OPTIENGINE | MRO SIMULATION TOOL

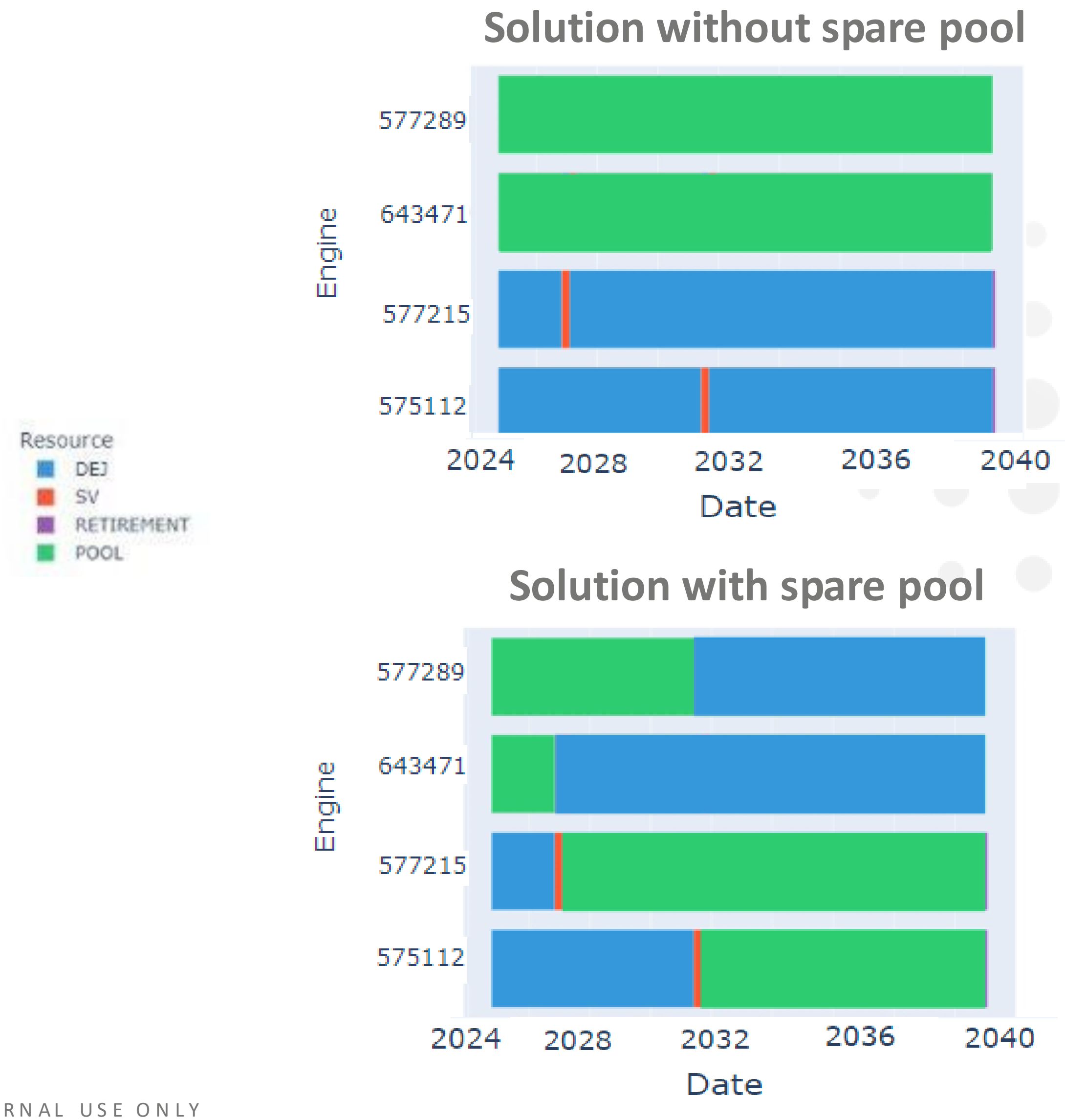
Metaheuristic Optimization

2025

IAG.ai

CONTEXT

- Maintenance planning problem
  - Decisions:
    - When to performe a SV and workscope for each SV
    - When to redeliver an engine
    - Which aircraft an engine will fly in
  - Objective:
    - Minimize total maintenance cost
  - Example of constraints:
    - No aircraft downtime if there are engines in the pool
    - Maximum SV per month = 2
    - Maximum SV per year = 15
    - Maximum redeliveries per year = 10
    - No SVs in summer months
    - ....



LINEAR PROGRAMMING APPROACHES

- Budget forecasting (IAG - MRO)
  - Baseline
    - SV dates and workscopes that maximize the use of the engine parts and the contractual redelivery dates
  - Generation of scenarios
    - 200 scenarios per engine where one or more parameters can change respect of the baseline (SV dates, SV workscopes or redelivery dates)
  - Optimization model
    - Output: scenario per engine that satisfy the global operational constraints and minimize the total maintenance cost
    - Solver: free solver
- Vueling solution
  - Optimization model 1: subproblem
    - One optimization model execution per engine to plans the SV dates.
    - Optimal solution + 99 non-optimal solutions
  - Optimization model 2: master problem
    - Output: scenario per engine that satisfy the global operational constraints and minimize the total maintenance cost
    - Solver: Gurobi

None of them works with the spare pool nor apply a global optimization

IAG.ai

PROBLEM OVERVIEW (I)

- The Job Shop Scheduling Problem (JSSP) is a classic optimization problem in operations research and manufacturing.
  - It involves scheduling a set of jobs, each consisting of a sequence of tasks, on a set of machines with the goal of optimizing certain criteria, such as minimizing the total time to complete all jobs (makespan), minimizing delays, balancing machine utilization, or reducing costs.
- Key Components:

1. **Jobs:** Distinct entities, each comprising multiple tasks.
2. **Tasks:** Operations that make up a job, with predefined processing times.
3. **Machines:** Resources required to process tasks. Each task requires a specific machine.
4. **Constraints:**
   - Precedence constraints: Tasks in a job must follow a specific order.
   - Machine constraints: Each machine can process only one task at a time.
   - No preemption: Once a task starts on a machine, it cannot be interrupted.

