Solving the Slot Allocation Problem with Flexible Scheduling of Series of Slots

Jamie Fairbrother[1] Stephen Maher[1] Konstantinos Zografos[1] Kevin Glazebrook[1]

[1] Centre for Transport and Logistics, Lancaster University

INFORMS Annual Conference, 2018-11-04

Introduction

Airport congestion

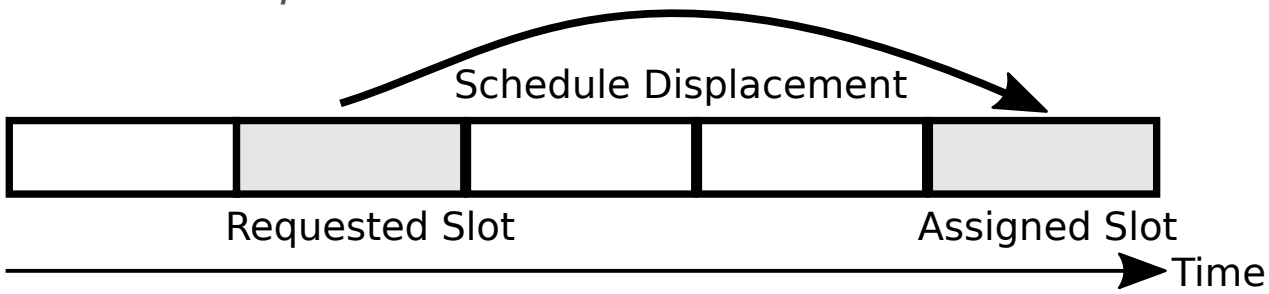
- Outside of the US, demand at congested airports is managed through the IATA WSG
- Airports are designated as *coordinated* and airlines must obtain slots to use an airport
- A slot is a time interval during which an aircraft can use an airport infrastructure for the purposes of landing or take-off
- A *coordinator* proposes an initial allocation for slots to airlines based on their requests

Slot Allocation Problem: Key Features

- Slots at a coordinated airport are allocated for a six month season
- Requests are typically made in *series*:
  - A series of slots consists of 5 or more requests for the same time and (usually) the same day
  - Request series are mostly allocated slots the same time
- Allocation should satisfy airport capacity constraints
- Allocation should satisfy turnaround constraints
- Slots are allocated according to the following main priority groups
  - Historics
  - New entrants
  - Others

Schedule Displacement

- Capacity restrictions mean that not all requests can be allocated their preferred slot
- The difference between an preferred and allocated time is called the *schedule displacement*



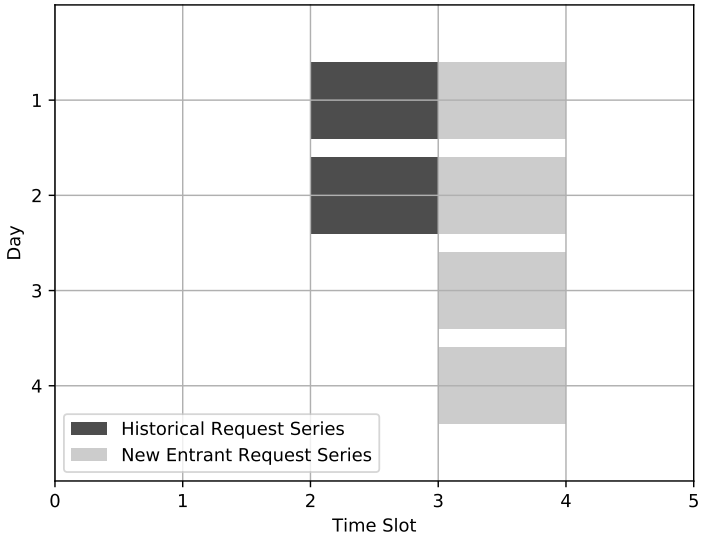
- The allocation should in some way minimize the total *schedule displacement*

Literature Review

- The complexity of allocating slots under IATA scheme has lead to the development of integer programming models
- Main features of IATA scheme first modeled as an MIP in [Zografos et al., 2012] which minimized total displacement
- Incorporation of fairness:
  - Displacement proportional to requests [Zografos and Jiang, 2017]
  - Displacement proportional to peak-time requests [Fairbrother and Zografos, 2018]
- [Zografos et al., 2017] introduced objectives to improve acceptability:
  - Minimize maximum displacement
  - Maximize slots allocated to acceptable time windows
- [Ribeiro et al., 2018] proposed new multiobjective model with:
  - Multiple objectives: number of rejections maximum displacement, total displacement, number of displacement
  - Differentiation between “historical” and “change to historical” requests

