GTOC9: Results from the University of Colorado at Boulder

Nathan L. Parrish, Daniel J. Scheeres, Simon Tardivel, Chandrakanth Venigalla, Jonathan Aziz, Marielle Pellegrino, Oscar Fuentes, Stijn De Smet

Problem Statement

- Deorbit 123 pieces of debris in ~sun-synchronous orbit at ~7000 km altitude for the minimum cost
  - J2 dynamics
  - Impulsive maneuvers
  - Time constraints
  - Mass constraints
- Cost function:

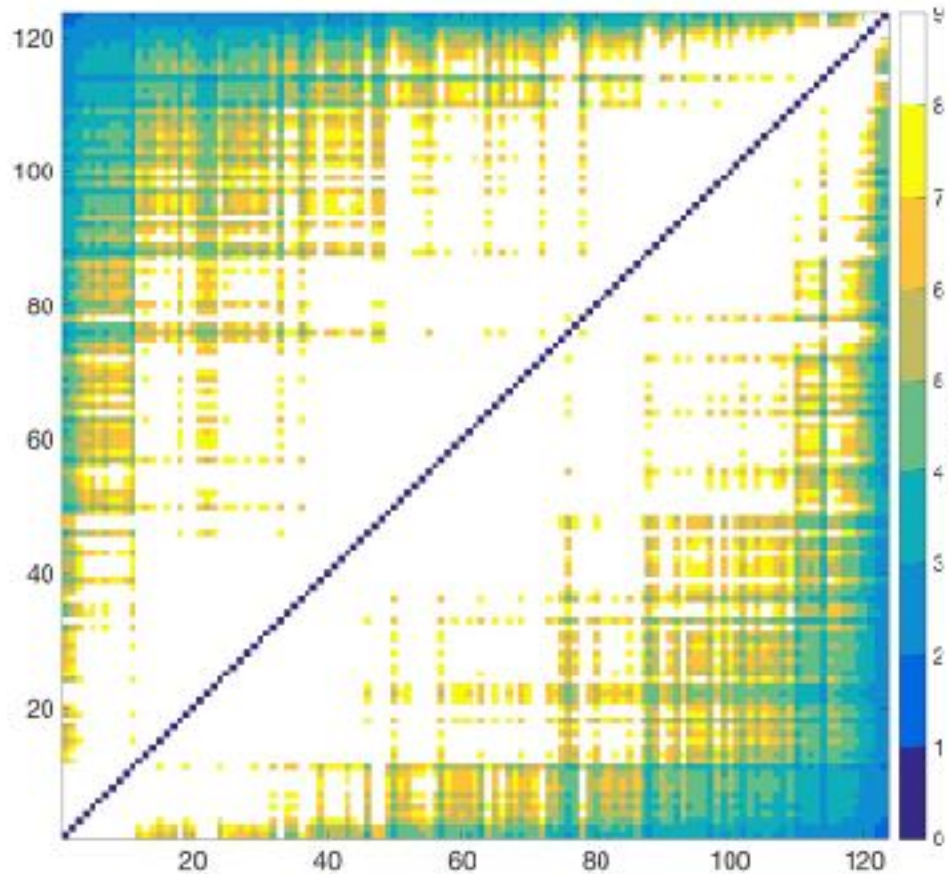
$$J = \sum_{i=1}^{n} C_i = \sum_{i=1}^{n} \left[c_i + \alpha (m_{0_i} - m_{dry})^2 \right]$$

Understanding the problem

- How often does each debris cross the others' RAAN?
- Fastest: every ~1 year
- Most: well beyond the 8 year time limit

- Most debris-to-debris transfers only exist (cheaply) once or twice in the entire competition window
- Debris transfer options are largely driven by RAAN drift

Synodic period of RAAN phasing (years)