PROBLEM OVERVIEW (II)

- Is this problem similar to ours?

1. Jobs: Distinct entities, each comprising multiple tasks. → **Engine lifecycle**
2. Tasks: Operations that make up a job, with predefined processing times. → **Dividing the engine lifecycle into segments to assign each segment to a specific "machine" for a defined period.**
3. Machines: Resources required to process tasks. Each task requires a specific machine. → **Assignment of each task (engine+period) to an aircraft, shop visit, or pool.**
4. Constraints:
   - Precedence constraints: Tasks in a job must follow a specific order. → **Shop visits can only be planned under specific conditions, and an engine cannot be assigned to two different aircraft consecutively, except in certain cases (e.g., aircraft retirement).**
   - Machine constraints: Each machine can process only one task at a time. → **Each aircraft must be assigned two engines.**
   - No preemption: Once a task starts on a machine, it cannot be interrupted.

This problem is more complex than a classic JSSP, as the number of tasks and their durations (e.g., lifecycle segments and time in each segment) are not known beforehand.

IAG.ai

PROBLEM OVERVIEW (III)

- JSSP is an NP-hard problem, meaning that finding an exact solution for large instances is computationally challenging. As a result, heuristic and metaheuristic approaches (e.g., genetic algorithms, simulated annealing) are often used for practical solutions.

- Metaheuristic optimization refers to a class of algorithms designed to find good (often near-optimal) solutions to complex optimization problems in a reasonable time.
  - They are inspired by natural processes, such as evolution, swarm behavior, or thermodynamics.
  - They are problem-independent and can be applied to various optimization problems without requiring major modifications. Thus, they are robust to the introduction of new constraints or changes in the cost function.
  - They are widely used in scheduling, routing, machine learning, logistics, and other domains where exact methods are impractical.

- Balancing exploration and exploitation:
  - Exploration: Searching broadly across the solution space to avoid getting stuck in local optima.
  - Exploitation: Refining promising solutions to improve their quality.

