradient method.

5. Data collection

5.1. Traffic data

We tested the model on real traffic data collected from the Demand Data Repository (DDR2) of EUROCONTROL. In particular (Niarchakou and Sfyroeras, 2021),

- Each regulation is characterised by its associated traffic volume[2], the sub-periods in which it is divided and the capacities of each sub-period.
- Each flight is characterised by: aircraft type, airline, ATFM delay, most penalising regulation, departure airport and destination airport, Estimated Take-Off Time (ETOT) and Estimated Time of Arrival (ETA), and the list of *intersections*, i.e., the estimated times of entry into each sector crossed along the flight route according to the *Initial Trajectory*, also called M1 trajectory, which is based on the Last Filed Flight Plan from the aircraft operator.

[2] A *traffic volume* is a tool used in ATFM to select a specific volume of air traffic. It is related to an aerodrome or an airspace sector, and to one or more traffic flows that can be either included or excluded from the regulation.

5.2. Data extraction

We processed the data about all flights and regulations on the 4th of July 2019 in all Europe. There were a total of 39080 flights and 203 regulations during that day. We selected data relative to one day because there are typically few regulations at night, so it is unlikely to have a regulation straddling the midnight.

As a preprocessing stage, we excluded flights having the same origin and destination airport, flights with unknown origin or destination airport, and regulations with zero capacity. The resources used to execute each flight f are the intersected airspace sectors, the departure airport and the destination airport; we are interested only in the regulated resources. In order to form the set R_f (see Section 3.1), we selected the resources for which the following three conditions hold:

(i) the resource is regulated;

(ii) the time of entry into the resource falls within the regulation period;

(iii) the flight is not excluded from the regulation by the corresponding traffic volume definition.

In the case the resource is an airspace sector, the time of entry in point (ii) is the ETO over the sector taken from the intersections; in the case the resource is the departure airport, the time of entry is the ETOT, and in the case the resource is the arrival airport, the time of entry is the ETA. The times of entry in each resource will constitute the inputs E_f of the model together with R_f. Point (ii) requires another specification: if the flight is subject to more than one regulation, we actually require that the interval $[e_i, e_i + MaxDel_f]$ (where e_i is the time of entry) intersects with the regulation period, because due to the delay in the Most Penalizing Regulation the flight may be pushed inside the regulation period even if the time of entry was before the start of the regulation. That is why in Section 3.1 we included a "dummy" time window at the beginning of a regulation, and not only at the end.

Point (iii) means that we check whether the flight is captured by the flows associated to the traffic volume. In addition, if the resource is the departure airport, we check that the traffic volume does not capture only inbound flights, and vice versa for the destination airport.

Since airspace sectors are sometimes non-convex, it can happen that a flights enters into a sector more than once. In these cases, we kept only the first intersection with the sector.

Regulations can be applied to collapsed sectors, but intersections of flights are always with elementary sectors. We verified that for our test day (4 July 2019), for each regulation associated with a collapsed sector, the opening schemes indicate that all elementary sectors of that collapsed sector were active during the time interval in which that regulation was in force. Since an elementary sector can be included in more than one collapsed sector, but only one of them can be active at a given time, we could associate each collapsed sector with the elementary sectors composing it (the configuration, i.e. the structure of collapsed sectors, was available from the data) and we considered a flight subject to a regulation on a collapsed sector if the flight crosses an elementary sector inside it.

Let F be the subset of flights which are found to be associated to at least one regulation according to our analysis, i.e. the flights for which $R_f \neq \varnothing$. We obtained $|F| = 11354$. Finally, let $F' \subset F$ be the subset of flights departing from outside the ATFM Area (which includes States receiving the full ATFM service from EUROCONTROL) and the ATFM Adjacent Area (which includes FIRs adjacent to the ATFM Area) and exempted flights (e.g. official, humanitarian and emergency flights). These categories of flights are not subject to ATFM measures. We did not include airborne flights (which are also not subject to the ATFM slot allocation) in F' because the data did not specify the time of creation of regulations. We obtained $|F'| = 1016$. For each flight $f \in F'$, for each $r_i \in R_f$, we blocked the TW of r_i containing e_i, meaning that no other flight can be assigned this TW in the optimal allocation. Flights in F' will not participate in the market mechanism. If a flight *does not wish* to join, but prefers to keep

its FPFS time window, it can be excluded from the market by placing it in the set F' (see also Appendix A). For each $f \in F \setminus F'$, a bundle containing any blocked time window is considered not feasible.

On our test day, on average, a flight in $F \setminus F'$ crosses 1.6 regulations and 39% of flights are subject to more than one regulation (see Figure 1, which shows the number of regulations crossed by a flight). Figure 2 shows a cumulative histogram of the number of flights subject to a regulation. The average is 85, and only 16% of cases comprise more than 150 flights. Based on these histograms, to test the market mechanism on meaningful real instances, in Section 6 we used the examples of two regulations with a number of flights each not too small but also not too large as the latter cases are rare (see also Table 3). Applications on larger instances can be found in Appendix E. We extended the analysis of traffic data for seven days (from the 1st to the 7th of July 2019) and found that a regulated flight crosses on average 1.9 regulations and 47% of regulated flights are subject to more than one regulation, thus motivating the need for a multiple regulation allocation system such as the one proposed in this work.

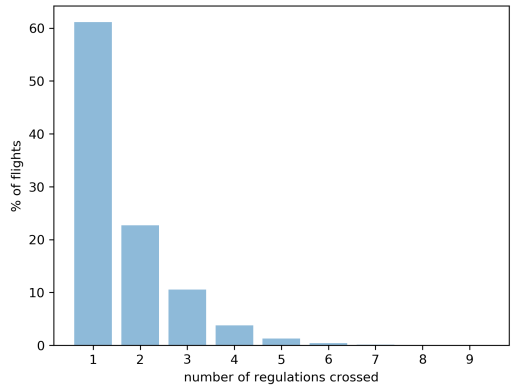

Figure 1: Normalised histogram of the number of regulations crossed by flights in $F \setminus F'$, 4 July 2019

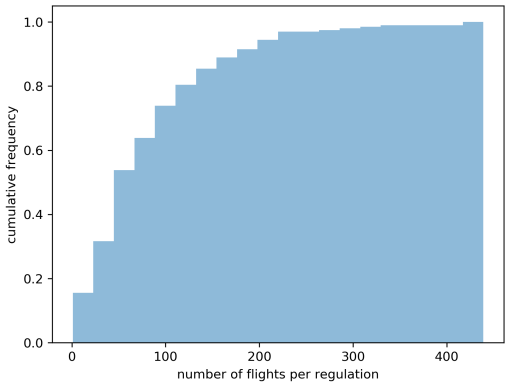

Figure 2: Normalised cumulative histogram of the number of flights in $F \setminus F'$ subject to a regulation, 4 July 2019

In principle, it would have been possible to use the ATFM delay from the data to obtain the FPFS TW allocation. However, the resulting allocation would not respect the capacities of time windows, because it was impossible to reproduce exactly the real environment due to the intricacies of the ATM system rules, which we tried nevertheless to take into account to the best of our possibilities. Thus we applied the algorithm in Appendix C instead.

As a check, we compared the set R_f obtained for each flight with the Most Penalising Regulation indicated in the data. For 98% of the flights the MPR was included in R_f. In addition, there was a small number of flights for which we found $R_f \neq \varnothing$ but the MPR was not present in the data, meaning that the flight was not subject to any regulation in reality; we decided to ignore these flights.

For reasons of computational efficiency, the size of Q_f was limited by setting $MaxDel_f = 60$ min $\forall f \in \mathcal{F}$. This is reasonable since the typical ATFM delays are below 60 minutes.

As a last note, it is straightforward to adapt the construction of TW allocation lists for regulations having more than one sub-period. We build a TW list as described in Section 3.1 for each sub-period, and then the allocation list L_r is the union of all these TW lists.

5.3. Cost data

As anticipated in Section 2.2, delay costs can be estimated relying on the values reported in Cook et al. (2021). To reflect the likely range of costs, they are assigned under three scenarios ("low", "base" and "high") for four flight phases (at-gate, taxi, en-route and arrival management), and are calculated for 18 common types of aircraft. In this work, we only consider the at-gate tactical delays.

To assign delay costs to flights whose aircraft is not among the 18 reference types, aircraft in the data were clustered in 18 groups whose centroid is the reference type, based on the square root of the Maximum Take-Off Weight (MTOW), as in Bolić et al. (2017).

Flights of low cost airlines were assigned to the low cost profile, flights into a hub airport were assigned to the high profile, and all other flights to the base profile, as in Bolić et al. (2017).

We attached the cost of delay specifically to each regulated flight, based on the aircraft type, company, destination, and flight length, which are available from the data. We also attached the cost of cancellation which was estimated in Cook et al. (2021) as well. Due to the clustering, many flights have the same cost of delay, which is unrealistic, thus we added a small Gaussian zero-mean noise to each cost $C(f, q)$.

6. Computational results

All algorithms were coded in Python and making use of NumPy, a package for scientific computing. All experiments were performed using the FICO XPRESS optimisation software, version 8.12.3. It is a software specifically devoted to solving mixed-integer linear programming problems. We ran it on a 64 bit Intel(R) Xeon(R) W-2145 @3.70GHz 16 core CPU computer, having 31 GB of RAM memory and Ubuntu 20.04 operating system. On this architecture, the most computationally demanding example instance presented here (Section 6.5) took 105 seconds to execute.

