

Supporting
European
Aviation



Are All the Requested Air Traffic Flow Management Regulations Actually Indispensable?

Identifying effective regulations with adaptive tabu search and strategic oscillations

Ramon Dalmau-Codina, Leila Zerrouki, Camille Anoraud, Darren Smith & Benjamin Cramet

EUROCONTROL Innovation Hub (EIH)

08/12/2021

11th SESAR Innovation Days



Outline



- Problem formulation
- Mathematical formulation
- Optimisation algorithm
- Illustrative example
- Conclusions and future work



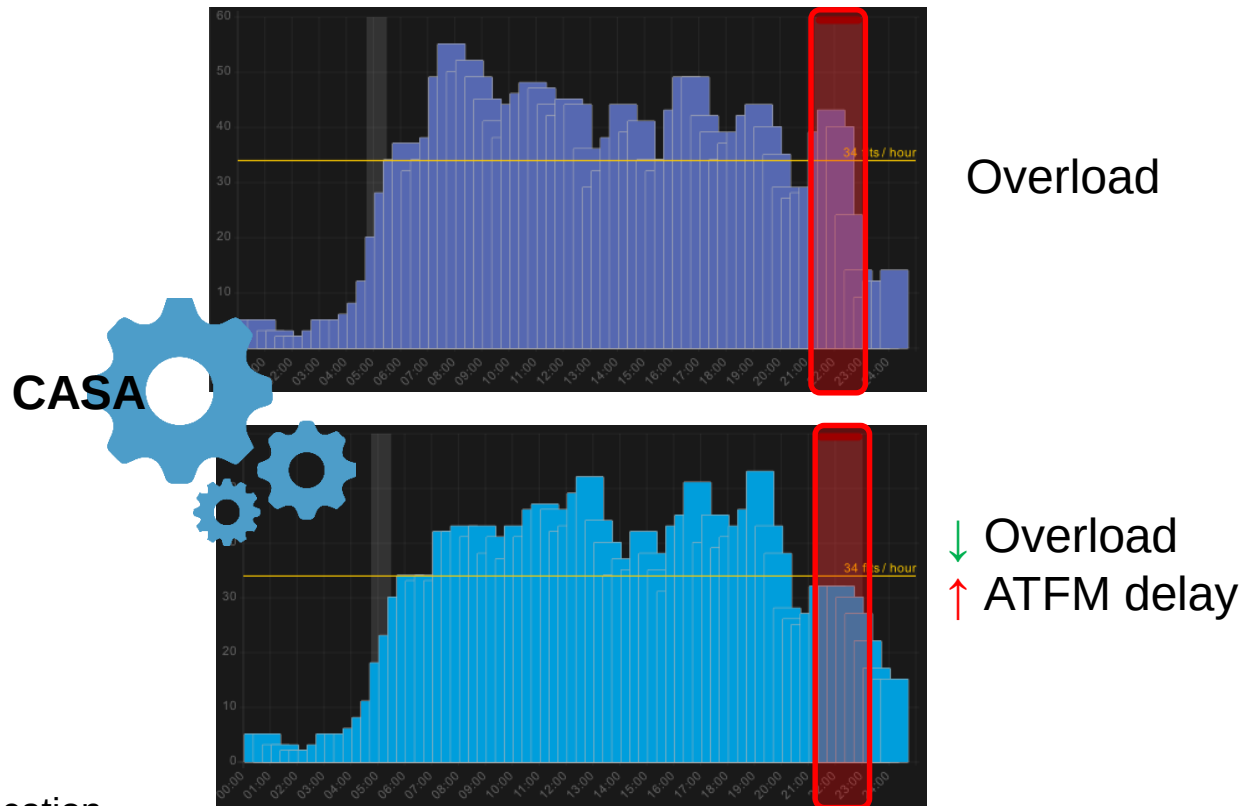
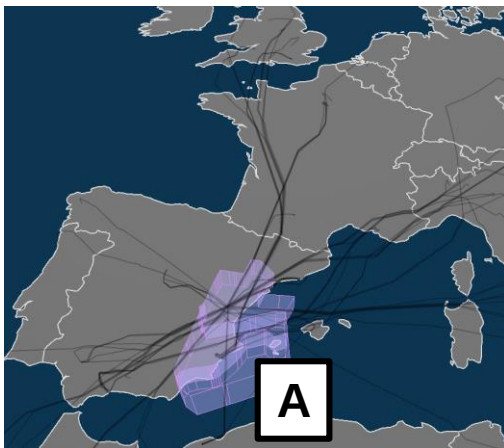
11th SESAR Innovation Days



