National Aeronautics and Space Administration

# Multi-objective Hybrid Optimal Control for Interplanetary Mission Planning

code 595

Jacob Englander

4-1-2015

NAVIGATION & MISSION DESIGN BRANCH
NASA GSFC

www.nasa.gov

# The Team

- NASA Goddard Space Flight Center
  - Jacob Englander
  - Frank Vaughn
- a.i. solutions
  - Matthew Vavrina
- University of Illinois
  - Professor Bruce Conway
  - Dr. Alexander Ghosh
  - Donald Ellison
  - Ryne Beeson
- University of Vermont
  - David Hinckley

# Introduction to the General Interplanetary Mission Design Problem

- The interplanetary design problem is composed of both discrete and real-valued decision parameters:
  - Choice of destination(s), number of planetary flybys, identities of flyby planets
  - Launch date, flight time(s), epochs of maneuvers, control history, flyby altitudes, etc.
- For example, for a main-belt asteroid mission, the designer must choose:
  - The optimal asteroid from a set of scientifically interesting bodies provided by the customer
  - Whether or not to perform planetary flybys on the way to the main belt and, if so, at which planets
  - Optimal trajectory from the Earth to the chosen asteroid by way of the chosen flyby planets

# Unique Characteristics of Low-Thrust Interplanetary Mission Design

- Low-thrust electric propulsion is characterized by high power requirements but also very high specific impulse ($I_{sp}$), leading to very good mass fractions
- Low-thrust trajectory design is a very different process from chemical trajectory design
  - Chemical thrusters fire for minutes during a mission time-scale of years
    - Chemical maneuvers may be approximated as occurring instantaneously (impulsively)
    - This allows a mission designer to parameterize the cost of a chemical mission in units of “change in velocity,” or $\Delta v$
    - $\Delta v$ is invariant to changes in spacecraft hardware or launch vehicle, and mass fraction may be computed using only $I_{sp}$ and $\Delta v$
  - Electric thrusters fire for months or years, sometimes the entire mission duration
    - The mission designer therefore must choose a time-history of thrust control rather than discrete maneuvers
    - The impulsive approximation does not apply, and so the affect of the thruster must be continuously integrated as the spacecraft flies
    - The propellant flow rate required to provide a given thrust is dependent on the available power at that instant
    - Therefore different choices of power and thruster systems change the trajectory solution!

# Traditional Methods of Low-Thrust, Multi-Flyby Trajectory Design

- Several methods of picking the destination and flyby sequence:
  - Grid search over all possible choices of destinations, flyby sequence, propulsion system, power system, etc. (very expensive and often impractical)
  - Intuition-guided manual design of the trajectory (even more expensive, can miss non-intuitive solutions)
- Several methods of designing the trajectory:
  - Local optimization from an initial guess provided by a chemical mission design (but sometimes the optimal chemical trajectory does not resemble the optimal low-thrust trajectory)
  - Local optimization from an initial guess provided by a low-fidelity approximation to the low-thrust model, i.e. shaped-based methods (but sometimes the shape-based method cannot accurately approximate the true trajectory)

# Brief History of Automated Interplanetary Trajectory Design

- Gage, Braun, and Kroo, 1994 – autonomous chemical design with variable mission sequence (no deep-space maneuvers)
- Vasile and de Pascale, 2005 – autonomous chemical design for fixed mission sequence
- Vinko and Izzo, 2008 – autonomous chemical design for fixed mission sequence
- Wall and Conway, 2009 – autonomous low-thrust design for fixed mission sequence (no planetary flybys)
- Chilan and Conway, 2009 – autonomous low-thrust and chemical design for fixed mission sequence (no planetary flybys)
- Yam, di Lorenzo, and Izzo, 2011 – autonomous low-thrust design for fixed mission sequence
- Abdelkhalik and Gad, 2011, 2012, and 2013 – autonomous chemical design with variable mission sequence
- Englander, Conway, and Williams, 2012 – autonomous chemical design with variable mission sequence
- Englander (dissertation) 2013 – autonomous low-thrust design with variable mission sequence

# Automated Mission Design via Hybrid Optimal Control

- Break the mission design problem into two stages, or “loops”
  - “outer-loop” picks sets of destinations, planetary flybys, sizes the power system, can pick propulsion system – a <u>discrete</u> optimization problem
  - “inner-loop” finds the optimal trajectory for a given candidate outer-loop solution – a <u>real-valued</u> optimization problem
  - For the outer-loop to work, the inner-loop must function autonomously (i.e. no human interaction)

# Multi-Objective Hybrid Optimal Control

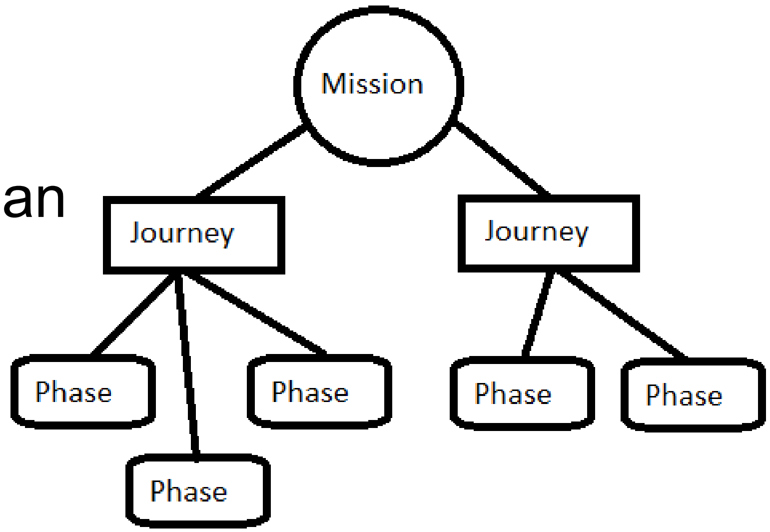
- The customer (scientist or project manager) most often does not want just one point solution to the mission design problem
- Instead, an exploration of a multi-objective trade space is required
- For a typical main-belt asteroid mission the customer might wish to see the trade-space of:
  - Launch date vs
  - Flight time vs
  - Deliverable mass
  - While varying the destination asteroid, planetary flybys, solar array size, etc
- To address this question we use a multi-objective discrete outer-loop which defines many single objective real-valued inner-loop problems

# Outer-Loop Transcription and Optimization

- The outer-loop finds the non-dominated trade surface between any set of objective functions chosen by the user
- Non-dominated surface means "no point on the surface is superior to any other point on the surface in all of the objective functions"
- The outer-loop solver may choose from a menu of options for each decision variable
- The choices made by the outer-loop solver are used to define trajectory optimization problems to be solved by the inner-loop

# Anatomy of a Mission

- Break mission into a set of “journeys,” each of which in turn is broken into “phases”
- The endpoints of a journey are chosen in the problem assumptions
- The endpoints of a phase (i.e. a flyby target) may be chosen by the user or an Outer-Loop solver



# Outer-Loop Transcription: An Example

**Asteroid Choices**

| Code | Body |
|---|---|
| 0 | Ceres |
| 1 | Pallas |
| 2 | Juno |
| 3 | Vesta |
| 4 | Astraea |
| 5 | Hebe |
| 6 | Iris |
| 7 | Flora |
| 8 | Metis |
| 9 | Hygiea |
| 10 | Parthenope |
| 11 | Victoria |
| 12 | Egeria |
| 13 | Irene |
| 14 | Eunomia |
| 15 | Psyche |
| 16 | Thetis |

**Flight time upper bound**

| Code | Flight Time (y) |
|---|---|
| 0 | 6 |
| 1 | 6.5 |
| 2 | 7 |
| 3 | 7.5 |
| 4 | 8 |
| 5 | 8.5 |
| 6 | 9 |
| 7 | 9.5 |
| 8 | 10 |
| 9 | 10.5 |
| 10 | 11 |
| 11 | 11.5 |
| 12 | 12 |

**Array Size**

| Code | Power (kW) |
|---|---|
| 0 | 10 |
| 1 | 11 |
| 2 | 12 |
| 3 | 13 |
| 4 | 14 |
| 5 | 15 |
| 6 | 16 |
| 7 | 17 |
| 8 | 18 |
| 9 | 19 |
| 10 | 20 |