First of all, we recall the formula (41) for the stepsize in the subgradient method elaborated in Section 4.2.2:

$$\mu_t = \epsilon_t \frac{RES^t}{\|SG^t\|^2}. \tag{45}$$

In addition to formula (45), we experimented with a heuristic method to adapt the stepsize, which we now describe. The idea is not to use RES^t directly in the stepsize formula, but just to monitor the trend of the Lagrangian function during iterations. We initialise μ_0 as in (45). Then, we half μ_t every time the objective function fails to improve after a given number of iterations n according to the estimate given by RES^t, and at least n iterations have passed since the last time we halved μ_t. That is:

$$\mu_{t+1} = \begin{cases} \frac{\mu_t}{\gamma} & \text{if } RES^{t-i} \geq RES^{t-n} \text{ and } \mu_t = \mu_{t-i} \quad \forall i = 0, \dots, n-1 \\ \mu_t & \text{otherwise} \end{cases} \tag{46}$$

with $\gamma = 2$, and $\mu_0 = \epsilon_0 \frac{RES^0}{\|SG^0\|^2}$. This piece-wise constant stepsize can be seen as a modification of a common rule used in the subgradient method, which halves the stepsize every ν iterations for some fixed ν. In this case, ν is dynamically adapted based on the behaviour of the function. The parameters n and ϵ_0 need to be tuned, and we found that $n = 4$ and $\epsilon_0 = 3$ work well in general. We will use these values in the rest of this Section if not otherwise stated.

A rule of thumb to initialise λ^t is to set λ^0 at random uniformly in an interval corresponding roughly to the range of equilibrium prices, which can be determined after a number of trials on similar-sized instances. We observed that generally setting λ^0 at random works better than setting $\lambda^0 = 0$. We believe that this is because in this way initial prices are on average closer to their final equilibrium value, which makes convergence faster.

To illustrate the computational experience gained, we describe the results obtained on some instances involving two regulations and extracted from the real data (Section 5), which are representative of the average case (see Figures 1 and 2). We first discuss the convergence of the subgradient depending on the different stepsizes (45) - (46) and conclude that the best choice is to rely on the heuristic approach (section 6.1). This is the stepsize used in the subsequent examples. In the first case, we present an instance producing a weakly budget balanced solution (section 6.2), then a case where the subgradient does not converge and therefore TW capacity is not respected (section 6.3). Since we observe a decreasing

number of violated time windows as the subgradient advances, we deem the final solution acceptable from an operational point of view. In the third case, the optimal solution obtained through the market mechanism is illustrated in more detail (section 6.4). We conclude with a greedy algorithm to derive a feasible solution when the duality gap is not equal to zero (section 6.5). Appendix E reports some additional computational experiments performed on data instances of various sizes.

6.1. Subgradient convergence - Regulations LBSAU04 and LBSCU04

First of all, we consider two regulations, LBSAU04 and LBSCU04, affecting two collapsed sectors in Bulgaria, that were both limiting the rate to 40 flights per hour due to adverse weather, the first from 13:00 to 14:31 and the second from 13:00 to 16:15. We take as $\mathcal{F}$ the set of all flights affected by at least one of the two regulations, and $\mathcal{L}$ the set of time windows on the two regulations. The set $\mathcal{F}$ comprises a total of 111 flights, of which 15 flights were affected by both regulations. The cost of the initial FPFS allocation is € 960.20. We verified that the duality gap for this instance is zero, and the cost of the TW allocation (5) is € 263.36 leading to 72.6% cost savings.

Figure 3 shows with a blue line the decrease in the Lagrangian function in the first 100 iterations of the subgradient algorithm applied to such data instance, using stepsize (45). After 100 iterations, the objective value reaches 100.83% of its minimum value (the pink dotted line), and the prices are still not exactly at their equilibrium. The algorithm does not converge to an optimal solution of problem (IP) even after 300 iterations.

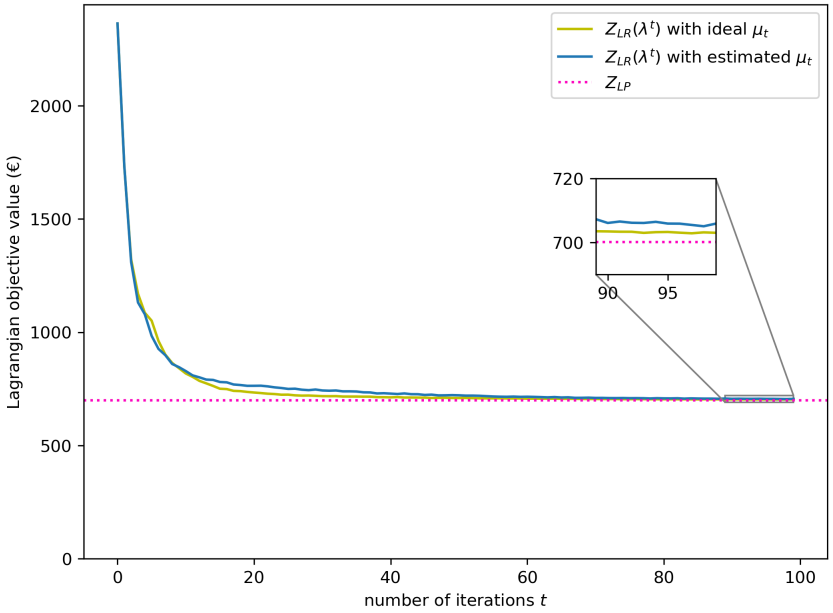

Figure 3: Lagrangian objective value $Z_{LR}(\lambda^t)$ vs number of iterations t (with zoom over the last 10 iterations)

This behaviour is not caused by the approximation we did by employing the lower bound in the stepsize formula. As a comparison, we report in the same plot (Figure 3) the result when we run the subgradient algorithm using the ideal stepsize (22), pretending that costs are known (yellow line). The convergence profile is very similar. In this case also, convergence to an optimal solution of (IP) does not happen, even if the final value is slightly lower, 100.42% of the minimum.

We found that this is a general trend on other data instances. Using the ideal formula (22) for the stepsize, the subgradient algorithm rarely converges to a capacity-compliant solution within a reasonable number of iterations, because prices converge too slowly.

In instances for which the duality gap is zero, the subgradient algorithm always converges as discussed in Section 4.1.1, at least asymptotically if the ideal stepsize is used.

Notice that we used the highest value possible for ϵ_t that guarantees convergence, i.e. $\epsilon_t = 2$. If we use a higher value, on some instances a faster convergence can be achieved, but on some others the objective value diverges. With (45) we used $\epsilon_0 = 3$ to compensate for the fact that RES^t is an underestimate of the true residual.

When we use the heuristic formula (46) on our example instance (Figure 4) the algorithm finds an optimal solution of (IP) at iteration 83 and terminates. We often achieved finite termination at optimality on other small-size instances (two regulations and comparable number of flights) with the heuristic formula.

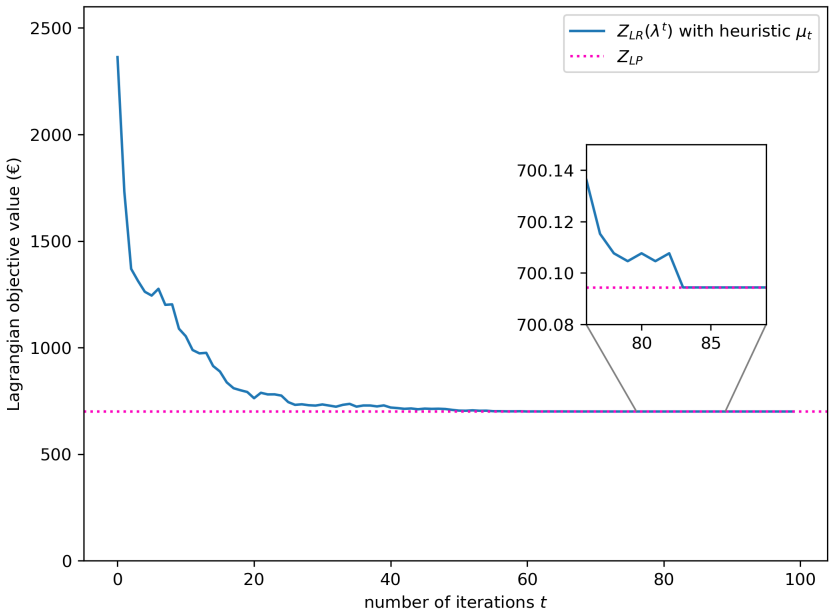

Figure 4: Lagrangian objective value $Z_{LR}(\lambda^t)$ vs number of iterations t (with zoom where finite convergence is attained)

Figure 5 shows the true value of the residual $Z_{LR}(\lambda^t) - Z_{LP}$ together with its underestimate RES^t. It is evident that this underestimate is able to reproduce the trend in the real residual. When the real residual increases, the underestimate increases too and when the real residual decreases, the underestimate decreases, even if the magnitude of the two is quite different. This provides justification to the empirical formula (46), as the underestimate is able to correctly capture when the real objective function fails to improve for n iterations.

Figure 6 shows the stepsize sequence obtained when we run the subgradient algorithm with the ideal stepsize rule (22), and the stepsize sequence obtained when we use the heuristic rule (46). The heuristic stepsize tends to be higher, and we believe that this is the reason of the success of this formula: if the magnitude of the step is higher, prices λ^t tend to converge faster, provided that they converge at all. Of course, the drawback is that convergence is not guaranteed, so this represents a more aggressive strategy for scheduling the stepsize than (22), or actually its version (45) in the case of unknown costs, but less robust.