Inflexibilty and Flexibility

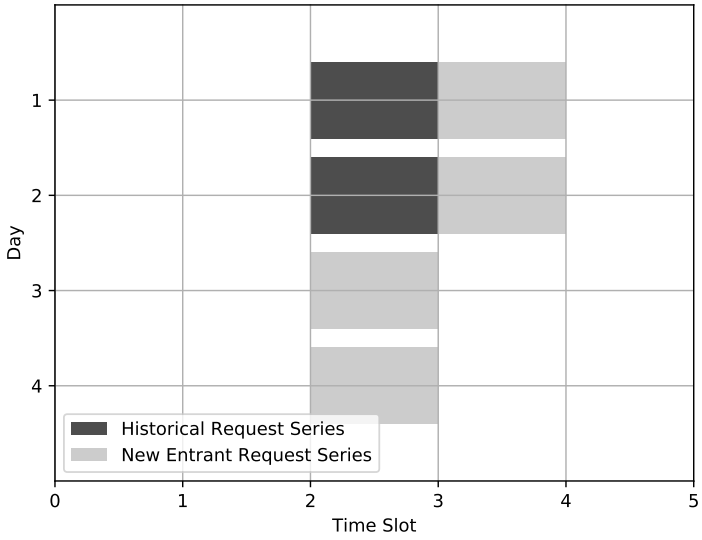
- A request series which coincides on at least one day may block another for all it's requested days
- Toy example:
  - One historical request series (2 days)
  - One new entrant request series (4 days)
  - Both prefer slot at time period 2
  - Capacity of 1 slot per time period



- Historical request series is given priority and blocks new entrant

Inflexibilty and Flexibility

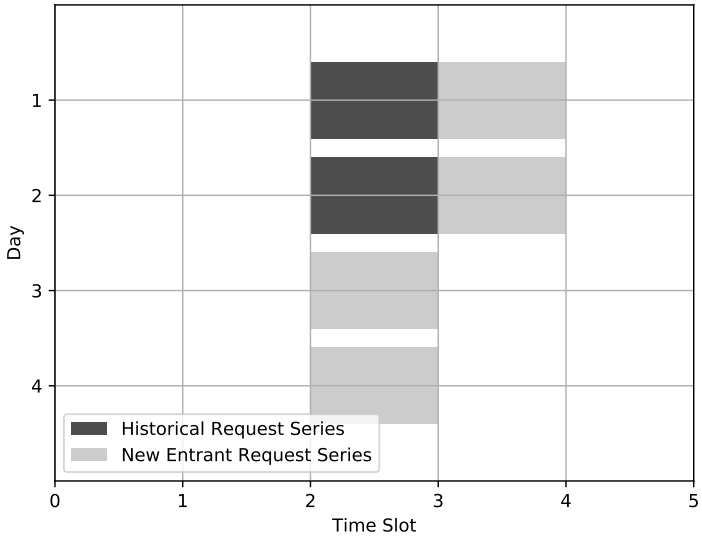
- A request series which coincides on at least one day may block another for all it's requested days
- Toy example:
  - One historical request series (2 days)
  - One new entrant request series (4 days)
  - Both prefer slot at time period 2
  - Capacity of 1 slot per time period



- Historical request series is given priority and blocks new entrant
- Allowing flexibility, new entrant series can be allocate slots for days not used by historic request series

Inflexibilty and Flexibility

- A request series which coincides on at least one day may block another for all it's requested days
- Toy example:
  - One historical request series (2 days)
  - One new entrant request series (4 days)
  - Both prefer slot at time period 2
  - Capacity of 1 slot per time period



- Historical request series is given priority and blocks new entrant
- Allowing flexibility, new entrant series can be allocate slots for days not used by historic request series

- By allowing some flexibility in the slots allocated to a series, we can reduce significantly the overall displacement

Aims

1. Develop a model which allows some flexibility in the slots allocated to a series
2. Incorporate demand-based fairness into new model
3. Study trade-off between flexibility, fairness and efficiency using real slot request data

Flexibility Model and Demand-based Fairness

Slot assignment variables

- Model is generalisation of that of [Zografos et al., 2012]
- Let:

$$\begin{aligned}
\mathcal{M} &= \text{set of request (series)} \\
\mathcal{D} &= \text{set of days in scheduling season} \\
\mathcal{D}_m \subseteq \mathcal{D} &= \text{set of days which make up request } m \\
\mathcal{T} = \{1, \ldots, T\} &= \text{set of time coordination intervals}
\end{aligned}$$

- Then, the main decision variables for our model are:

$$y_m^{dt} = \begin{cases} 1 & \text{if request } m \text{ is allocated slot at time } t \text{ on day } d, \\ 0 & \text{otherwise} \end{cases}$$

Range Constraints

- It is necessary that the slots allocated to a request series are close to each other
- Introduce auxiliary variables:

$$\underline{\tau}_m = \text{ earliest slot allocated to request series } m$$
$$\overline{\tau}_m = \text{ latest slot allocated to request series } m$$

- We require that each request series is assigned slots at times within a given range of each other:

$$\overline{\tau}_m - \underline{\tau}_m \leq r_m, \; m \in \mathcal{M}$$

where r_m is a tolerance specified by the airline making the request m

Fairness - Peak Request Proportionality Principle

- For a schedule to be acceptable it is important that it distributes schedule displacement in a fair way to the airlines
- We define fairness based on the following principle [Fairbrother and Zografos, 2018]

proportion of schedule displacement $\approx$ proportion of peak requests

- Solving base model may yield schedule which distributes schedule displacement disproportionately