Problem formulation

- Air traffic flow management (**ATFM**) regulations are used by the Network Manager to solve demand and capacity balance (**DCB**) problems

Example: 01/09/2019 - LECBCTAW

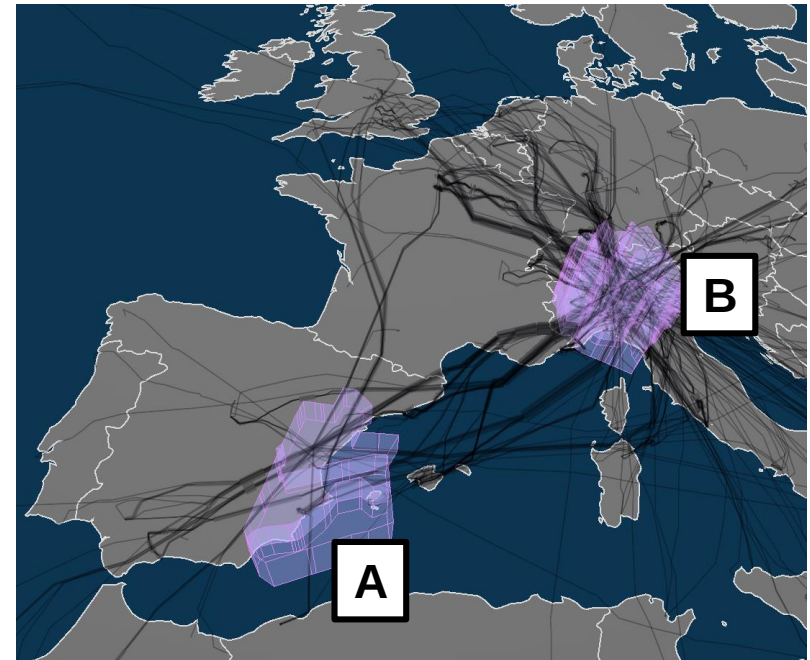


CASA: Computer Assisted Slot Allocation

11th SESAR Innovation Days

Problem formulation

- During busy days, many regulations are proposed by the flow managers
- A flight subject to several regulations is assigned the *most penalising* (highest) delay, and is *pushed* into the other regulations with this delay
- Regulations with flights in common influence each other



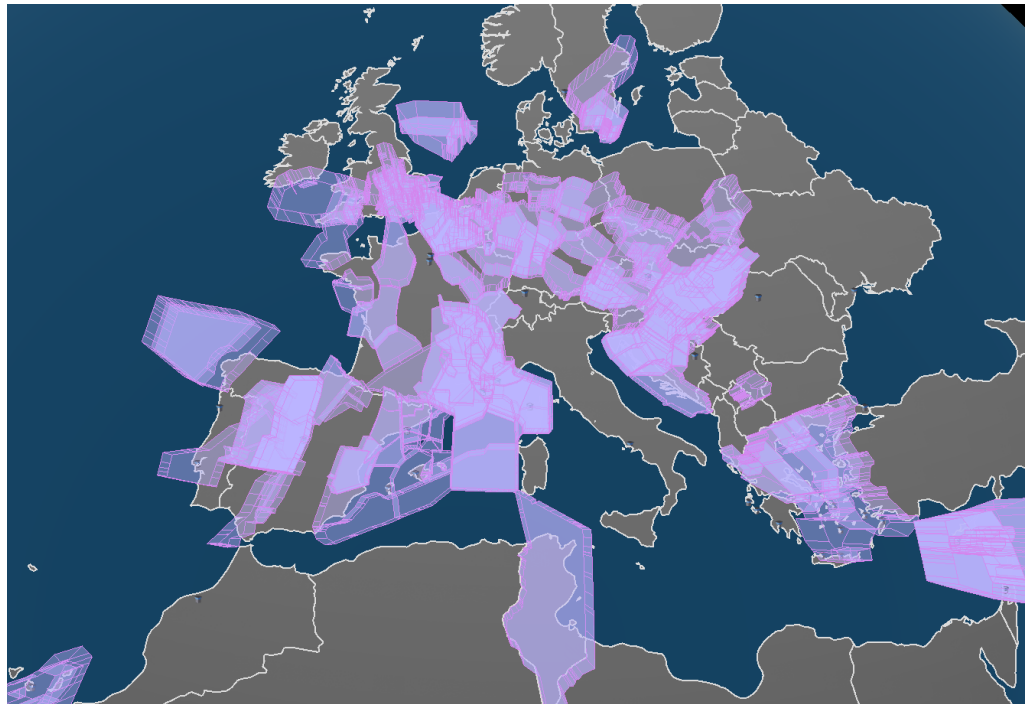
Can we cancel regulation A?