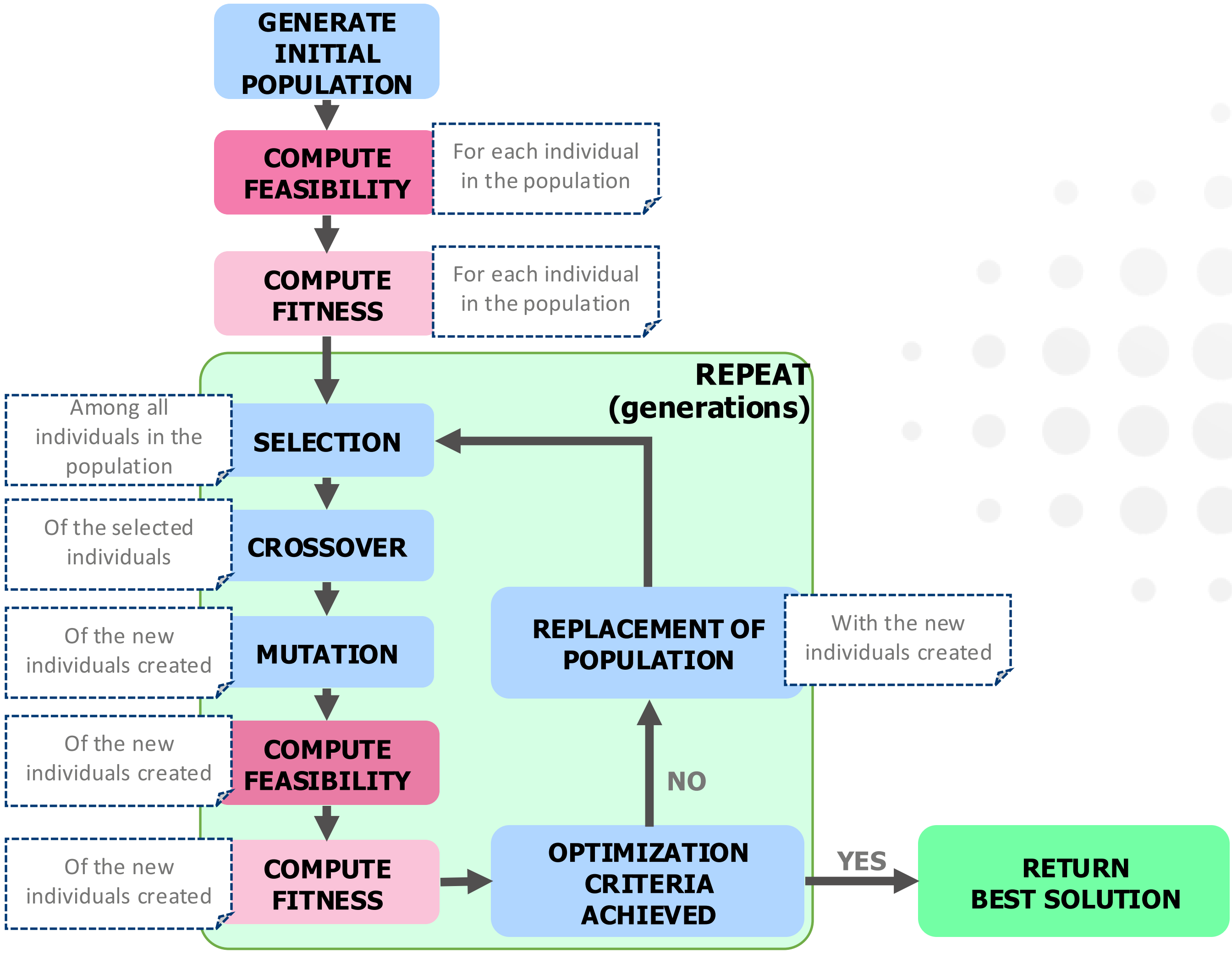
PROBLEM OVERVIEW

REFERENCES

- Amjad, M. K., Butt, S. I., Kousar, R., Ahmad, R., Agha, M. H., Faping, Z., ... & Asgher, U. (2018). Recent research trends in genetic algorithm based flexible job shop scheduling problems. *Mathematical Problems in Engineering*, 2018(1), 9270802.
- Benni, R., Umarani, S. R., & Totad, S. (2024). A Comprehensive Study of Meta-Heuristic Algorithms for Job Shop Scheduling Optimization. In *2024 15th International Conference on Computing Communication and Networking Technologies (ICCCNT),* 1-10. IEEE.
- Bhatt, N., & Chauhan, N. R. (2015). Genetic algorithm applications on job shop scheduling problem: A review. In *2015 International Conference on Soft Computing Techniques and Implementations (ICSCTI),* 7-14. IEEE.
- Nicoara, E. S., Filip, F. G., & Paraschiv, N. (2011). Simulation-based optimization using genetic algorithms for multi-objective flexible JSSP. *Studies in Informatics and Control*, 20(4), 333-344.
- Pimapunsri, K., Weeranant, D., & Riel, A. (2022). Genetic algorithms for the resource-constrained project scheduling problem in aircraft heavy maintenance. *arXiv*. https://arxiv.org/abs/2208.07169

ALGORITHM

- A **Genetic Algorithm** (GA) is a metaheuristic technique inspired by natural selection, where potential solutions (i.e., individuals) evolve through selection, crossover, and mutation to find the best solution according to feasibility (constraints) and fitness (objective function).



SOLUTION

INDIVIDUAL DEFINITION – ALLOCATION PRIORITY

Engines

12	23	35	37	43	22	…	13	62	27	2	0	55

Priority to allocate engines:
[0, 1, 2..n]

<u>**Interpretation**</u>

Engines

- Each engine has a value that defines the priority of allocation from 0 to the max number of engines.
- No engines can share the same priority.

AIRCRAFT 3	PRSV	POOL	AIRCRAFT 1	RETIREMENT
AICRAFT 1	PRSV	POOL	AIRCRAFT 2	RETIREMENT

1. The allocation of engines is ordered by date.
2. If two engines share the same dates of allocation the priority decides which one goes first.
3. If an engine doesn't share allocation dates with another engine the pool will act as a First In, First Out.

Objective function: Minimize total maintenance costs by scheduling engine entries and exits from the pool.

SOLUTION

INDIVIDUAL DEFINITION – ENGINE SV

Engine 1			Engine 2				Engine 60			Engine 61		
0	2	3	1	4	3	…	5	4	6	0	2	0

- Permitted months in advance for PRSV before engine life ends: [0..6]
- maximum PRSVs per engine : 3

Interpretation

Engine 1

- The 1st scheduled PRSV won't change.
- The 2nd scheduled PRSV will be done 2 months in advance before having an LLP out of life.
- The 3rd scheduled PRSV will be done 3 months in advance before having an LLP out of life.

Engine 2

- The 1st scheduled PRSV will be done 1 months in advance before having an LLP out of life.
- The 2nd scheduled PRSV will be done 4 months in advance before having an LLP out of life.
- The 3rd scheduled PRSV will be done 3 months in advance before having an LLP out of life.

…

1. The maximum PRSVs per engine can be defined by the user.
2. The permitted months in advance for each PRSV before engine life ends can be defined by the user.
3. The PRSVs will be defined by the model when it finds that an engine can't fly the next month because it has no remaining life left.

Objective function: Minimize the total maintenance cost by moving PRSV dates.

SOLUTION

INDIVIDUAL DEFINITION – PRSVs WORKSCOPES & RESTORATION

Engine 1			Engine 2				Engine 60			Engine 61		
2	1	4	3	2	1	…	3	0	2	4	1	0

Type of restoration done in a PRSV:
- Max-life: 0
- Half-life: 1
- 7K: 2
- 5K: 3
- Fanswap: 4

<u>**Interpretation**</u>

Engine 1
- The 1st scheduled PRSV will try to do a workscope with 7k restoration on LLPs.
- The 2nd scheduled PRSV will try to do a workscope with half-life restoration on LLPs.
- The 3rd scheduled PRSV will try to do a fanswap workscope.

Engine 2
- The 1st scheduled PRSV will try to do a workscope with 5k restoration on LLPs.
- The 2nd scheduled PRSV will try to do a workscope with 7k restoration on LLPs.
- The 3rd scheduled PRSV will try to do a workscope with half-life restoration on LLPs.

…

Note*: each restoration type have different workscopes (LPC,LPT or HPC,HPC or HPC,HPT,LPC…)

1. The model asumes that we have an unlimited supply of second-hand LLPs.
2. The workscope will be selected by the model between all the possible workscopes on each restoration type to repair the engine, having in mind that each LLP < 500 RFC needs to be repaired.
3. If a defined restoration type is not possible the model will try another one that fits better.

Objective function: Minimize the total maintenance cost by using different workscopes.

SOLUTION

INDIVIDUAL DEFINITION – ENGINE RETIREMENTS

Aircraft 1		Aircraft 2		Aircraft 3			Aircraft 25		Aircraft 26		Aircraft 27	
7	8	4	5	6	0	…	21	32	37	34	22	26

1st engine of the aircraft

2nd engine of the aircraft

<u>**Interpretation**</u>

Aircraft 1

- Retired with tittled engines. (engine 7, engine 8)

Aircraft 2

- Retired with tittled engines. (engine 4, engine 5)

…

1. Ensure that all aircraft are retired with their tittled engines.

Objective function: Respect the retirements defined in the contract.

