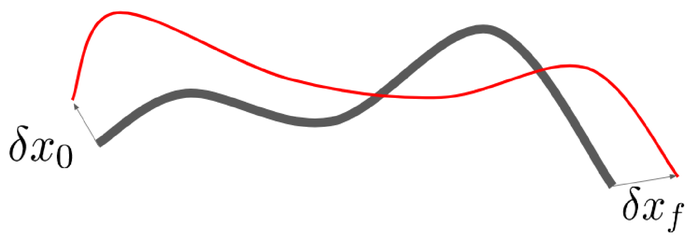
Precomputation and Interpolation of the Matrizant for Starshade Slewing

Jackson Kulik [1] Dmitry Savransky [2]

[1]Cornell University, Center for Applied Mathematics [2]Cornell University, Department of Mechanical and Aerospace Engineering

The State Transition Matrix

The state transition matrix (STM) or matrizant, $\Phi(t_f, t_0)$, gives a linear approximation to the behavior of a dynamical system in the vicinity of some reference trajectory.



Variations in initial conditions, $\delta\mathbf{x}_0$, at time t_0 are mapped into variations in the final state, $\delta\mathbf{x}_f$, at time t_f.

$$\delta\mathbf{x}_f \approx \Phi(t_f, t_0)\delta\mathbf{x}_0$$

Calculating the STM

Suppose a dynamical system for the state $\mathbf{x} \in \mathbb{R}^n$ is described by the vector field $\mathbf{F}$

$$\frac{d}{dt}\mathbf{x} = \mathbf{F}(\mathbf{x})$$

The state transition matrix associated with a given trajectory arises from simultaneous integration of the first order variational equations

$$\frac{d\Phi(t,0)}{dt} = \frac{\partial \mathbf{F}(\mathbf{x})}{\partial \mathbf{x}}\Phi(t,0), \quad \Phi(0,0) = I_n$$

- No analytical solutions exist in the three body problem
- Numerical integration required to calculate STM

Impulsive Relative Transfers

Suppose the state, $\mathbf{x}$, is partitioned into position, $\mathbf{r}$, and velocity, $\mathbf{v}$. The final relative position $\delta\mathbf{r}_f$ can be expressed in terms of initial variations.

$$\delta\mathbf{r}_f \approx \Phi^{\mathbf{r}}_{\mathbf{r}}(t_f, t_0)\delta\mathbf{r}_0 + \Phi^{\mathbf{r}}_{\mathbf{v}}(t_f, t_0)\delta\mathbf{v}_0$$

Given an initial relative position, r_0, and a desired final relative position, r_f, the initial relative velocity to achieve that final position is given by

$$\delta\mathbf{v}_0 \approx \left(\Phi^{\mathbf{r}}_{\mathbf{v}}(t_f, t_0)\right)^{-1}\left(\delta\mathbf{r}_f - \Phi^{\mathbf{r}}_{\mathbf{r}}(t_f, t_0)\delta\mathbf{r}_0\right)$$

Final relative velocity can be calculated from the STM

$$\delta\mathbf{v}_f \approx \Phi^{\mathbf{v}}_{\mathbf{v}}(t_f, t_0)\delta\mathbf{v}_0 + \Phi^{\mathbf{v}}_{\mathbf{r}}(t_f, t_0)\delta\mathbf{r}_0$$

- initial and final $\delta\mathbf{v}$ impulses calculated from difference with desired values
- desired final velocity is zero relative inertial velocity relative to telescope
- similar technique to find initial costates for continuous thrust optimal control

References

[1] Egemen Kolemen and N Jeremy Kasdin. Optimization of an occulter-based extrasolar-planet-imaging mission. *Journal of guidance, control, and dynamics*, 35(1):172–185, 2012.

[2] Gabriel J Soto, Dmitry Savransky, Daniel Garrett, and Christian Delacroix. Parameterizing the search space of starshade fuel costs for optimal observation schedules. *Journal of Guidance, Control, and Dynamics*, 42(12):2671–2676, 2019.

Acknowledgments

The authors thank the DOD for its generous support through the NDSEG fellowship.