Fairness indices and constraints

- For each airline, we define the following fairness index:

$$\mu_a = \frac{\text{proportion of schedule displacement}}{\text{proportion of peak requests}}$$

- Fairness is then ensured through the addition to the base model of a constraint on the maximum deviation from absolute (MDA) fairness:

$$\max_{a \in \mathcal{A}} |\mu_a - 1| \leq \epsilon$$

Base Model

$$\text{minimize} \quad \sum_{m \in \mathcal{M}} |t - t_m| y_m^{td} \qquad \text{(minimize total displacement)}$$

$$\text{subject to} \quad \sum_{t \in T} y_m^{td} = 1, \; m \in \mathcal{M}, d \in \mathcal{D}_m \qquad \text{(assign slot to each request)}$$

$$\sum_{m \in \mathcal{M}} \sum_{t \in \mathcal{T}_c^s} a_m^d b_{mc} y_m^{td} \leq u_c^{ds}, \; c \in \mathcal{C}, \; d \in \mathcal{D}, \; s \in \mathcal{T}_c \qquad \text{(rolling capacity constraints)}$$

$$\underline{l}_{m_1 m_2} \leq \sum_{t \in \mathcal{T}} t y_{m_2}^{td} - \sum_{t \in \mathcal{T}} t y_{m_1}^{td} \leq \bar{l}_{m_1 m_2}, \; (m_1, m_2) \in \mathcal{P}, \; d \in \mathcal{D}_m \qquad \text{(turnaround time constraints)}$$

$$\underline{\tau}_m \leq \sum_{t \in \mathcal{T}} t y_m^{td} \leq \overline{\tau}_m, \; m \in \mathcal{M}, \; d \in \mathcal{D}_m \qquad \text{(definition of } \underline{\tau}_m, \; \overline{\tau}_m\text{)}$$

$$\overline{\tau}_m - \underline{\tau}_m \leq r_m, \; m \in \mathcal{M} \qquad \text{(range constraints)}$$

$$\max_{a \in \mathcal{A}} |\mu_a - 1| \leq \epsilon \qquad \text{(fairness constraints)}$$

$$y_m^{td} \in \{0, 1\}$$

Notation

Sets	
$\mathcal{A}$	set of airlines
$\mathcal{D}$ ($\mathcal{D}_m$)	set of days (for which movement m requested)
$\mathcal{M}(\mathcal{M}_a)$	set of movement requests (by airline a)
$\mathcal{P} \subset \mathcal{M} \times \mathcal{M}$	set of arrival-departure pairs (m_a, m_d)
$\mathcal{C}$	set of airport capacity constraints
$\mathcal{T} = \{1, \ldots, T\}$	set of coordination time intervals
Parameters	
t_m	requested time for movement m
δ_c	duration of constraint c
$\underline{l}_p(\overline{l}_p)$	minimum (maximum) turnaround time for movement pair $p \in \mathcal{P}$
a_m^d	indicates whether constraint $c \in \mathcal{C}$ is active on day $d \in \mathcal{D}$
b_{mc}	contribution of movement m to constraint c
u_c^{ds}	capacity for constraint $c \in \mathcal{C}$ on day $d \in \mathcal{D}$ at time period s
r_m	maximum range of slot times for request m
ν_m	marginal cost of request m
Decision variables	
y_m^{td}	indicates whether movement m is assigned slot t on day d
$\underline{\tau}_m(\overline{\tau}_m)$	earliest (latest) slot time assigned to m
μ_a	fairness index for airline a

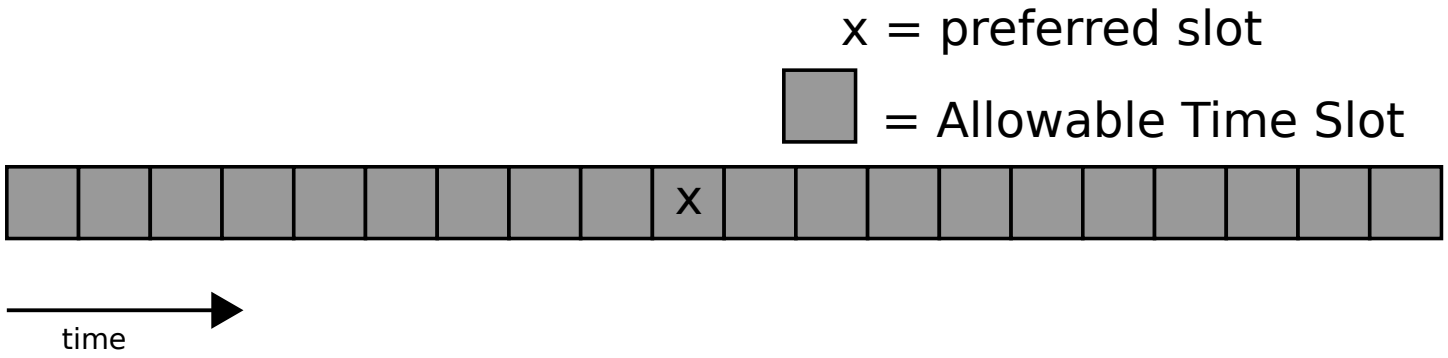
Solution Approaches

Problem Size

- The flexibility slot allocation model has $\mathcal{O}(|\mathcal{T}| \sum_{m \in \mathcal{M}} |\mathcal{D}_m|)$ variables as opposed to $\mathcal{O}(|\mathcal{T}||\mathcal{M}|)$ variables in previous models [Zografos et al., 2012, Ribeiro et al., 2018]
- It also has many more turnaround constraints as these are now required for every day and every movement ($\sum_{m \in \mathcal{M}} |\mathcal{D}_m|$) rather than just every movement