MOVING FORWARD

With the genetic algorithm

New constraints

1. Maximum number of PRSVs per month
2. Months without PRSVs

Better modelling

1. Installation logic for CFM56-5B4/P and CFM56-5B4/3
2. PRSV in the first month if required due to engine status
3. No PRSV if engine RFC > 500

New options in PRSV planning

1. Use of second-hand LLPs (mid-life, 7000 RFC and 5000 RFC)
2. Module swap if workscope is only LPC

PRSV and EOLC costs

1. Calculation of EOLC and PRSV costs according to input data
2. Calculation of cashflow
3. Calculation of provisions and depreciation for building the P&L:
   - Between events
   - From delivery to redelivery (only for provisions)
   - Fixed period of time (only for depreciation)

NEW CONSTRAINTS

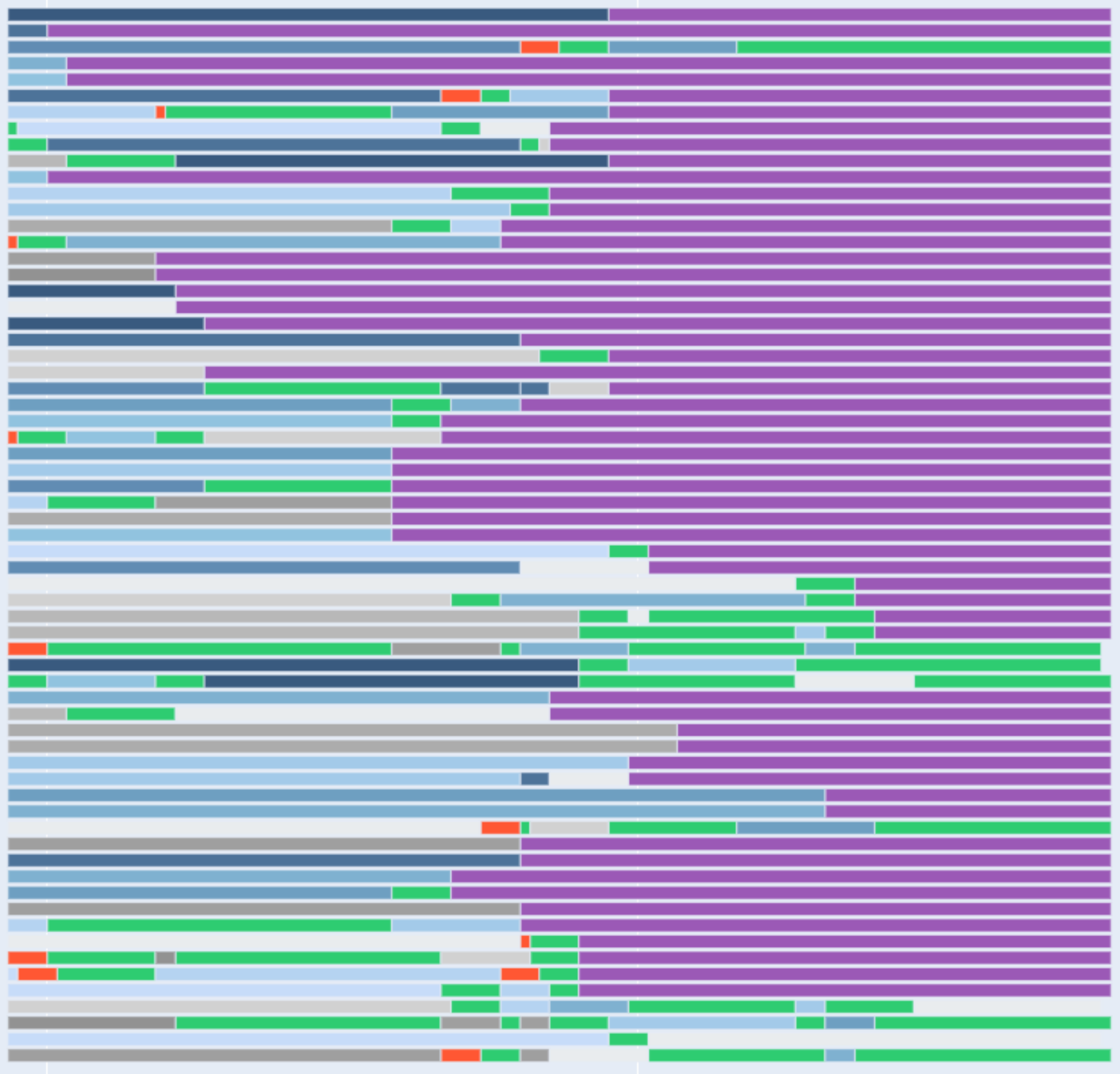
Comparison of an execution with and without constraints

1. Maximum number of PRSVs per month (with max 2 PRSVs in May)

NEW

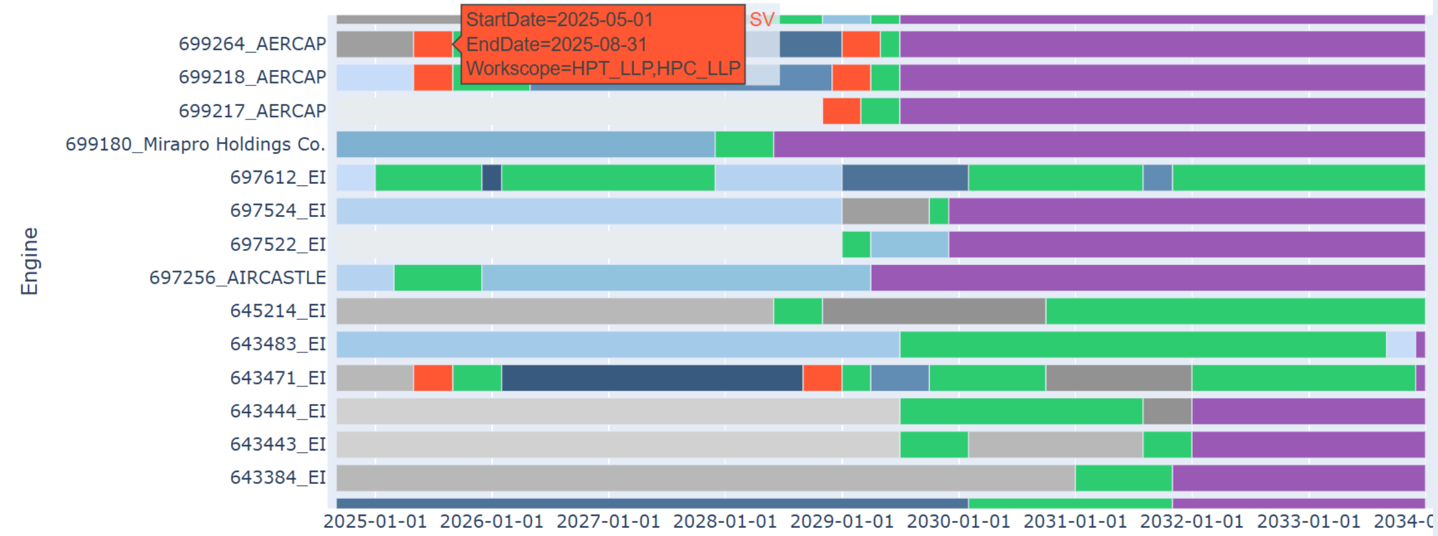
Can set maximum PRSVs for each month, for example:
{1: 2, 2: 2, 3: 1, 4: 2, 5: 2, 6: 1, 7: 0, 8: 0, 9: 1, 10: 2, 11: 2, 12: 2}

