University of Strathclyde Engineering

# On the generation and evolution of multiple debris removal missions

**Carlos Ortega Absil*, Lorenzo A. Ricciardi*, Marilena Di Carlo*, Cristian Greco†, Romain Serra*, Mateusz Polnik*, Aram Vroom†, Annalisa Riccardi*, Edmondo Minisci*, Massimiliano Vasile***

* Department of Mechanical and Aerospace Engineering
University of Strathclyde, Glasgow (United Kingdom)

† Delft Institute of Technology, Delft (The Netherlands)

June 6, 2017

# The team

Dr Annalisa Riccardi

Dr. Edmondo Minisci

Prof Mx Vasile

Dr. Romain Serra

Carlos Ortega Absil

Lorenzo Ricciardi

Marilena Di Carlo

Mateusz Polnik

Cristian Greco

Aram Vroom

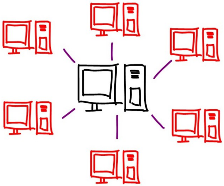
# Tools

University of Strathclyde Engineering

- **Slack** - instant messages, creation of discussion channels (#combinatorial_optimisation, #impulsive_transfer, #the_bakery, ...)
- **GitHub** - code developing
- **Wikispace** - sharing of information (problem considerations, plots, tutorials on parallel computing)
- **GDrive** - sharing of data (solution files, databases)

# Computing resources

- Archie-West: teaching cluster for the West of Scotland $\sim$ 12400 core/hours used
- Torque resource manager: 11 heterogeneous commodity hardware $\sim$ 60 cores



# Final rank



Table 1: Solution Rankings for the Kessler Run (GTOC9)

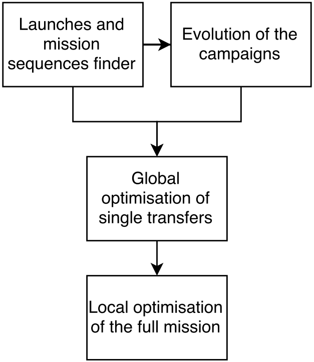
| Rank | Team Name | Missions | Removed | $J$ in MEUR |
|---|---|---|---|---|
| 1 | Jet Propulsion Laboratory | 10 | 123 | 731.2756 |
| 2 | NUDT Team | 12 | 123 | 786.21452 |
| 3 | XSCC-ADL | 12 | 123 | 821.37966 |
| 4 | Tsinghua-LAD | 12 | 123 | 829.57987 |
| 5 | NPU | 13 | 123 | 878.99821 |
| 6 | Strathclyde++ | 14 | 123 | 918.9808 |
| 7 | DLR | 14 | 123 | 949.85389 |
| 8 | Missions Learners | 14 | 123 | 964.51134 |
| 9 | The Aerospace Corporation | 14 | 123 | 1004.4860 |
| 10 | Team Jena | 15 | 123 | 1022.9063 |
| 11 | UT Austin | 15 | 122 | 1044.1787 |
| 12 | NJU Team | 16 | 123 | 1047.9685 |
| 13 | EFLMAN TEAM | 14 | 119 | 1107.6936 |
| 14 | CU Boulder | 17 | 123 | 1150.8439 |
| 15 | CAS-NUAA | 14 | 123 | 1182.0632 |
| 16 | MTU-UoM | 16 | 122 | 1192.7433 |
| 17 | NSSC-THU | 16 | 122 | 1210.3333 |
| 18 | Brute WORHP | 18 | 123 | 1229.5475 |
| 19 | The Goonies | 15 | 122 | 1238.6396 |
| 20 | NablaZeroLabs | 16 | 123 | 1267.7501 |
| 21 | TYSE | 16 | 123 | 1336.8590 |
| 22 | TM | 18 | 123 | 1490.9659 |
| 23 | Occitania | 22 | 120 | 1493.8567 |
| 24 | ARGoPS | 20 | 123 | 1512.6017 |
| 25 | Personal team | 23 | 123 | 1588.5770 |
| 26 | GO to space | 20 | 112 | 1819.1391 |
| 27 | UofI and Goddard | 23 | 123 | 1951.6797 |
| 28 | LSPirates | 20 | 105 | 2164.2321 |
| 29 | Astro-ASAP-UC3M | 13 | 85 | 3141.1951 |
| 30 | Cal Poly SLO | 39 | 84 | 4467.8746 |
| 31 | Team STAR Lab | 12 | 57 | 4481.7781 |
| 32 | Nicolas RAVE | 13 | 18 | 6453.0254 |
| 33 | National University of Colombia | 2 | 7 | 6511.5471 |
| 34 | MeltedCode | 1 | 5 | 6594.1105 |
| 35 | AMSS_GTOC | 1 | 4 | 6619.3569 |
| 36 | Bremen optimizers | 1 | 2 | 6760.20 |