Problem Simplification - Time Windows

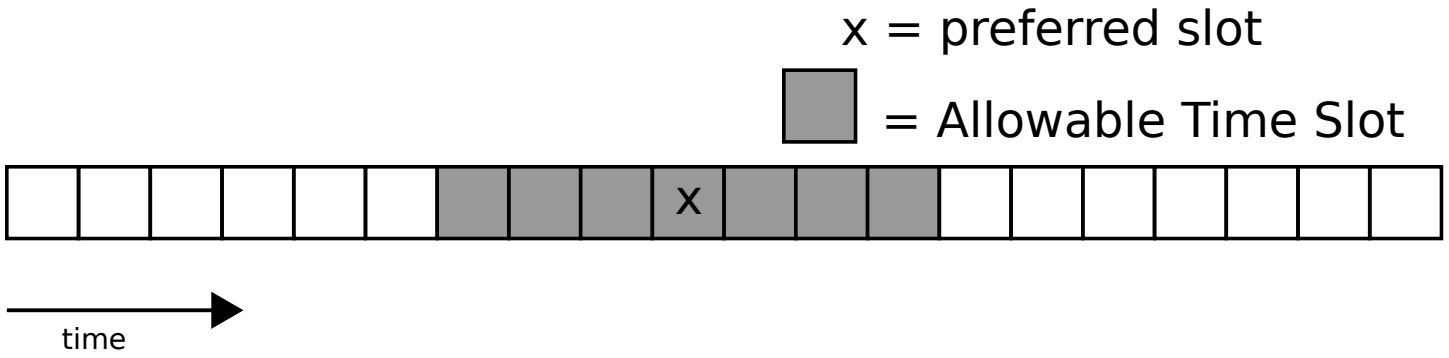
- The base model assumes that a request can be allocated a slot any time in the day
- In reality, slots will only be acceptable if they are close to the requested time
- We therefore enforce that each request series is allocated slots in the time window $[t_m - R, t_m + R]$



- This reduces number of variables to $\mathcal{O}(2R \sum_{m \in \mathcal{M}} |\mathcal{D}_m|)$
- Suitable value for R can be found by solving inflexible problem

Problem Simplification - Time Windows

- The base model assumes that a request can be allocated a slot any time in the day
- In reality, slots will only be acceptable if they are close to the requested time
- We therefore enforce that each request series is allocated slots in the time window $[t_m - R, t_m + R]$

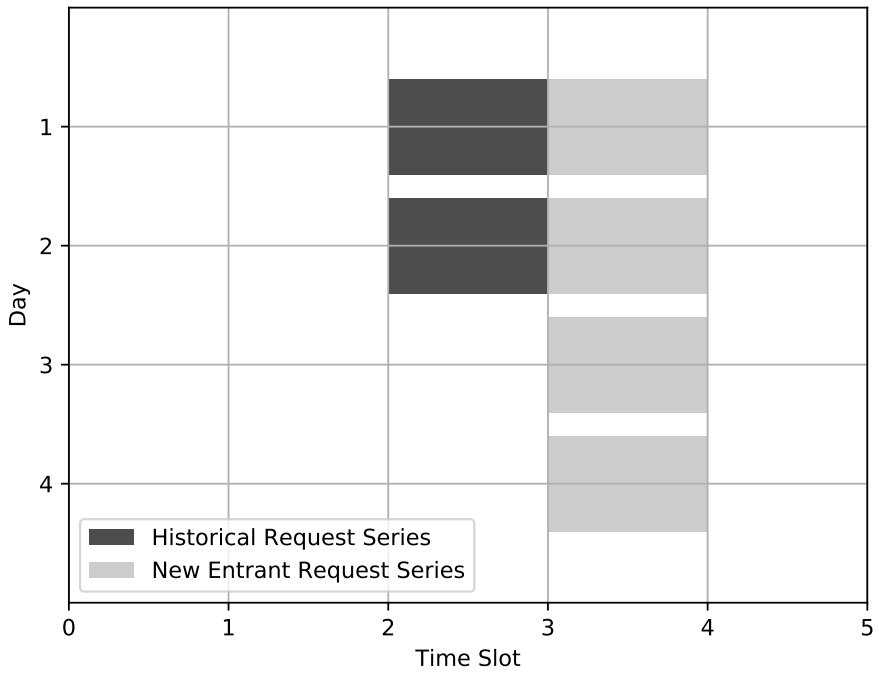


- This reduces number of variables to $\mathcal{O}(2R \sum_{m \in \mathcal{M}} |\mathcal{D}_m|)$
- Suitable value for R can be found by solving inflexible problem