Any flight affected by A will impact the sequence in B

The new solution must reduce the total ATFM delay (objective)
Without creating overloads elsewhere in the network (constraints)

Problem formulation

This **binary combinatorial optimisation problem** cannot be solved *by hand* ...



Example: with *only* 25 regulations, there exist 2^{25} possible combinations*

**Assuming that each regulation can be either cancelled (0) or active (1)*

Mathematical formulation

Binary combinatorial optimisation problem

Without exceeding the rate of the regulations during the regulated periods

$$\min_{\mathbf{x} \in \{0,1\}^{\|\mathcal{R}\|}}$$

$$\sum_{i \in \mathcal{R}} f_i(\mathbf{x})$$

Minimise total ATFM delay

$$\text{s.t. } \sum_{j \in \mathcal{W}_i} \max(0, d_{ij}(\mathbf{x}) - r_i) \leq 0, \forall i \in \mathcal{R},$$

$\mathbf{x}$: binary solution vector

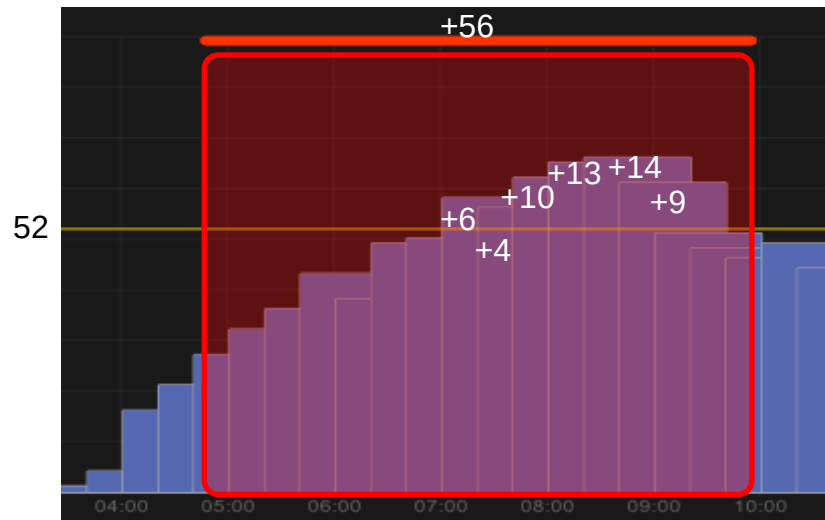
$f_i(\mathbf{x})$: ATFM delay generated by regulation i

$d_{ij}(\mathbf{x})$: demand in the time window j of regulation i

r_i : entry rate of regulation i

$\mathcal{R}$: set of ATFM regulations subject to optimisation

$\mathcal{W}_i$: time windows within the period of regulation i



Mathematical formulation

Binary combinatorial optimisation problem

Without exceeding
the rate of the
regulations during the
regulated periods

$$\min_{\mathbf{x} \in \{0,1\}^{\|\mathcal{R}\|}}$$

$$\sum_{i \in \mathcal{R}} f_i(\mathbf{x})$$

Minimise total ATFM delay

$$\text{s.t. } \sum_{j \in \mathcal{W}_i} \max(0, d_{ij}(\mathbf{x}) - r_i) \leq 0, \forall i \in \mathcal{R},$$

