deimos
elecnor group

7th International Conference on Astrodynamics Tools and Techniques

A STOCHASTIC CONSTELLATION REPLENISHMENT PLANNER

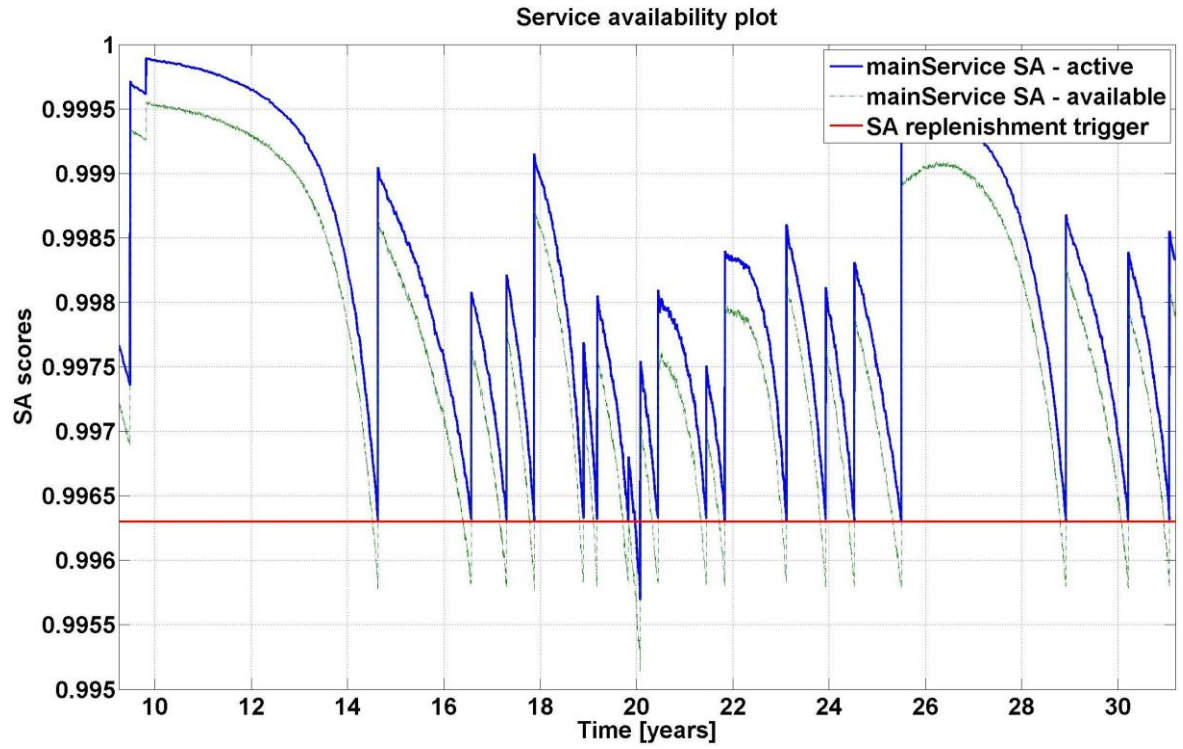
6-9 November 2018
DLR Oberpfaffenhofen, Germany

@ElecnorDeimos

AGENDA

A STOCHASTIC CONSTELLATION REPLENISHMENT PLANNER

- Introduction
- Problem Description
- Tool Implementation
- Use Cases
- Conclusions



deimos
elecnor group

INTRODUCTION

- Long-term planning for a constellation must take into account the need to replace individual satellites

- The Replenishment Planner evaluates the **replenishment needs of a constellation** based on **the quality of the service** provided
- It is a **highly flexible simulator** able to
  - Provide **statistical information** about the **replenishment needs** of a constellation
  - Assess the **goodness** of different **launch scenarios** and **replenishment strategies**
  - Measure the quality of the service (**service availability**)
  - Generate a **replenishment plan**
  - Evaluate the **sensitivity of the service availability** and the **replenishment plan** to multiple parameters
    - Launcher reliability, service outages, etc.

PROBLEM DESCRIPTION – SERVICE AVAILABILITY

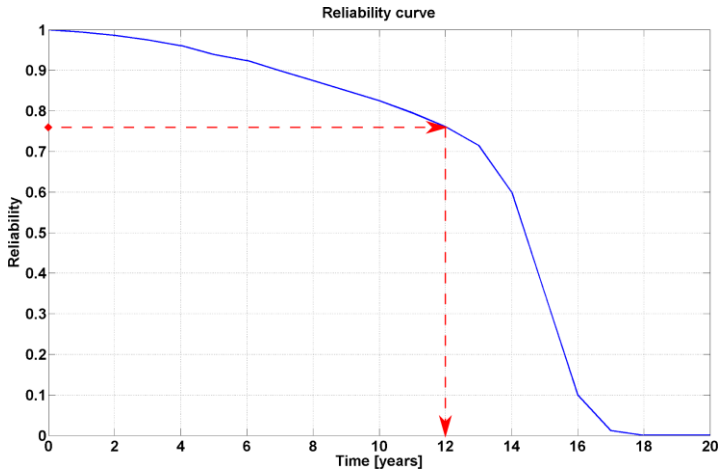
- The simulator **provides a replenishment plan that maintains the service availability** above a user-defined threshold