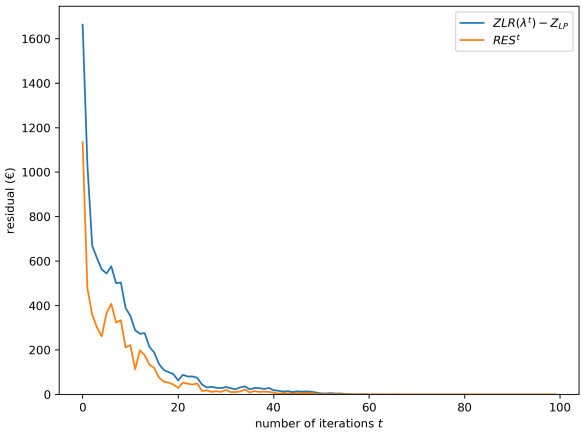

Figure 5: Goodness of the lower bound on the residual

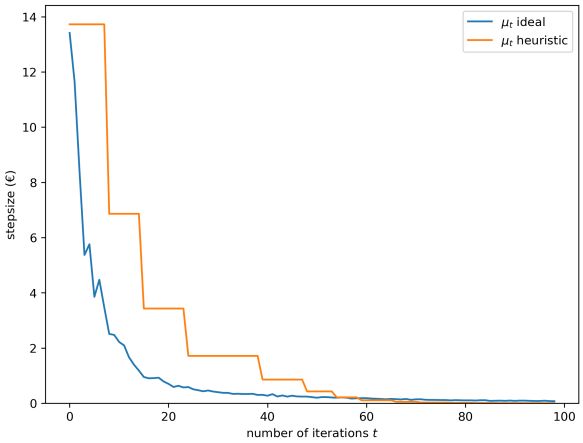

Figure 6: Comparison between stepsize sequences

6.2. Weakly budget balanced solution - Regulations LOVS04N and LOWB304A

We present an example that shows that the subgradient algorithm finds a capacity-compliant solution that is also weakly budget balanced, even if it does not satisfy the complementary slackness conditions.

The example is relative to two interacting regulations, LOVS04N and LOWB304A, and a total of 145 flights affected by at least one of them. We verified that the duality gap for this instance is zero.

Figure 7 displays the converge profile on this instance, with a zoom on the last iterations to show that convergence is not attained at the minimum of the Lagrangian function, but at a slightly higher value. Nonetheless, at iteration 100 the solution of the Lagrangian problem turns out to be capacity-compliant, and also weakly budget balanced, even if it does not satisfy complementary slackness. Although the solution is not optimal, the subgradient algorithm may be stopped, as the resulting market mechanism implied by the current solution satisfies the desired properties.

We remark that we can assess the degree to which this near-optimal solution departs from optimality. Specifically, we have that $Z_{IP} - \sum_{f \in \mathcal{F}} V(f, q_f^{*t}) \leq \lambda^t \cdot SG^t$ (Geoffrion, 1974).

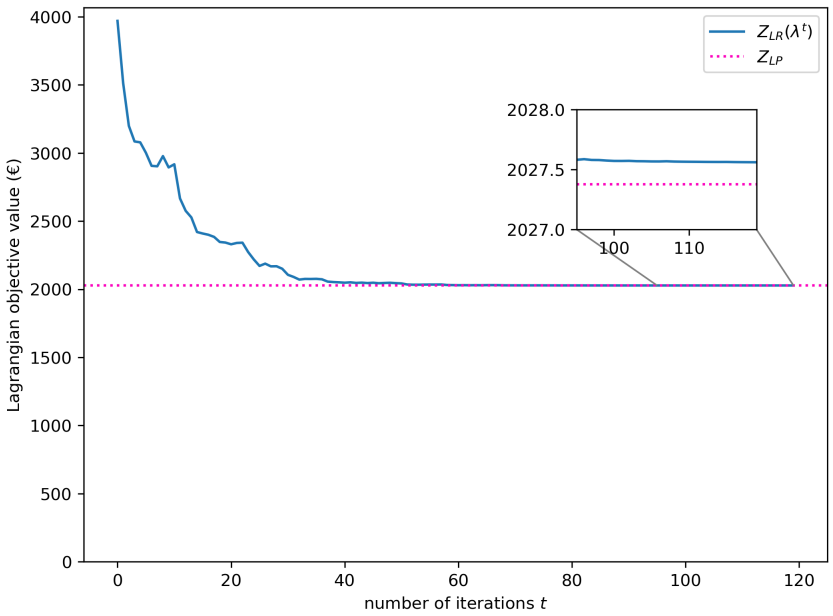

Figure 7: Convergence profile (with detail of the last iterations)

6.3. *Capacity constraint violation - Regulations LHSUH04A and LONE3504*

We introduce an example that shows that when the subgradient algorithm does not converge to a solution which respects the capacity constraint, often the violation of the capacity constraint is small. We consider the regulations LHSUH04A and LONE3504 which affect a total of 233 flights. Figure 8 shows the convergence profile for our third example instance. The Lagrangian objective value reaches 100.59% of its minimum after 150 iterations.

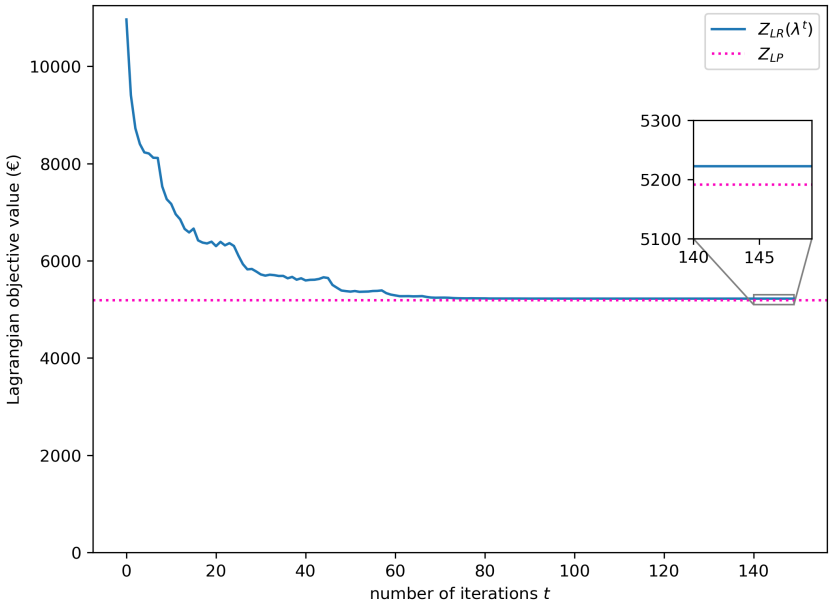

Figure 8: Convergence profile

As we discussed in Section 4.1, due to the way the subgradient algorithm is designed, it tends to reduce capacity violations during the course of iterations. A step of the subgradient algorithm tends to raise the price of time windows for which the demand exceeds the capacity, so that at the next iteration the demand will be lower.

More quantitatively, let us define the *total overload* of a time window allocation as the total number of flights exceeding the capacity of time windows (i.e. 1 flight per time window). Recall that the number of flights whose demanded bundle at iteration t contains time window k is $1 - SG_k^t$. Then, at iteration t, the overload OL_k^t of time window k is given by $OL_k^t = \max(0, -SG_k^t)$ and the total overload is given by $\sum_{k \in \mathcal{L}} OL_k^t$.

Figure 9 reveals that the total overload indeed tends to decrease during iterations. If the subgradient algorithm would converge to an optimal solution, then the total overload would be zero. At the same time, as shown in Figure 10, the surplus $\sum_{f \in \mathcal{F}} \left(p^t(q_f^t) - p^t(a_f) \right)$ tends to increase, and after 60 iterations it rises above zero.

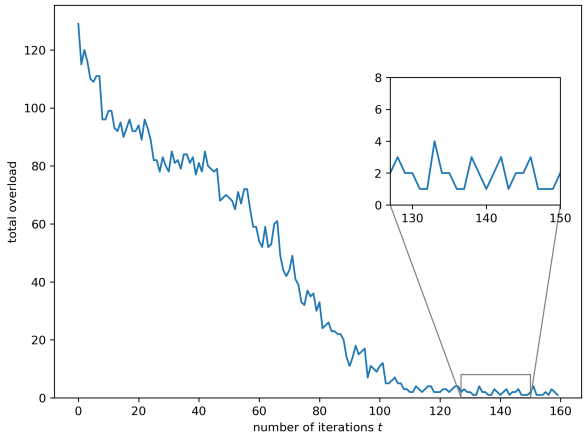

Figure 9: Total overload

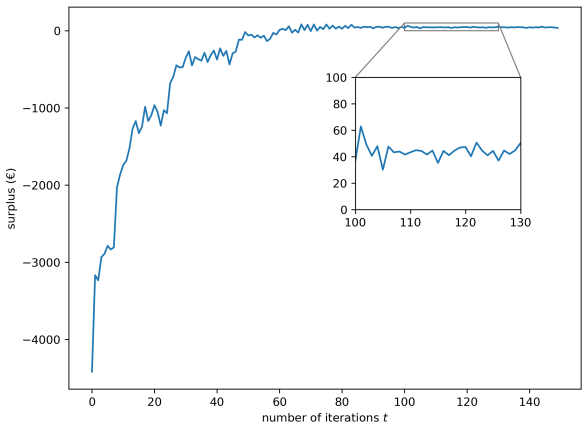

Figure 10: Surplus

At iteration $t = 131$, there is only one flight in excess with respect to the capacity, i.e. the total overload is 1, and the surplus is positive, in particular it amounts to 44.57 €. Although one flight could not be appropriately accommodated by the algorithm, in practice it might happen that the ATFM controllers and/or the airlines could re-allocate such flight in cost-efficient manner. Therefore, even with this small capacity violation, we can be content with this solution.

