Optimising airport slot allocation decisions with stability considerations

Fotios A. Katsigiannis[a,*], Konstantinos G. Zografos[a]

[a] Lancaster University, Lancaster, United Kingdom
f.katsigiannis@lancaster.ac.uk, k.zografos@lancaster.ac.uk
*Presenter

Extended abstract submitted for presentation at the 11th Triennial Symposium on Transportation Analysis conference (TRISTAN XI)
June 19-25, 2022, Mauritius Island

March 18, 2022

Keywords: *airport slot allocation, mixed integer programming, stable matching*

1 INTRODUCTION

Airport Slot Allocation (ASA) is the main airport demand management mechanism currently being applied in 198 airports (as per the 2022 summer season (IATA/ACI/WWACG, 2020)) which concern more than half of global passenger demand (despite the impact of COVID-19). The ASA process is defined by the World Airport Scheduling Guidelines (WASG)(IATA/ACI/WWACG, 2020). During the ASA process airlines submit requests for airport slots based on their commercial interests and preferences, while coordinators allocate slots to requests following a set of primary and secondary criteria stemming from WASG. Existing ASA studies propose formulations that consider airlines' preferences for each available time slot (Zografos *et al*., 2017; Jacquillat and Vaze, 2018; Fairbrother *et al*., 2019), but they do not explicitly consider the interactions between the two sides of the coordination process, i.e., the explicit consideration of airlines' utility by the coordinator when applying WASG.

In this paper we propose an *Integer Programming* (IP) formulation and a *Deferred Acceptance* (DA) algorithm (Gale and Shapley, 1962) that consider the ASA problem as a two-sided matching game and model each submitted airline request and available airport slot as distinct entities. In doing so, time-dependent functions express the airlines' utility for each submitted request and the priority assigned by the coordinators based on WASG. The IP formulation considers the equilibria between the airline and the coordinators' side but introduces increased computational complexity. Vis-à-vis this complication, the proposed DA algorithm generates multiple airport slot schedules within tractable computational times. The generated schedules guarantee that the utility achieved for each request cannot be improved without compromising the request-to-slot assignment of a more important request (based on the airlines' preferences and the WASG-defined prioritisation considered by the coordinator), ergo ensuring that airlines and coordinators have no incentives to reject or alter the proposed allocations. Consequently, the request-to-slot assignments proposed by our approach are said to be *stable* and achieve improved schedule acceptability.

2 FORMULATION

The paper introduces an IP formulation (expressions 1-5). The notation required for formulating the IP and the DA algorithm is presented in Table 1. The expressions used for the definition of the IP formulations are provided as follows.

$$\sum_{t \in T_{\tilde{c}}} x_{t,m} \leq 1 \qquad m \in M \qquad (1)$$

$$\sum_{m\in M^k}\sum_{t'\in[t,t+c-1]} H_m^d x_{t',m} \le CAP_{d,t,c}^k \qquad c\in C, d\in D, k\in K, t\in[0,|T_{\tilde{c}}|-c] \tag{2}$$

$$\sum_{t\in[0,\kappa)} x_{md,} + \sum_{t\in[\kappa-TT_m,|T_{\tilde{c}}|]} x_{ma,} \le 1 \qquad (ma,md)\in M\times M, \kappa\in[TT_m,|T_{\tilde{c}}|) \tag{3}$$

$$\sum_{t\in T_{\tilde{c}}} x_{t,m}|t-t_m| \le \Omega \qquad m\in M \tag{4}$$

$$min\ Z = \beta_1\frac{Z_1}{|M||D|}+\beta_2\frac{Z_2}{|T_{\tilde{c}}|}+\beta_3\frac{Z_3}{|M||D|}+\beta_4\frac{Z_4}{|M||T_{\tilde{c}}||D|} \tag{5}$$