**Flyby Choices (Journey 1)**

| Code | Body |
|---|---|
| 0 | Earth |
| 1 | Mars |
| 2 | none |
| 3 | none |

**Flyby Choices (Journey 2)**

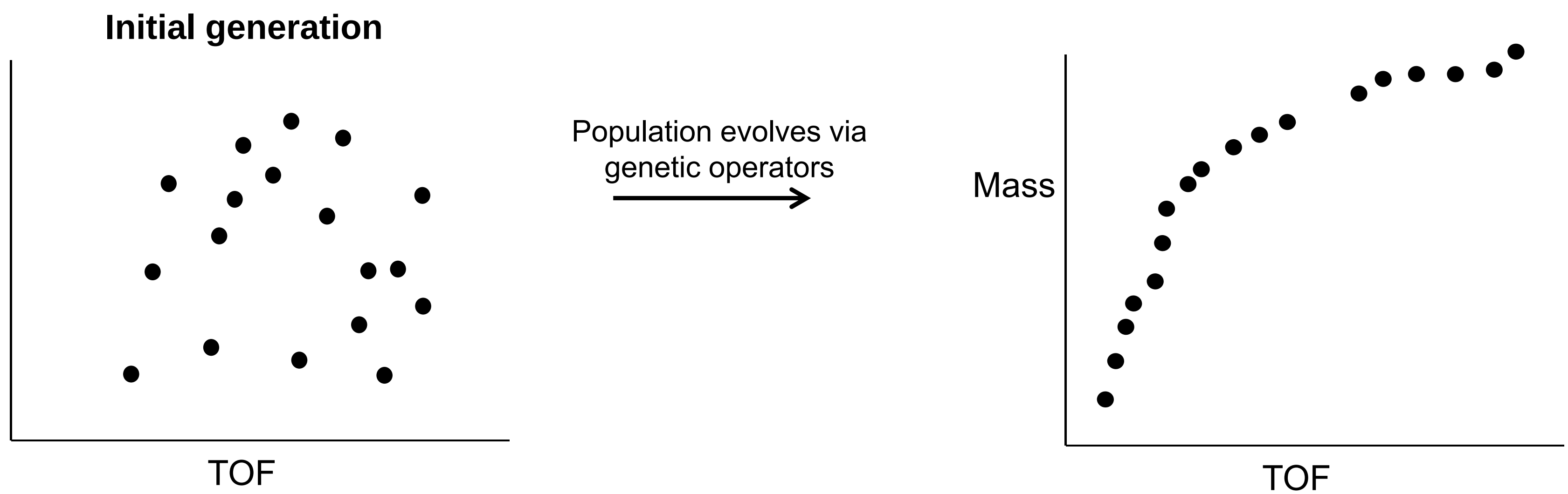
| Code | Body |
|---|---|
| 0 | Mars |
| 1 | none |

**Sample Mission**

| | Array Size | Flight Time Upper Bound | Asteroid 1 | Potential Planetary Flyby 1 | Asteroid 2 | Potential Planetary Flyby 2 |
|---|---|---|---|---|---|---|
| **Code** | 8 | 4 | 0 | 1 | 1 | 1 |
| **Translation** | 18 kW | 8 y | Ceres | Mars | Pallas | none |

# Multi-Objective Optimization via NSGA-II

- The outer-loop optimization problem is solved using a discrete multi-objective solver, in this case Non-Dominated Sorting Genetic Algorithm II (NSGA-II)
- NSGA-II finds the non-dominated front, surface, or hyper-surface between any number of objectives chosen by the user

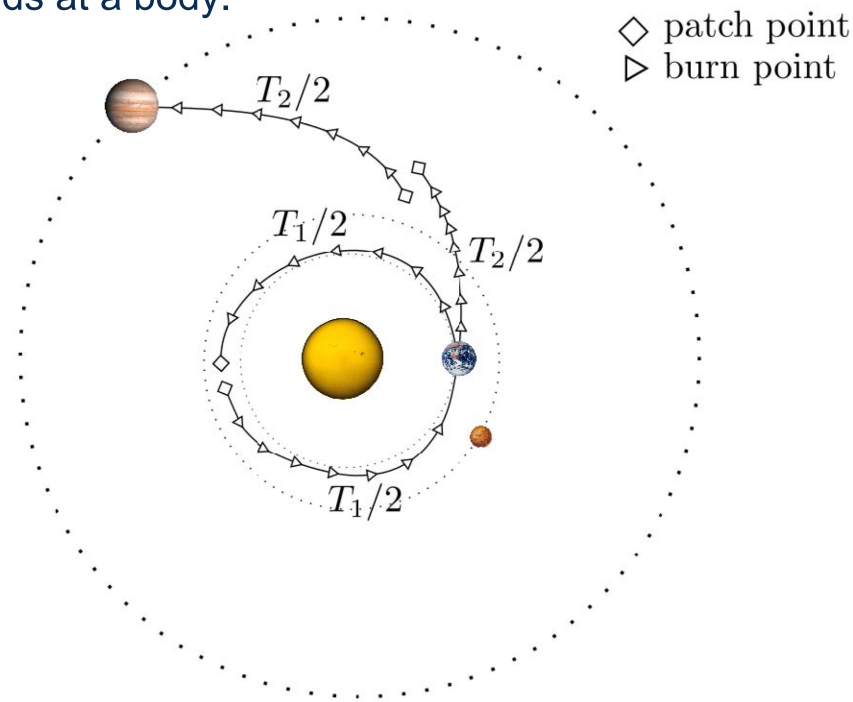


# Inner-Loop Modeling and Optimization

- The inner-loop solves a real-valued trajectory optimization problem which is defined by each candidate solution to the outer-loop problem
- The inner-loop must function autonomously because the problems are generated in real time and there is no opportunity for human intervention
- The outer-loop is only as good as the solutions to the inner-loop problem, so the inner-loop must be robust
- A given run of the outer-loop may require hundreds or even thousands of runs of the inner-loop, so the inner-loop must be fast
- If the individual inner-loop runs are independent then many of them can be run in parallel

# Multiple Gravity Assist with Low-Thrust (MGALT) via the Sims-Flanagan Transcription

- Break mission into phases. Each phase starts and ends at a body.
- Sims-Flanagan Transcription
  - Break phases into time steps
  - Insert a small impulse in the center of each time step, with bounded magnitude
  - Optimizer Chooses:
    - Launch date
      - For each phase:
        - Initial velocity vector
        - Flight time
        - Thrust-impulse vector at each time step
        - Mass at the end of the phase
        - Terminal velocity vector
- Assume two-body force model; propagate by solving Kepler's problem
- Propagate forward and backward from phase endpoints to a "match point"
- Enforce nonlinear state continuity constraints at match point
- Enforce nonlinear velocity magnitude and altitude constraints at flyby



# Power, Propulsion, and Ephemeris Modeling

- Medium-fidelity mission design requires accurate hardware modeling
- Launch vehicles are modeled using a polynomial fit

$$m_{delivered} = (1 - \sigma_{LV})\left(a_{LV}C_3^5 + b_{LV}C_3^4 + c_{LV}C_3^3 + d_{LV}C_3^2 + e_{LV}C3 + f_{LV}\right)$$

where $\sigma_{LV}$ is launch vehicle margin and $C_3$ is hyperbolic excess velocity

- Thrusters are modeled using either a polynomial fit to published thrust and mass flow rate data

$$\dot{m} = a_F P^4 + b_F P^3 + c_F P^2 + d_F P + e_F$$
$$T = a_T P^4 + b_T P^3 + c_T P^2 + d_T P + e_T$$

or, when detailed performance data is unavailable

$$T = \frac{2\,\eta P}{I_{sp} g_0}$$

- Power is modeled by a standard polynomial model

$$\frac{P_0}{r^2}\left(\frac{\gamma_0 + \frac{\gamma_1}{r} + \frac{\gamma_2}{r^2}}{1 + \gamma_3 r + \gamma_4 r^2}\right)(1-\tau)^t$$

where $P_0$ is the power at beginning of life at 1 AU and $\tau$ is the solar array degradation constant