- **Service Availability** (SA) of a constellation
  - A_n: fraction of time when service is available in the target areas
    - Depends on constellation geometry and S/C available or providing service (n)
  - P_n: probability of having n spacecrafts available
- **Reliability curve** defines the S/C probability to provide a service at a certain age
- P_n are computed with a stochastic approach
  - Simulate the evolution of a given number of constellations
  - Count the constellations with "n" S/C providing service
  - Analytical computation with binomial distribution discarded
    - very time consuming, especially for large constellations

$$SA = \sum_{n=1}^{N} A_n \cdot P_n$$

PROBLEM DESCRIPTION – CONSTELLATION EVOLUTION

- Simulates Walker constellations: number of planes, nominal and spare slots
  - S/C in nominal slots provide one or multiple services
  - Spare S/C for backup → **relocation** to nominal slots when a nominal S/C dies
- Service model
  - Defined with one/multiple **reliability curves**
    - Death date of the service is a random variable
    - Dependent services are supported
- Other S/C features
  - Decommissioning on-demand at given age defined as N(μ,σ)
  - Different types of outages (temporary loss of service)
    - Scheduled outages: at known date
    - Unscheduled outages: defined with failure rate



PROBLEM DESCRIPTION – REPLENISHMENT STRATEGY

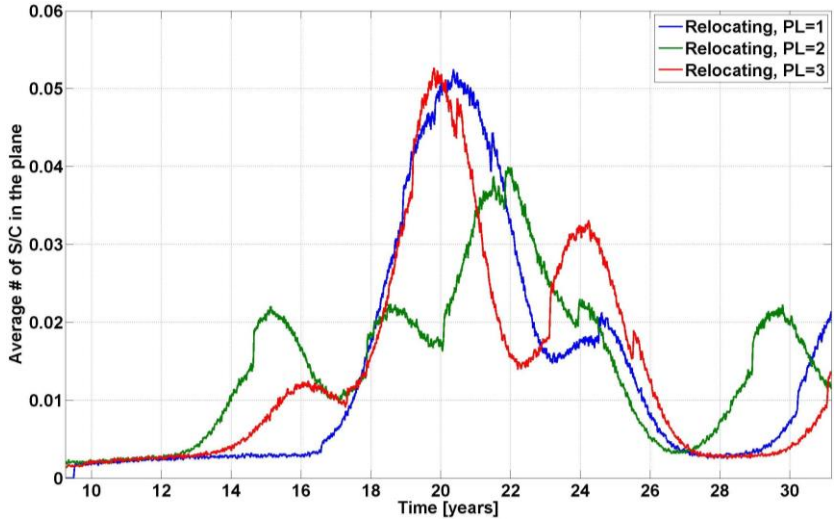
- Launches are scheduled when a certain FoM violates a user-given condition
- **Corrective strategy (C)**
  - Launch requested when a constellation of the statistical sample loses its spare satellites in any of its planes
  - Delay between launch request and launch date
- **Preventive strategy (P)**
  - Launch scheduled (in advance) according to SA violation of a given threshold
  - The target plane is the one with lower spare capability (i.e. probability of having more than the nominal number of S/C)
- **Preventive-corrective strategy (PC)**
  - To guarantee the spare capability on each plane
  - Launch when spare capability violates a given threshold
- **Combined strategies (P & PC, P & C)**