(binary) weighted constraints satisfaction problem (**WCSP**)

$$\{f_i(\mathbf{x}) \leq 0 \mid i \in \mathcal{R}\},$$

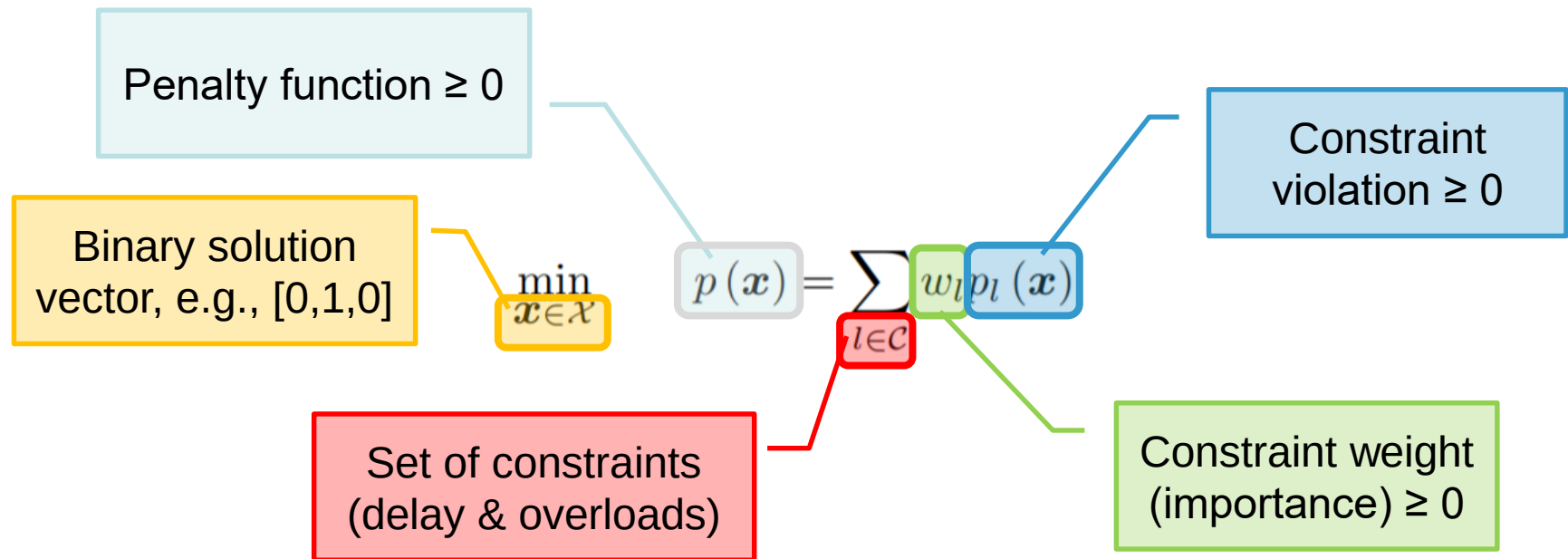
Set of delay constraints

$$\left\{ \sum_{j \in \mathcal{W}_i} \max(0, d_{ij}(\mathbf{x}) - r_i) \leq 0 \mid i \in \mathcal{R} \right\}.$$

Set of overload constraints

Mathematical formulation

(binary) weighted constraints satisfaction problem (**WCSP**)



- Each element of the solution vector $\mathbf{x}$ corresponds to one regulation r in R indicating whether the regulation r is cancelled (0) or active (1)
- Each regulation r in R is associated with two constraints:
 - Delay generated by $r \leq 0$
 - Sum of overloads in $r \leq 0$

Mathematical formulation

- How to set the weight w_l associated to each constraint l ?
 - Constraints with small weights tend to be ignored and
 - too much emphasis placed on solving constraints with large weights
 - What is more important? delays or overloads? how much? are all regulations equally important?
- Weights should be dynamically adjusted

(original) penalty function ≥ 0

(search) evaluation function ≥ 0

$$\min_{\mathbf{x} \in \mathcal{X}} \quad p(\mathbf{x}) = \sum_{l \in \mathcal{C}} w_l p_l(\mathbf{x})$$

Fixed weight
(importance) ≥ 0

$$\min_{\mathbf{x} \in \mathcal{X}} \quad q(\mathbf{x}, \mathbf{v}) = \sum_{l \in \mathcal{C}} v_l p_l(\mathbf{x})$$

Adaptive weight
(importance) ≥ 0

$$0 \leq v \leq w$$

- How to minimise $q(\mathbf{x}, \mathbf{v})$?
- How to dynamically adjust the weights in $\mathbf{v}$?