- Ephemeris data for solar system bodies is provided via the SPICE toolkit

# Inner-Loop Solver: Nonlinear Programming (NLP)

Minimize $f(\boldsymbol{x})$

Subject to:

$$\boldsymbol{x}_{lb} \le \boldsymbol{x} \le \boldsymbol{x}_{ub}$$
$$\boldsymbol{c}(\boldsymbol{x}) \le \boldsymbol{0}$$
$$\boldsymbol{A}\boldsymbol{x} \le \boldsymbol{0}$$

where:

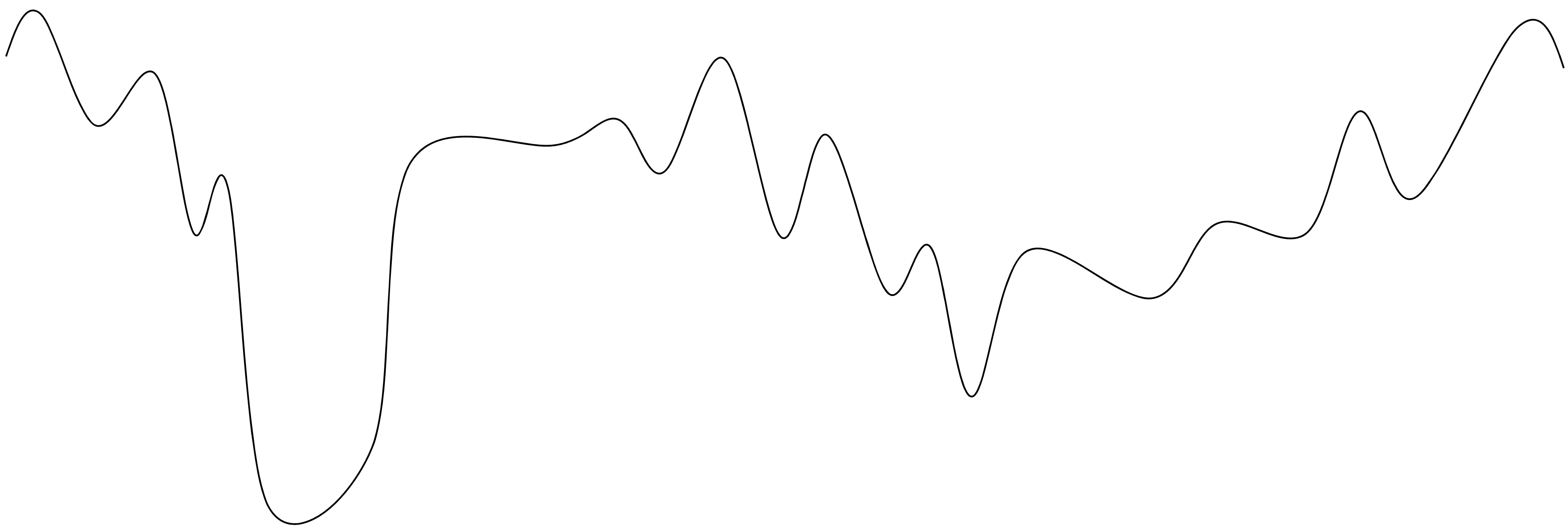
$\boldsymbol{x}_{lb}$, $\boldsymbol{x}_{ub}$ are lower and upper bounds on the decision variables

$\boldsymbol{c}(\boldsymbol{x})$ is a vector of nonlinear constraints

$\boldsymbol{A}\boldsymbol{x}$ is a vector of linear constraints

- There are several third party solvers that do this (SNOPT, IPOPT, fmincon, vf13AD)
- But all of these methods require an initial guess…

# Inner-Loop Solver: Monotonic Basin Hopping (MBH)



Leary, 2000
Vasile, Minisci, and Locatelli, 2009
Yam, di Lorenzo, and Izzo, 2011
Englander (dissertation), 2013
Casioli *et al.*, 2013
Englander and Englander, 2014

Improved from standard MBH by:

1. “Feasible point finder” aggregate penalty method
2. Non-uniform (Pareto) perturbation step
3. “Time-hop” operator (Casioli *et al.*)

# Example: Main-Belt Two Asteroid Tour

| | |
|---|---|
| **Mission Objective** | Visit two main-belt asteroids with diameter greater than 50 km (475 bodies meet this filter) |
| **Launch Vehicle** | Atlas V 401 |
| **Power System** | |
| Array power at 1 AU | 15 kW |
| Cell performance model | $1/r^2$ |
| Spacecraft bus power | 800 W |
| Power margin | 15% |
| **Propulsion System** | |
| Thruster | NEXT (throttle table 11, high-Isp mode) |
| Number of thrusters | 1 |
| Duty cycle | 90% |
| Propellant tank | unconstrained |
| **Mission Sequence** | up to two planetary flybys are permitted before the first asteroid and up to one between the first and second asteroids |
| **Inner-Loop Objective Function** | Maximize delivered mass to second asteroid |
| **Outer-Loop Objective Functions** | Delivered mass to second asteroid<br>Launch year<br>Flight time |

# Main-Belt Two Asteroid Tour: Outer-Loop Menu

**Launch Year**

| Code | Year |
| --- | --- |
| 0 | 2020 |
| 1 | 2021 |
| 2 | 2022 |
| 3 | 2023 |
| 4 | 2024 |
| 6 | 2025 |
| 7 | 2026 |
| 8 | 2027 |
| 9 | 2028 |
| 10 | 2029 |

**Flight Time Upper Bound**

| Code | # Years |
| --- | --- |
| 0 | 5 |
| 1 | 6 |
| 2 | 7 |
| 3 | 8 |
| 4 | 9 |
| 5 | 10 |
| 7 | 11 |
| 8 | 12 |

**First Asteroid**

| Code | Body |
| --- | --- |
| 0 | Ceres |
| 1 | Pallas |
| 2 | Juno |
| 3 | Vesta |
| 4 | Astraea |
| 5 | Hebe |
| 6 | Iris |
| 7 | Flora |
| … | (475 choices) |

**Second Asteroid**

| Code | Body |
| --- | --- |
| 0 | Ceres |
| 1 | Pallas |
| 2 | Juno |
| 3 | Vesta |
| 4 | Astraea |
| 5 | Hebe |
| 6 | Iris |
| 7 | Flora |
| … | (475 choices) |

**First Journey First Flyby**

| Code | Body |
| --- | --- |
| 0 | Earth |
| 1 | Mars |
| 2 | Jupiter |
| 3 | No flyby |
| 4 | No flyby |
| 5 | No flyby |