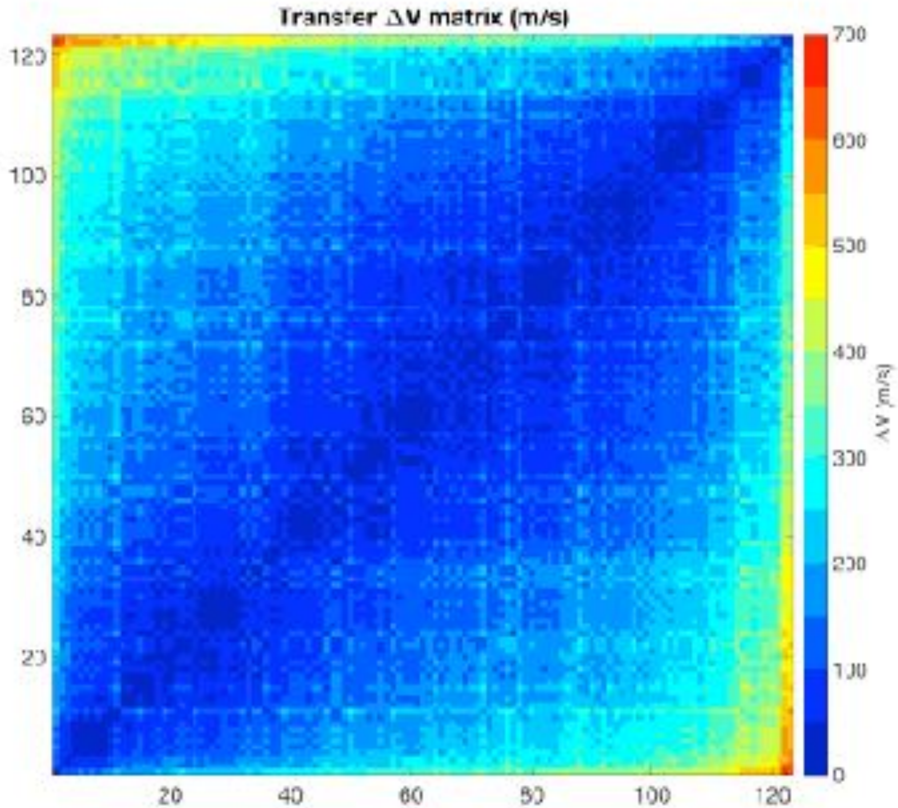


Debris ID # (sorted by inclination)

Debris ID # (sorted by inclination)

Understanding the problem

ΔV transfer cost, assuming fixed Ω



Debris ID # (sorted by inclination)

Debris ID # (sorted by inclination)

- With fixed RAAN, problem becomes standard TSP (Traveling Salesman Problem)
- Optimal ΔV is ~3500 m/s
- Need absolute minimum of 2 launches to achieve
- To fly this solution would take millions of years to wait for RAAN phasing

Solution outline

1. Analytical ΔV estimate
2. "Building Blocks": find all the low-ΔV chains of debris, and pick a set of chains that visit most of the debris
3. "Stitcher": add a few debris (usually high-ΔV) on to the starting chains
4. "Finisher": add 2-3 more launches which each visit 2-3 debris at very high cost (but cheaper than single launches)
5. "Wiggler": slightly adjust the dates of each launch to reduce the ΔV
6. Integrated Solution: shooting algorithm with random restart to optimize the numerically-integrated solution and satisfy final constraints

Analytical ΔV

- Given: transfer time, Ω_1, Ω_2, $\dot{\Omega}_1$ → $\Delta\dot{\Omega}$
- Assume:
  - Circular orbits
  - Small changes in i, a
- Change in node rate due to change in i:

$$\Delta_i\dot{\Omega} = \frac{\partial\dot{\Omega}}{\partial i}\Delta i = -\dot{\Omega}\tan i\,\Delta i$$

- Change in node rate due to change in a:

$$\Delta_a\dot{\Omega} = \frac{\partial\dot{\Omega}}{\partial a}\Delta a = -\frac{7}{2}\dot{\Omega}\frac{\Delta a}{a}$$

- Change in i as a function of ΔV:

$$\Delta i = \sqrt{a/\mu}\Delta V$$

- Change in a as a function of ΔV:

$$\Delta a = 2a\sqrt{a/\mu}\Delta V$$

Analytical ΔV

- ΔV required to change node rate, via changing i:

$$\Delta V_i = \Delta_i \dot{\Omega} \frac{2}{3 J_2 R^2} \frac{a^3}{\cos i}$$

- ΔV required to change node rate, via changing a:

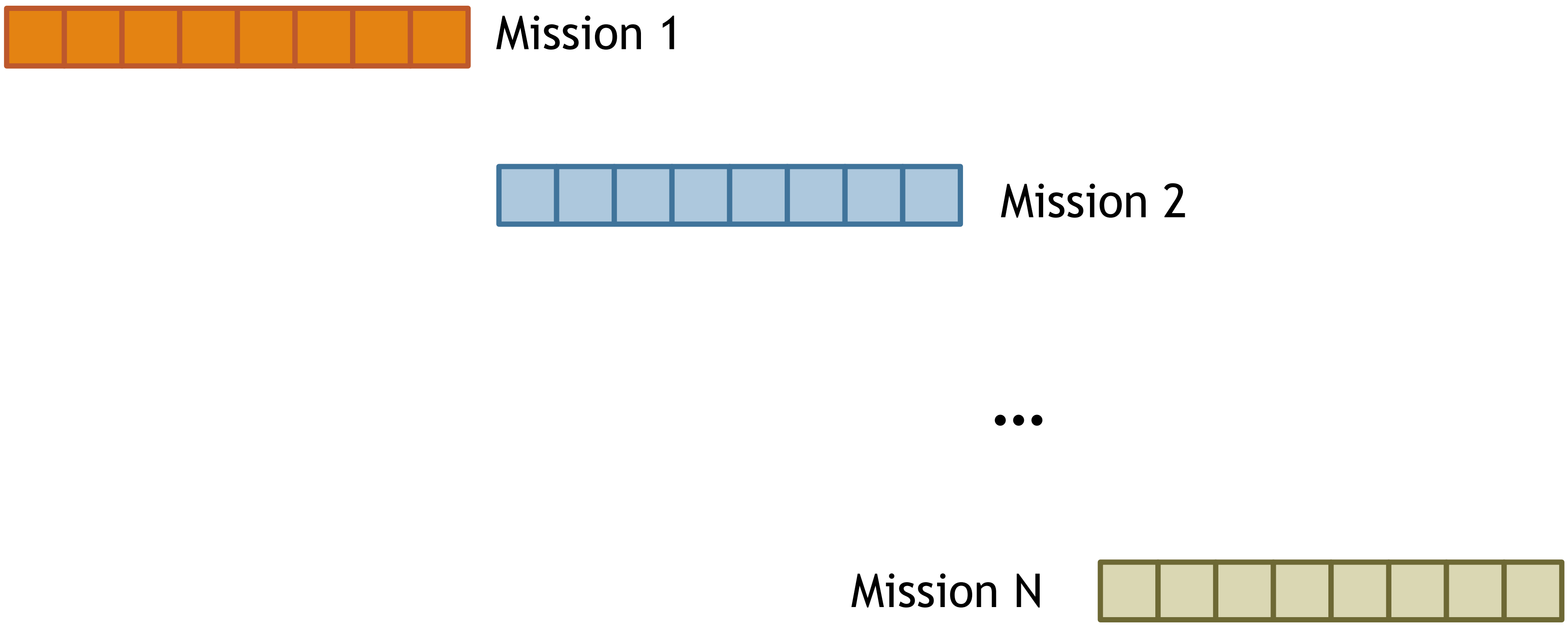
$$\Delta V_a = \Delta_a \dot{\Omega} \frac{2}{21 J_2 R^2} \frac{a^3}{\cos i}$$

- Comparing efficiency, found they are equally efficient at $\tan i = -7$, or $i = 98.13°$
  - Better to use a to change node rate when $i_1 > 98.13°$
  - Better to use i to change node rate when $i_1 < 98.13°$
  - Also must consider i_2, a_2
- Accurate enough when $\Delta V < \sim 300\ m/s$

Building Blocks

- Find all the chains of debris with average ΔV <200 m/s per transfer
  - Brute force search
  - Assume fixed 20 days transfer time
  - Find al chains which visit 10 debris each, 9 debris each, 8 debris each…
  - Yields ~100,000 possible chains
- Choose a set of chains that visit most of the debris
  - Sort the chains by the “common-ness” of the debris each visits
  - Prioritize the chains that remove the least common debris – which are the hardest to access
- Result from this stage: ~75-95 debris removed out of 123