Warmstart - Improvement Heuristic

- Warm-start can help speed-up solution signifantly
- Heuristic takes solution to inflexible problem and improves solution greedily

```
input  : Schedule
output: Improved schedule
Order requests by duration;
repeat
    for each request do
        Remove request from current schedule;
        Find minimum cost insertion of request into schedule;
        Reinsert request into schedule;
    end
until No improvement made;
```

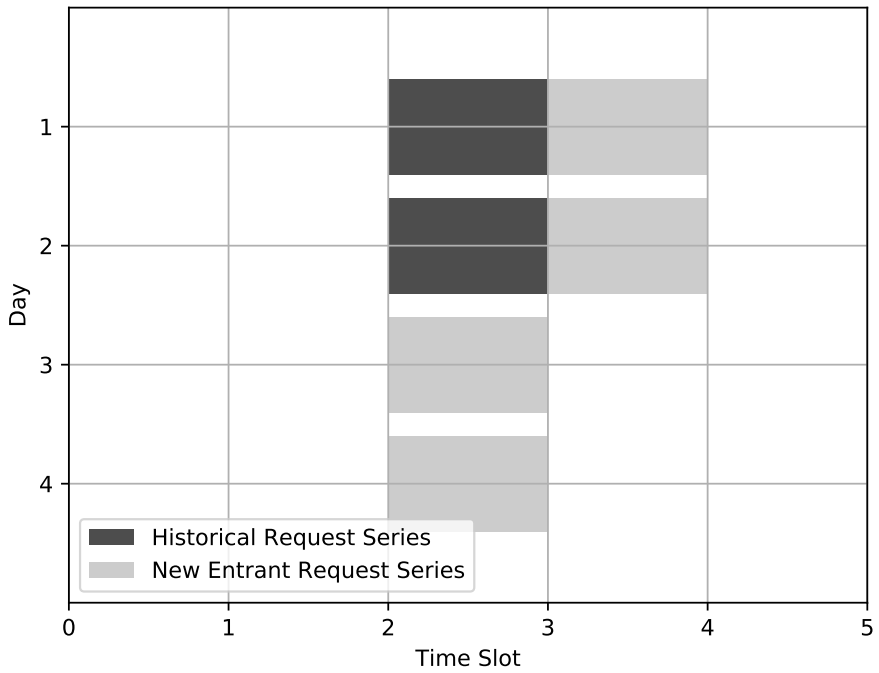


- Minimum displacement insertions found by solving small ILP

Warmstart - Improvement Heuristic

- Warm-start can help speed-up solution signifantly
- Heuristic takes solution to inflexible problem and improves solution greedily

```
input  : Schedule
output: Improved schedule
Order requests by duration;
repeat
    for each request do
        Remove request from current schedule;
        Find minimum cost insertion of request into schedule;
        Reinsert request into schedule;
    end
until No improvement made;
```



- Minimum displacement insertions found by solving small ILP

Branch-and-cut

- For most time periods in the scheduling season, there is sufficient capacity to meet demand, and so most capacity constraints are inactive
- We can speed up the solution of our model by adding capacity constraints as *lazy constraints*
- This is achieved through a callback in the branch-and-bound algorithm

Overview of Solution Approach

1. Impose time windows on slot allocations
2. Aggregate dates in preprocessing stage
3. Solve inflexible problem and run improvement heuristic to construct warm start solution
4. Run branch-and-bound with lazy capacity constraints

Numerical Tests

Problem Data

- Slot allocation problem solved using real request data for two medium-sized airports

Airport 1:

	Requests	Total movements
Historics	252	4304
Changes to Historics	154	3012
New Entrants	48	660
Others	444	7412
Total	898	15388

(a) Number of requests

	15 minutes	1 hour
Arrivals	-	4 (5)
Departures	-	6
Total	3 (4)	10 (11)

(b) Rolling Capacity

Airport 2:

	Request	Total Movements
Historics	266	5480
Changes to Historics	432	8554
New Entrants	76	2680
Others	1090	18000
Total	1864	34714

(a) Number of requests

	15 minutes	1 hour
Arrivals	-	8
Departures	-	12
Total	5	20

(b) Rolling Capacity

Solution Approach

Hierarchical solve

- To take into account the different priority classes we solve the model *hierarchically*
- That is, problem is solved first for highest priority group, the airport capacities are updated, the problem is solved for the next highest priority group and so on.