To sum up what we discussed in this section and in Section 4.1.1, these are the factors that prevent convergence of the subgradient algorithm to an optimal solution of our allocation problem:

1. The duality gap may be non-zero, in which case the prices which solve the Lagrangian dual do not support the optimal allocation.

2. There may be multiple solutions of the Lagrangian relaxation subproblem, so even when the duality gap is zero and the prices have converged exactly, the subgradient algorithm may not find the optimal allocation.

3. Even if costs were known and the ideal stepsize sequence was used, finite convergence would not be guaranteed, and asymptotic convergence does not yield the solution to the allocation problem.

4. By using an heuristic stepsize rule, in not all cases finite convergence is attained.

When the subgradient algorithm stops without converging to the optimal solution, two situations can arise. Either the subgradient algorithm solution provides a set of bundles that do not cause slot overloads (i.e., no time windows are assigned to more than one flight), or conversely, some slot overload does occur (i.e., some time windows are assigned to more than one flight). In the former case, flights may decide to accept the proposed solution if it is weakly budget balanced. In the latter case, the presence of slot overloads can be handled with the heuristic presented in Section 6.5.

We also note, as we have already pointed out, that it is not uncommon in operations to accept some slot overloads, i.e., ~~more than one flight in the same time window, are accepted~~ two flights in the same sequence time window are often accepted in a sector, especially if there is a nearby empty slot, typically in the range between -20 min and +60 min, that can compensate for the overload. Therefore, the acceptability of this type of "imperfect" solution may be worth further investigation in the future.

6.4. Market mechanism solution - Regulations ME1204 and MKK04

Here we introduce an example on an instance of smaller size (in terms of number of flights), for which the space limitation of the paper allows us to report the solution.

The interested regulations are ME1204 and MKK04, affecting two en-route sectors near Marseille, France. The regulation reason for both was ATC Staffing. Regulation MKK04 was active from 11:40 to 13:00 with a capacity of 38 flights/hour, and regulation ME1204 was active from 12:00 to 13:40 with a capacity of 30 flights/hour. There were 39 flights subject to regulation ME1204 and 40 flights subject to MKK04, of which 15 to both of them, for a total of 64 flights. Regulation ME1204 has 50 time windows and MKK04 has 51 time windows. At first sight, it may seem weird that the number of flights subject to a regulation is smaller than the number of TWs available. But the point is that, if no delay was assigned to flights, some time windows would have more than one flight passing through them, and some other time windows would have no flights at all: the purpose of a regulation is to smooth the traffic over the regulation period.

With the subgradient algorithm an optimal solution is found after only 37 iterations. The total delay under the FPFS allocation is 105 minutes and the total cost of delay is 1159.90€. Under the optimal allocation the total delay is slightly higher, 106 minutes, but the cost is almost halved, 619.72€. The surplus from the market mechanism is 7.26€. The sum of profits (utility variations) of all flights is thus $\sum_{f\in\mathcal{F}} \Delta u(f) = \sum_{f\in\mathcal{F}} (V(f, q_f^*) - p(q_f^*) + p(a_f)) = 1159.90€ - 619.72€ - 7.26€ = 532.92€$.

It is interesting to look closely at the optimal solution. Table 2 lists all the monetary transactions between flights and the central authority (CA) prescribed by the market mechanism. The first column is the TW being exchanged, the second column is the flight selling the TW (releasing the TW from its FPFS allocation), the third column is the flight buying the TW (taking up the TW under the optimal allocation), and the fourth column is the payment associated with the TW exchange, i.e. the price of the TW (see Section 3.3). Time windows not appearing in the table either are not assigned under the FPFS policy nor under the optimal policy, or they are assigned to the same flight in both policies (we omitted virtual transactions from a flight to itself from the table). Out of the total of 64 flights, 24 flights were allocated with the same time windows under both FPFS and optimal policy. These are the flights who are willing to participate in the process but for whom it is most convenient to keep the FPFS bundle.

As an example, consider the two rows of the table involving time windows r1 k48 and r1 k49. Flights f28 and f54 exchange their FPFS TWs; f28 increases its delay and thus receives a net amount of 23.12€ - 10.8€ = 12.32€ from f54 as a compensation. In many cases the transactions occur between a pair of flights swapping a time window, as in the example, but in other cases they consist in more complex trading cycles involving three or more flights.

Most trades occur between flights, and only 5 trades occur between a flight and the central authority (see again cases (ii) and (iii) in Section 3.3). Of these, all are 0€ except one in which flight f63 pays 7.26€ to the central authority for TW r1 k18 and this represents the surplus of the market mechanism. In the last case, we are in the presence of an airline that is willing to pay for a currently free TW. In such a situation, the central authority may decide to allocate it free of charge, so that in practice the exchange would be fully budget balanced. However, a similar policy should be agreed upon with all airlines in advance. In fact, reducing the price paid by f63 for TW r1 k18 from 7.26€ to 0.00€ may be unfair, since other flights may then prefer this TW at zero price to their allocated TW, and (19) would not hold.

We remark that some TW exchanges or some central authority TW allocations may naturally occur with price equal to 0€. This is always the case when a flight sells a TW to the central authority, due to the complementary slackness condition. It can also happen when a flight exchanges a pair of slots with another flight. For instance, considering Table 2 again, f18 sells its FPFS time window r2 k5 at 0€ to f19 so that this f19 can sell r2 k4 to f18 and allows it to decrease its delay. In this type of TW exchange, and in general in circular TW exchanges, the difference in TW prices may be more important than the price of each individual TW, especially if no other flight is interested in the TWs being considered. As another example, the circular exchange of TWs r2 k45, k46, and k47 between f3, CA, and f23 at 0€ is justified even if f3 gets a TW with a greater delay. It allows f3 to be compensated in regulation r1, where

its profit is positive because it receives compensation of 24.03€ for the increase in delay.

TW	seller	buyer	payment
r1 k1	f33	f48	22.81€
r1 k3	f48	f11	0.00€
r1 k4	f24	CA	0.00€
r1 k5	f29	f24	2.70€
r1 k6	f11	f29	0.00€
r1 k18	CA	f63	7.26€
r1 k21	f63	f13	0.00€
r1 k22	f13	CA	0.00€
r1 k23	f16	f23	10.80€
r1 k24	f23	f16	0.00€
r1 k30	f3	f14	24.03€
r1 k31	f14	f59	10.80€
r1 k32	f59	f3	0.00€
r1 k37	f41	f27	12.65€
r1 k38	f27	f38	10.75€
r1 k39	f38	f41	0.00€
r1 k44	f62	f26	8.99€
r1 k45	f26	f5	7.27€
r1 k46	f5	f62	2.70€
r1 k48	f28	f54	23.12€
r1 k49	f54	f28	10.80€
r2 k4	f19	f18	5.40€
r2 k5	f18	f19	0.00€
r2 k7	f10	f33	76.36€

TW	seller	buyer	payment
r2 k10	f33	f48	46.51€
r2 k11	f24	f2	39.56€
r2 k12	f48	f10	21.61€
r2 k13	f2	f24	10.80€
r2 k14	f43	f34	5.40€
r2 k15	f22	f43	1.01€
r2 k16	f34	f22	0.34€
r2 k24	f60	f12	143.94€
r2 k25	f12	f60	104.63€
r2 k28	f36	f44	30.51€
r2 k31	f15	f17	17.53€
r2 k32	f44	f63	21.27€
r2 k33	f17	f36	4.05€
r2 k34	f63	f15	2.70€
r2 k35	f16	f20	2.03€
r2 k36	f20	f52	13.94€
r2 k37	f52	f49	2.38€
r2 k38	f49	f16	0.68€
r2 k45	f3	f23	0.00€
r2 k46	f23	CA	0.00€
r2 k47	CA	f3	0.00€
r2 k48	f50	f61	44.34€
r2 k49	f61	f50	22.53€
r2 k50	f21	f1	13.59€

Table 2: Transactions (in the first column, "r1" stands for ME1204, "r2" stands for MKK04; "k1" is the first TW, "k2" is the second etc.)

Finally, we show how strict are the bounds on costs obtained by the central authority and used to compute the stepsize. Figure 11 shows a histogram of the ratios $\widetilde{LB}^t(f,q)/C(f,q)$ for the last iteration $t = 37$ for all bundles $q \in Q_f, f \in \mathcal{F}$. The bounds $\widetilde{LB}^t(f,q)$ are computed as in problem (32). For 96% of the bundles, the lower bound is less than 90% of the true cost value. Figure 12 shows a histogram of $C(f,q)/\widetilde{UB}^t(f,q)$ for $t = 37$. The first bin of the histogram highlights that for 86% of the bundles no upper bound can be obtained (i.e. problem (32) is unbounded). This means that in practice the cost elicitation procedure described in Section 4.2.1 does not pose a privacy risk for the airlines.

