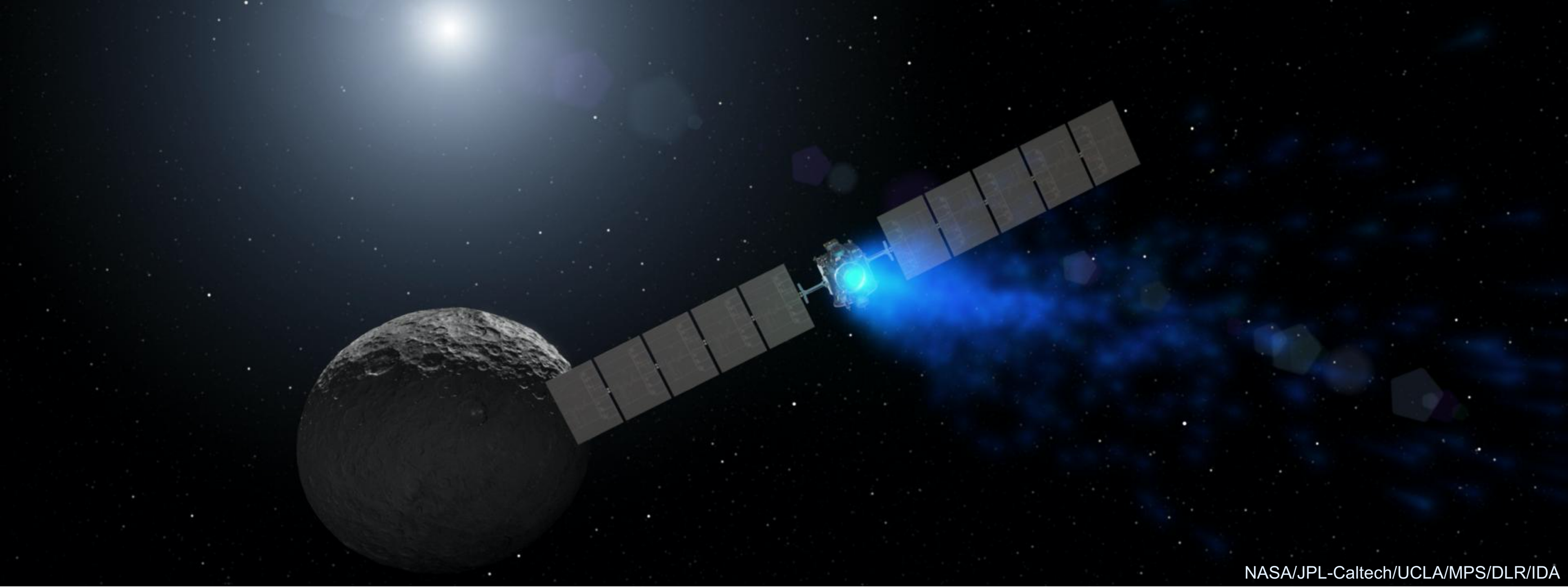


Virtual Research Presentation Conference

Low-Thrust Control Parameterization for Fast, Robust Optimization

Principal Investigator: Zubin Olikara (392)
Program: Spontaneous Concept
Presentation RPC-115