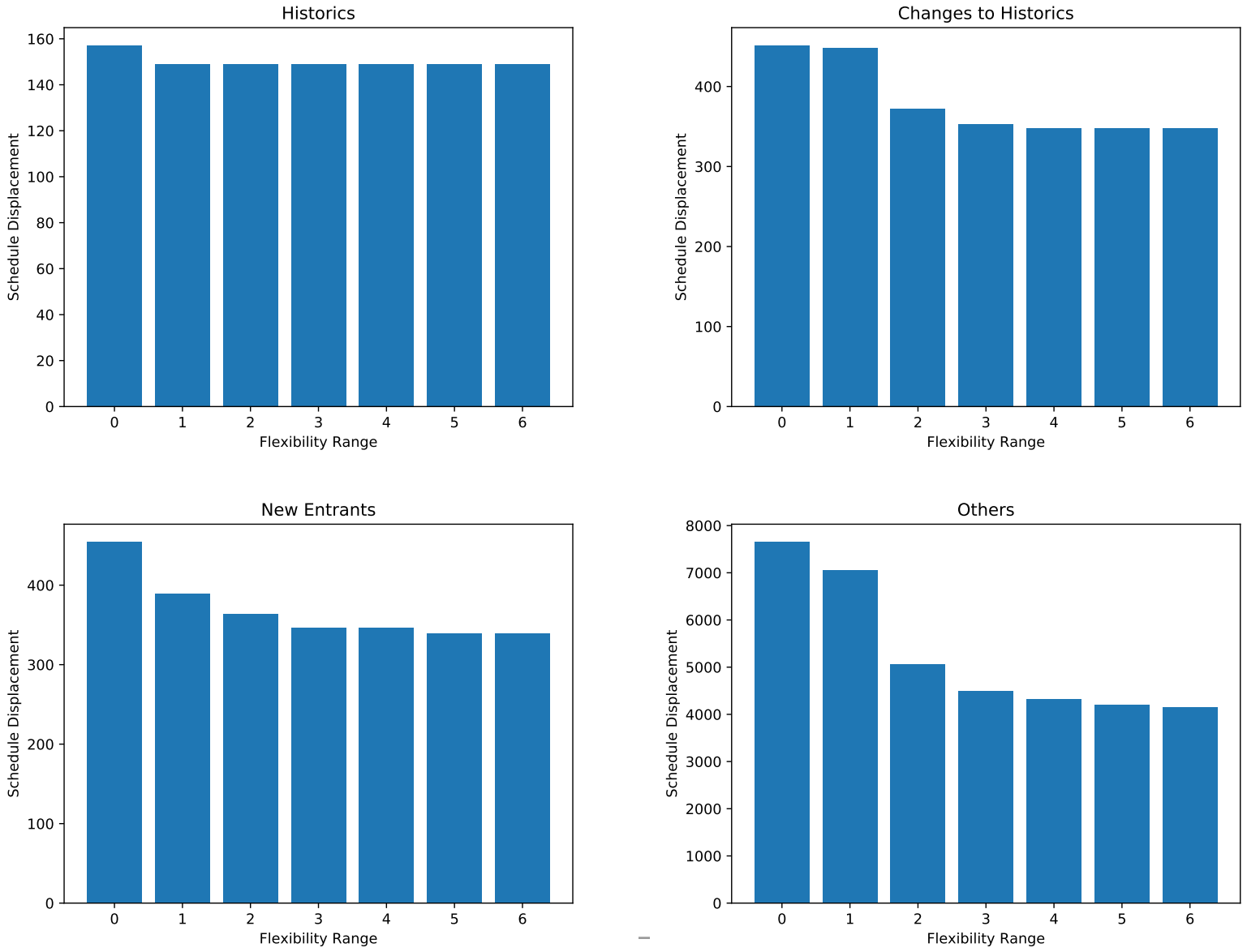
Preprocessing

- 181 (out of 212) scheduling days after date aggregation for Airport 1
- 203 (out of 212) scheduling days after date aggregation for Airport 2