Table 1 – *Notation of the IP formulation*

Sets	M	Set of request series denoted by m
	$M^{Arr(Dep)}$: $M^{Arr}\cup M^{Dep}=M^{Total}$	Set of arrival (departure) series
	$M\times M$	Set of paired requests (md,ma)
	$D\ (D_m)$	Set of days in scheduling season denoted by d (of request m)
	C: $\{5,15,60\}$	Set of capacity time intervals indexed by c
	$T_{\tilde{c}}$: $\{1,2,\dots,n\}$	Set of time intervals per day based on interval $\tilde{c}$ indexed by t, t'
	K: $\{Arr, Dep, Total\}$	Set of movement types denoted by k
Parameters	t_m	Requested time for slot series m
	TT_m	Maximum and minimum turnaround times of paired request p
	$CAP_{d,s,c}^k$	Capacity for movements k for a period $[s,s+c]$ on day d based on time interval c
	$H_{d,m}$	Indicates whether series m is requested on day d or not
	$\Xi_m(t)$	Preference assigned by request m to slot t
	$\Psi_m(t)$	Priority assigned by the coordinator to request m with respect to t
	β_j	Relative importance assigned to objective j
Decision variables and objectives	$x_{t,m}$	1 if request m is allocated to time t; 0 otherwise
	Z_1	Number of rejected requests’
	Z_2	Maximum displacement
	Z_3	Number of displaced requests
	Z_4	Total displacement

Expressions (1) are assignment constraints. Expressions (2) define rolling capacity constraints for arrival/departure and total movements. Expressions (3) are turnaround time constraints. (4) are auxiliary expressions defining the maximum displacement objective. Expression (5) defines the scalar objective function of our model which consists of relative importance weights (β_j) for each considered objective. The model defined by (1)-(5) is complemented by expressions which guarantee that there will be no request-to-slot assignments where a request may prefer another available slot, or a slot would rather be matched to another request of higher priority. These are referred to as *stability inequalities* consider $\Psi_m(t)$ for determining the requests that will be assigned to each slot, while $\Xi_m(t)$ is used for considering airlines’ preferences.

3 SOLUTION METHODOLOGY

The solution of the IP formulation using commercial solvers (airport instance concerning 2491 requests series which translate to 72000 movements) requires days so as to produce a single airport schedule. In response, in what follows we detail a fast DA algorithm that can generate multiple airport slot schedules that are Pareto optimal with respect to each submitted request. DA algorithms were first introduced by Gale and Shapley (1962) and are currently being considered in several scheduling and resource allocation problems (Abdulkadiroglu and Sönmez, 2013).

In contrast to the MIP defined in section 2, the proposed algorithm considers the rules and priorities of WASG using Ξ_m and Ψ_m as soft constraints and operates based on the following logic. At the first iteration of the algorithm, all requests demand the slot that has the best spot on their preference list (step 1 in Algorithm 1). Based on the slot's capacity, the requests with highest priority are provisionally matched to the resource (step 2). If the number of applications exceeds the capacity of the slot, then the applications with the lowest priority (as per $\Psi_m(t)$) are rejected. In the case that the capacity of the slot suffices to accommodate all applicants, then all requests are matched. The preference lists of the requests are updated (step 3) and the algorithm proceeds to the next iteration.

During the following iterations, each request that was rejected in previous iterations proposes to the best-remaining and feasible slot (the best option remaining in Ξ_m after the update process described in step 3) and the requests with the highest priority are matched with the slot. At this point a subset of requests from among the previously matched requests and the newly proposing requests is selected. This signifies, that if a request proposed to t, and t prefers request m to a previously matched request m_1 ($\Xi_{m_1}(t) < \Xi_m(t)$), then m_1 is unmatched and request m takes their place. The algorithm terminates when either all requests receive a slot, or requests that remain unmatched have empty preference lists (the request has been rejected by all slots).