Building Blocks



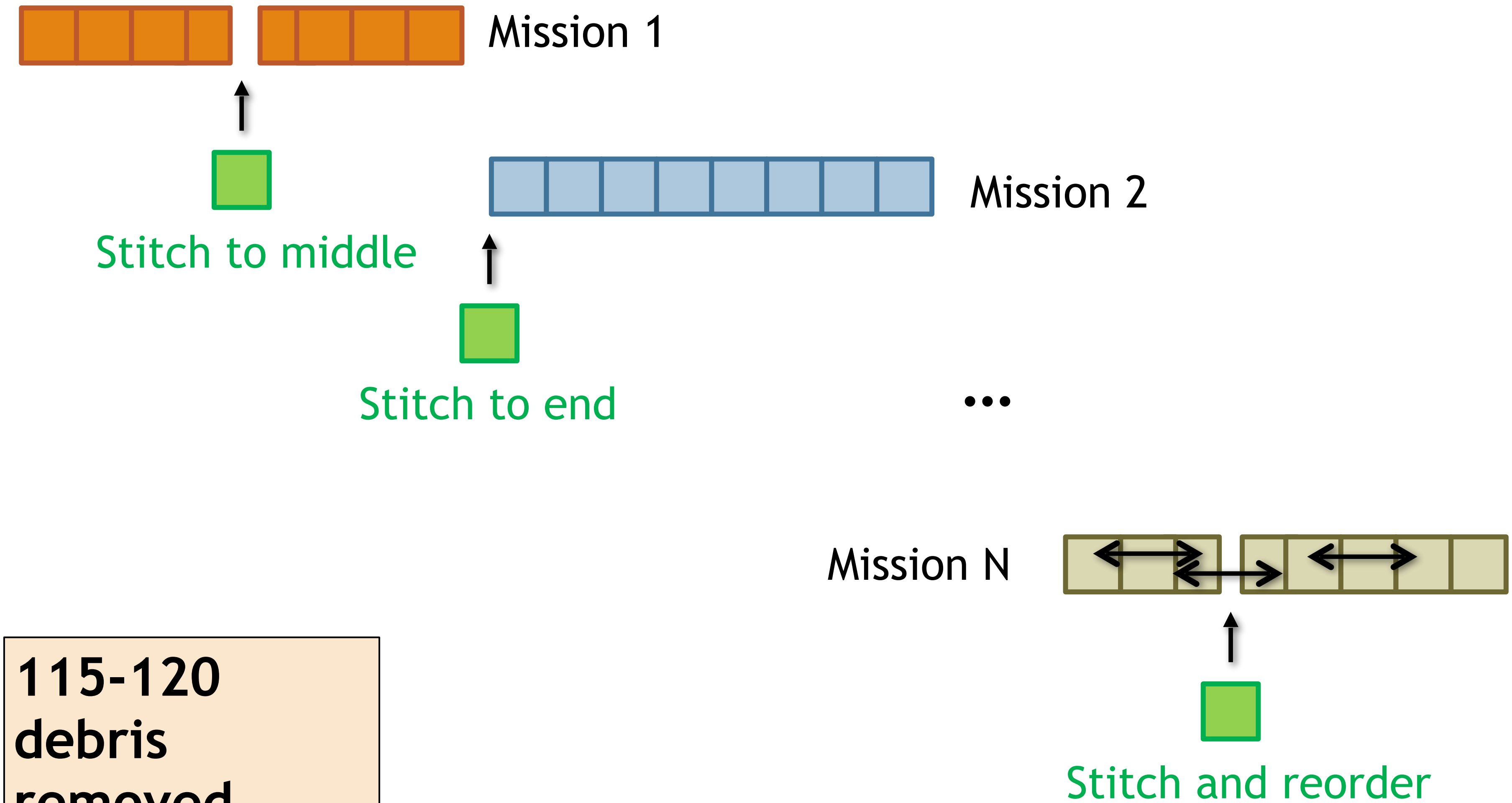
75-95 debris removed

Stitcher

- For the remaining debris, want to “stitch” them onto the existing chains
- Which debris to stitch where?
  - Have ~20 debris to stitch to ~12 existing chains
  - New debris rendezvous can be added anywhere in existing chains
  - Reduced problem from the original, but still too big to search with brute force
- Hard to know a priori which addition will be most productive
- Our solution: Greedy tree search with randomization, run many times in parallel