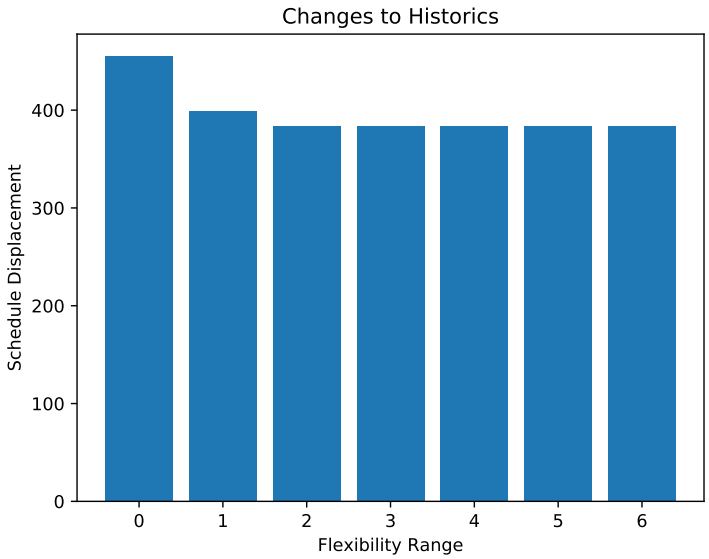
Parameters

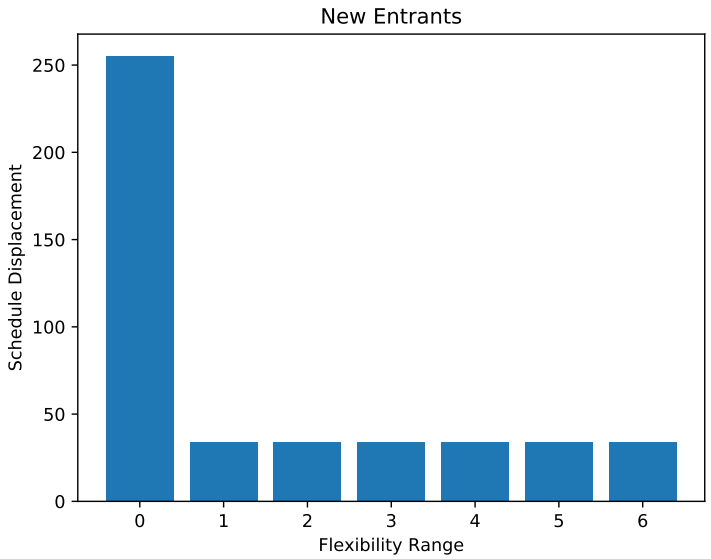
- One coordination time interval corresponds to 15 minutes
- Problems solved using Gurobi 8.1 on Intel(R) Core(TM) i5-4690 CPU @ 3.50GHz
- For time windows we use $R = 3$ hours