No more than 2 PRSV starting at the same month
(instead of forced PRSV at 1st month)



3 engines starting PRSV in May

OLD

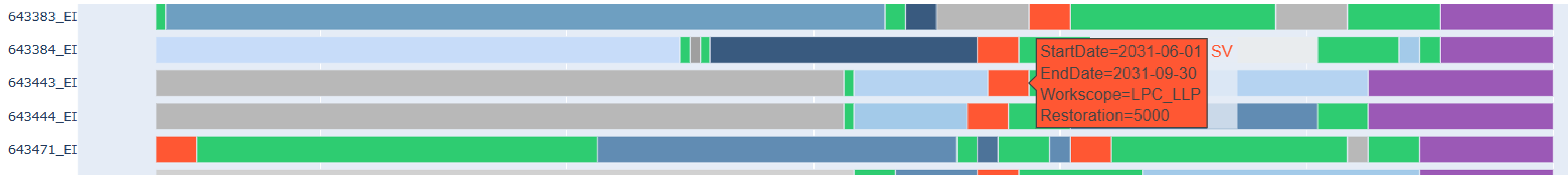


NEW CONSTRAINTS

Comparison of an execution with and without constraints

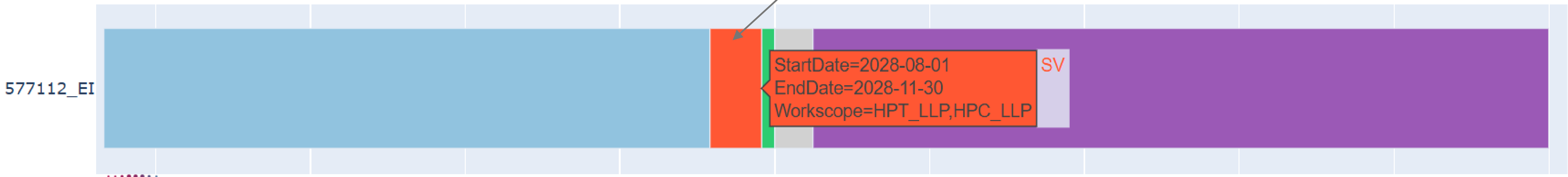
2. Months without PRSVs

Avoidable months → June & September (1 PRSV allowed if needed)



Forbidden months → July & August (0 PRSVs allowed)