# Solution approach

Three-step process involving:

- Low & high fidelity models
- Global & local optimisers

1. Beam search & sequence patching method
2. Global evolutionary optimisation
3. Local optimisation



# Dynamical models

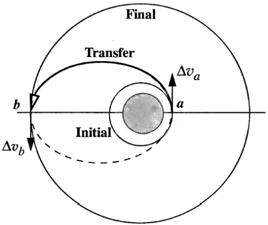
## Low-fidelity cost estimation of a transfer

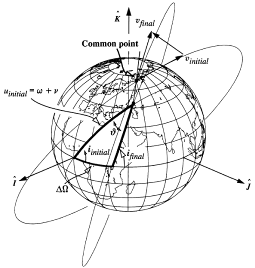
Assumptions:

- Keplerian motion
- Circular orbits
- No phasing

Combination of two parts:

- Hohmann transfer (2 in-plane maneuvers)
- Change of orbital plane ($i$ and $\Omega$ simultaneously)




The cheapest combination of the two is chosen as the predicted cost.

# Dynamical models

## Trajectory design for rendezvous

Ideas:

- 5 maneuvers (including 3 'deep-space' ones)
- Incremental complexity of the model

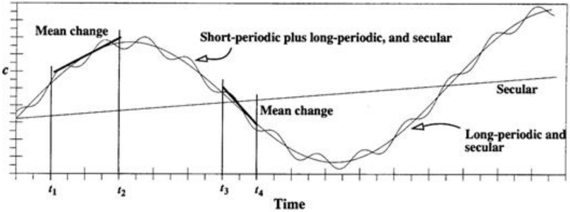
**1st model**: $J_2$ **secular perturbations** only

- $\Delta V$s applied on velocity obtained from osculating elements
- Propagation performed analytically with mean elements after conversion (same model as debris except for $M$)

$$\dot{M} = n \left[ 1 + \frac{3}{2} J_2 \left( \frac{R_\oplus}{p} \right)^2 \sqrt{1 - e^2} \left( 1 - \frac{3}{2} \sin^2 i \right) \right]$$

**2nd model**: $J_2$ **complete model**

- Numerical integration (predictor-corrector and Runge-Kutta)



# Combinatorial search

## Incremental construction of launch campaigns

- Itinerary $\{(D_j, t_j)\}$
- At level $k$, extend with either:
  - transfer to new target in active mission
  - **new launch** in available mission time intervals
- Base algorithm: *Beam Search*
  - easy to control memory and runtime $(Br, Be)$
  - heuristics to avoid **excess of permutations**

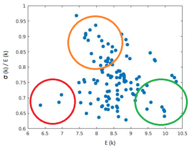
(a) Breadth-first-search (BFS) (b) Depth-first-search (DFS) (c) Beam-search (BS)

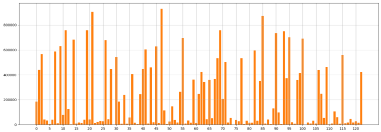
- $J$ estimated with low-fidelity $\Delta V$ model (precomputed).
- Various additional heuristics, corrections to the cost function, etc. gave **solutions with different properties**.