**First Journey Second Flyby**

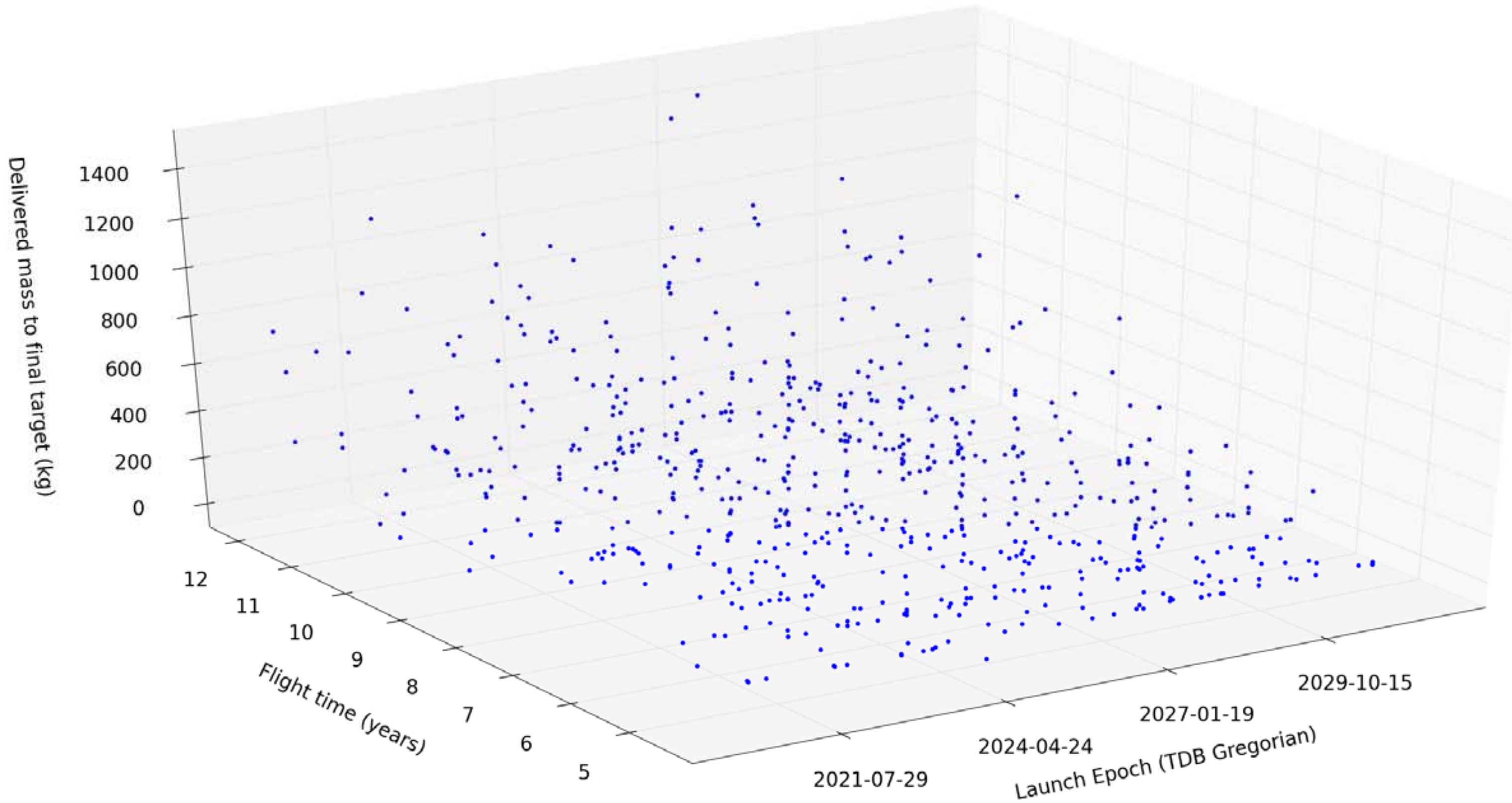
| Code | Body |
| --- | --- |
| 0 | Earth |
| 1 | Mars |
| 2 | Jupiter |
| 3 | No flyby |
| 4 | No flyby |
| 5 | No flyby |

**Second Journey Flyby**

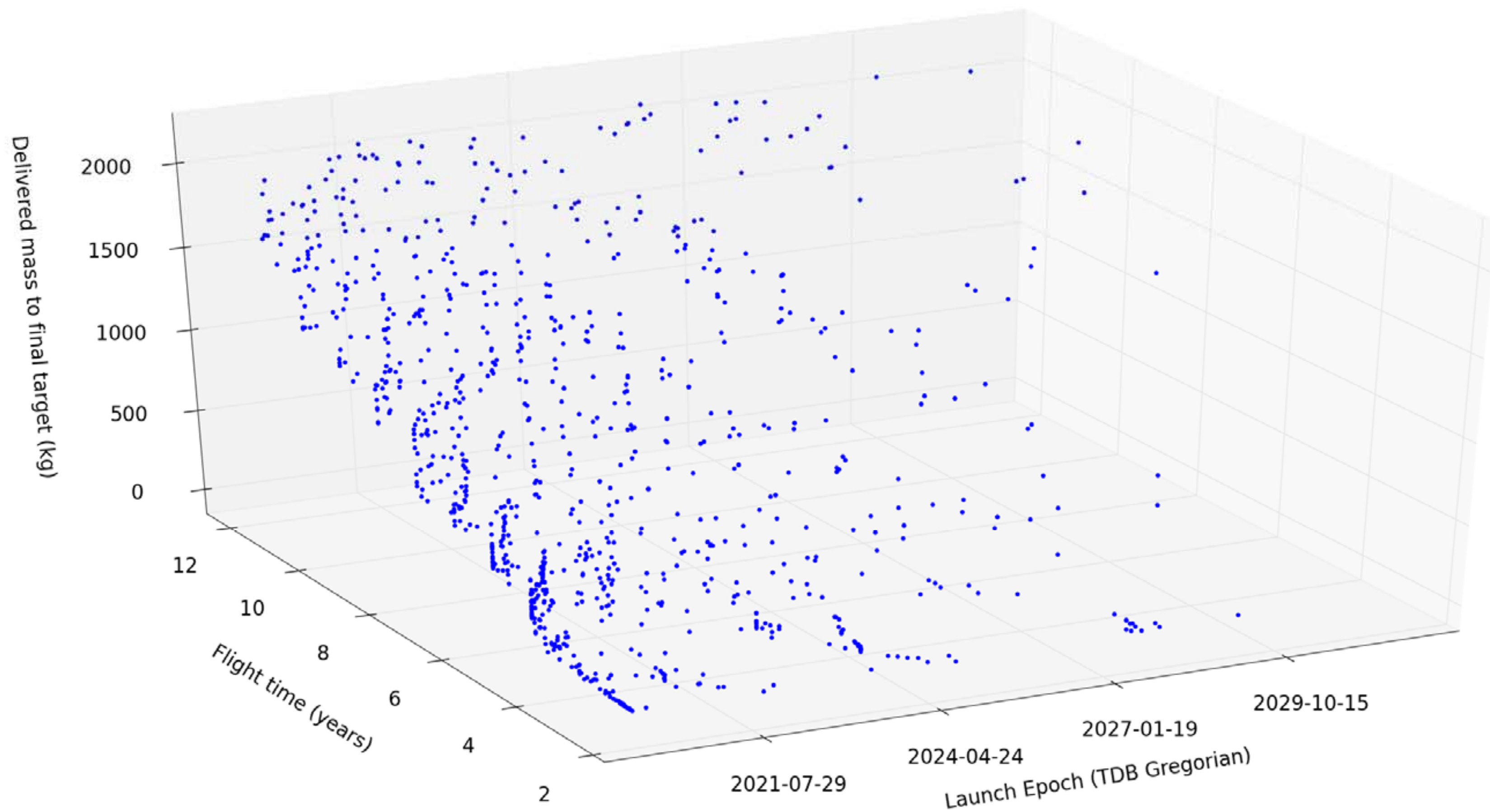
| Code | Body |
| --- | --- |
| 0 | Earth |
| 1 | Mars |
| 2 | Jupiter |
| 3 | No flyby |
| 4 | No flyby |
| 5 | No flyby |

$1.16 \times 10^9$ possible combinations, $4.82 \times 10^9$ with duplicates

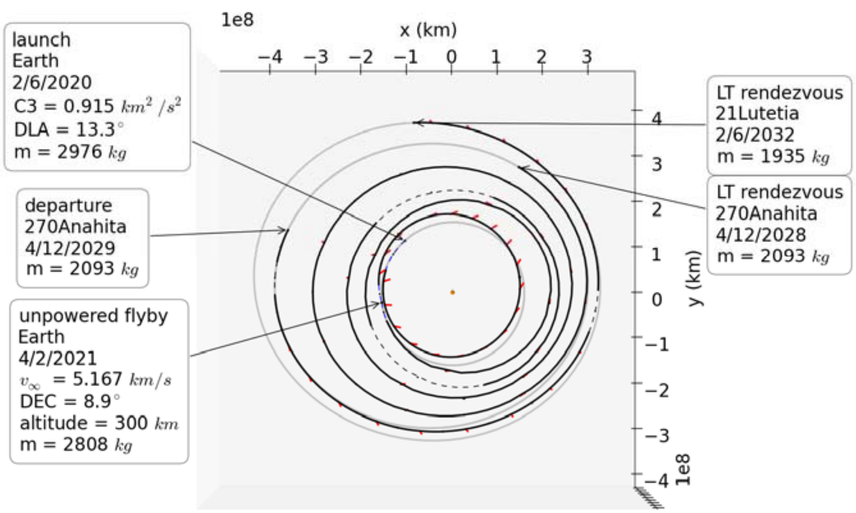
# Main-Belt Two Asteroid Tour: First Generation Trade Space