Jet Propulsion Laboratory
California Institute of Technology

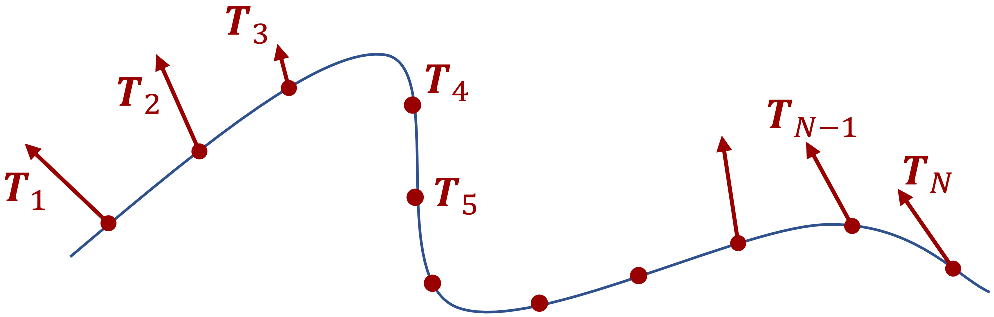
Tutorial introduction

- Electric propulsion can be mission enabling technology
  - Applicable from cubesats to flagship missions
  - MaSMi and MEP thruster development at JPL
- Low-thrust trajectories have extended periods of thrusting
  - Optimization can be more complex than trajectories with a few high-thrust maneuvers
  - Design space searches can require years of CPU time
- Goal: incorporate indirect optimization ideas and new control representations into a direct optimization framework in order to
  - Speed up convergence to optimal trajectory
  - Converge more reliably

Problem description

Direct methods:

Discretize trajectory and have optimizer determine all thrust vectors
(large-scale nonlinear programming (NLP) problem ➞ would like to reduce size, solve faster)



- Easy to add other optimization parameters (e.g., launch energy)
- Easy to impose constraints (e.g., flyby conditions)

For minimum-fuel transfers, common to have optimal coast arcs, but:

$$\dot{m} = \frac{\|\boldsymbol{T}\|}{T_{\max}} \cdot \dot{m}_{\max}$$ not differentiable when $\boldsymbol{T} = \boldsymbol{0}$;
Workaround: add fictitious leak ➞ $$\frac{\sqrt{\|\boldsymbol{T}\|^2 + \delta}}{T_{\max}} \cdot \dot{m}_{\max}$$ differentiable but poorly conditioned for small δ

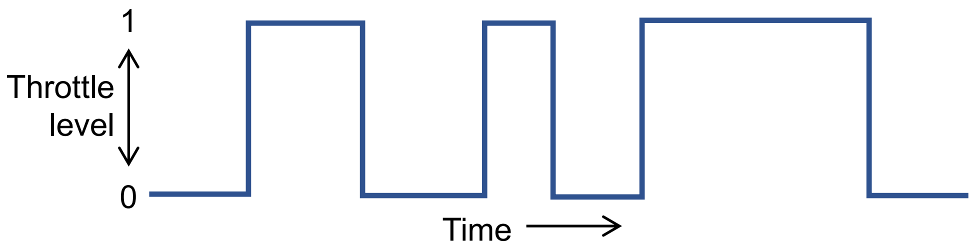
[MALTO ADD, 2013]

★ Methods based on Bellman's principle of optimality often uses mass leak as well

Problem description

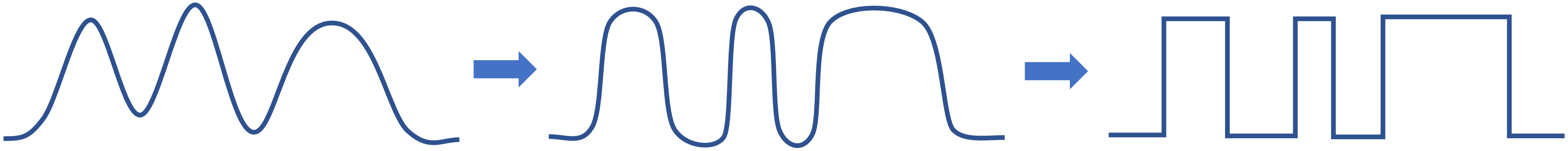
Indirect methods:

Propagate dynamics of states and auxiliary variables ("costates");
Solve boundary-value problem rather than NLP problem



- Challenging to optimize additional parameters or directly impose constraints
- Convergence can be quite sensitive to initial guess

Dynamics not differentiable when engine turns on or off (poor numerical behavior);
Workaround: add smoothing that is progressively reduced (re-solving problem)



[Bertrand & Epenoy, 2002]