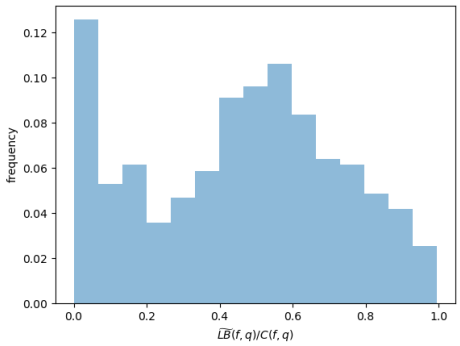

Figure 11: Histogram of the ratio between the lower bound on the cost and the true cost

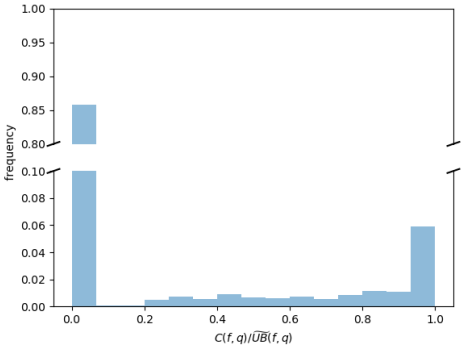

Figure 12: Histogram of the ratio between the true cost and the upper bound on the cost (broken y axis)

Appendix E.1 reports some additional computational experiments performed on real instances of different sizes with the duality gap equal to zero.

6.5. Resolution of duality gaps

The duality gap was always zero in all the previous examples. However, our experiments confirm the expectations that the percentage of problem instances for which the duality gap is zero decreases as the instance size (in terms of number of flights and regulations) increases. Here, we discuss how to handle these situations, and also the cases where the subgradient algorithm returns a solution that provides bundles that cause some slot overloads.

A straightforward way to find a feasible solution is to allow only a subset of the flights to exchange time windows, while the rest of the flights will maintain their FPFS allocation. We run the subgradient algorithm until the objective value is close to its minimum. We consider the time windows for which the overload is $OL_k^t > 0$. For each of these, we remove from the market mechanism one of the OL_k^t flights demanding that time window at time t, and assign the FPFS bundle to them. This means that the size of the set $\mathcal{F}$ on which the optimisation is performed is reduced. The time windows belonging to these FPFS bundles are removed from the set $\mathcal{K}$ of time windows available for exchange, meaning that the feasible bundles of all flights in $\mathcal{F}$ are filtered from these. Then, the subgradient algorithm is run again on this reduced instance, with the hope that the duality gap will be now zero. In the case it is, the feasible solution of the reduced optimisation problem is implemented. In the case the duality gap is still different from zero, the above procedure of constructing a smaller instance is repeated one or more times. This procedure obviously guarantees to terminate with a feasible solution, since in the limit all flights are removed from $\mathcal{F}$ and the FPFS allocation for all flights is implemented. In practice, we found that just one iteration is often sufficient.

As an example, we used an instance with 5 regulations and 832 flights, depicted in Figure 13 as a graph. A node corresponds to a regulation in $\mathcal{R}$, and an edge is drawn between every pair of regulations which are connected by at least one flight, in the sense that the flight crosses the two regulations consecutively. The edge weight represents the number of connections. The delay cost of the initial FPFS allocation is equal to € 24364.04 and the savings from the TW allocation (5) are equal to 82.9%. The duality gap for this instance is 0.14.

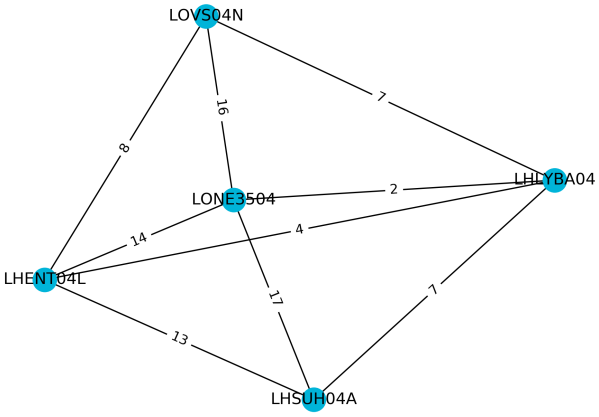

Figure 13: Instance structure

For such large instance, it is necessary to decrease the parameter γ appearing in equation (46), because convergence is expected to be slower with respect to the four cases encountered before. We set $\gamma = 1.2$. After 371 subgradient iterations on the entire instance, the objective function reaches 100.0002% of its minimum value and there are only 4 overloaded time windows. At this point, after removing the 4 flights in excess with respect to the capacity, the duality gap becomes zero. Restarting the subgradient algorithm, an optimal solution is obtained after 410 iterations, with an optimal cost of € 4184.67. By adding the cost of the FPFS bundles of the 4 removed flights, we obtain the cost of the feasible solution for the entire instance, € 4184.67 + € 6.79 = € 4191.46. This cost is very close to the optimal cost for the entire instance, € 4170.23. In other words, the value of this feasible solution is only 21.23€ lower than the value of the optimal solution. It is also worth noting that the prices obtained are close to the optimal prices. More precisely, for 70% of the time windows the price is less than 10% away from the optimal price. The maximum price is 179.81€, and 25% of the time windows have prices equal to zero.

Figure 14 shows a histogram of the non-zero prices.

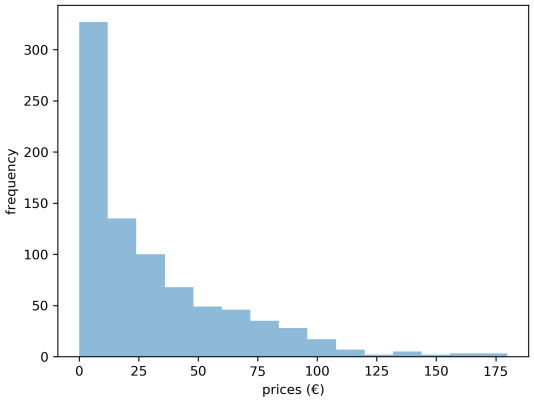

Figure 14: Histogram of non-zero prices

Appendix E.2 reports other computational experiments performed on real instances of different sizes with the duality gap different zero.

6.6. Concluding remarks on computational experiments

Summing up, the five examples introduced in this section provide some guidelines on how to deal with our problem in practice, when it is not known if we are dealing with zero dual-gap instances.

When the subgradient algorithm is stopped three situations may occur. If we find a feasible solution that satisfies the complementary slackness conditions, we are in presence of a zero dual-gap instance and the algorithm has individuated a Walsarian equilibrium. If the solution does not satisfy the complementary slackness conditions but it is feasible for (IP), this solution is capacity compliant and individual rational, but may or may not be weakly budget balanced (see Section 6.2 for a weakly budget balanced instance). If the solution is not feasible for (IP) the heuristic proposed in Section 6.5 can be applied.

Table 3 summarises the properties of the optimal solution ("opt. sol.") compared to the FPFS solution in the four cases with two regulations presented from Section 6.1 to Section 6.4, respectively. The total delay is not the sum of the two regulations' delay because the delay of flights passing through both regulations does not have to be double counted, and similarly for the number of flights and the cost. We notice that cost savings are always significant (from 33% to 85%), despite sometimes a small increase in the delay is experienced. The table also reports the number of flights delayed beyond the upper bound of each regulation period ("n° flights out of reg."), both in the FPFS and in the optimal allocation. These are flights occupying a dummy TW. Since their number, if any, is always very low, the presence of these TWs is not expected to pose problems from an operational point of view. Similarly to capacity violations addressed in Section 6.3, in practice it might happen that the ATFM controllers and/or the airlines could re-allocate such flights in a cost-efficient manner.

		section 6.1	section 6.2	section 6.3	section 6.4
		LBSAU04, LBSCU04	LOVS04N, LOWB304A	LHSUH04A, LONE3504	ME1204, MKK04
first regulation	n° flights	39	60	159	39
	cost FPFS	€ 319.82	€ 1,783.97	€ 5,421.41	€ 713.04
	cost opt. sol.	€ 121.13	€ 340.64	€ 1,005.92	€ 477.66
	% of change opt.sol. vs. FPFS	−62%	−81%	−81%	−33%
	delay FPFS	23 min	171 min	554 min	66 min
	delay opt. sol.	27 min	172 min	556 min	64 min
	n° flights out of reg. in FPFS	0	0	2	1
	n° flights out of reg. in opt. sol.	0	0	1	1
second regulation	n° flights	87	112	116	40
	cost FPFS	€ 746.96	€ 2,027.90	€ 1,755.64	€ 1,020.92
	cost opt. sol.	€ 236.26	€ 426.82	€ 264.08	€ 564.30
	% of change opt.sol. vs. FPFS	−68%	−79%	−85%	−45%
	delay FPFS	84 min	192 min	172 min	87 min
	delay opt. sol.	100 min	162 min	189 min	92 min
	n° flights out of reg. in FPFS	1	1	0	1
	n° flights out of reg. in opt. sol.	1	0	0	1
both regulations	n° flights	111	145	233	64
	cost FPFS	€ 960.20	€ 2,689.30	€ 6,328.80	€ 1,159.90
	cost opt. sol.	€ 263.36	€ 661.93	€ 1,121.86	€ 619.72
	% of change opt.sol. vs. FPFS	−73%	−75%	−82%	−47%
	delay FPFS	100 min	268 min	651 min	105 min
	delay opt. sol.	109 min	280 min	666 min	106 min

Table 3: Number of flights, cost and delay of the FPFS solution and the optimal solution, for the two single regulations and in total

7. Conclusions and future perspectives

In this paper, we propose a market-based mechanism for the allocation of time windows in case of multiple interacting ATFM regulations. We show that significant delay cost reductions are possible for airlines with respect to the First Planned First Served allocation policy currently adopted in Europe.