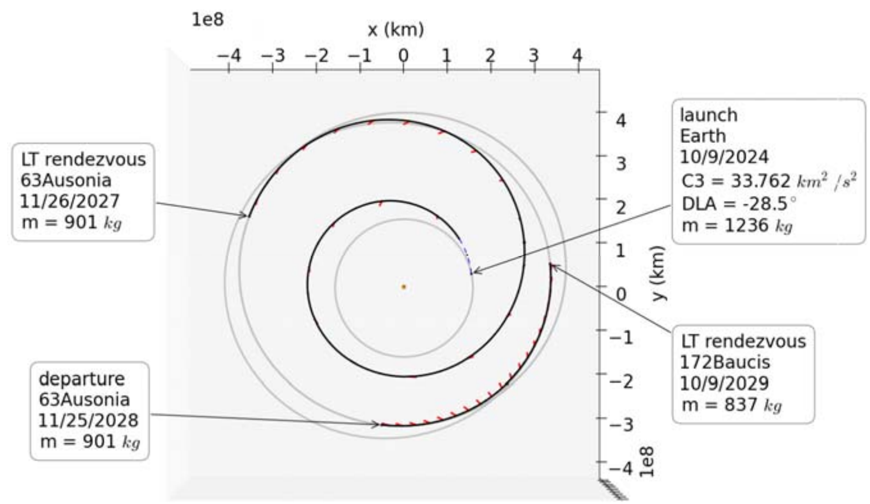
# Main-Belt Two Asteroid Tour: Final Generation Trade Space



# Main-Belt Two Asteroid Tour: Example Trajectories




A 12-year mission to Anahita and Lutetia delivers a very large science payload




A 5-year mission to Ausonia and Baucis delivers a smaller payload in less time and at a lower cost

**Both of these candidate missions, and many others, would be valuable data points for our scientist customers**

# Example: Deimos Large-Mass Sample Return

| | |
|---|---|
| **Mission Objective** | Return a large boulder from Deimos |
| **Launch Vehicle** | Delta IV Heavy with lunar flyby (C3 2.0) |
| **Power System** | |
| Array power at 1 AU | chosen by optimizer |
| Cell performance model | $1/r^2$ |
| Spacecraft bus power | 2.0 kW |
| Power margin | 0% |
| **Propulsion System** | |
| Thruster | chosen by optimizer (high-Isp or high-thrust versions of a large Hall thruster) |
| Number of thrusters | chosen by optimizer (3, 4, 5, or 6) |
| Duty cycle | 90% |
| Propellant tank | unconstrained |
| **Mission Sequence** | Direct travel to Mars followed by direct return to C3 2.0 for lunar flyby capture<br>Mars arrival/departure is modeled using an Edelbaum spiral |
| **Inner-Loop Objective Function** | Maximize sample return mass |
| **Outer-Loop Objective Functions** | Sample return mass<br>Solar array size<br>Launch epoch<br>Flight time |

# Deimos Sample Return: Outer-Loop Menu

**Power Supply at 1 AU**

| Code | Array Output |
|---|---|
| 0 | 40 |
| 1 | 41 |
| 2 | 42 |
| 3 | 43 |
| 4 | 44 |
| 5 | 45 |
| 6 | 46 |
| 7 | 47 |
| 8 | 48 |
| 9 | 49 |
| 10 | 50 |
| 11 | 51 |
| 12 | 52 |
| 13 | 54 |
| 14 | 56 |
| 15 | 58 |
| 16 | 60 |
| 17 | 62 |
| 18 | 64 |
| 19 | 66 |
| 20 | 68 |
| 21 | 70 |

**Launch Year**

| Code | Year |
|---|---|
| 0 | 2019 |
| 1 | 2020 |
| 2 | 2021 |
| 3 | 2022 |
| 4 | 2023 |
| 6 | 2024 |
| 7 | 2025 |
| 8 | 2026 |
| 9 | 2027 |

**Flight Time Upper Bound**

| Code | Days |
|---|---|
| 0 | 800 |
| 1 | 900 |
| 2 | 1000 |
| 3 | 1100 |
| 4 | 1200 |
| 5 | 1300 |
| 7 | 1400 |
| 8 | 1500 |
| 9 | 1600 |
| 10 | 1700 |
| … | |
| 26 | 3300 |

**Thruster Type**

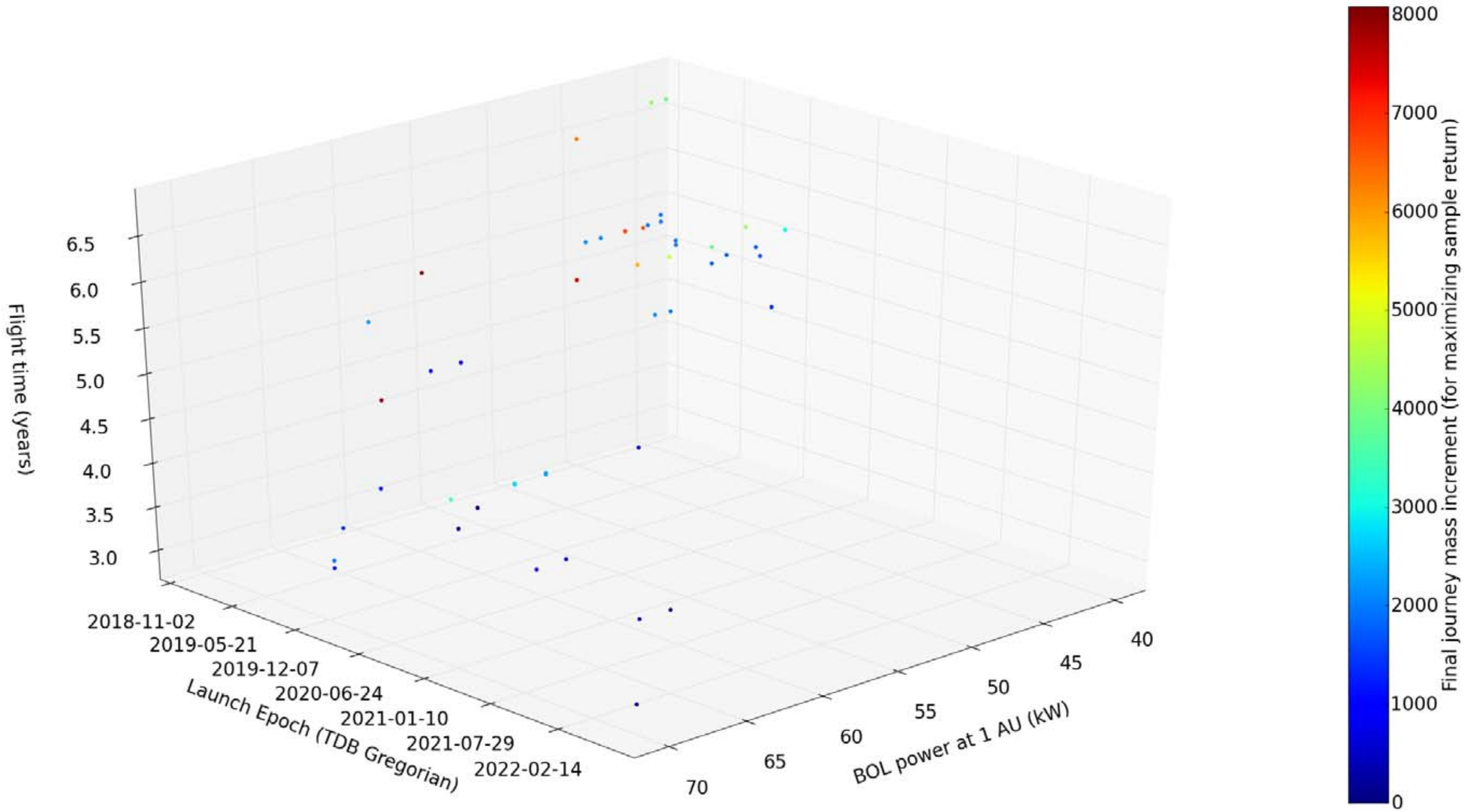
| Code | Thruster |
|---|---|
| 0 | 13 kW Hall (High-Isp) |
| 1 | 13 kW Hall (medium-thrust) |
| 2 | 13 kW Hall (High-thrust) |