Algorithm 1 – *A deferred acceptance algorithm for ASA*

Input: Set of requests: M indexed by m
Set of slots: T indexed by t
Preference list of each request (m) with respect to each slot (t): $\Xi_m = \{\Xi_m(t) | \forall t \in T, \Xi_m(t)\}$
Slot-dependent priority of each request (m) by the coordinator: $\Psi_m = \{\Psi_m(t) | \forall t \in T, \Psi_m(t)\}$

Output: Set of stable request-to-slot assignments

Define the sets of paired requests (PR) and the list of request-slot pairs (P) to be empty $(PR = \{\ \}, P = [\,]\ )$
Initialise the list of unmatched requests to be equal to the request set $(UR = M)$
while there are unmatched requests with non-empty preference lists **do:**
(*step 1*) Each m applies for the slot t such that $\underset{t \in T_{\tilde{c}}}{\operatorname{argmax}} \Xi_m(t)$
(*step 2*) $\forall\ t \in T_{\tilde{c}}$, the algorithm determines which requests will be provisionally matched to t
(*step 3*) The preference lists of all requests that applied in this iteration are updated by removing $\Xi_m(t)$ from Ξ_m
(*step 4*) UR, PR and P are updated
return P

4 RESULTS

Preliminary results using preference list lengths of alternative sizes, indicate that the DA algorithm reports objective values that are comparable to the values reported by the IP formulation. In addition, the sets of schedules generated by Algorithm 1 are not only acceptable and non-dominated with respect to each submitted request, but also constitute an efficient frontier between the spilled airline/passenger demand and maximum displacement. The DA and the IP can be used in conjunction so as to produce airport schedules comprising Pareto optimal request-to-slot assignments in more tractable computational times. This set of observations suggest that the DA algorithm may grasp the specificities of ASA, generate multiple stable schedules, and provide crucial decision-support to the ASA decision-making process.

5 DISCUSSION

The proposed IP and DA algorithm provide request-to-slot assignments that are Pareto optimal *per se*, meaning that a request cannot receive an improved allocation without compromising the allocation efficiency of a more important request. Hence, the reported schedules are stable based on the perspectives of both airlines and coordinators. Our computational results to date exhibit that the concurrent consideration of airlines' preferences and the priorities assigned by the coordinators allows the scheduling of additional movements throughout the entire scheduling season. This is achieved by scheduling requests that operate for longer effective periods and serve more passengers. Despite the computational complexity of the problem, Algorithm 1 may produce a large set of schedules, albeit requiring a fraction of the computational times required for the solution of the IP formulation.

Our presentation will elaborate on the computational implications of the proposed methodology. Comparisons among alternative solution approaches and slot allocation regimes/prioritisations will enable us to discuss the method's implications for policy and decision-making.

6 REFERENCES

Abdulkadiroglu, A. and Sönmez, T., 2013. Matching markets: Theory and practice. *Advances in Economics and Econometrics*, 1, pp.3-47.

Fairbrother, J., Zografos, K.G., Glazebrook, K., (2019). A slot scheduling mechanism at congested airports which incorporates efficiency, fairness and airline preferences. *Transportation Science,* 54 (1) pp. 115-138.

Gale, D., Shapley, L.S., (1962). College Admissions and the Stability of Marriage. *The American Mathematical Monthly* 69, 9–15.

IATA/ACI/WWACG, (2020). Worldwide Airport Slot Guidelines [WWW Document]. URL https://www.iata.org/en/policy/slots/slot-guidelines/ (accessed 9.22.20).

Jacquillat, A., Vaze, V., (2018). Interairline Equity in Airport Scheduling Interventions. *Transportation Science*, 52 (4) pp. 941-964.

Zografos, K.G., Androutsopoulos, K.N., Madas, M.A., (2017). Minding the gap: Optimizing airport schedule displacement and acceptability. *Transportation Research Part A: Policy and Practice*, 114 pp. 203-221.

ACKNOWLEDGEMENTS

The research reported in this paper has been supported by the UK's Engineering and Physical Sciences Research Council (EPSRC) through the Programme Grant EP/M020258/1 "*Mathematical models and algorithms for allocating scarce airport resources* (OR-MASTER)". The opinions expressed in this extended abstract reflect the authors' views.