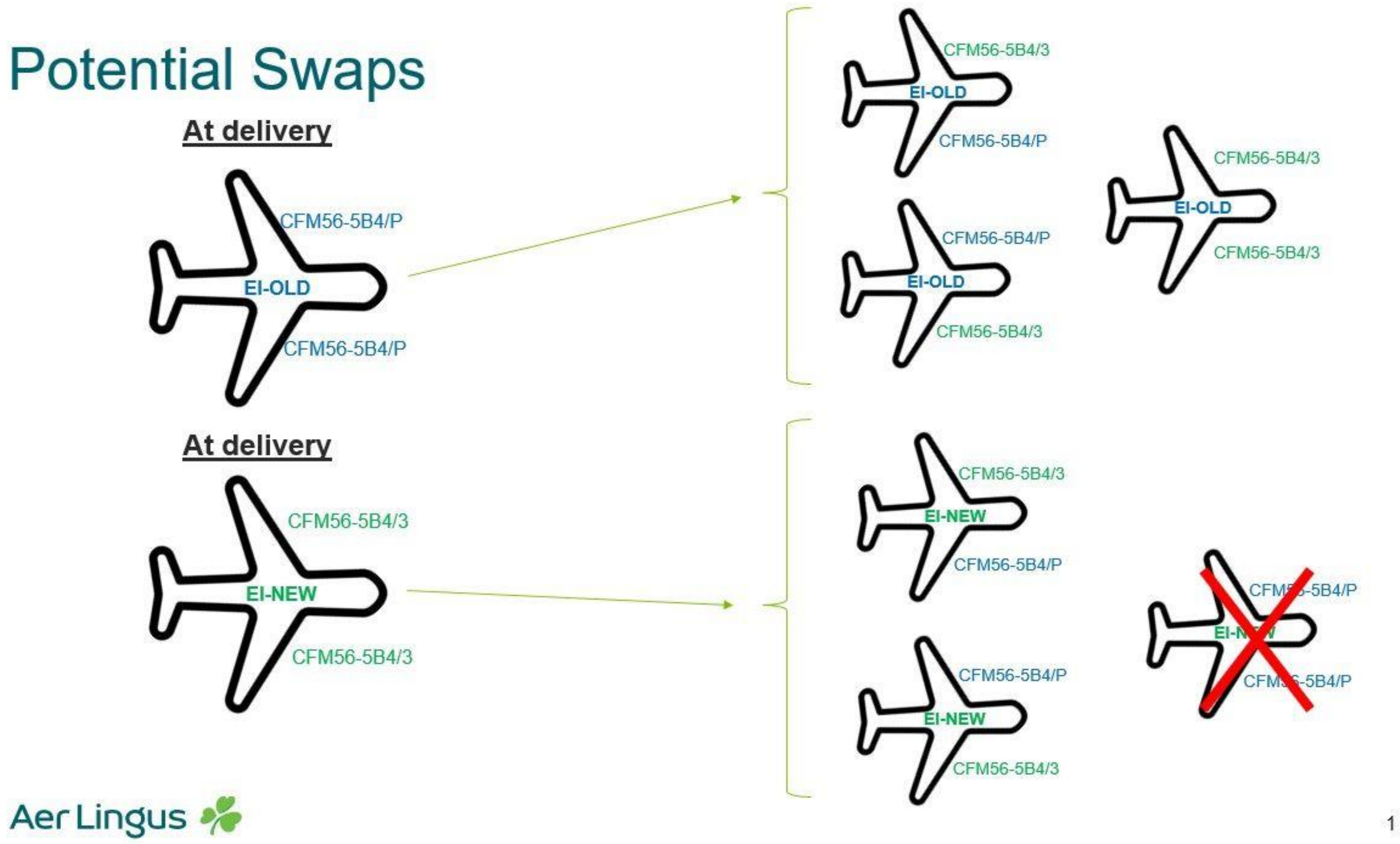
The model will avoid this situation



BETTER MODELLING

Application of new logics in the model

1. Installation logic for CFM56-5B4/P and CFM56-5B4/3



BETTER MODELLING

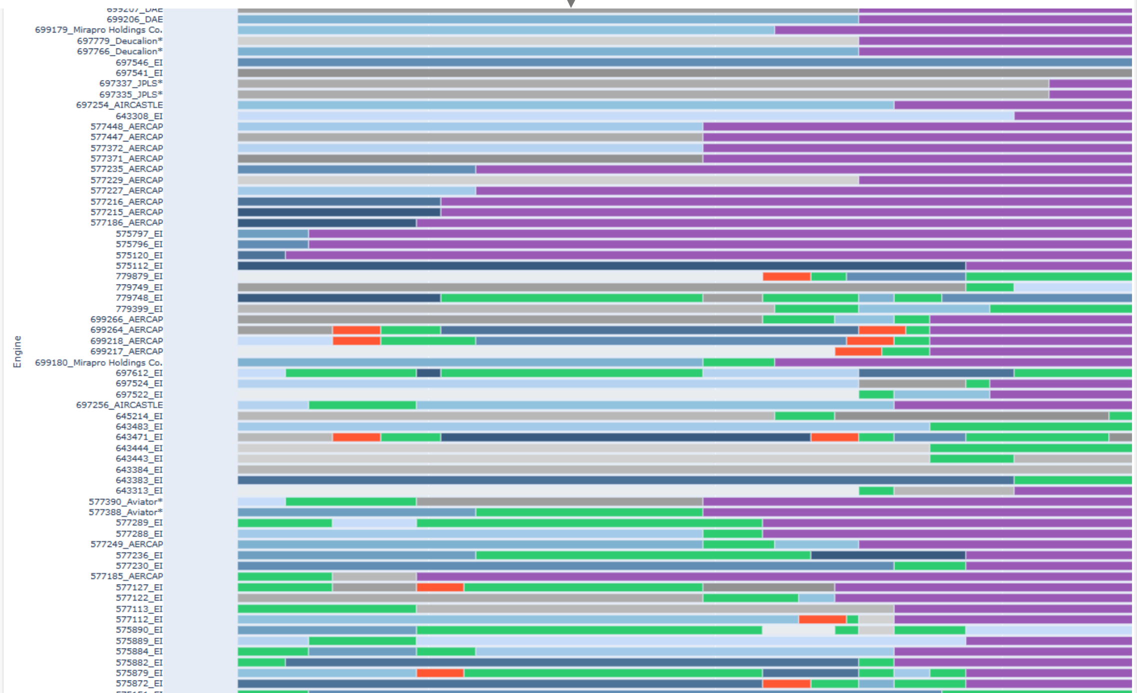
Application of new logics in the model

2. PRSV in the first month if required due to engine status

If an engine can't fly in the first month we force a PRSV

Forced to modify input to not have any PRSV in the first month

OLD



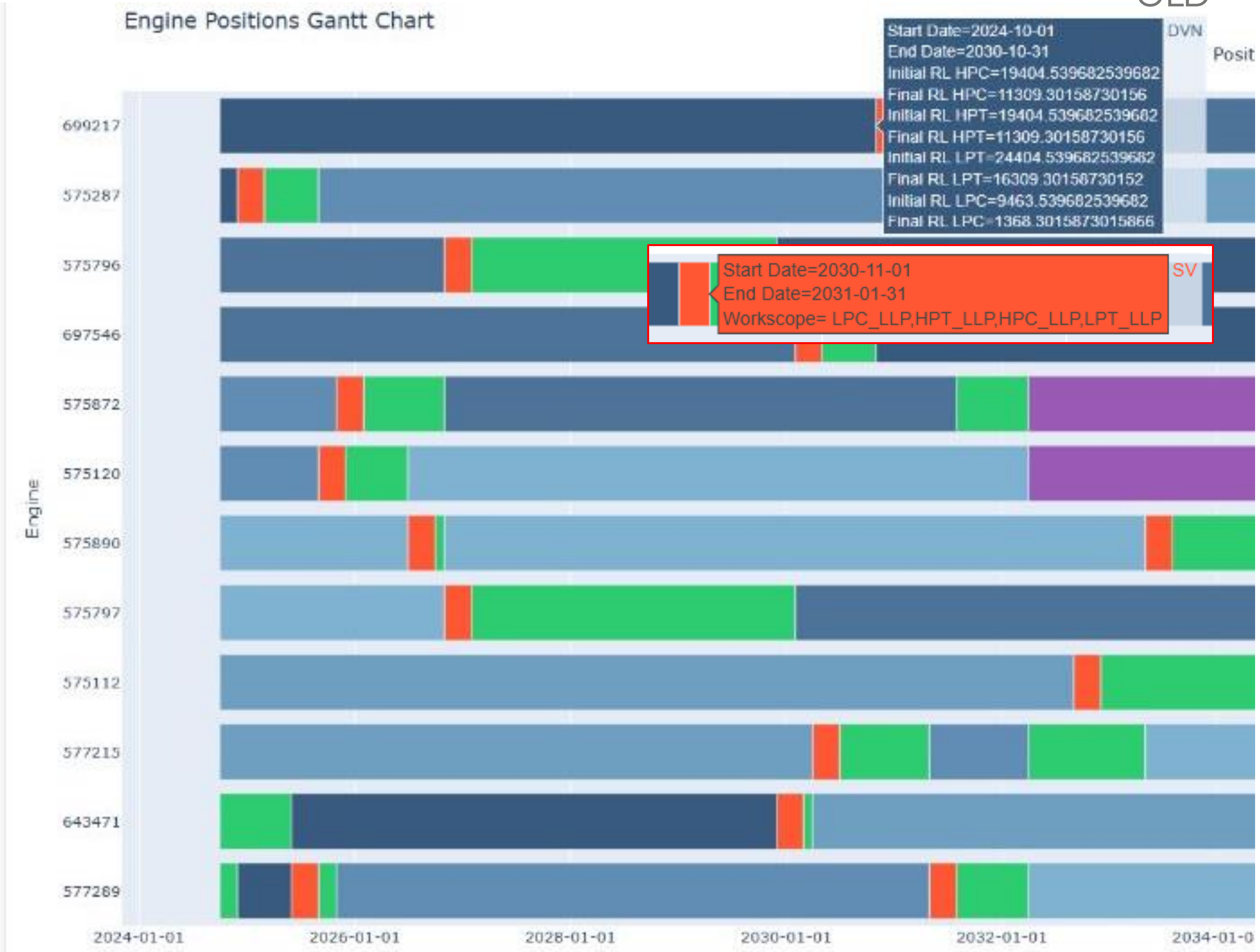
NEW

BETTER MODELLING

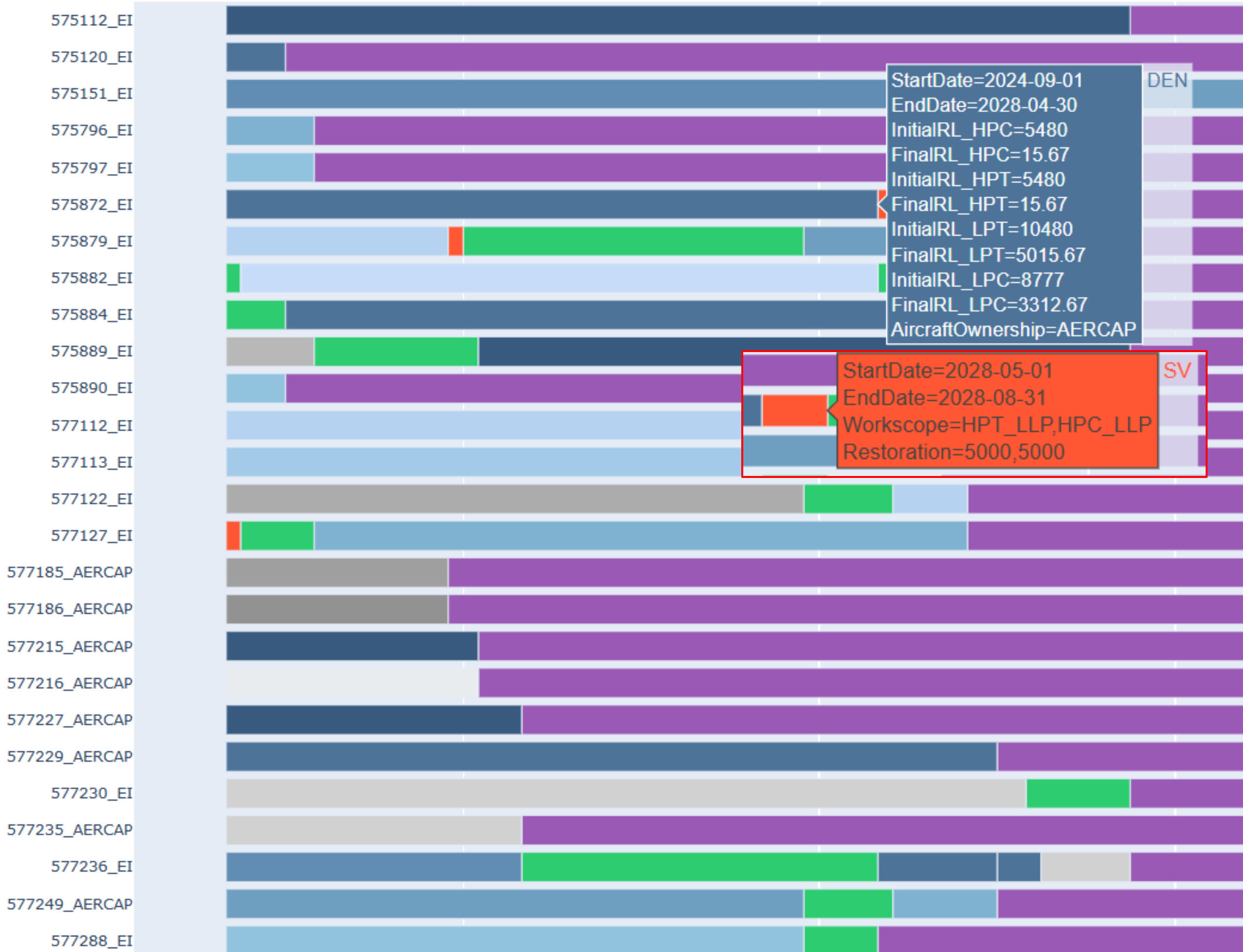
Application of new logics in the model

3. No PRSV if engine RFC > 500

OLD



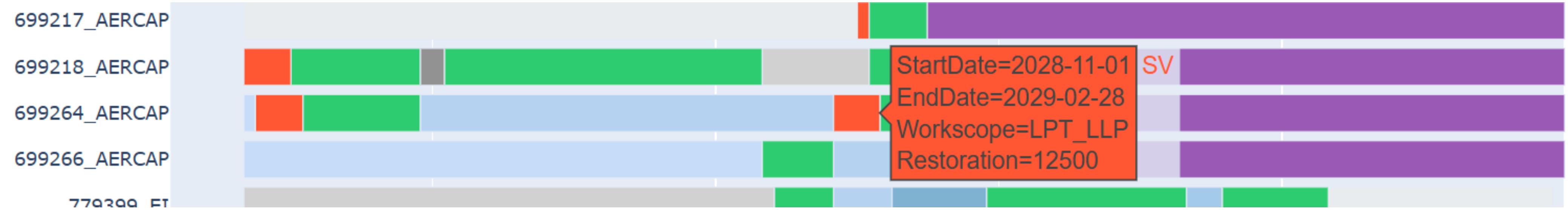
NEW



NEW OPTIONS IN PRSV PLANNING

Comparison of an execution with and without options