Optimisation algorithm

- How to minimise $q(\mathbf{x}, \mathbf{v})$? Any optimisation method, e.g., tabu search
- How to dynamically adjust the weights in $\mathbf{v}$? Based on the current violations

Algorithm 3 Adaptive Tabu Search with Strategic Oscillations

Require: $\mathbf{x}, \mathbf{w}$

```

1:  $\mathbf{x}_* \leftarrow \mathbf{x}$ 
2:  $\mathbf{v} \leftarrow \mathbf{w}$ 
3: repeat
4:    $(\mathbf{x}_{\text{local}}, \mathbf{x}_{\text{global}}) \leftarrow \text{TABUSEARCH}(\mathbf{x}, \mathbf{v})$ 
5:    $\mathbf{x} \leftarrow \mathbf{x}_{\text{local}}$ 
6:   if  $p(\mathbf{x}_{\text{global}}) < p(\mathbf{x}_*)$  then
7:      $\mathbf{x}_* \leftarrow \mathbf{x}_{\text{global}}$ 
8:   end if
9:    $\mathbf{v} \leftarrow \text{PENALTYWEIGHTUPDATE}(\mathbf{v}, \mathbf{x}, \mathbf{x}_*)$ 
10: until TERMINATION
11: return  $\mathbf{x}_*$ 

```

Update current solution $\mathbf{x}$ with tabu search to minimise the evaluation function $q(\mathbf{x}, \mathbf{v})$

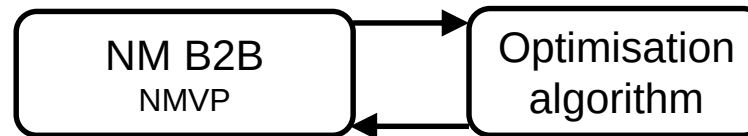
Update incumbent $\mathbf{x}^*$ to minimise the penalty function $p(\mathbf{x})$