- Result from this stage: ~115-120 debris removed out of 123

Stitcher

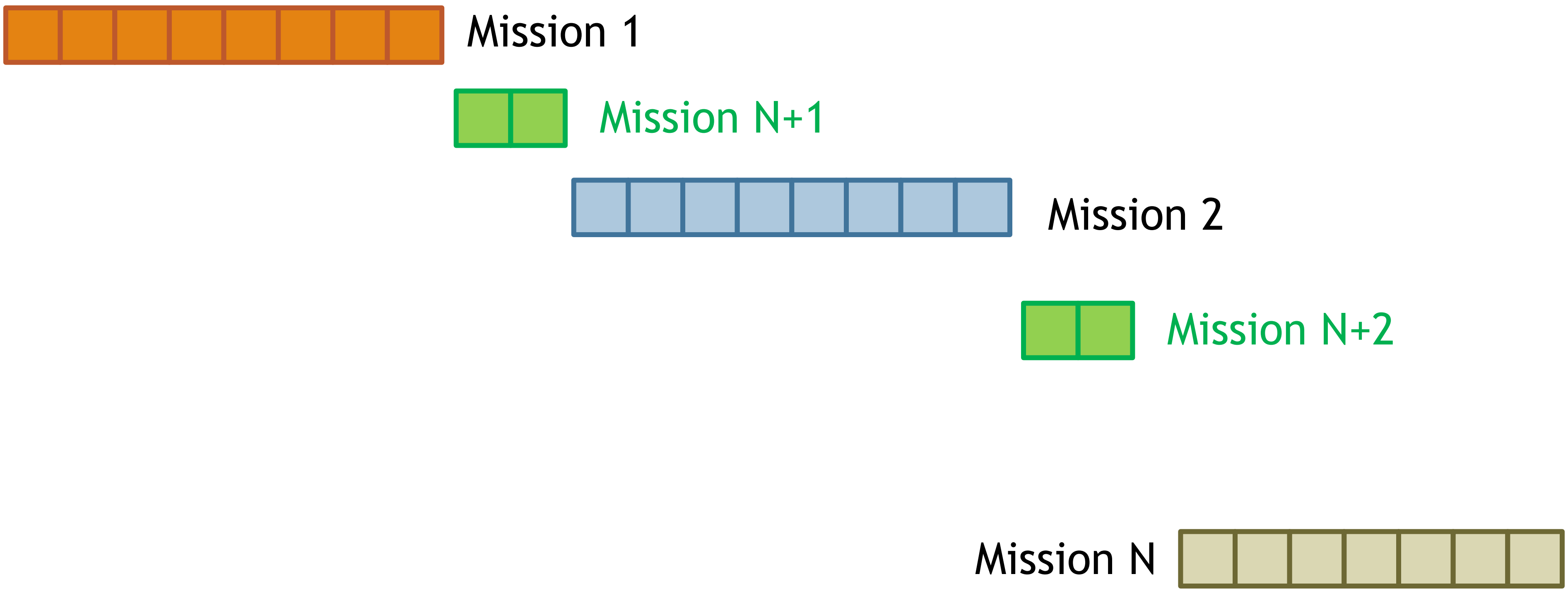


115-120 debris removed

Finisher

- After stitching, there are still some debris that could not be attached to any existing mission
- The remaining debris would cost 55.0018 MEU each with single launches
- Even a launch with high ΔV that visits just 2 debris is better than single launches
- The "finisher" adds new missions that each visit 2-3 debris
- Problem is small now, so a complete brute force search is effective