# Combinatorial search

## Incremental construction of launch campaigns

- New launch heuristics:
  - **Cyclic exploration** of mission timeline (>95% solutions).
  - Other used: concurrent search, cheap transfer density, etc.
- **Naïve initialisation:**
  - Each search from all 123 debris.
  - Total $\sim$ 150 searches in a grid of launch times $t_0$.
  - Keep a database, give priority to promising $t_0$ values.
- Additional heuristics:
  - Maximise length of shorter mission ($\rightarrow$ 13 launches!).
  - Per-debris **rarity bonus**, based on:
    - Cluster size first and second statistical moments.
    - Debris frequency in database of unexpensive missions.



# Combinatorial search

## Sequence patching

- **Input** Feasible sequences
- **Output** Launch campaign
- **Algorithm**
  Find a clique in the unidirected graph $(N, A)$
  - $N$ sequences
  - $A$ pairs of compatible sequences
- Extracted campaigns covering up to 116 debris from a database of 2.2 mln samples without single launches

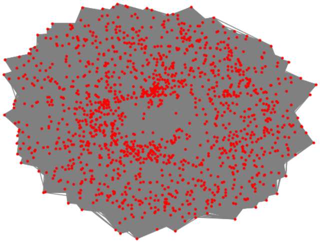


Figure: Partial graph of 1500 compatible sequences

# Evolution of the solutions

## The Time-shuffler encoding

Idea 1

- each debris associated to a (free) departure time
- times are then sorted in ascending order $\rightarrow$ ID sequence
- $\Delta t \leq 30$ between sorted times define missions
- within each mission, $5 \leq \Delta t \leq 30$ is imposed
- between missions $\Delta t \geq 43$ is imposed (3 days safety margin, 10 days for removal of first and last)

| | | | | | |
|---|---|---|---|---|---|
| $t_{deb}$ | 69.2 | 4.4 | 41.6 | 17 | 3.1 |
| ID | 1 | 2 | 3 | 4 | 5 |

| | | | | | |
|---|---|---|---|---|---|
| | | <5 | >5,<30 | >30,<43 | >5,<30 |
| $t_{sort}$ | 3.1 | 4.4 | 17 | 41.6 | 69.2 |
| ID | 5 | 2 | 4 | 3 | 1 |

| | | | | | |
|---|---|---|---|---|---|
| $t_{corr}$ | 3.1 | 8.1 | 17 | 60 | 69.2 |
| ID | 5 | 2 | 4 | 3 | 1 |
| | | M1 | | M2 | |

# Evolution of the solutions

## Reformulation of the problem

Idea 2

- The Time-shuffler encoding allows working only on continuous variables $\rightarrow$ timings are treated explicitly, combinatorial problem is treated implicitly, gradient based refinement possible
- Debris sequence, visit times and mission lengths are optimised concurrently $\rightarrow$ holistic global optimisation of the whole campaign (transfers evaluated with low fidelity model)
- To reduce number of missions, promising mass-unfeasible solutions should be retained $\rightarrow$ multi-objective problem, with mass constraint violation as second objective

$$\min_{\mathbf{L_t} \leq \mathbf{t} \leq \mathbf{U_t}} \mathbf{J}^*(\mathbf{t}), \qquad \mathbf{t} = (t_1, \dots, t_j, \dots, t_{123})$$

s.t.

$$5 \leq t_{s_j+1} - t_{s_j} \leq 30 \qquad \forall s_j \in M_i, \forall M_i$$

$$t_{s_1, M_{i+1}} - t_{s_{\mathrm{end}}, M_i} \geq 43 \quad \forall M_i$$

- solved with MACS, initialised with solutions from Beam Search

# Global optimisation

## Minimisation of the cost of the debris-debris transfer

- Objective: $\min \sum_{i=1}^{5} \Delta V_i$
- Optimisation variables: **times of application and magnitude and direction of impulsive $\Delta V$s**
- Population-based algorithm **MP-AIDEA** (Multi-Population Adaptive Inflationary Differential Evolution Algorithm): *https://github.com/strath-ace/smart-o2c*