This market mechanism is allocative efficient, individual rational and weakly budget balanced. It is based on the Lagrangian relaxation of an integer optimisation problem that can be implemented in a distributed way with the subgradient method. We have successfully tested the mechanism on real data instances and discussed possible problems that can arise in convergence of the subgradient method. In particular, the choice of stepsize for the subgradient method is a delicate matter, which requires to find a compromise between fast convergence and "robust" convergence (i.e. convergence in all instances).

It should be noted that optimised hardware can partially help to resolve this trade-off, but the inevitable uncertainty that plagues cost values makes it questionable to spend excessive computational time on finding a solution whose value is approximated anyway.

We have shown that convergence to an optimal solution, or even just a feasible solution, is not always possible. A possible future development would be to try to design a Lagrangian heuristic (Fisher, 2004) to slightly modify nearly feasible Lagrangian solutions to satisfy the capacity constraint while maintaining individual rationality and weak budget balance.

Another possibility for future experiments could be to leverage the fact that delay costs are not completely unknown and that recent advances in predicting rerouting costs (Khan et al., 2021) provide reliable thresholds for airlines to assess whether the acceptance of the delay is economically viable. For example, before running the subgradient method, to set the stepsize sequence we could initialise prices of time windows with the optimal dual prices of the allocation problem solved with the estimated costs available by the central authority. This would facilitate and speed up convergence towards optimal solutions.

A challenging perspective would be to try to apply the model on a large-scale instance, possibly a whole day of air traffic data. The difficulty is that the duality gap for such large instance is probably non-zero. Also, this would require to find another rule for the stepsize, because solving a linear optimisation problem at each iteration would be too computationally expensive.

References

Barnhart, C., Bertsimas, D., Caramanis, C., Fearing, D., 2012. Equitable and efficient coordination in traffic flow management. Transportation science 46, 262–280.

Berechet, I., Debouck, F., Castelli, L., Ranieri, A., Rihacek, C., 2009. A target windows model for managing 4-d trajectory-based operations. AIAA/IEEE Digital Avionics Systems Conference - Proceedings , 3.D.21–3.D.29.

Bikhchandani, S., Mamer, J.W., 1997. Competitive equilibrium in an exchange economy with indivisibilities. Journal of economic theory 74, 385–413.

Bolić, T., Castelli, L., Corolli, L., Rigonat, D., 2017. Reducing atfm delays through strategic flight planning. Transportation Research Part E: Logistics and Transportation Review 98, 42–59.

Bolić, T., Castelli, L., Corolli, L., Scaini, G., 2021a. Flexibility in strategic flight planning. Transportation Research Part E: Logistics and Transportation Review 154, 102450.

Bolić, T., Castelli, L., Scaini, G., Frau, G., Guidi, S., 2021b. Flight flexibility in strategic traffic planning: visualisation and mitigation use case. CEAS Aeronautical Journal 12, 847–862.

Castelli, L., Corolli, L., Lulli, G., 2011a. Critical flights detected with time windows. Transportation research record 2214, 103–110.

Castelli, L., Pesenti, R., Ranieri, A., 2011b. The design of a market mechanism to allocate air traffic flow management slots. Transportation research part C: Emerging technologies 19, 931–943.

Castelli, L., Pesenti, R., Ranieri, A., 2011c. Short-term allocation of time windows to flights through a distributed market-based mechanism. Journal of Aerospace Operations 1, 29–40.

Cook, A., 2016. European air traffic management: principles, practice, and research. Routledge.

Cook, A.J., Tanner, G., 2015. European airline delay cost reference values.

Cook, A.J., Tanner, G., Anderson, S., 2004. Evaluating the true cost to airlines of one minute of airborne or ground delay.
Cook, A.J., Tanner, G., Bolic, T., 2021. D3.2 industry briefing on updates to the european cost of delay.
Dalmau, R., 2022. Predicting the likelihood of airspace user rerouting to mitigate air traffic flow management delay. Transportation Research Part C: Emerging Technologies 144, 103869.
Delgado, L., Gurtner, G., Bolić, T., Castelli, L., 2021. Estimating economic severity of Air Traffic Flow Management regulations. Transportation Research Part C: Emerging Technologies 125, 103054.
Fisher, M.L., 2004. The lagrangian relaxation method for solving integer programming problems. Management Science 50, 1861–1871.
Geoffrion, A.M., 1974. Lagrangean relaxation for integer programming, in: Approaches to integer programming. Springer, pp. 82–114.
Granberg, T., Polishchuk, V., 2012. Socially optimal allocation of ATM resources via truthful market-based mechanisms. 2nd SESAR Innovation Days, 27-29 November 2012, Braunschweig, Germany.
Han, F., Wong, W.B.L., Gaukrodger, S., 2010. Improving future air traffic punctuality:"pinch-and-pull" target windows. Aircraft Engineering and Aerospace Technology 82, 207–216.
Khan, W.A., Ma, H.L., Ouyang, X., Mo, D.Y., 2021. Prediction of aircraft trajectory and the associated fuel consumption using covariance bidirectional extreme learning machines. Transportation Research Part E: Logistics and Transportation Review 145, 102189.
Krishna, V., 2009. Auction theory. Academic press.
Lawler, E.L., Lenstra, J.K., Rinnooy Kan, A.H.G., 1980. Generating all maximal independent sets: Np-hardness and polynomial-time algorithms. SIAM Journal on Computing 9, 558–565.
Liu, Y., Liu, Y., Hansen, M., Pozdnukhov, A., Zhang, D., 2019. Using machine learning to analyze air traffic management actions: Ground delay program case study. Transportation Research Part E: Logistics and Transportation Review 131, 80–95.
Lulli, G., Odoni, A., 2007. The european air traffic flow management problem. Transportation science 41, 431–443.
Margellos, K., Lygeros, J., 2013. Toward 4-D Trajectory Management in Air Traffic Control: A Study Based on Monte Carlo Simulation and Reachability Analysis. IEEE Transactions on Control Systems Technology 21, 1820–1833.
Mehta, R., Vazirani, V., 2020. An incentive compatible, efficient market for air traffic flow management. Theoretical Computer Science 818, 41–50.
Myerson, R.B., Satterthwaite, M.A., 1983. Efficient mechanisms for bilateral trading. Journal of economic theory 29, 265–281.
Niarchakou, S., Sfyroeras, M., 2021. ATFCM Operations Manual.
Odoni, A.R., 1987. The flow management problem in air traffic control, in: Flow control of congested networks. Springer, pp. 269–288.
Performance Review Commission, 2019. Performance Review Report 2019. Technical Report. EUROCONTROL.
Pilon, N., Guichard, L., Cliff, K., 2019. Reducing Impact of Delays using Airspace User-Driven Flight Prioritisation. 9th SESAR Innovation Days, 2-6 December 2019, Athens, Greece.
Pilon, N., Ruiz, S., Bujor, A., Cook, A., Castelli, L., 2016. Improved flexibility and equity for airspace users during demand-capacity imbalance: An introduction to the user driven prioritisation process. 6th SESAR Innovation Days, 8-10 November 2016, Delft, The Netherlands.
Ranieri, A., 2010. Combinatorial exchange models for a user-driven air traffic flow management in Europe. Ph.D. thesis. Università degli studi di Trieste.
Rodríguez-Sanz, Á., Comendador, F.G., Valdés, R.M.A., Pérez-Castán, J.A., García, P.G., Godoy, M.N.G.N., 2019. 4D-trajectory time windows: definition and uncertainty management. Aircraft Engineering and Aerospace Technology 91, 761–782.
Rodríguez-Sanz, Á., Puchol, C.C., Pérez-Castán, J.A., Comendador, F.G., Valdés, R.M.A., 2020. Practical implementation of 4D-trajectories in air traffic management: system requirements and time windows monitoring. Aircraft Engineering and Aerospace Technology 92, 1357–1375.
Rosenthal, E., Eisenstein, E., 2016. A rescheduling and cost allocation mechanism for delayed arrivals. Computers and Operations Research 66, 20–28.
Ruiz, S., Guichard, L., Pilon, N., Delcourte, K., 2019a. A new air traffic flow management user-driven prioritisation process for low volume operator in constraint: Simulations and results. Journal of Advanced Transportation 2019.
Ruiz, S., Kadour, H., Choroba, P., 2019b. An innovative safety-neutral slot overloading technique to improve airspace capacity utilisation, in: 9th SESAR Innovation Days.
Ruiz, S., Kadour, H., Choroba, P., 2019c. A novel air traffic flow management model to optimise network delay, in: The 13th USA/Europe Air Traffic Management Research and Development Seminar, Vienna, Austria.
SESAR, 2019. SESAR Solution PJ07.02 SPR-INTEROP/OSED for V2 - Part I.
Sherali, H.D., Hill, J.M., McCrea, M.V., Trani, A.A., 2011. Integrating slot exchange, safety, capacity, and equity mechanisms within an airspace flow program. Transportation Science 45, 271–284.
Vossen, T., Ball, M., 2006a. Optimization and mediated bartering models for ground delay programs. Naval research

logistics (NRL) 53, 75–90.
Vossen, T.W., Ball, M.O., 2006b. Slot trading opportunities in collaborative ground delay programs. Transportation Science 40, 29–43.
Zhang, Q., Le, M., Xu, Y., 2021. Collaborative delay management towards demand-capacity balancing within User Driven Prioritisation Process. Journal of Air Transport Management 91, 102017.