Finisher

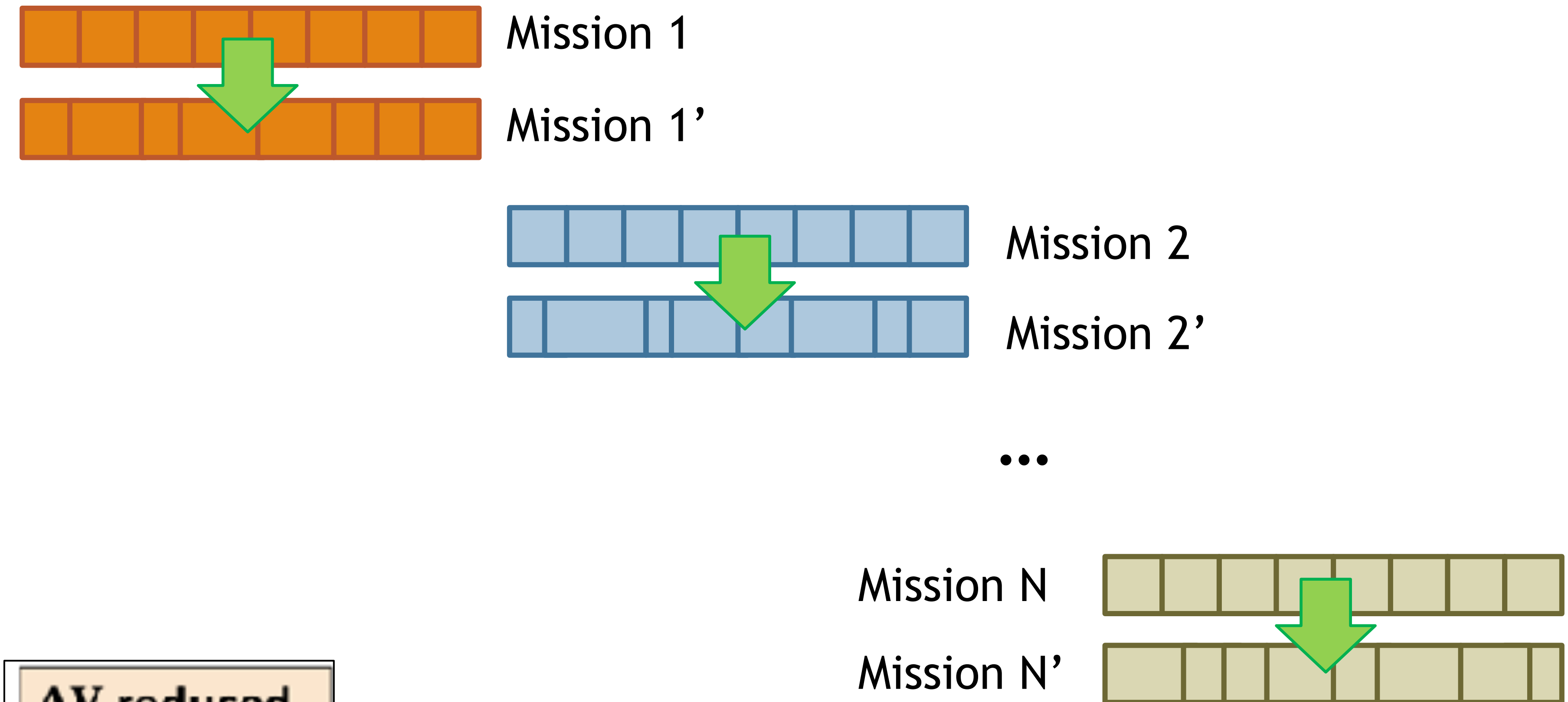


120-123 debris removed

Wiggler

- Within each mission, the dates of rendezvous with each debris were slightly adjusted
- Saved 5-10% ΔV
- Interior Point method used to minimize ΔV while enforcing time of flight constraints