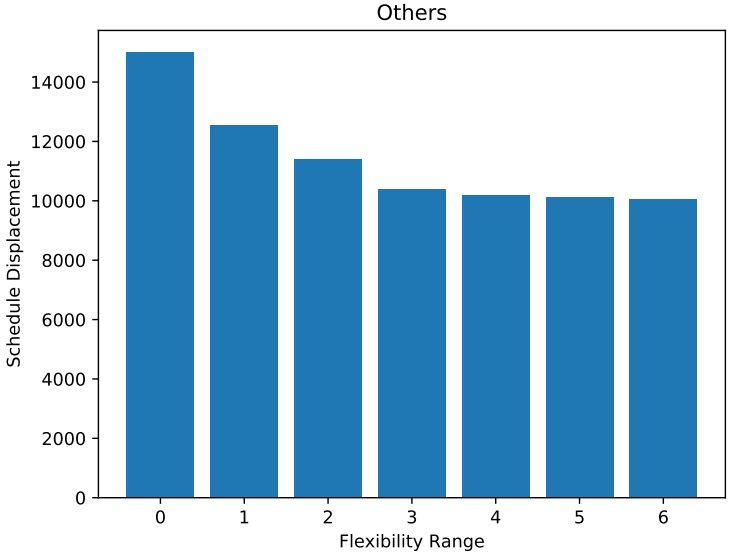
Total Displacement - Airport 1



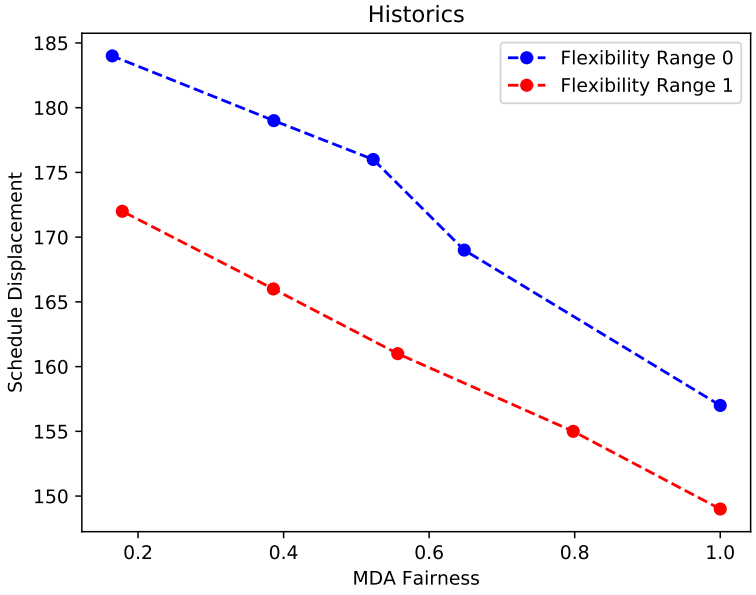
Total Displacement - Airport 2

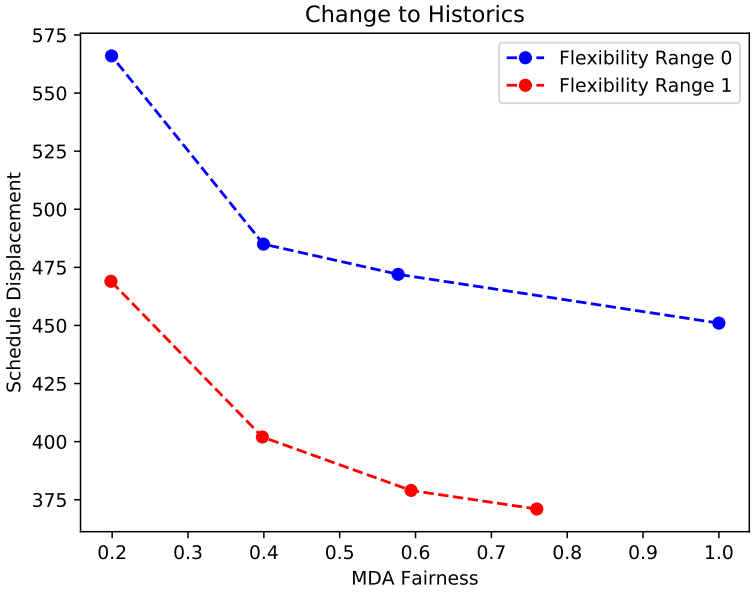


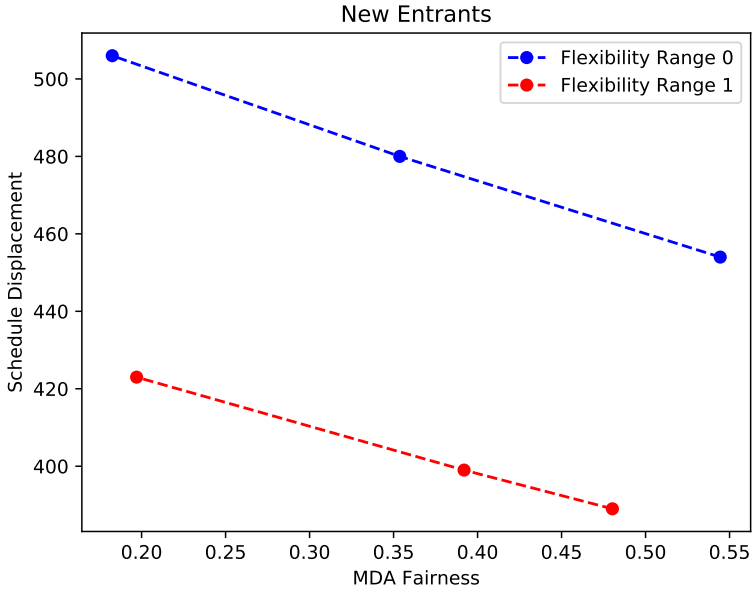


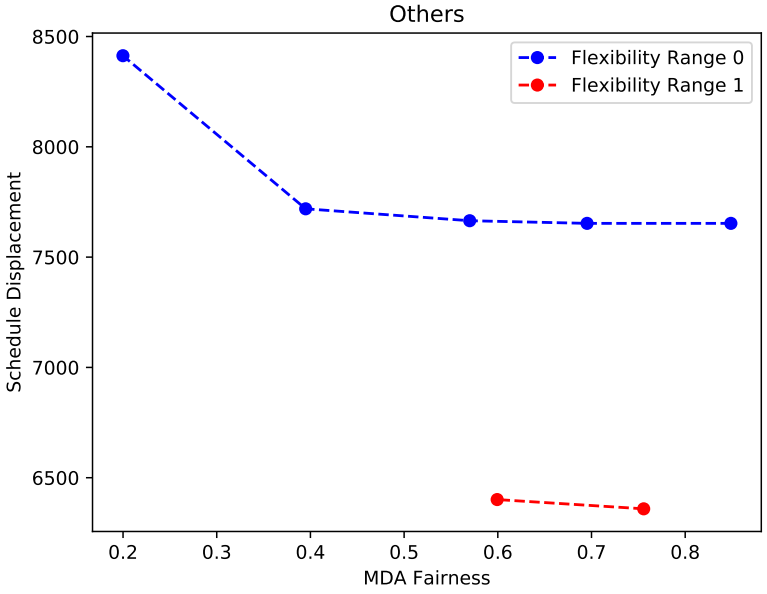


Fairness - Airport 1









Conclusions

- We introduced a new model which incorporates efficiency, fairness and flexibility
- Flexibility in slots allocated to request series subject to range constraints
- Model tested using request data for medium-sized airports:
- Flexibility signficantly reduces displacement, but benefit attenuates for greater ranges
- Flexibility also improves outcomes of fairness

Future Work

- Develop faster algorithms to allow solution for larger problems:
  - Benders decomposition by scheuling day (in progress)
  - Dantzig-Wolfe decomposition by request series
- Development of model which ensures greater schedule consistency

Bibliography I

Fairbrother, J. and Zografos, K. G. (2018).
On the development of a fair and efficient slot scheduling mechanism at congested airports.
Presented at 97th Transportation Research Board Annual Meeting, last accessed May 2018.

Ribeiro, N. A., Jacquillat, A., Antunes, A., Odoni, A., and Pita, J. (2018).
An optimization approach for airport slot allocation under {IATA} guidelines.
Transportation Research Part B: Methodological, 112:132 – 156.

Zografos, K. and Jiang, Y. (2017).
Modelling fairness in slot scheduling decisions at capacity-constrained airports.
presented at Transportation Research Board Annual Conference.

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

Bibliography II

Zografos, K. G., Salouras, Y., and Madas, M. A. (2012).
Dealing with the efficient allocation of scarce resources at congested airports.

Transportation Research Part C: Emerging Technologies, 21(1):244–256.