**Number of Thrusters**

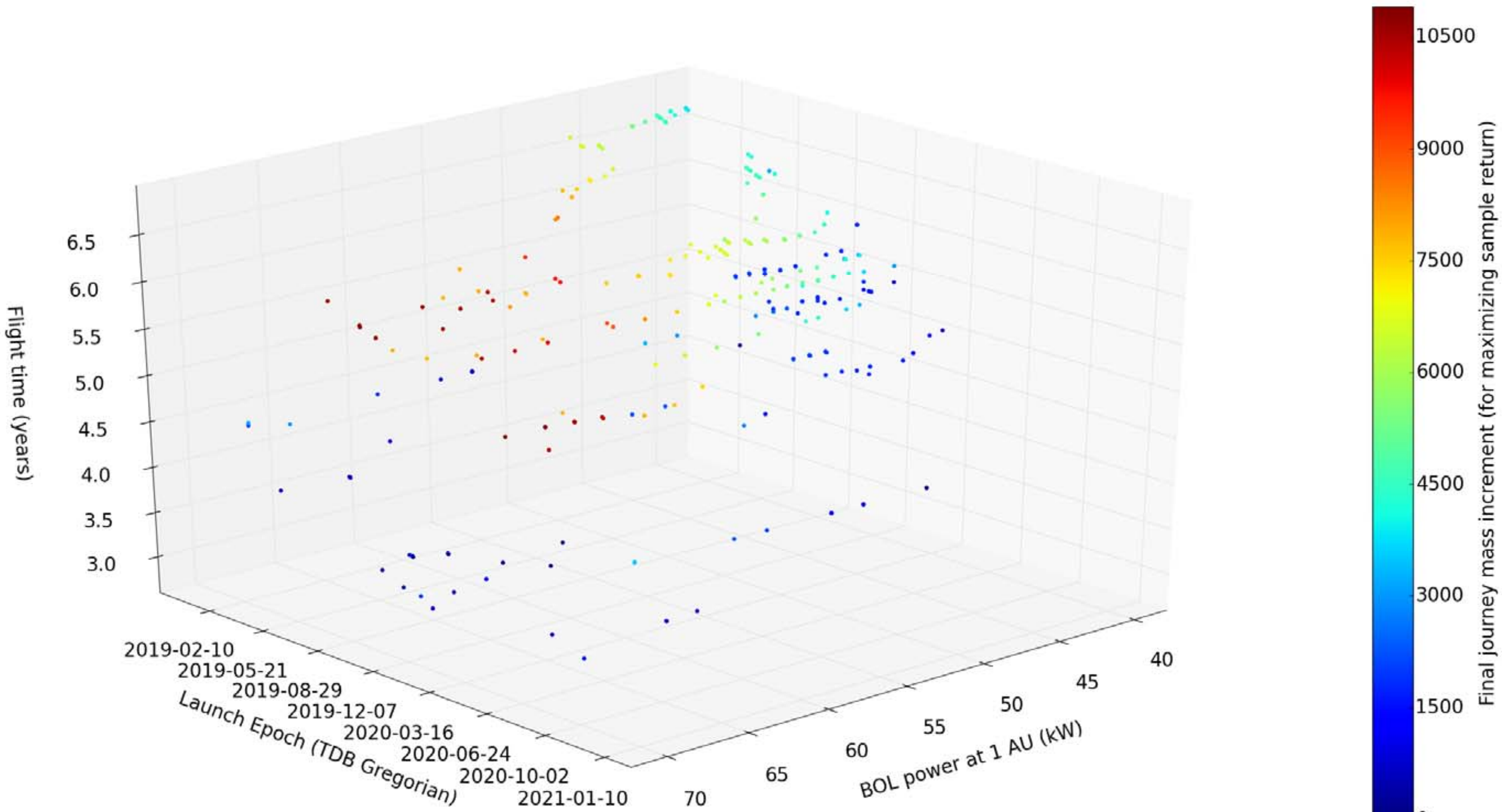
| Code | # Thrusters |
|---|---|
| 0 | 3 |
| 1 | 4 |
| 2 | 5 |
| 3 | 6 |

71280 possible combinations

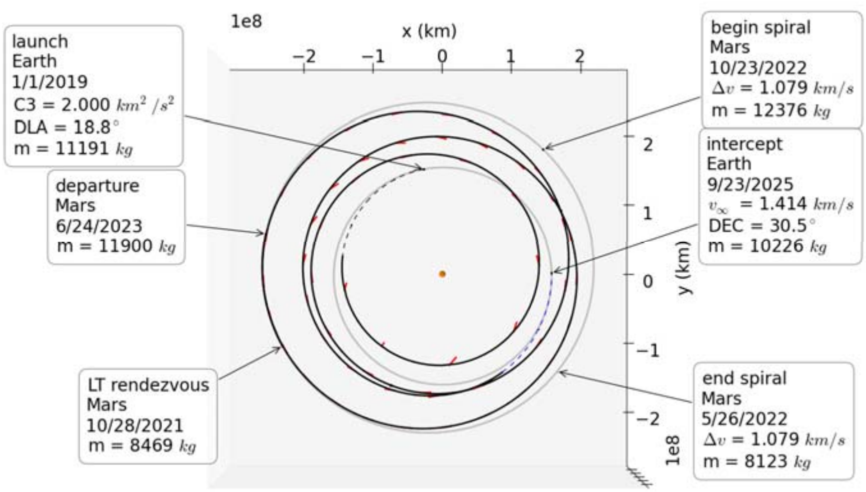
# Deimos Sample Return: First Generation Trade Space



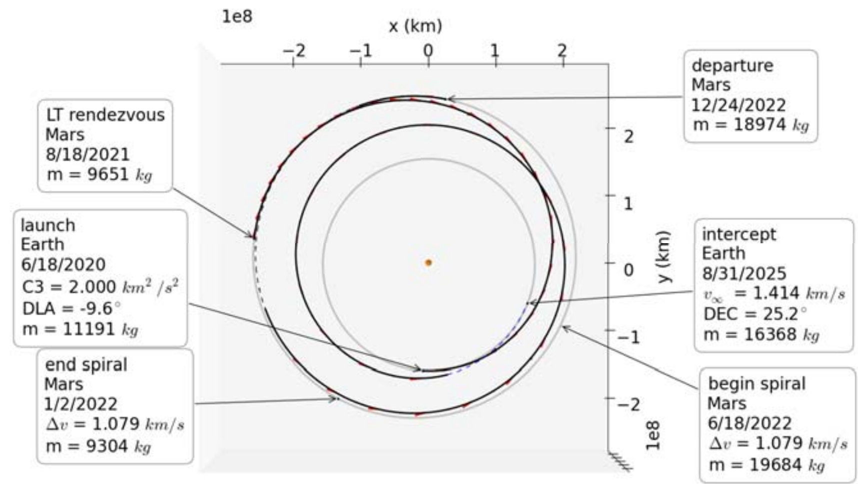
# Deimos Sample Return: Final Generation Trade Space



# Deimos Sample Return: Two Trajectories




A 7-year mission with a 41 kW solar array returns a 4 ton boulder




A 5-year mission with a 68 kW solar array returns a 10 ton boulder

**Both of these candidate missions, and many others, would be of interest to our customers**

# Conclusions

- The low-thrust interplanetary mission and systems design problem may be posed as a multi-objective hybrid optimal control problem
- The combination of a multi-objective discrete NSGA-II outer-loop with a MBH+NLP inner-loop is a very powerful way to explore a mission and systems trade space in an efficient, automated manner
- The algorithm described here has revolutionized the low-thrust interplanetary mission design process at NASA Goddard Space Flight Center
  - We can now study multiple mission design cases simultaneously, limited only by available computing power
  - Mission design engineers can now spend more time with the customer and with spacecraft hardware engineers so that we can fully understand the scientific and engineering context of our work
  - Good mission ideas are much less likely to be rejected due to lack of time to work on mission design, and bad ideas are much more likely to be rejected before they consume too many resources
- Skilled analysts are expensive. With a multi-objective HOCP automaton, analysts can focus on understanding the customer's needs and the spacecraft's capabilities and also detailed design work, leaving repetitive tasks to the computer