Update adaptive weights $\mathbf{v}$

Illustrative example

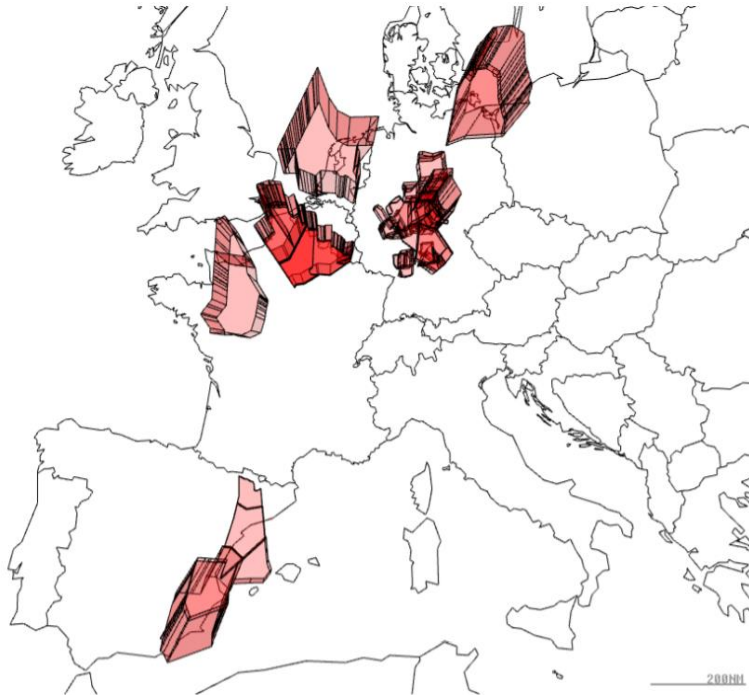
Real CASA and data

In Python

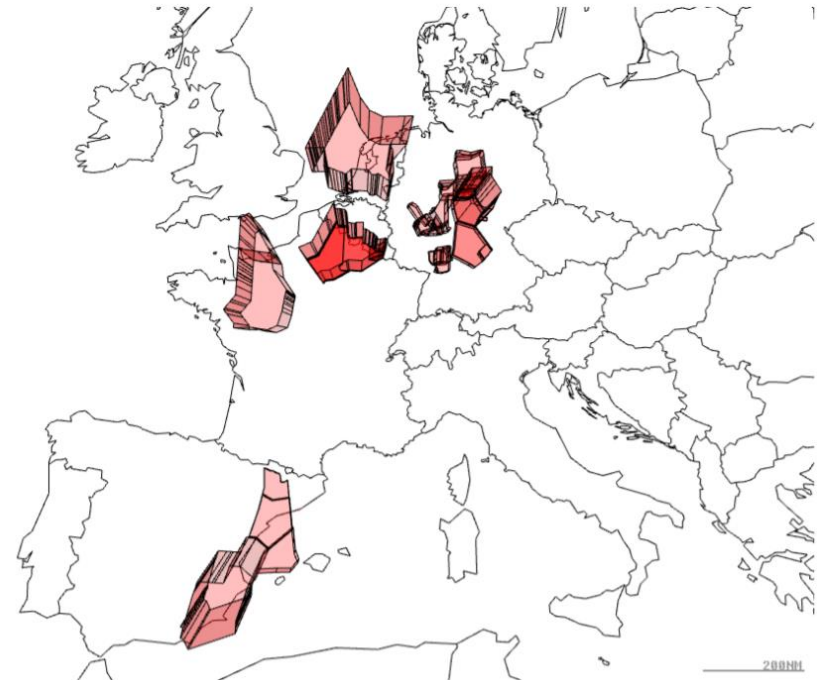


- **Simulator:** Network Manager validation platform (**NMVP**), which fully replicates the real Network Management operational systems (including CASA)
- **Replay date:** A busy day (27/06/2019) in real-time, starting at 07:40 AM
 - 120 regulations in the network, generating 43K minutes of delay
- **Regulation cause:** Air traffic control (**ATC**) capacity
- **Scope:** Spain, France, Belgium, Germany, Luxembourg and the Netherlands
 - 21 regulations, generating 5343 minutes of delay
 - $2^{21} \approx 2000000$ possible solutions x...
- **Maximum number of solutions:** 100 (<0.01% of the search space)
- **Procedure:** Local optimisation was followed by a network impact assessment

Illustrative example



Before optimisation (21 regulations)



After optimisation (13 regulations)

Illustrative example – Local optimisation results

Delay difference for the 21 regulations subject to optimisation

Regulation <i>i</i>	Flights		Delay		
	Before	After	Before	After	Difference
KWUR327E	183	181	634	262	-372
KFFM2427	132	127	345	0	-345
LELVL27	93	58	280	0	-280
E4N27	246	175	278	0	-278
EDG7NO27	57	35	236	3	-233
EDG4DK27	51	22	103	0	-103
EDG3GH27	33	26	57	20	-37
KOSE1O27	52	46	87	54	-33
LEBLA27	44	44	107	107	0
E5R27	278	278	0	0	0
EDG3GI27	47	47	222	222	0
KWUR1C27	67	67	132	132	0
K1UFX27	67	67	50	50	0
EDG2NT27	44	44	165	165	0
LEGE1V27	72	72	132	145	+13
KWUR2427	127	127	537	552	+15
RQXU27E	51	51	145	170	+25
EDG3TA27	34	34	365	399	+34
EDWHRZ27	70	70	271	312	+41
LELVU27	98	98	518	561	+43
YD6WH27M	145	145	679	809	+130
Total	1991	1814	5343	3963	-1380

What about the other 99 regulations active in the network?

What about the traffic volumes not monitored during the optimisation?

11th SESAR Innovation Days

Illustrative example – Network Impact Assessment

Delay difference for the 120 regulations active in the network

Regulation <i>i</i>	Flights		Delay		
	Before	After	Before	After	Difference
KWUR327E	183	181	634	262	-372
KFFM2427	132	127	345	0	-345
LHLYBA27	290	290	1410	1074	-336
LELVL27	93	58	280	0	-280
E4N27	246	175	278	0	-278
EDG7NO27	57	35	236	3	-233
KNTM3C27	156	156	1583	1356	-227
LOWB3527	65	65	577	395	-182
LOWB1227	155	155	2021	1906	-115
EDG4DK27	51	22	103	0	-103
...					
EDG3TA27	34	34	365	399	+34
LHWSLM27	177	177	571	610	+39
EDWHRZ27	70	70	271	312	+41
LPPTA27	120	120	3679	3721	+42
LELVU27	98	98	518	561	+43
EHAMA27M	224	224	2306	2349	+43
LHENU27	116	116	241	296	+55
EGLLA27M	112	112	2476	2556	+80
KNTM1C27	90	90	1386	1479	+93
YD6WH27M	145	145	679	809	+130
Total	7304	7127	43051	40842	-2209