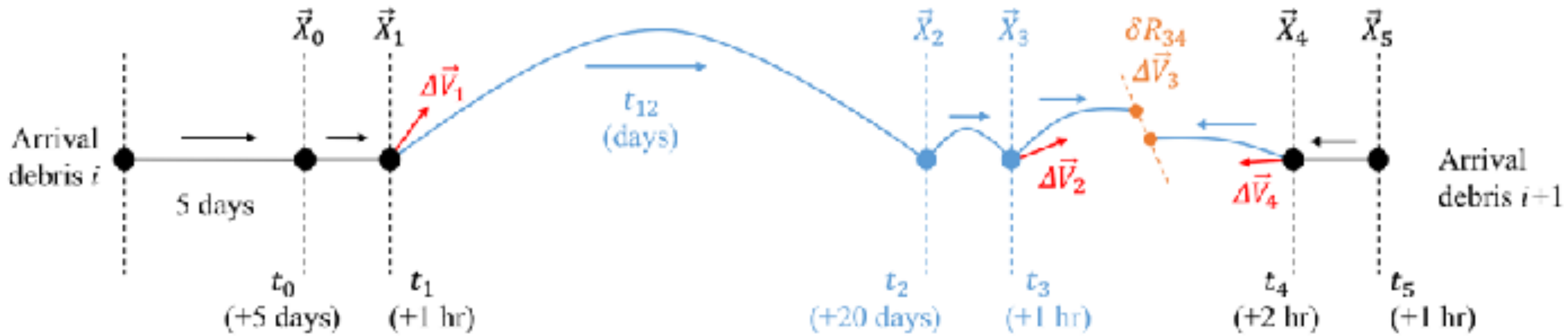
Wiggler



ΔV reduced 5-10%

Optimize Integrated Solution

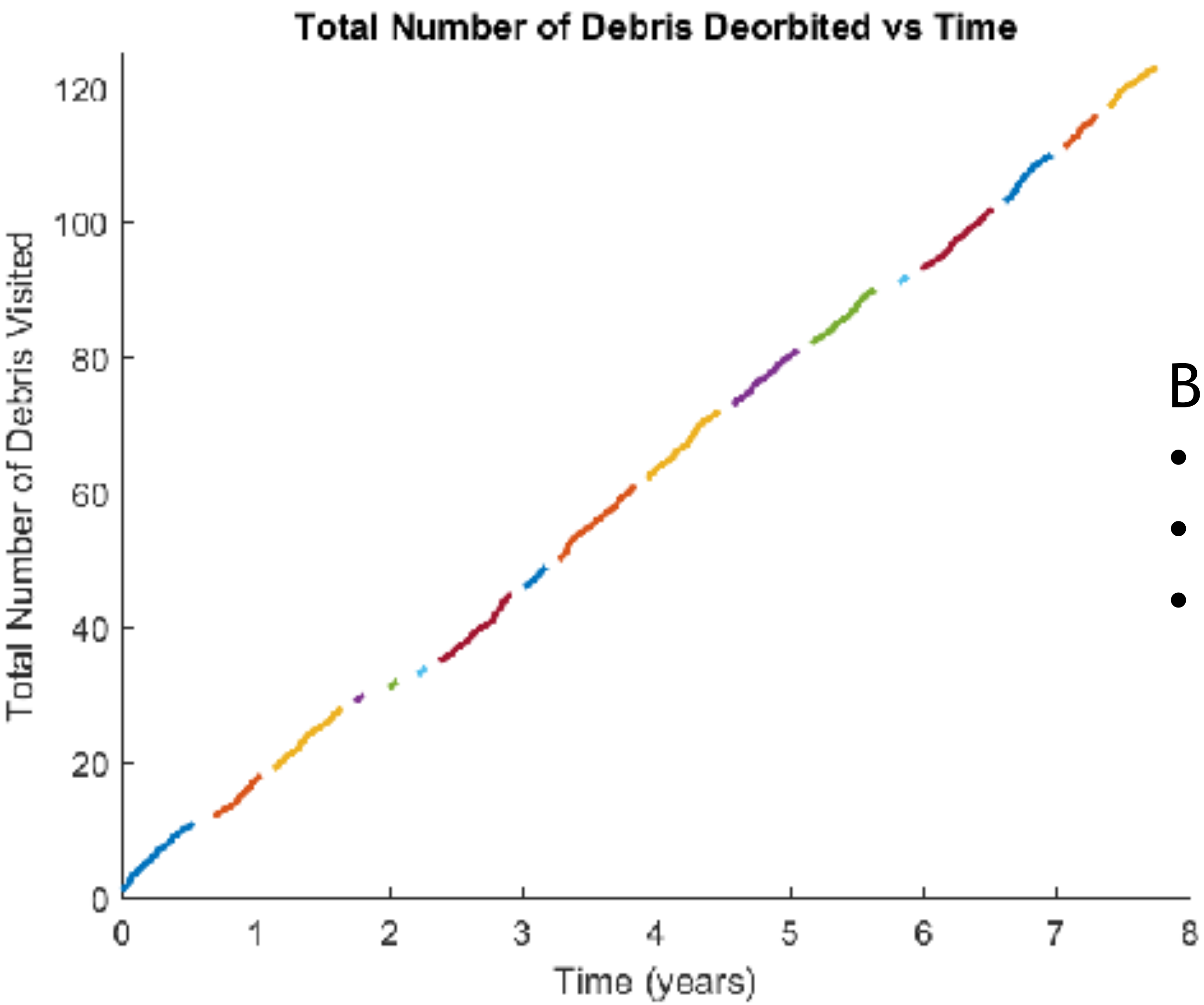
- Transition from approximate, analytical model to "high-fidelity" numerically integrated solution
  - 4 maneuvers
  - Optimization problem with 8 variables: $t_{23}, t_{45}, \Delta\vec{V}_2, \Delta\vec{V}_4$
  - ΔV_1 chosen deterministically to match Ω, u at t_2