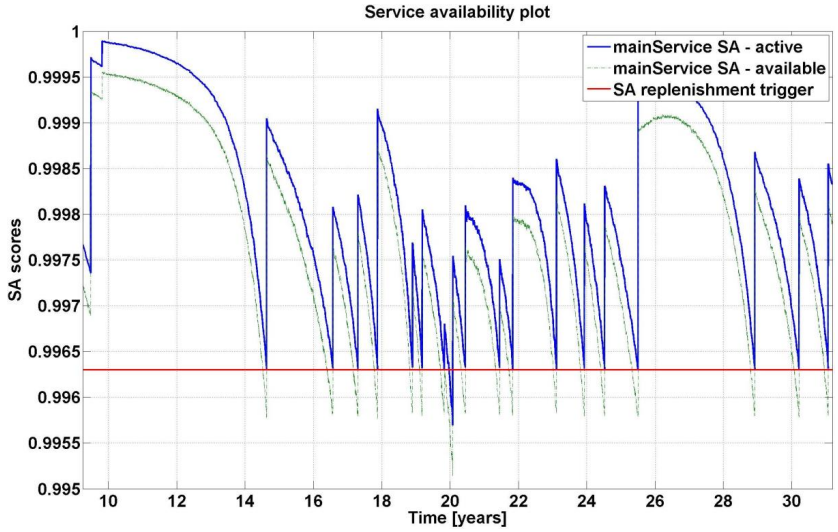
TOOL IMPLEMENTATION

- Developed in C++ for a **Linux OS**
- Running and tested in **Windows**
- **Integrated in openSF** (ESA generic simulation framework) providing capabilities to perform Monte Carlo, batch and parametric analysis as well as input edition and managing external tools for output post-processing (e.g. plot generation)
- **Parametric and Monte Carlo analysis**
  - Sensitivity analysis of SA and replenishment plan
  - Robustness analysis of a reference replenishment plan
- **Performances**
  - Variable time-step to improve runtime performances
  - openSF distributes the computational processes across cores → computational time drastically reduced
- **Highly customizable post-processing module** compatible with Octave 4.2 and Matlab 2013b to load and plot the simulation results

PREVENTIVE STRATEGY

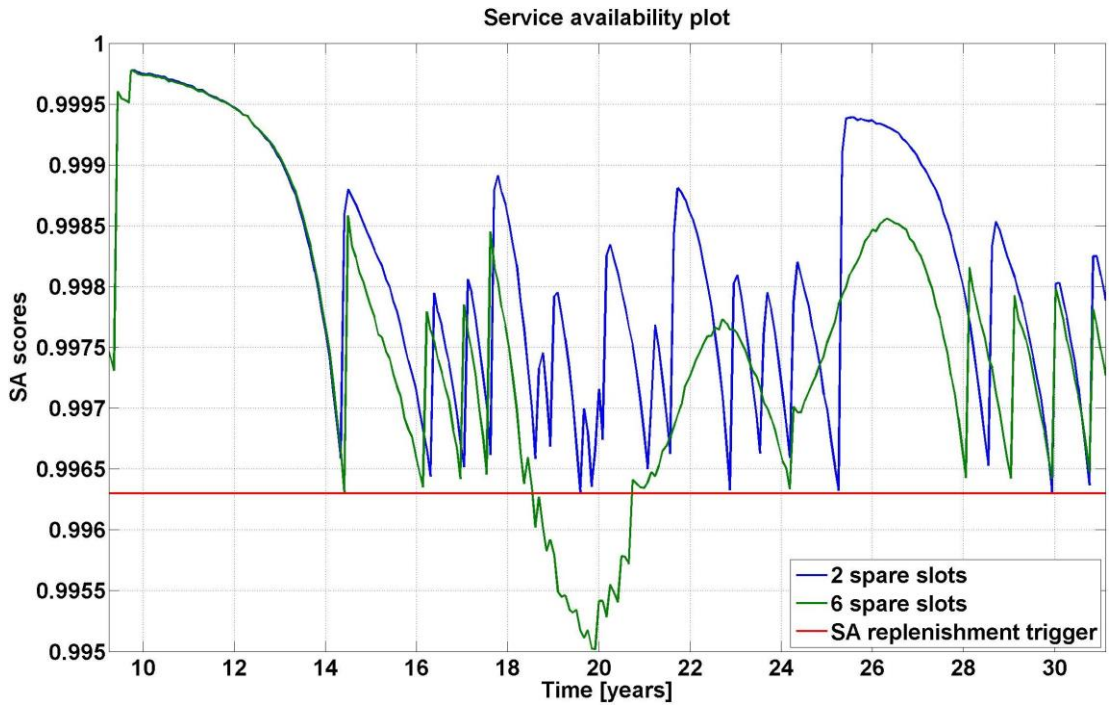
- Launch scheduled (in advance) according to SA violation of a given threshold
- The target plane is the one with lowest spare capability
  - Lowest probability of having more than the nominal number of S/C
- Example case of constellation: 3 planes, 8 nominal + 6 spare slots per plane
  - Unscheduled outage of 1 day with constant rate of 0.001 failures per day
  - One violation due to the 90 days minimum time between launches constraint
  - High death rate → high number of relocating S/C