**1st GLOBAL OPTIMISATION**
Model: $J_2$ secular perturbation
Solver: MP-AIDEA

↓

**2nd GLOBAL OPTIMISATION**
Model: $J_2$ complete model
Solver: MP-AIDEA

↓

**LOCAL OPTIMISATION**
Model: $J_2$ complete model
Solver: Matlab fmincon active-set

# Solution refinement

## Local optimization and tolerance matching

University of Strathclyde Engineering

### Direct multiple-shooting transcription scheme

- Initial guess: solution from local single-shooting
- $J_2$ complete dynamical model
- Runge-Kutta 4 integration scheme
- WORHP as NLP solver

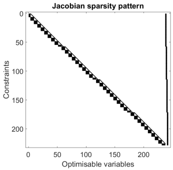
### Accuracy and computational efficiency enhancement

- Augmented variational dynamics to compute first and second-order derivative information

$$\frac{\partial \mathbf{G_x}(t,\mathbf{x})}{\partial t} = \frac{\partial \mathbf{f}(\mathbf{x})}{\partial \mathbf{x}} \cdot \mathbf{G_x}, \qquad \mathbf{G_x}(t_0, \mathbf{x}_0) = \boldsymbol{I}$$

where $\mathbf{G_x} = \frac{\partial \mathbf{x}(t)}{\partial \mathbf{x}_0}$ is the first-order sensitivity matrix.

- Sparsity patterns exploited
  - Constraints' Jacobian and Hessian non-zero elements: $< 0.1\%$

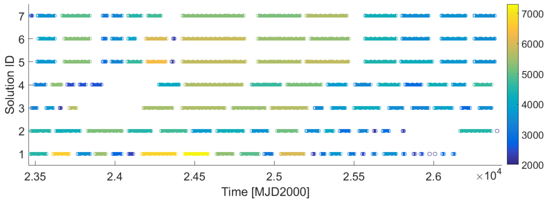


# Results

## Submitted solutions

| Solution ID | N. Launches | $\hat{J}$ | Improvements in solution process |
|---|---|---|---|
| 7 (The Mistery) | 14 | 918.98 | Larger population including diverse features. |
| 6 (The One) | 14 | 945.15 | Multi-objective formulation of evolution |
| 5 (Ghostbuster) | 14 | 967.49 | Added evolution algorithm to solution process, |
| | | | Small population of best submitted solutions. |
| 4 (Anibal) | 16 | 1028.72 | Further relaxation of search overconstraints. |
| 3 (Donald) | 16 | 1059.54 | Improved Cyclic Beam Search heuristics. |
| *"Make Strathclyde great again"* | | | Improved low-fidelity model. |
| | | | Improved global optimisation on high-fidelity model. |
| 2 (Wrappy) | 18 | 1133.94 | Cyclic Beam Search. |
| | | | Improved high-fidelity model. |
| | | | Added single and multiple-shooting refinement. |
| 1 (Wary) | 26 | 1713.07 | Beam Search in the first 100 mission days. |
| | | | No thorough trajectory refinement. |

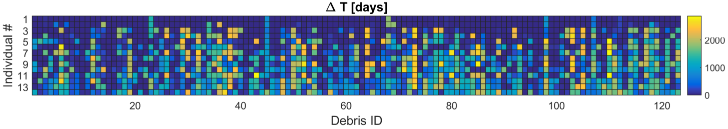
Time and mass distribution of the missions of the submitted campaigns:



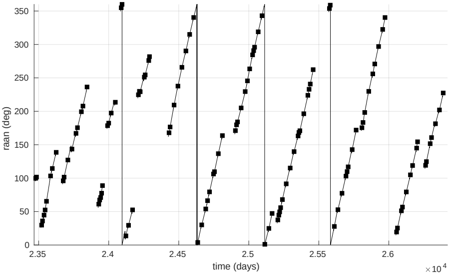
# Results

## Final solution

Difference in departure time from each debris between final solution and the initial 14 individuals used by MACS:



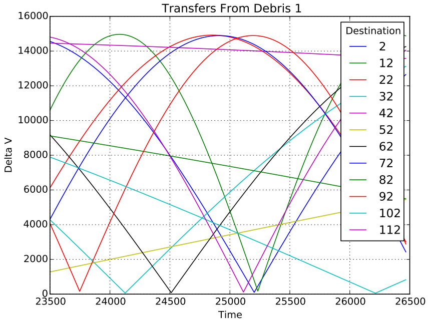

RAAN evolution of the spacecraft:

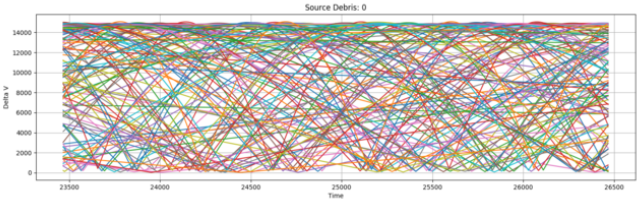


# Results

## Low-fidelity estimation of the $\Delta V$ for debris-to-debris transfer

Cost of the transfer from debris 1 to other debris in the database:



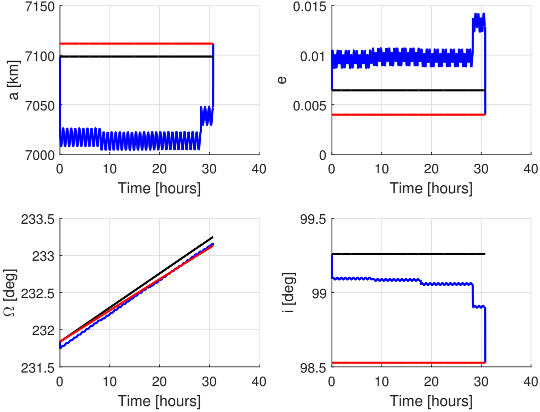



# Results

## Example of transfer between debris

- Black: departure orbital element
- Red: arrival orbital element
- Blue: High-fidelity orbital elements variation during transfer between debris

The optimiser reduces the semi-major axis to exploit the natural $J_2$ drift



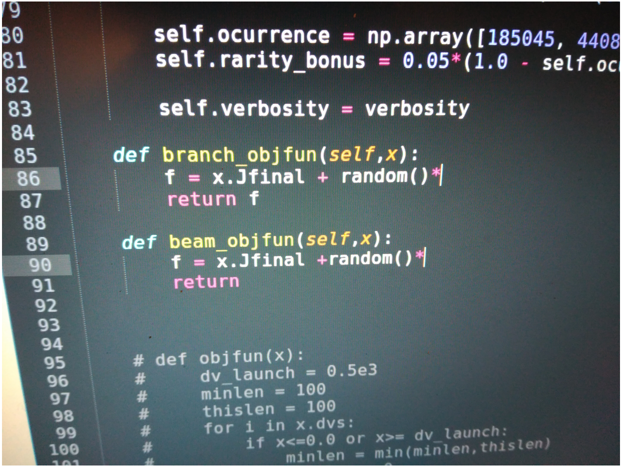
# Future ideas

University of Strathclyde Engineering

- Complete the construction of the **surrogate model** for the low fidelity transfer estimation → Multi-layer Perceptron Neural Network with 12 inputs (6 departure orbital elements, 6 arrival orbital elements), 2 outputs: time of transfer and $\Delta V$
- Solve the problem as an integer/mixed integer programming problem: example **bin-packing** problem formulation
- Study the best generation of **sequences** for sequence patching: allow the best variety of sequences
- Study the evolutionary **(non-) combinatorial** approach and its possible applications
- Investigate further the best **number of impulses** for the problem
- Win GTOC next year :)

# GTOC last day

while continuing pickling solutions...



# GTOC art

Part of the GTOC team enjoying some space debris art... and wine, the day after

by Dr Stuart Grey