# Thank you to our backers:

- NASA Goddard Space Flight Center Independent Research and Development (IRAD) program
- NASA Goddard Space Science Mission Operations (SSMO)
- NASA Graduate Student Researchers Program (GSRP)
- Illinois Space Grant
- Various customers at Goddard

# Thank You

EMTG is available open-source at

https://sourceforge.net/projects/emtg/

# Backup – Tuning Monotonic Basin Hopping

# Tuning Monotonic Basin Hopping (MBH)

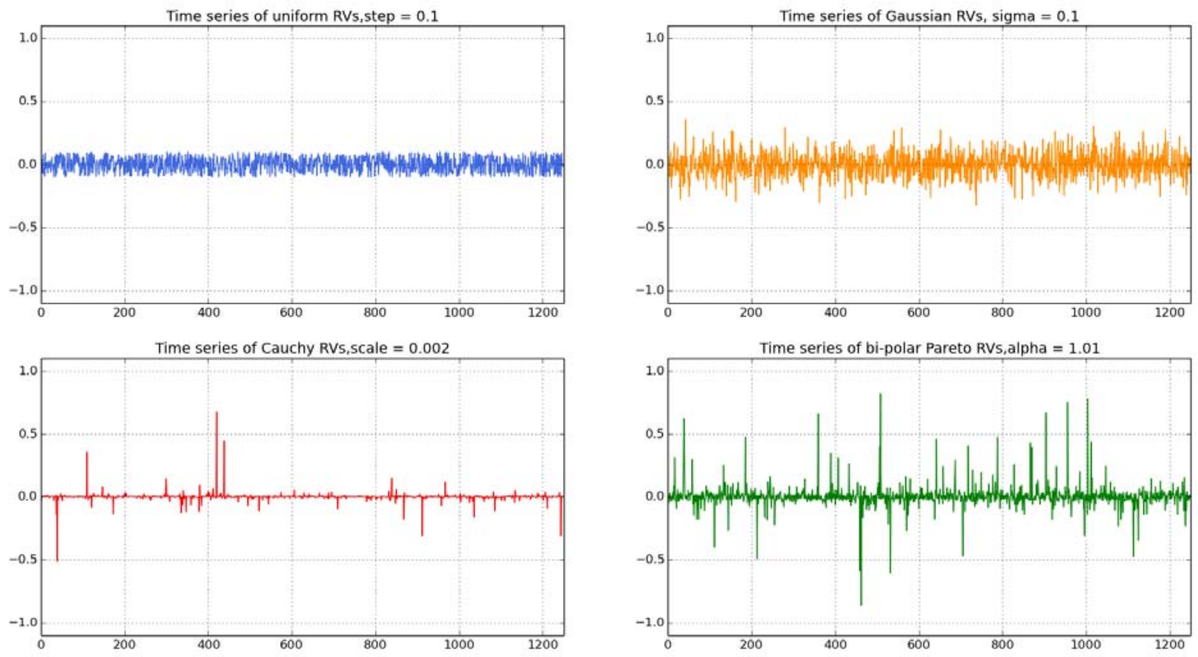
- We examined two components of classical MBH:
  - the random hops are driven by a uniform probability distribution; hops can occur in a ball of some user-defined radius about the current best point
  - There is a concept of "impatience" – a certain number of iterations where the solution does not improve, after which the algorithm resets
- In this work we consider:
  - Alternative probability distributions (Gaussian, Cauchy, Pareto) which have the ability to "hop" over the entire solution space
  - Given the above, that the concept of "impatience" may not be necessary when using alternative probability distributions
- Our objective was to find a version of MBH that would be:
  - Efficient (find better solutions in less time)
  - Robust (work well on highly constrained problems and not be sensitive to tuning parameters)

# Probability Distributions and Their Tuning Parameters

| Distribution | RV Generator | Excursion Parameter |
|---|---|---|
| Uniform | $2\rho(r - 0.5)$ | $\rho$: ball size, impatience |
| Gaussian | $\frac{s}{\sigma\sqrt{2\pi}} e^{-\frac{r^2}{2\sigma^2}}$ | $\sigma$: standard deviation |
| Cauchy | $\rho \tan\big(\pi(r - 0.5)\big)$ | $\rho$: scale |
| Pareto | $\frac{s}{\epsilon} \frac{(\alpha - 1.0)}{\left(\frac{\epsilon}{\epsilon + r}\right)^{-\alpha}}$ | $\alpha$: "parameter" |

$r = uniform(0.0,1.0)$, $s$ is a fair coin flip, $\epsilon = 1.0 * 10^{-13}$

# RV Generators

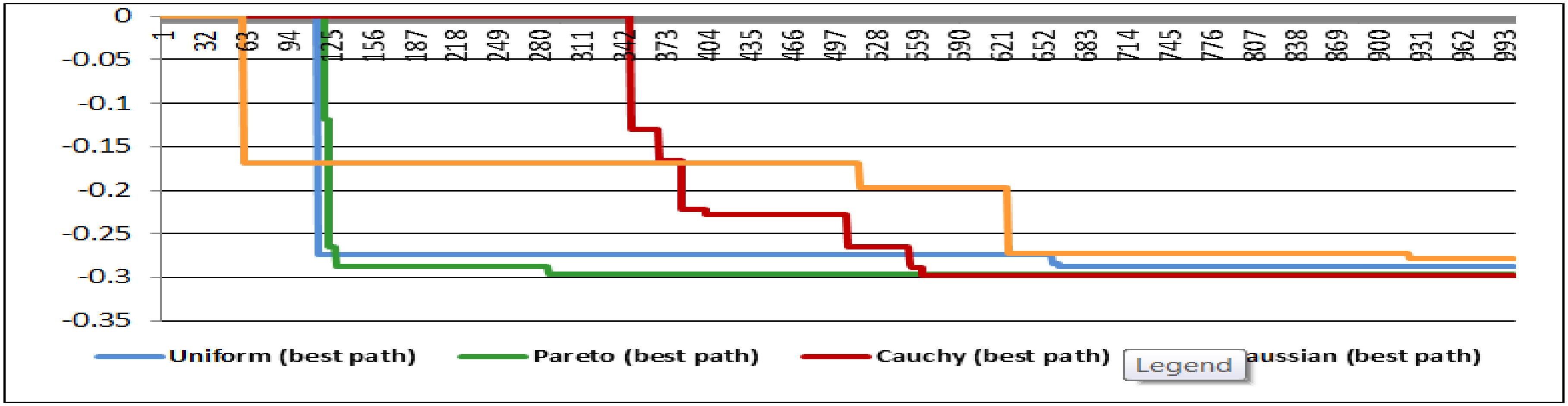


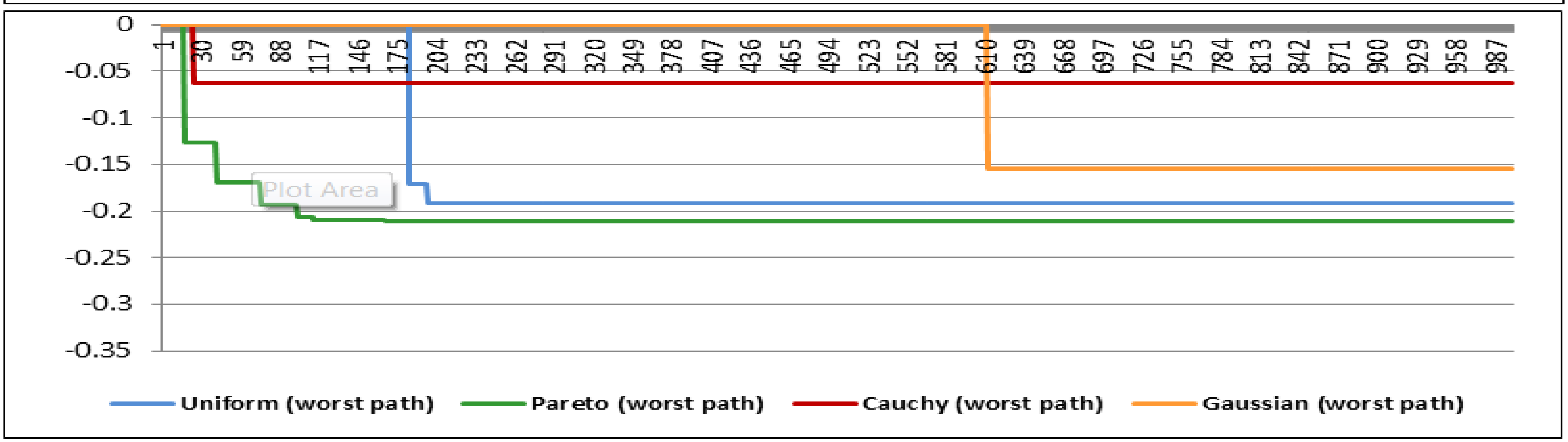

We want a distribution which takes lots of small steps to "exploit" the local region but also takes frequent large steps to "explore" the rest of the space.