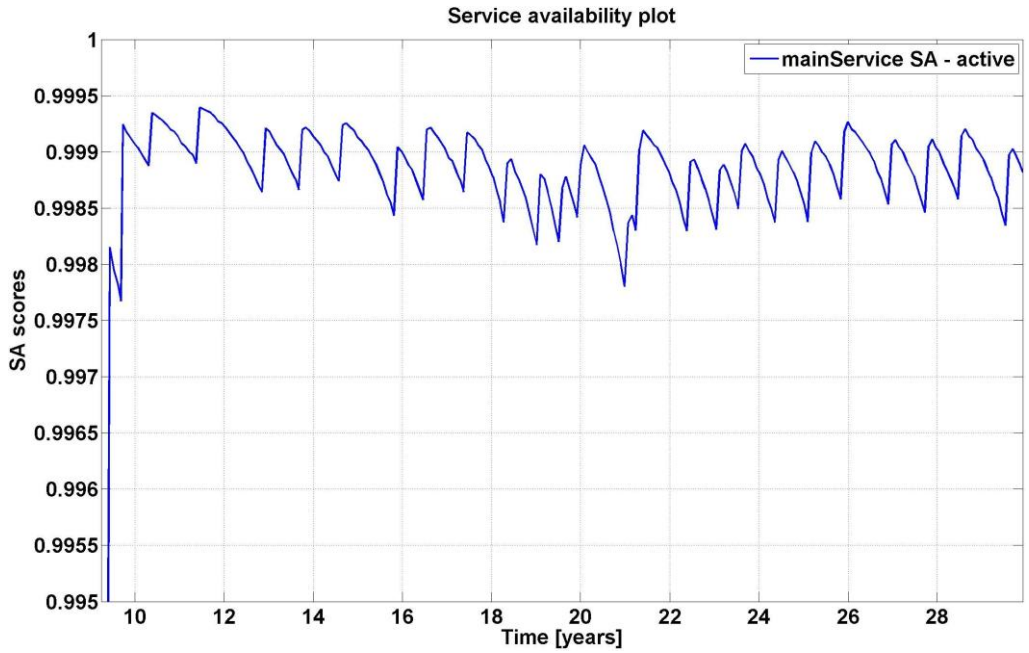
PREVENTIVE STRATEGY: SPARE BUFFER EFFECT

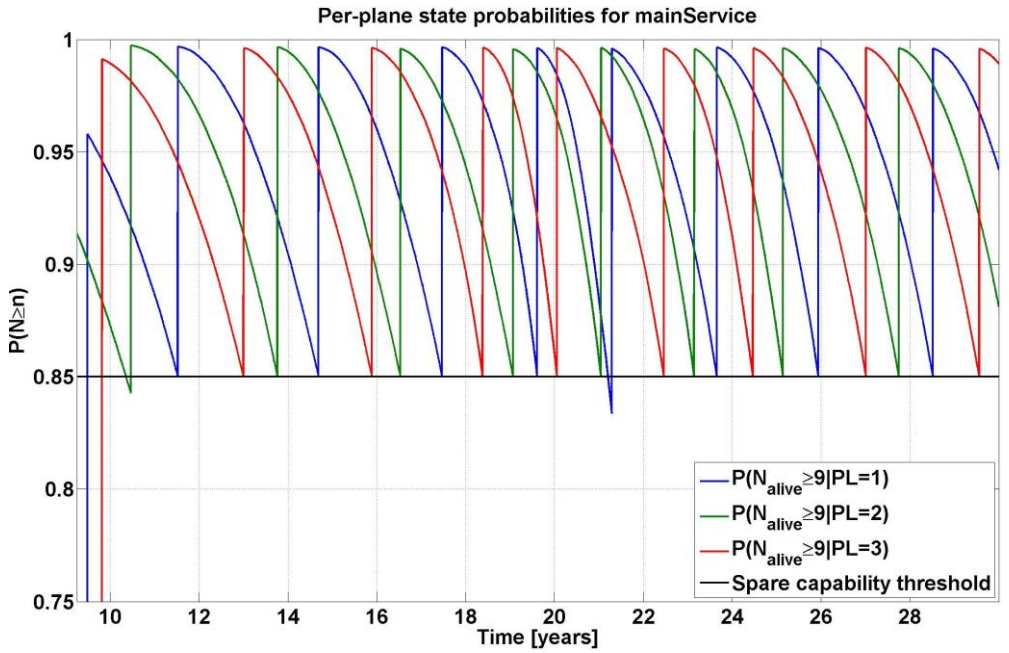
- With high number of spare S/C (6) and longer relocation time (25 days), during high death rate phases SA is recovered by relocation (not by arrival of new S/C)
- If spare S/C are available → relocations are triggered as soon as a nominal S/C dies → arriving S/C head to spare slots, not to the nominal ones
- Response to the arrival of new satellites is muted
- The simulation with 6 spares needs one more launch and has a violation of ~3 years



PREVENTIVE CORRECTIVE STRATEGY

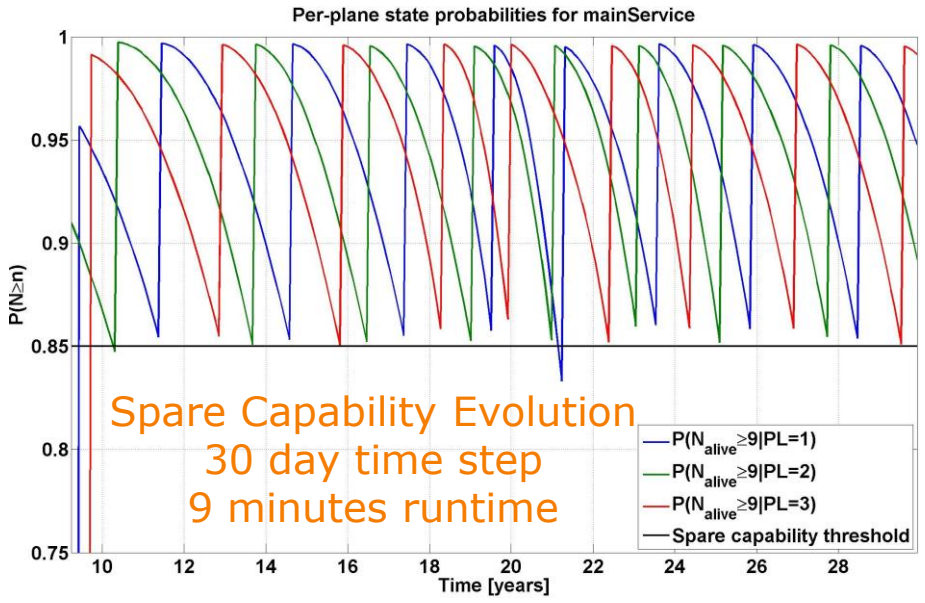
- Launch when spare capability violates a given threshold → aims to guarantee the spare capability on each plane
- Example case of a constellation: 3 planes, 8 nominal slots + 2 spares per plane
  - Spare capability threshold of 85%
  - 150,000 constellations simulated for 30 years