Appendix A Discussion on incentive compatibility

In the paper, we assume the incentive compatibility property is always respected, i.e., all participants in the market report their preferences honestly. Some of the consequences that may occur if this assumption is violated are discussed below, extending similar considerations already made for the case of a single regulation, i.e., $|\mathcal{R}| = 1$ (Castelli et al., 2011b).

In the centralised mechanism (Section 3), each flight communicates the value of each bundle to the central authority which allocates them to all flights, maximising the overall value. Some airline could therefore be tempted to communicate false values to gain an advantage (i.e., declare costs greater than the real $C(f, q)$ in order to receive a better position in the optimal allocation). This possibility is however mitigated by the fact that, operating in a competitive environment, an airline would need to have perfect knowledge of other participants' costs in order to be sure that its utility would not decrease when cheating.

Another action that could be a consequence of dishonest attitudes is the rejection of the allocated bundle because it does not match the desired one. This possibility is mitigated by not requiring anyone to participate in the mechanism. Even if the market mechanism is individual rational (each participant has a non-negative profit from entering the market), there should be no obligation from an airline to participate in it. Exactly as with the UDPP (Pilon et al., 2016), if an airline does not want to participate, it remains with the slots allocated through the FPFS. Only those airlines that wish to be in the market participate. In our setting, this is possible by including in the set F' introduced in Section 5.2 (flights that are not requested to participate in the market mechanism) also the flights that *do not wish* to participate in it. Those who participate are then required to accept the solution provided. Of course, an airline that agrees to participate in the centralised mechanism could still provide false cost values to the authority. However, this case falls under the one described in the previous paragraph.

Alternatively, applying an iterative procedure similar to the one illustrated in section 6.5 ("The idea is to allow only a subset of the flights to exchange time windows, while the rest of the flights will maintain their FPFS allocation"), one could also think of a multi-step scheme in which an airline can reject the solution proposed by the market mechanism. In the first step, the TW allocation is made to all flights. Within a certain instant of time, the airline decides whether to accept or keep the FPFS slot and then the allocation is performed again only on those who accepted. Reasons for refusal could be either because you have tried to cheat the system or because, in good faith, you have made an incorrect assessment of your costs. The flight dispatcher might not accept the market solution because he/she knows from experience that FPFS is better. Clearly, this whole process should be regulated (e.g., a maximum number of refusals or penalties for refusals could be envisaged) in order to avoid continual attempts by airlines to test the system until the desired bundle is obtained, and to give stability to the solution of those who do adhere. A detailed design of this scheme is, however, outside the scope of this study and may be the subject of future work.

In the distributed case (Section 4), at each iteration the central authority communicates the TW prices and each flight identifies the optimal bundle for it at those prices. For example, for flight f_1 the optimal bundle contains the TW called a_1 available at price p_1. f_1 could actually communicate another bundle that, for example, contains a_2 instead of a_1 to lower the price of the latter at the next iteration. This operation is risky, however, since a_1 could also be requested by another flight f_2 willing to pay p_1

for it, and thus f_1 has to make content with a_2 from which it obtains a lower payoff. As in the case of the centralised mechanism, the existing competition for critical resources makes opportunistic behaviour complex.

Appendix B Construction of feasible bundles

Algorithm 1 describes how to generate the set of feasible bundles Q_f for any flight f. Bundles are constructed in sequence, from the one having the smallest delay to the one with the largest delay. The first bundle in Q_f is the bundle whose time windows contain the expected times of entry E_f. This bundle will have zero delay. Then, the idea is to shift a sequence of time instants t, initialised with E_f, towards the right of the discretised timeline and keep track of which time windows contain t in each element of R_f; as soon as the time windows change, a new bundle will be appended to Q_f. Starting from $t = E_f$ and shifting t towards higher times, you will remain inside the same time windows until you encounter the first upper border of a time window. When an upper bound (or more than one) is encountered, you take the next time window in the slot allocation list of this resource(s), while the other time windows stay the same, and append this new bundle to Q_f. The delay will be the temporal distance between the lower bound of the new time window and the expected time over the corresponding resource. This procedure is repeated iteratively until either the maximum delay $MaxDel_f$ is reached, or the end of all slot allocation lists is reached. In the case that $MaxDel_f$ is reached before all time windows become dummy time windows, the empty bundle corresponding to flight cancellation is added to Q_f.

Algorithm 1: Construction of feasible bundles

Input: $R_f, E_f, \{\hat{L}_r \text{ for } r \in R_f\}, MaxDel_f$
Output: Q_f

$n \leftarrow |R_f|$;
$t \leftarrow E_f$;
$q \leftarrow [TW_1, \ldots, TW_n]$ where TW_i is such that $I_{TW_i} \leq e_i \leq U_{TW_i}$ for $i = 1, \ldots, n$;
$delay \leftarrow 0$ seconds;
$Q_f \leftarrow \varnothing$;
while $delay < MaxDel_f$ **do**
  $d_{q_f} \leftarrow delay$;
  $Q_f \leftarrow Q_f \cup q$;
  if *all* $TW_i \in q$ *are dummy* **then** break;
  $interval \leftarrow [U_{TW_1} - t_1, \ldots U_{TW_n} - t_n]$;
  $gap \leftarrow min(interval)$;
  $q \leftarrow [TW_i + 1 \text{ if } interval_i = gap \text{ else } TW_i \text{ for } i = 1, \ldots, n]$;
  $shift \leftarrow gap + 1$ second;
  $t \leftarrow [t_1 + shift, \ldots, t_n + shift]$;
  $delay \leftarrow delay + shift$;
end
if *any* $TW_i \in q$ *is not dummy* **then**
  $Q_f \leftarrow Q_f \cup [\,]$;
end

Appendix C Algorithm for FPFS

This appendix provides an algorithm, adapted from Ranieri (2010), for computing the FPFS bundle a_f for all $f \in \mathcal{F}$. Here we will denote e_r^f the ETO of flight f in regulation r. We will consider the set

Q_f as ordered by increasing delay, and with a little abuse of notation we will identify a bundle q with its index in the ordered list Q_f, for example the most-preferred bundle is $q = 1$. We will denote $TW_r \in q$ the time window in q which belongs to regulation r.

The algorithm begins with variable initialisation at lines 1-9. Variable $provalloc(f)$ is the tentative FPFS allocation for flight f. During the course of the algorithm, a time window can be assigned to multiple flights according to *provalloc*, but at the end of the algorithm at most one flight has assigned any time window. Variable $processed(f, r)$ is $True$ whenever the time window assigned to f in regulation r is not assigned to another flight whose ETO is smaller.

The algorithm applies the FPFS policy independently on each regulation (lines 11-15). This implies that the bundle assigned by the FPFS in one regulation might not respect the FPFS order in another regulation, thus the algorithm iteratively adjusts the allocation (lines 16-26) until the FPFS property is respected for each flight (line 10) and the capacity constraint is satisfied. At the end of the while loop, also the Most Penalizing Regulation rule is satisfied. The second while loop (lines 31-42) checks whether some flights can be assigned a bundle with a smaller delay without breaking the FPFS rule and the capacity constraint.

Algorithm 2: FPFS

```
for f ∈ F do
    provalloc(f) ← None;
    for r ∈ R_f do
        processed(f, r) ← False;
    end
end
for r ∈ R do
    F_r ← ordered_flight_list(r);
end
while all_processed() = False do
    for r ∈ R do
        for f ∈ F_r do
            if provalloc(f) = None then
                assign_first_feasible(f, r, 1);
                processed(f, r) ← True;
            else if processed(f, r) = False then
                if is_feasible(provalloc(f), f, r) = True then
                    processed(f, r) ← True;
                else
                    for z ∈ R_f : z ≠ r do
                        processed(f, z) ← False;
                    end
                    assign_first_feasible(f, r, provalloc(r));
                    processed(f, r) ← True;
                end
            end
        end
    end
end
noimprovement ← False;
while noimprovement = False do
    noimprovement ← True;
    for f ∈ F do
        for q ∈ Q_f : q < provalloc(f) do
            if all_feasible(q, f, r) = True then
                do_assign(f, q);
                noimprovement ← False;
                break;
            end
        end
    end
end
for f ∈ F do
    a_f ← provalloc(f);
end
```

The function *ordered_flight_list*(r) returns the list of flights crossing r ordered by increasing e_r^f.

The function *all_processed*() returns $True$ if $processed(f, r) = True$ for all $f \in \mathcal{F}$, $r \in R_f$.

The function $all\_feasible(q, f, r)$ returns $True$ if $is\_feasible(q, f, r) = True$ for all $r \in R_f$.

The function $assign\_first\_feasible(f, r, i)$ assigns to f the first bundle in Q_f whose index is at least i and such that $TW_r \in q$ is either not assigned to any other flight, or for all flights g to which it is assigned it holds $e_r^g > e_r^f$. In this latter case, we also set $processed(g, r) \leftarrow False$.

The function $is\_feasible(q, f, r)$ returns $True$ if $TW_r \in q$ for flight f has not been assigned to any other flight.

The function $do\_assign(f, q)$ assigns bundle q to flight f.

For the sake of efficiency, the five functions described above make use of a data structure which maps every time window to the set of flights that are currently assigned to it according to *provalloc*.