Methodology

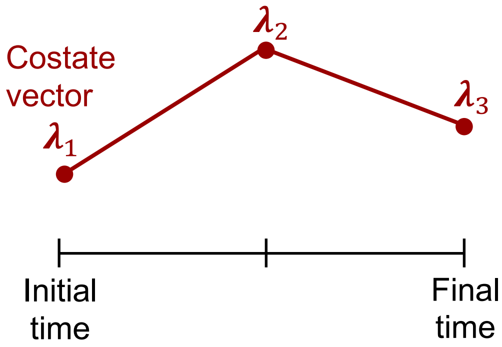
- **How to represent 6-dimensional spacecraft state?**
  - Cartesian elements:
    3 positions, 3 velocities, all varying quickly
  - ✓ Modified equinoctial elements:
    5 elements that change slowly, 1 that changes quickly
    - Like Keplerian elements (a, e, ω, Ω, i, ν) but fewer singularities
    - Their costates fit in well with this study's optimization approach

Makes optimizer's life easier

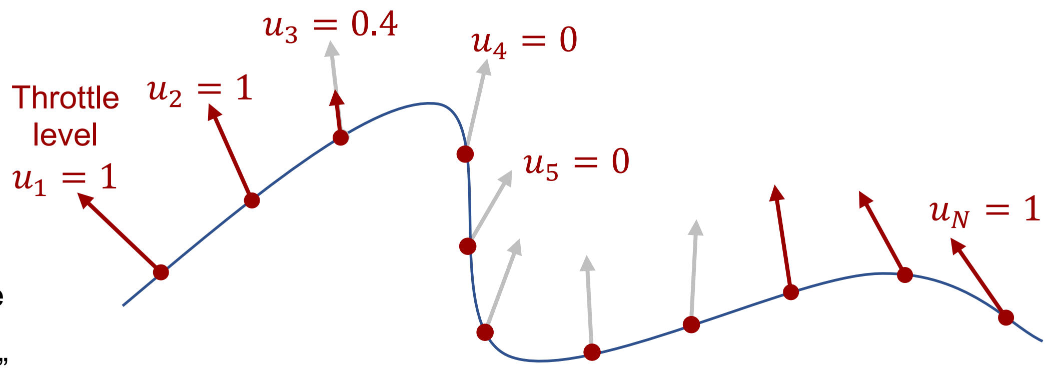
- **How to discretize trajectory?**
  - Shooting approach: propagate trajectory; need to drive a few "large" errors to zero
  - ✓ Collocation approach: represent trajectory as piecewise polynomial, enforce EOMs; need to drive many "small" errors to zero

For related approach with averaged many-rev trajectories, see [ZO, 2018]

Methodology



Costates encode thrust direction via "primer vector"



Which direction to thrust?

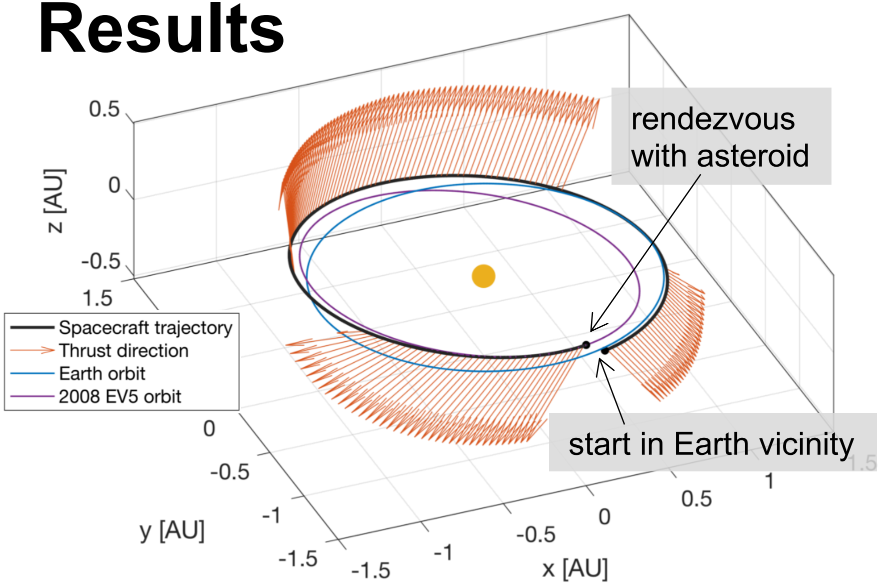
- Use costates as optimization variables (not propagated)
- Use very simple costate interpolation; only adds ~10 degrees of freedom to full optimization problem

Makes optimizer's life easier

How much to thrust?