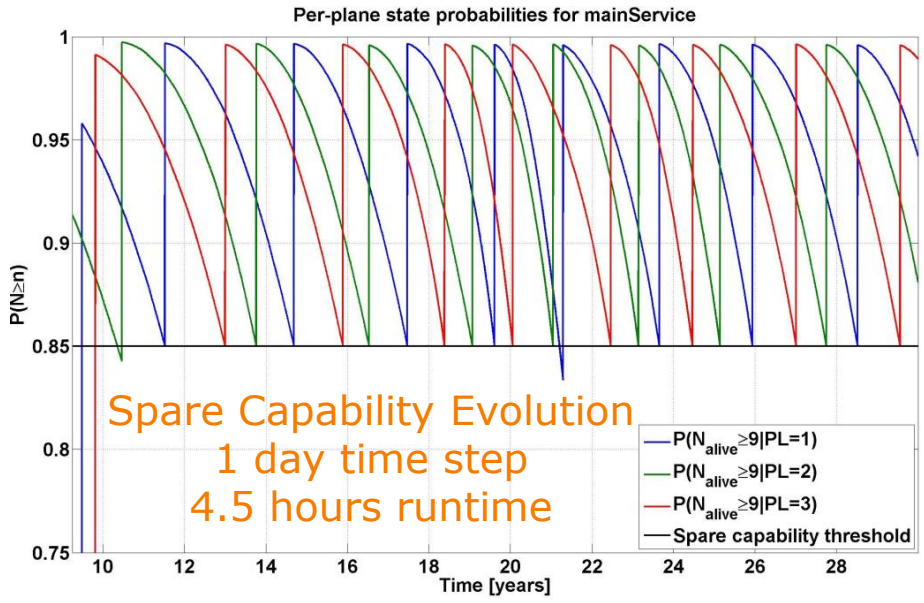




PREVENTIVE CORRECTIVE STRATEGY

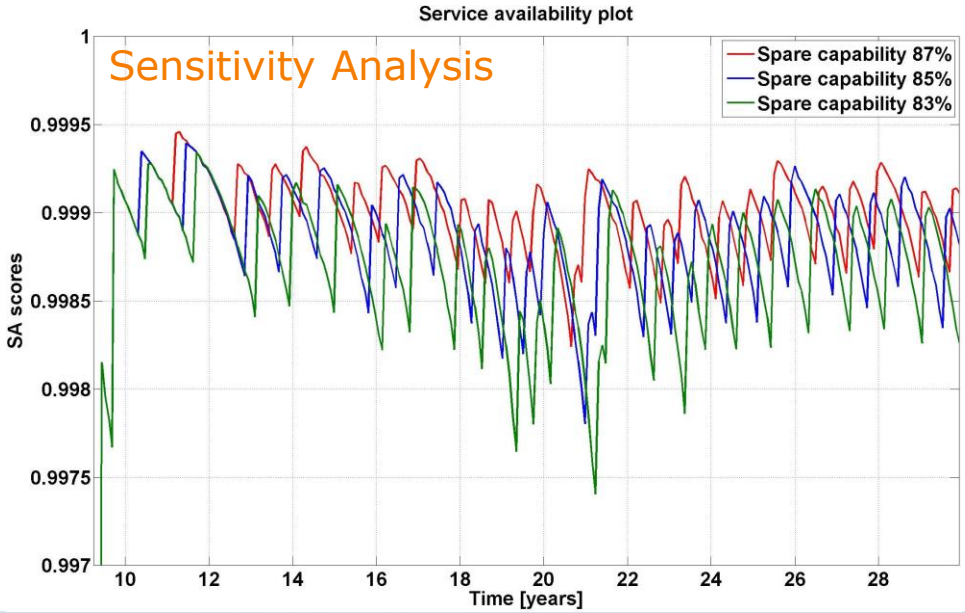
- Launch when spare capability violates a given threshold → aims to guarantee the spare capability on each plane
- Example case of a constellation: 3 planes, 8 nominal slots + 2 spares per plane
  - Spare capability threshold of 85%
  - 150,000 constellations simulated for 30 years
  - 1 day time step vs 30 days time step (x30 speed-up factor)