Overload alert difference for the traffic volumes active in the network

Traffic volume	Period	Before	After	Difference
LFEUBN	08:20 - 11:40	8	6	-2
EG12GCLE	06:20 - 08:40	7	5	-2
LEMDARR	08:20 - 10:40	2	0	-2
LOVVWB12	06:00 - 11:20	15	13	-2
LFEKR	11:00 - 11:40	2	1	-1
MASB5WL	05:40 - 11:00	10	9	-1
LOVVE15	07:20 - 12:20	13	12	-1
LHLYBALL	09:20 - 10:40	2	1	-1
LCWU	10:00 - 12:00	5	4	-1
LDULWX	07:40 - 08:20	2	1	-1
...				
LPCEU1	11:20 - 11:20	0	1	+1
LFMY34	09:20 - 10:20	2	3	+1
LFEKHR	08:40 - 15:00	17	18	+1
EDG1LA05	07:20 - 07:20	0	1	+1
LFMMOML	08:00 - 09:40	2	3	+1
EGGWARR	10:40 - 15:40	4	5	+1
EGDTS	05:40 - 07:20	4	5	+1
EDG4DKB	05:40 - 08:40	0	1	+1
LOVVWB35	07:00 - 11:20	14	16	+2
LYBATWES	06:40 - 11:20	14	16	+2
Total	-	1831	1828	-3

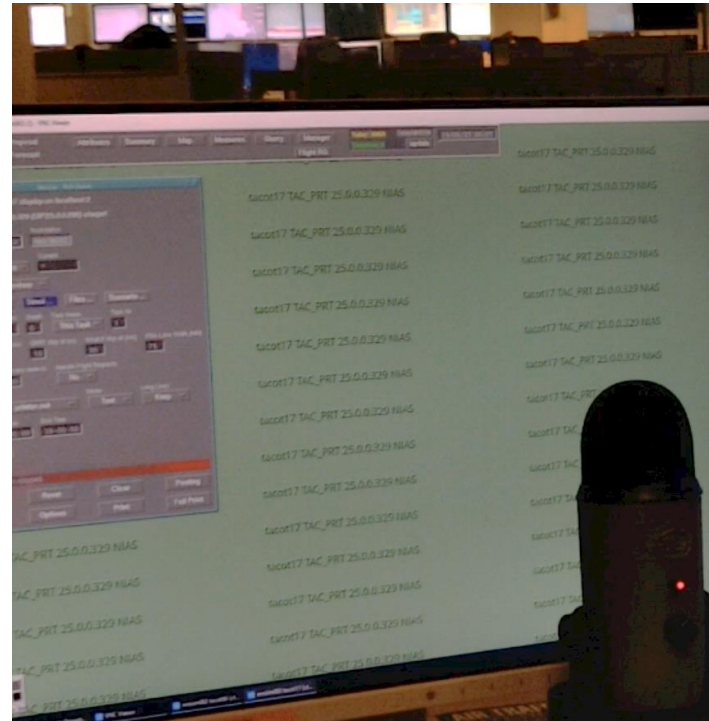
11th SESAR Innovation Days

Conclusions and future work

- "Are all the requested ATFM regulations actually indispensable?" Is a difficult problem due to the complexity of the network – cannot be solved by hand
- Adaptive tabu search with strategic oscillations can identify which regulations could be *safely* cancelled while minimising the total ATFM delay
- Other algorithms may be also good at solving this problem: adaptive large neighborhood search (ALNS), harmony search, branch & bound, ...
- Operational exercise on 16/11/2019 at the Network Manager Operations Centre (**NMPC**) - 8 runs executed by two different subjects
- Extend the algorithm to propose new regulations and / or modify existing ones (e.g., the rate, the start / end times) – Multi-agent reinforcement learning (work in progress)

Operational exercise at the NMOC

CFMU TACTICAL (TACT)



Algorithm HMI



11th SESAR Innovation Days

THANK YOU!



11th SESAR Innovation Days