1. Use of second-hand LLPs (mid-life, 7000 RFC and 5000 RFC)



We assume that we have an unlimited pool of second hand LLPs

2. Module swap if workscope is only LPC

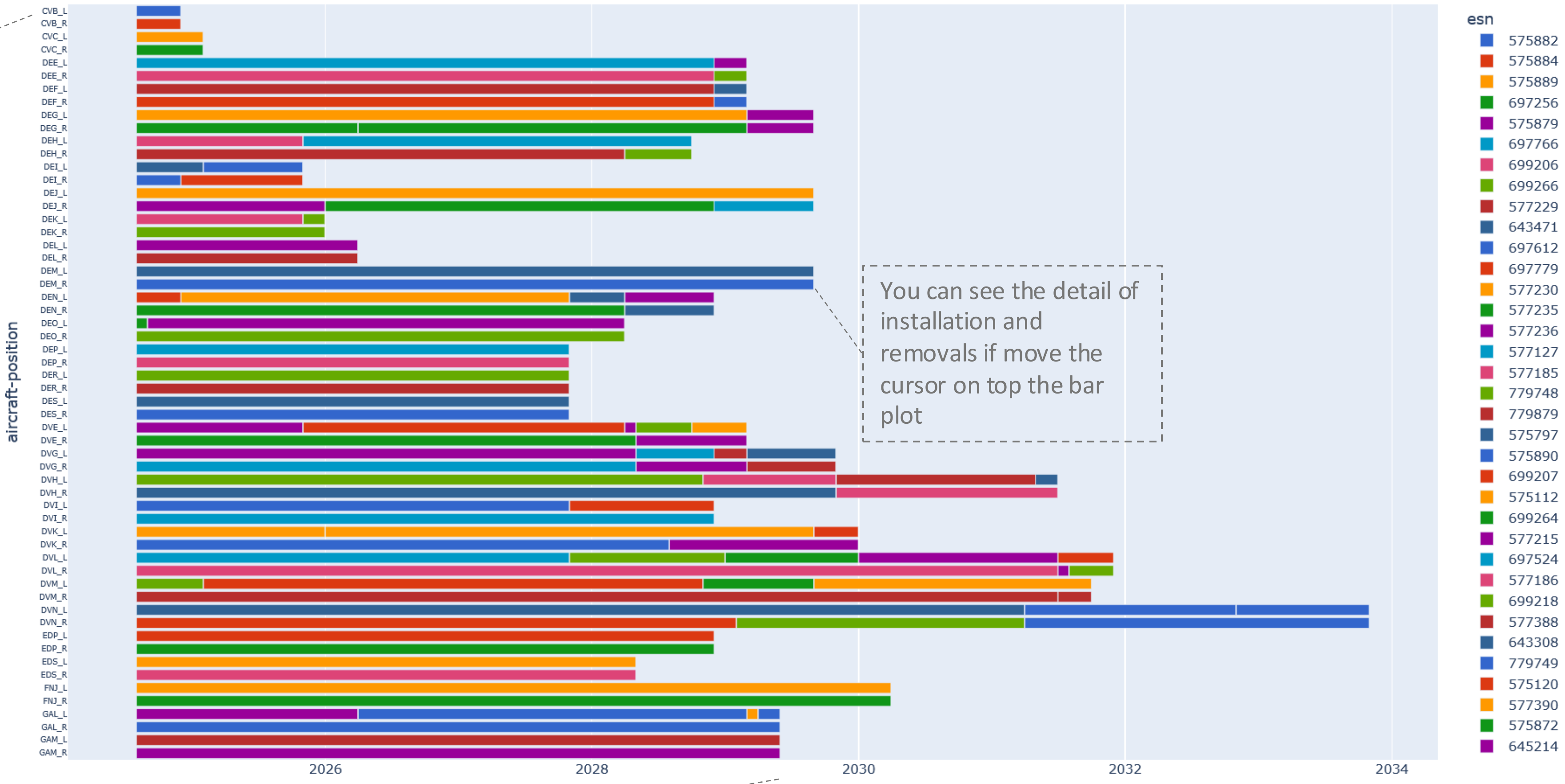
We assume that we have an unlimited pool of second hand LLPs, & the duration of the PRSV changes to 1 month

AIRCRAFT VIEW OF THE OPTIMIZATION

A new way to analyse better the results

CVB_L
CVB_R

Represantation of position of engine installed in aircraft



You can see the detail of installation and removals if move the cursor on top the bar plot

You can filter with double click on Esn number

The orizzontal axes is a representation of time

INTERNAL USE ONLY

TEST OF MULTI-OpCo OPTIMIZATION

GOAL: Using the current available data from IB that was used for the lineal model and fake data for the missing information from IB (spare engines and current position) and for VY CFM56 (EI data), launch an optimization to find out the following:

- We can launch a common optimization for two OpCos
- Which data we miss in the output to be able to show the maintenance plan, redelivery plan and costs (CF and P&L) per OpCo

EI Fleet: 27 aircrafts
61 engines

EI + IB Fleet: 71 aircrafts
151engines

object	stable_version	num_epochs	pop_size	Final epoch	Execution	Total Cost	Maintenance Cost	SV scheduled
Only_EI_27_aircraft_61_engines	04-03-2025	10	10	10	91.68	398,912,369	230,191,762	43
EI_IB_71_aircrafts_151_engines	04-03-2025	10	10	10	143.99	1,430,435,267	499,764,635	70
Only_EI_27_aircraft_61_engines	04-03-2025	50	50	50	1,576.21	349,511,323	184,781,830	39
EI_IB_71_aircrafts_151_engines	04-03-2025	50	50	50	3,364.87	1,393,382,202	436,821,259	68

IAG.ai

THANK YOU

MRO.ai