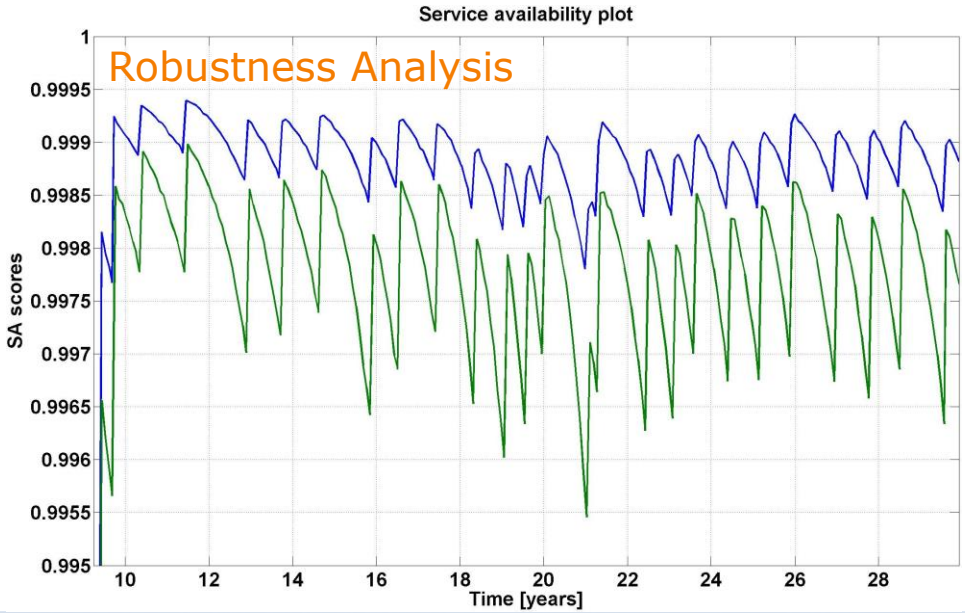




SENSITIVITY AND ROBUSTNESS ANALYSIS

- **Perturbation of nominal scenario**
  - The tool can be used to perform sensitivity analysis over different parameters
    - Replenishment with different levels of space capacity threshold
  - Robustness analysis using a reference replenishment plan
    - SA evolution with different reliability curves and the same plan
  - openSF distributes the analyses within the available cores