# The Experiment

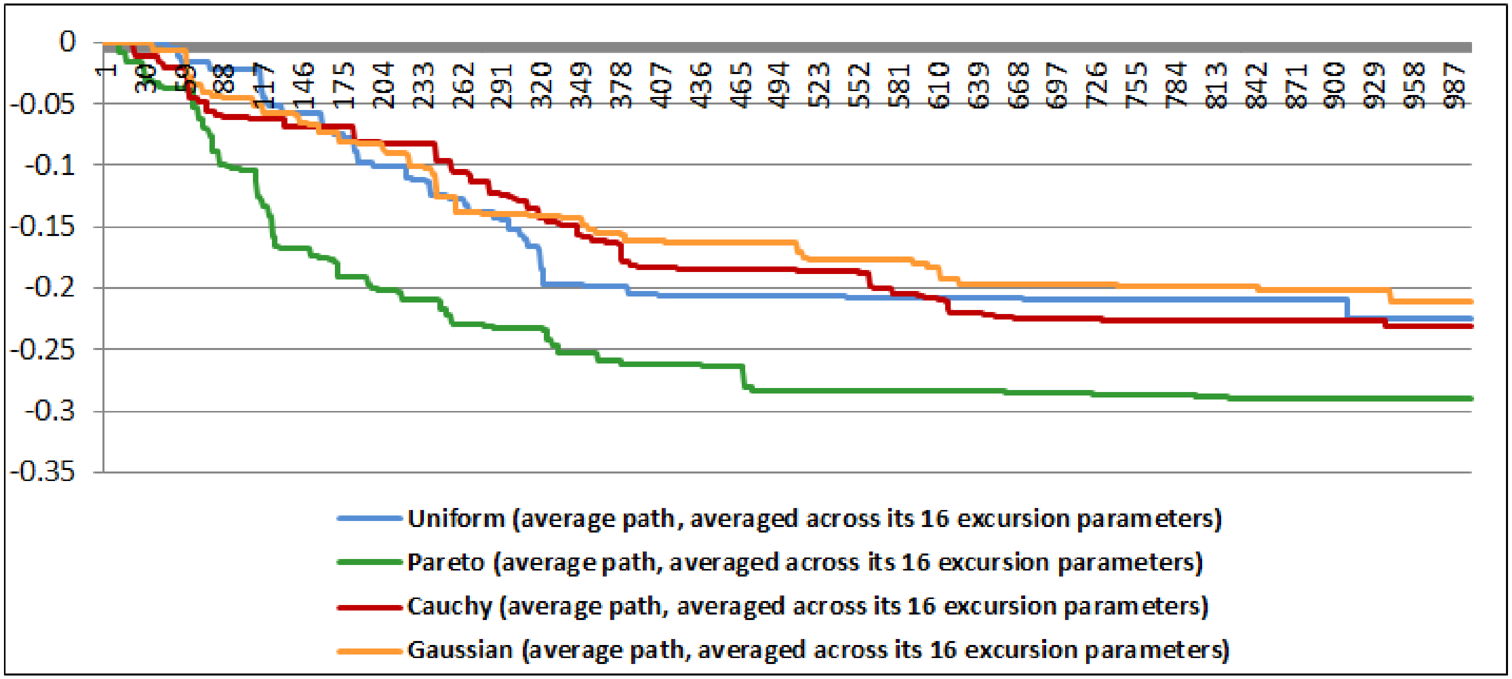
- 16 four-day (10000 step) runs of EMTG were conducted for each distribution
- Each of the 16 runs had a different value of the excursion parameter
- Impatience was turned off, i.e. MBH was never allowed to reset during the experiment
  - This was necessary to see how effectively each distribution could random-walk around the decision space

# Results – Best and Worst Path



# Results – Average Performance



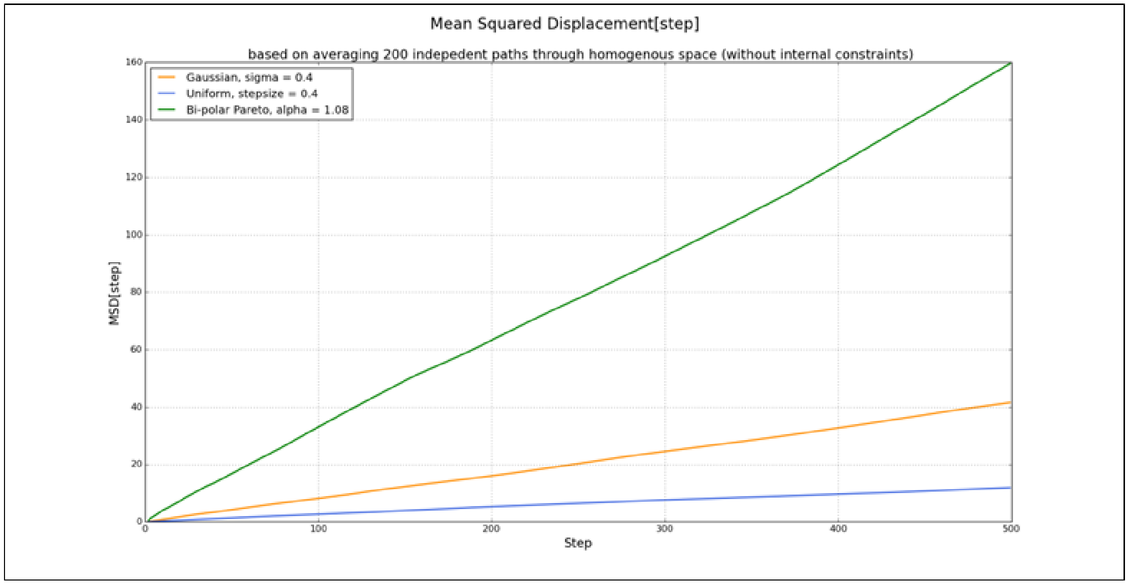

Pareto-driven MBH is most efficent (better solution in less time) and most robust (insensitive to tuning parameters)

# Why?

- Random walks (RWs) can be compared in terms of mean squared displacement (MSD)
  - A higher MSD means that an RW travels the problem space faster and more thoroughly than a lower MSD
- MSD can be used to describe RWs as diffusions through media
- In diffusion through homogeneous media (i.e. unconstrained problem spaces), RWs driven by independent identically distributed (i.i.d.) distributions with finite variance are considered “normally diffusive”
  - MSD proportional to the number of steps
- RWs driven by i.i.d. distributions with infinite variance are “super-diffusive”
  - MSD proportional to the number of steps raised to some power

# Why? Continued…

- In a simplified test problem the Pareto RW is super-diffusive while the uniform and Gaussian RWs are normally diffusive (or just barely super-diffusive for Gaussian)
- It is difficult to plot MSD of the Cauchy RW on the same graph because Cauchy distributions do not have a mean



# What about constraints?

- Constraints introduce serial negative auto-correlations
  - Constraint effectively restricts RW from moving in a certain direction, i.e. into the constraint
- Stochastic global search in constrained problem spaces can be described as diffusions through in-homogenous media
- When constraints are added to the simplified test problem, the uniform and Gaussian RWs become sub-diffusive but the Pareto distribution is still super-diffusive