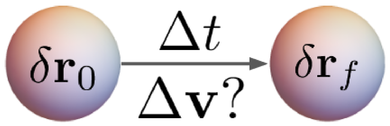
Starshades for Exoplanet Direct Imaging

A starshade satellite **formation flying** in coordination near a space telescope

- Enables direct imaging of exoplanets by blocking out starlight
- Requires consideration of **cost to maneuver starshade** for each observation [1]
- Assumes the telescope flies on a known stable orbit in Sun-Earth space

Optimal mission scheduling

- Balances potential science objectives with limited fuel to maneuver
- Takes potentially millions of cost function evaluations to approximate solution
- Requires **fast evaluation of the cost function** for a proposed mission schedule



Evaluating the maneuever cost function $\Delta\mathbf{v}$

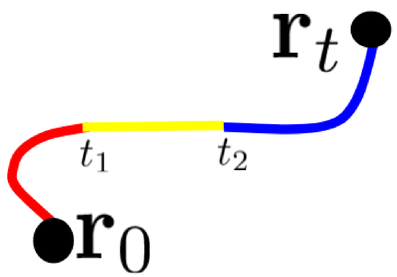
- Requires solution of a boundary value problem with many design variables
  - Transfer time Δt
  - Initial and final position along sphere around telescope $\delta\mathbf{r}_0$ and $\delta\mathbf{r}_f$
- Takes multiple numerical integrations of **42 equation system** to solve exactly
- Employs **costly numerical integration to find STM** even for linear approximation
- Existing interpolation approaches ignore potentially important spatial variables [2]

Our approach

- Avoid online costs by **precomputing STMs** along the telescope trajectory
- Combine and interpolate STMs online to solve for any maneuver cost

Composing STMs

Cocycle conditions allow for STMs along consecutive segments to be **composed into a single STM** along the combined trajectory.

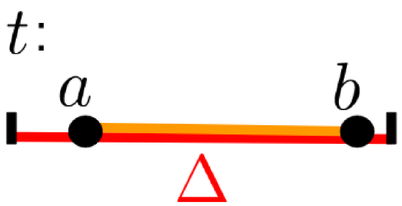


$$\Phi(t,0) = \Phi(t,t_2)\Phi(t_2,t_1)\Phi(t_1,0)$$

Interpolation of the STM

Let $(a,b) \subseteq \Delta$, then an **entrywise linear interpolant** of Φ is given by

$$\Phi(b,a) \approx I + \frac{b-a}{|\Delta|}(\Phi(\Delta) - I)$$



Error in any matrix norm is quadratic in $|\Delta|$.

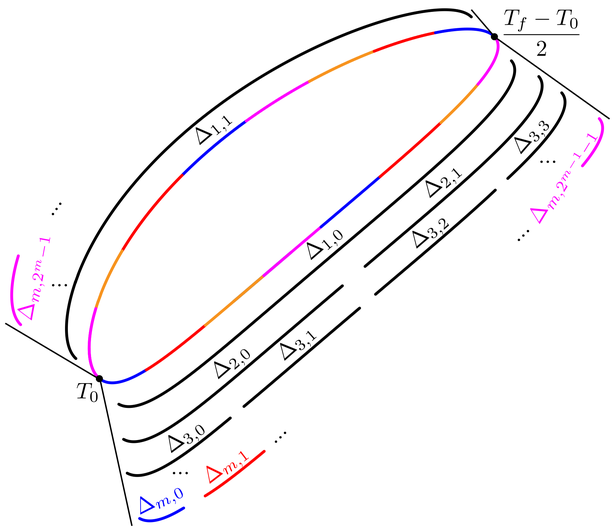
Precomputation and Online Computation

To quickly calculate the STM along any portion of the telescope's periodic orbit

Perform the following **ahead of time**

- Divide the trajectory into 2^m subintervals $\Delta_{m,j}$ evenly spaced from T_0 to T_f
- **Precompute and store the STM** $\Phi(\Delta_{m,j})$ along each subset of the trajectory
- Build up and store $\Phi(\Delta_{i,j})$ for i from 0 to $m-1$ using cocycle conditions

$$\Delta_{m,j} = \left(T_0 + \frac{T_f - T_0}{2^m}j, T_0 + \frac{T_f - T_0}{2^m}(j+1)\right)$$



During schedule optimization online

- **Follow logarithmic search** algorithm to build up STM for desired time interval
- Use cocycle conditions on precomputed STMs contained in desired time range
- Interpolate on end points of time intervals, and then use cocycle conditions
- Calculate approximate transfer costs using the resulting STM

Benefits

- Less than $2m$ matrix multiplications required for computation
- $2^{m+1} - 1$ matrices with 36 floating point entries each stored.
- On the order of **kilobytes and microseconds** for m chosen to keep interpolation error on the order of 0.1%

Delta-V Error

- Compare $\Delta\mathbf{v}$ calculated with exact STM vs an interpolated STM.
- Along a 6 month L2 Halo orbit
- 5000 example transfers to and from uniform random look vectors
- Interpolated STM has 1 day finest discretization, and 0.5 day interpolation is used on either side to demonstrate worst case error
- **Delta-v error is less than 0.05%**, and diminishes for longer transfers
- **Accurate delta-v estimates** with low memory requirements (**50Kb**) for **any nearby relative transfer**