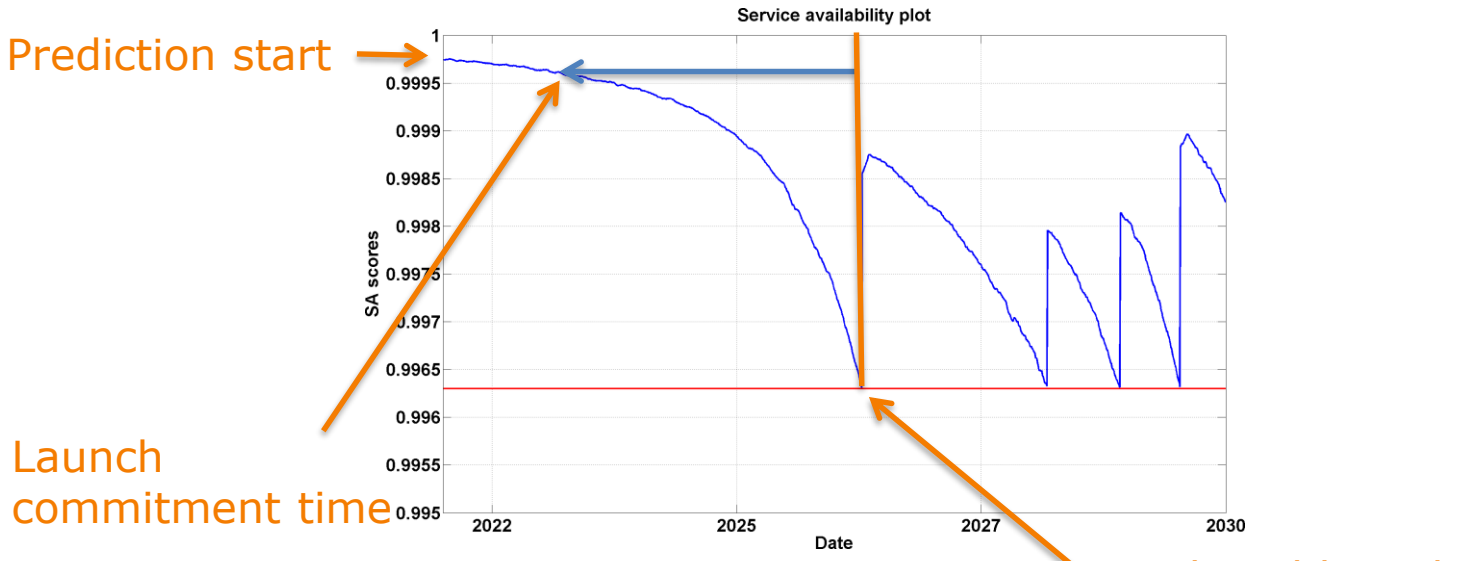


PREDICTOR-CORRECTOR REPLENISHMENT PLANNER

- Analyze the **sensitivity** and **robustness** of a replenishment plan against real scheduling constraints
  - Launches are announced with enough anticipation: **prediction**
  - Announced date is not the final one: flexibility to delay and advance the launch
  - A certain time before the announced date, the launch is fixed and committed: **correction**
- Uses a Monte Carlo approach
  - Each shot simulates a single constellation
  - Each shot runs the Replenishment Planner multiple times
    - Input: current **constellation state**, that evolves during the simulation
    - Constellation state: Nº spare, relocating and nominal S/Cs per plane
  - **Replenishment** performed with a **Predictor-Corrector sequence**

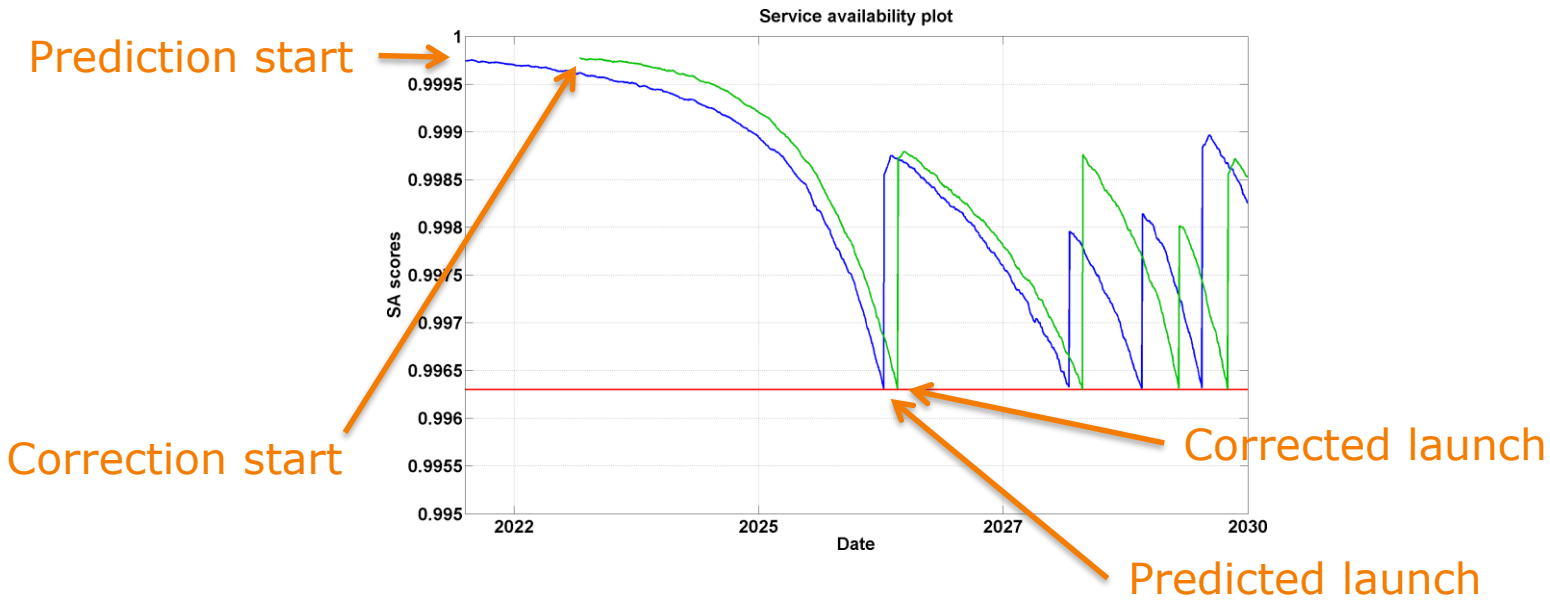
PREDICTOR-CORRECTOR REPLENISHMENT PLANNER

- Predictor-Corrector sequence
  - **Predict** the launches that will be needed
  - Evolve the constellation until launch commitment time
  - **Correct** launch date using the current state of the constellation
  - Evolve the constellation and **predict again**