- Let optimizer decide throttle level u; only 1 parameter to optimize in each thrust interval
- Numerically well behaved
  - No need for mass leak
  - No need for smoothing

Results

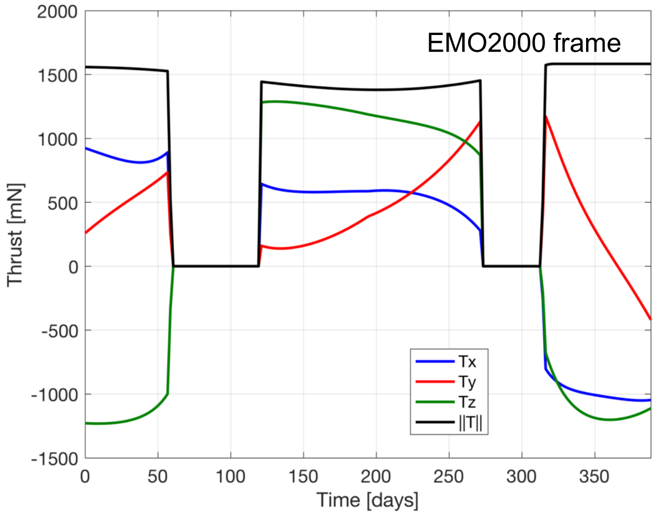


Example scenario:

3 HERMeS engines

47 kW array power @ 1 AU, 5 kW bus power

Inverse square solar array performance



- JPL software Mystic and MColl optimize this transfer in about 25 seconds
- This approach optimizes it as fast as 0.5 seconds with N = 40 throttle intervals, and 3 seconds with N = 160
  - Starting from very simple initial guess
  - Propellant cost accurately estimated even with coarse N

Not a full apples-to-apples comparison, but promising results

Results

- Also tried test case set [Lantoine & Russell, 2012], which provides comparison direct, indirect, and differential dynamic programming results
  - All cases (1, 2, 5, 9, 17-rev transfers) converged well using this study's approach including case where direct method failed, indirect method required manual tuning
  - Speedups of ~100x obtained (hardest case: about 30 minutes → 10 seconds)

- Looking forward:
  - Approach provides basis for quickly exploring low-thrust design space, and reliable convergence avoids missing out on attractive transfers
  - Formulation well suited to future extension (new constraints, new cost functions, new dynamical and spacecraft models)

References

- Paul Finlayson and Edward Rinderle, Jr., "Algorithm descriptions for MALTO: the Mission Analysis Low Thrust Optimizer," Revision 7, Jet Propulsion Laboratory (June 2013).
- Regis Bertrand and Richard Epenoy, "New smoothing techniques for solving bang-bang optimal control problems: numerical results and statistical interpretation," *Optimal Control Applications and Methods* **23** (2002): pp. 171-197.
- Zubin Olikara, "Framework for optimizing many-revolution low-thrust transfers," AAS/AIAA Astrodynamics Specialist Conference, Snowbird, Utah (August 2018).
- Daniel Grebow and Thomas Pavlak, "MColl: MONTE collocation trajectory design tool," AAS/AIAA Astrodynamics Specialist Conference, Stevenson, Washington (August 2017).
- Gregory Lantoine and Ryan Russell, "A hybrid differential dynamic programming algorithm for constrained optimal control problems. Part 2: Application," *Journal of Optimization Theory and Applications* **154** (2012): pp. 418-442.