Propagated with debris dynamics

Propagated with numerical integration

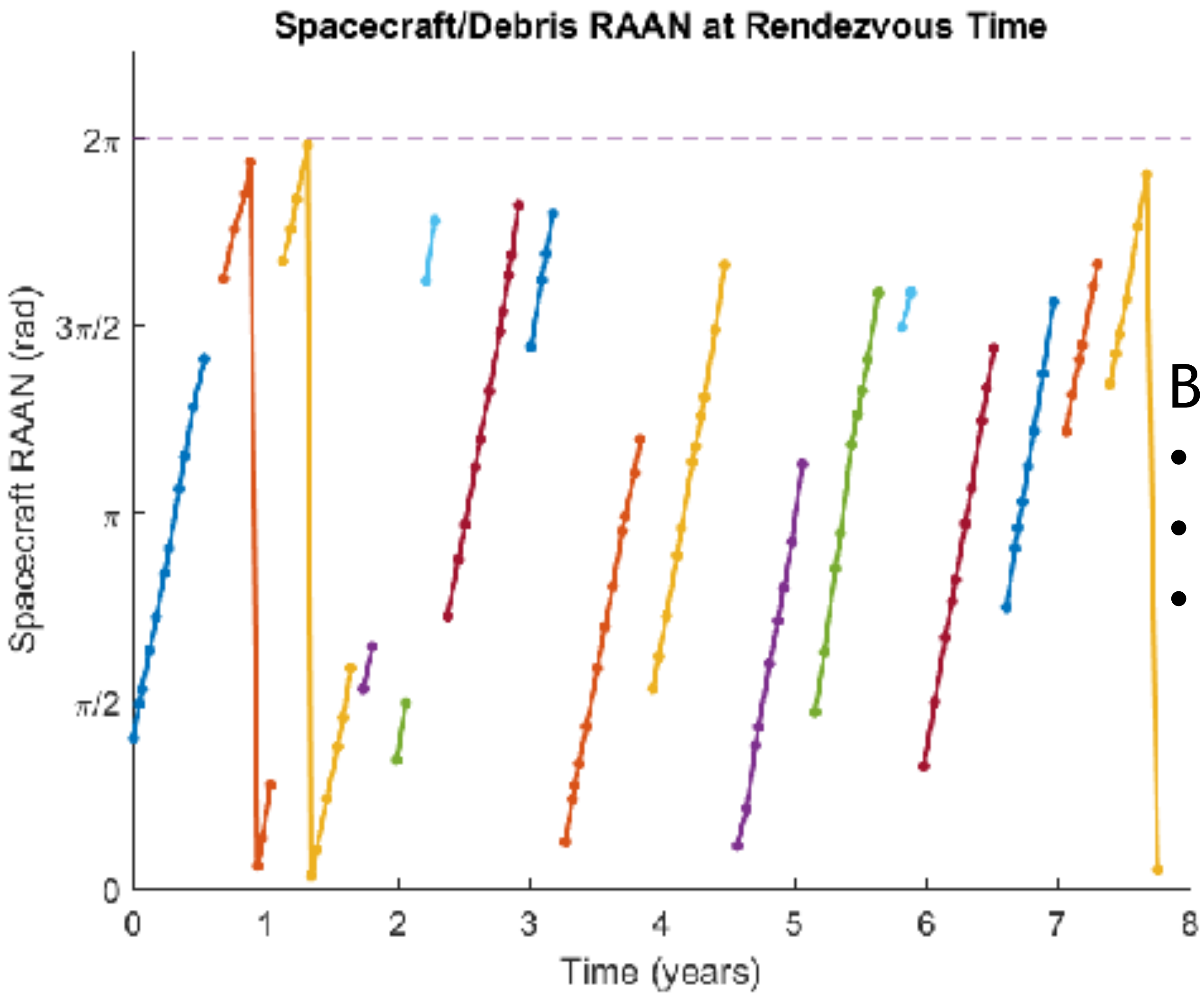
Final Solution



Best solution found:

- 1150.84 MEUR
- 17 launches
- 123 debris removed

Final Solution



Best solution found:
- 1150.84 MEUR
- 17 launches
- 123 debris removed

Conclusion

- Thank you to Dr. Dario Izzo for organizing
- Great learning opportunity for our team, mostly students

- We wish to thank the following for their help:
  - Prof. Natasha Bosanac (CU Boulder)
  - Dr. Jeffrey Parker (Advanced Space)
  - Prof. Christoffer Heckman (CU Boulder)

- This work was supported in part by a NASA Space Technology Research Fellowship