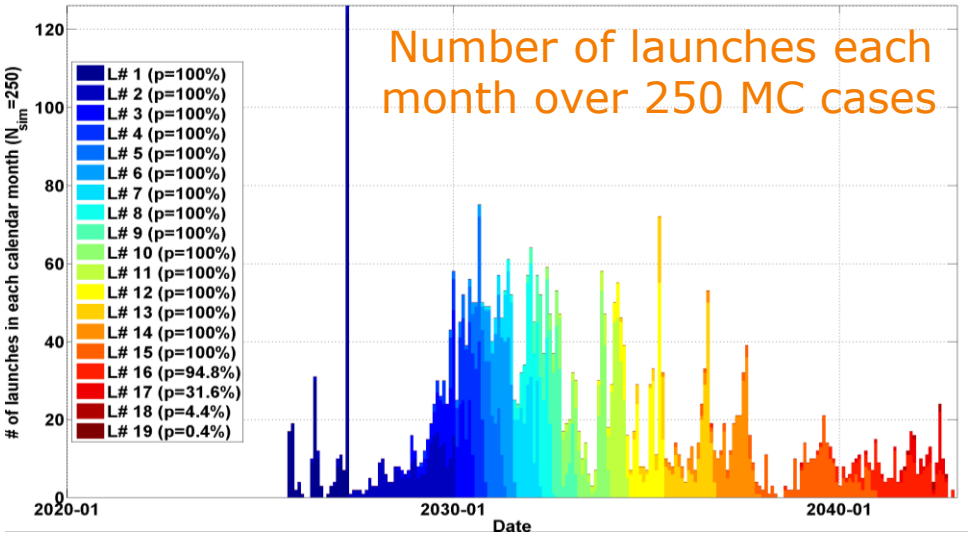
PREDICTOR-CORRECTOR REPLENISHMENT PLANNER

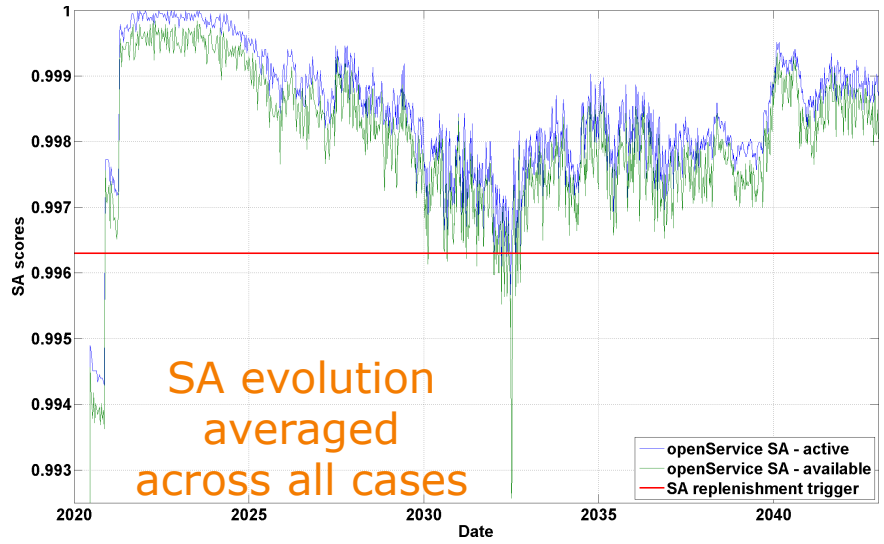
- Predictor-Corrector sequence
  - **Predict** the launches that will be needed
  - Evolve the constellation until launch commitment time
  - **Correct** launch date using the current state of the constellation
  - Evolve the constellation and **predict again**



PREDICTOR-CORRECTOR REPLENISHMENT PLANNER

- Outcome of each shot
  - Evolution of a single constellation
  - A replenishment plan considering real scheduling constraints
- Statistical information obtained from post-processing the shots
  - Sensitivity mode: variability of replenishment plan
  - Robustness mode: service availability evolution
    - Number of committed launches is restricted





FUTURE DEVELOPMENTS

- Mostly focused on **improving the fidelity** of the models used to compute the failure time of the spacecrafts

- **Services split-down in components**
  - Each component with its own reliability curve and outages definition
  - Services reliability implemented as reliability chain of components

- New **predictive strategy**, based on **estimation of expected lifetime** of the S/Cs
  - Trigger launches independently to each simulated constellation whenever the **expected number of S/Cs** is not enough to guarantee the service level
  - Computation of the expected lifetime might vary from very simple implementations to other more complex
    - Based on the monitoring of the status of the redundant critical components
  - **Combination** with **corrective** strategy will mitigate errors in prediction

CONCLUSIONS

- The Replenishment Planner is a **highly flexible** simulator able to provide **valuable statistical information** about the **replenishment needs** of a constellation and the **service quality**
- It can model a wide **variety of constellations** in terms of **geometry, spacecraft and services models**, operational **launch characteristics** and **constraints**
- It can analyze different **launch scenarios** and perform **sensitivity and robustness analyses** on multiple parameters to assess its **impact on the replenishment plan and SA**
- The **Predictor-Corrector Monte Carlo** analyses the **robustness** and **sensitivity** of a plan to **real scheduling constraints**
- Takes into account operational constraints
  - Performs **complex and reliable analyses** quickly (*minutes*)
  - More accurate results in reasonable computation time (*hours*)

deimos
elecnor group

THANK YOU

federico.letterio@deimos-space.com