Appendix D Proof of bounds

D.1 Bound on optimal value

Problem (39) is the same of the linear relaxation of problem (5) with the objective function coefficients $V(f, q)$ substituted by lower bounds $LB^t(f, q)$. The feasible region of (39) is the same as the feasible region of the linear relaxation of (5), and the objective function of (39) is everywhere lower than the objective function of (5). In fact, from

$$LB^t(f, q) \leq V(f, q) \quad \forall f \in \mathcal{F}, q \in Q_f$$

it follows, since x is feasible and so $x(f, q) \geq 0$,

$$LB^t(f, q)x(f, q) \leq V(f, q)x(f, q) \quad \forall f \in \mathcal{F}, q \in Q_f$$

and by summing over $f \in \mathcal{F}$ and $q \in Q_f$

$$\sum_{f\in\mathcal{F}} \sum_{q\in Q_f} LB^t(f, q)x(f, q) \leq \sum_{f\in\mathcal{F}} \sum_{q\in Q_f} V(f, q)x(f, q)$$

Therefore, $ZLB^t \leq Z_{LP}$. Analogously, if $LB^t(f, q)$ is substituted by an upper bound $UB^t(f, q)$, we obtain that the optimal value of (39) is an upper bound for the optimal value of the linear relaxation of (5).

D.2 Non-negativity of a bound

The lower bound RES^t on $Z_{LR}(\lambda^t) - Z_{LP}$ is given by

$$RES^t = SG^t \cdot \lambda^t - \max_{x\in S} \sum_{f\in\mathcal{F}} \sum_{q\in Q_f} \left(p^t(q) - p^t(q_f^{*t})\right) x(f, q). \tag{47}$$

If the duality gap $Z_{IP} - Z_{LP}$ is zero, the maximisation problem appearing in (47) has always an integer optimal solution, since its feasible region is the same as the feasible region of the linear relaxation of problem (5). Let $\{q_f^{\circ t}\}_{f\in\mathcal{F}}$ be the set of bundles corresponding to the optimal solution. Then we can write

$$\begin{aligned}
RES^t &= \sum_{k\in\mathcal{L}} \lambda_k^t \Big(1 - \sum_{f\in\mathcal{F}} \sum_{q\in Q_f : q\ni k} x^t(f, q)\Big) - \sum_{f\in\mathcal{F}} \left(p^t(q_f^{\circ t}) - p^t(q_f^{*t})\right) \\
&= \sum_{k\in\mathcal{L}} \lambda_k^t - \sum_{f\in\mathcal{F}} \sum_{q\in Q_f} \sum_{k\in q} \lambda_k^t x^t(f, q) - \sum_{f\in\mathcal{F}} \left(p^t(q_f^{\circ t}) - p^t(q_f^{*t})\right) \\
&= \sum_{k\in\mathcal{L}} \lambda_k^t - \sum_{f\in\mathcal{F}} p^t(q_f^{*t}) - \sum_{f\in\mathcal{F}} p^t(q_f^{\circ t}) + \sum_{f\in\mathcal{F}} p^t(q_f^{*t}) \\
&= \sum_{k\in\mathcal{L}} \lambda_k^t - \sum_{f\in\mathcal{F}} p^t(q_f^{\circ t}) \geq 0
\end{aligned}$$

where the inequality follows from the fact that $\{q_f^{\circ t}\}_{f\in\mathcal{F}}$ are capacity-compliant, thus they share no time windows.

Appendix E Additional computational results

E.1 Instances with zero duality gap

Tables 4 and 5 present some additional computational experiments performed. The table has been split for reasons of space. The first four rows (a, b, c and d) are the four cases already presented in sections 6.1, 6.2, 6.3 and 6.4. The duality gap was zero for all cases in the table. Table 4 contains the following columns: names of the regulations, number of regulations, total number of flights, number of flights subject to at least two of the regulations. Table 5 contains: number of iterations, final overload, whether or not the solution found is optimal, surplus, cost of the FPFS allocation, cost of the optimal allocation, cost of the allocation found (in case it is not optimal). Notice that if the overload is nonzero, the latter cost can be smaller than the optimal cost, since the allocation found is not a proper solution. All these costs are unknown to the central authority. The cost savings are between 47% and 85%, and the cost of the solution found is at most 6% higher than the optimal cost.

	regulations	num. reg.	num. flights	num. flights with $R_f > 1$
a	LBSAU04, LBSCU04	2	111	15
b	LOVS04N, LOWB304A	2	145	27
c	LHSUH04A, LONE3504	2	233	42
d	ME1204, MKK04	2	64	15
e	RMZU04E, RQXU04E	2	143	39
f	EGKKA04, RJS04	2	163	25
g	KCHI104A, LOW3504A	2	259	73
h	EDDLA04, EDWSUD04	2	119	10
i	LHNLMU04, LHWSEN04, LOVSC04	3	82	13
j	EDGN004M, GL67W04, YB3LL04M	3	114	20
k	GL67W04, PDMX04M, YB3LL04M	3	123	25
l	EHYR04M, RESTU04, RG04A	3	170	30
m	KDON1D04, KFFM2404, KWUR1C04	3	221	54
n	K11UFX04, LHENH04, LHENU04	3	275	39
o	EDMBBG04, EDMHAG04, K11UFX04	3	206	25
p	GL12W04, GL5W04, MGY04A, ZM3404	4	363	36
q	KDON104N, KFFM1C04, KFFMC04A	3	414	99

Table 4: Description of the instances, zero duality gap

E.2 Instances with nonzero duality gap

Tables 6 and 7 present additional computational experiments performed with the algorithm of Section 6.5. In all cases the duality gap was reduced to zero after the first iteration of the algorithm and the optimal solution of the reduced problem (which is a feasible solution of the whole problem) was found in the second iteration. The first row of the table is the case already presented in Section 6.5. The columns of Table 7 are: number of iterations performed on the whole problem, number of iterations on the reduced problem, magnitude of the duality gap, number of flights removed, cost of the FPFS allocation, cost of the optimal allocation of the whole problem, cost of the feasible solution found. The

	num. iter.	overload	is sol. opt.	surplus	FPFS cost	opt. cost	solution cost
a	83	0	true	€ 0.00	€ 960.20	€ 263.36	
b	100	0	false	€ 13.15	€ 2,689.30	€ 661.93	€ 662.93
c	131	1	false	€ 44.57	€ 6,328.80	€ 1,121.86	€ 1,068.50
d	37	0	true	€ 7.26	€ 1,159.90	€ 619.72	
e	141	0	false	€ 11.37	€ 6,559.48	€ 1,021.94	€ 1,070.14
f	122	0	true	€ 0.00	€ 9,537.10	€ 1,823.13	
g	184	1	false	€ 73.12	€ 10,932.82	€ 2,153.40	€ 2,228.41
h	111	1	false	€ 46.27	€ 11,640.81	€ 1,981.35	€ 1,897.21
i	66	0	true	€ 14.82	€ 2,055.24	€ 1,042.56	
j	109	0	true	€ 58.08	€ 3,761.87	€ 843.38	
k	141	0	true	€ 8.89	€ 4,484.10	€ 913.96	
l	138	0	false	€ 95.87	€ 6,375.28	€ 911.90	€ 932.05
m	157	0	false	€ 34.42	€ 8,675.39	€ 1,612.18	€ 1,660.24
n	169	0	false	€ 36.31	€ 21,496.61	€ 3,800.02	€ 4,031.79
o	183	0	true	€ 58.67	€ 13,792.50	€ 2,107.29	
p	191	0	true	€ 0.00	€ 12,195.43	€ 3,964.32	
q	223	2	false	€ 90.36	€ 18,380.43	€ 3,460.63	€ 3,518.02

Table 5: Computational results, zero duality gap

cost of the feasible solution is most often less than 10% more than the optimal cost, and at most 50% higher, and the savings with respect to the FPFS solution are between 71% and 89%.

	regulations	num. reg.	num. flights	num. flights with $Rf > 1$
a	LHENT04L, LHLYBA04, LHSUH04A, LONE3504, LOVS04N	5	832	81
b	CBHRE04, EGKKA04, EHYR04M, RJS04	4	345	85
c	MAB04L, MRAEE04L	2	61	13
d	LEBLA04A, MMF04A	2	82	10
e	EKHR04, RMZU04E, RQXU04E	3	165	50
f	EDMBBG04, EDMHAG04, K11UFX04	3	206	25
g	KFFM204A, KWUR04A	2	253	66
h	KFFM2404, KTGO1T04, KWUR04, KWUR2404, LHWSUH04	5	475	120
i	EDGN004M, GL67W04, PDMX04M, YB3EH04M, YB3LL04M	5	426	51
j	CPBDL04, K11UFX04, KHVL1H04, LHENLM04, LOE1504	5	703	80

Table 6: Description of the instances, nonzero duality gap

	num. iter. first round	num. iter. second round	duality gap	num. flights removed	FPFS cost	opt. cost	solution cost
a	371	410	0.14	4	€ 24,364.04	€ 4,170.23	€ 4,191.46
b	387	484	1.66	4	€ 31,779.53	€ 3,396.45	€ 3,499.92
c	83	233	0.06	1	€ 1,145.10	€ 171.72	€ 224.23
d	185	386	0.23	1	€ 1,741.77	€ 286.01	€ 286.14
e	552	366	2.35	1	€ 6,798.12	€ 1,104.81	€ 1,296.86
f	600	370	0.17	4	€ 13,777.47	€ 2,131.83	€ 2,276.11
g	154	204	8.97	6	€ 8,885.98	€ 1,685.85	€ 2,532.98
h	479	505	8.40	4	€ 15,992.62	€ 2,704.83	€ 2,836.05
i	563	341	0.88	6	€ 15,002.05	€ 3,376.16	€ 3,766.67
j	537	442	0.73	3	€ 36,963.43	€ 7,754.63	€ 7,890.24

Table 7: Computational results, nonzero duality gap