Decentralized Model Predictive Control of Swarms of Spacecraft

Daniel Morgan[1], Soon-Jo Chung[1], and Fred Y. Hadaegh[2]
Presented by Saptarshi Bandyopadhyay[1]

[1]University of Illinois at Urbana-Champaign
[2]Jet Propulsion Laboratory

Outline

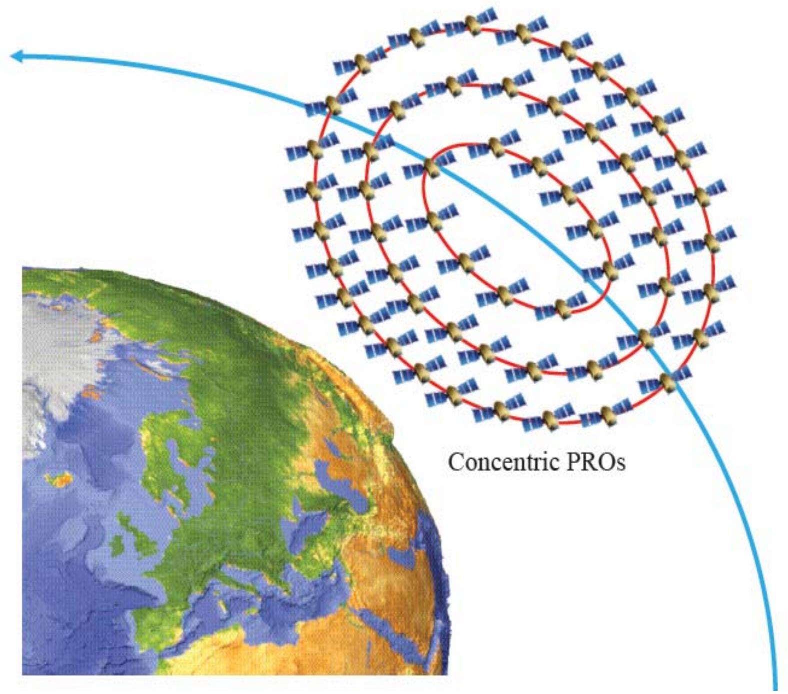
- Mission objectives
- Swarm reconfiguration
- Conversion to a convex program
- MPC-SCP formulation
- Numerical simulation
- Conclusions

JPL

ILLINOIS
UNIVERSITY OF ILLINOIS AT URBANA-CHAMPAIGN

Mission Objective

- To create a swarm of 1000 “fully functional” femtosats (~100g) which can work together in order to perform certain tasks better than a typical monolithic spacecraft



Swarm Reconfiguration Goals

- Transfer spacecraft from one J_2-invariant passive relative orbit to another
- Minimize fuel usage
- Avoid collisions
- Decentralize communication and computation requirements

Optimal Control Problem

- Objective: Minimize the total fuel used by all of the spacecraft

$$\min_{\mathbf{u}_j, j=1,\ldots,N} \sum_{j=1}^{N} \int_0^{t_f} \|\mathbf{u}_j(t)\| dt$$

- Constraints
  - Dynamics (nonlinear with J_2) [Xu, Wang, 2008]

$$\dot{\mathbf{x}}_j = \mathbf{f}(\mathbf{x}_j, \text{œ}) + B\mathbf{u}_j$$

  - Maximum allowable acceleration: $\|\mathbf{u}_j(t)\| \leq U_{\max}$
  - Collision avoidance: $\|C[\mathbf{x}_j(t) - \mathbf{x}_i(t)]\| \geq R_{\text{col}}$
  - Initial and terminal states: $\mathbf{x}_j(0) = \mathbf{x}_{j,0}$

$$\mathbf{x}_j(t_f) = \mathbf{x}_{j,f}$$

JPL

ILLINOIS
UNIVERSITY OF ILLINOIS AT URBANA-CHAMPAIGN

Convexifying Dynamics Constraints

- Linearize dynamics to get linear ODE
- Discretize problem
  - Results in a finite dimensional optimization problem
  - Dynamics equations become linear equality constraints, which are acceptable in convex programming

$$\mathbf{x}_j[k+1] = A_d(\bar{\mathbf{x}}_j, \boldsymbol{\oe})\mathbf{x}_j[k] + B_d\mathbf{u}_j[k] + c_d(\bar{\mathbf{x}}_j, \boldsymbol{\oe})$$

Convexifying Collision Avoidance Constraints

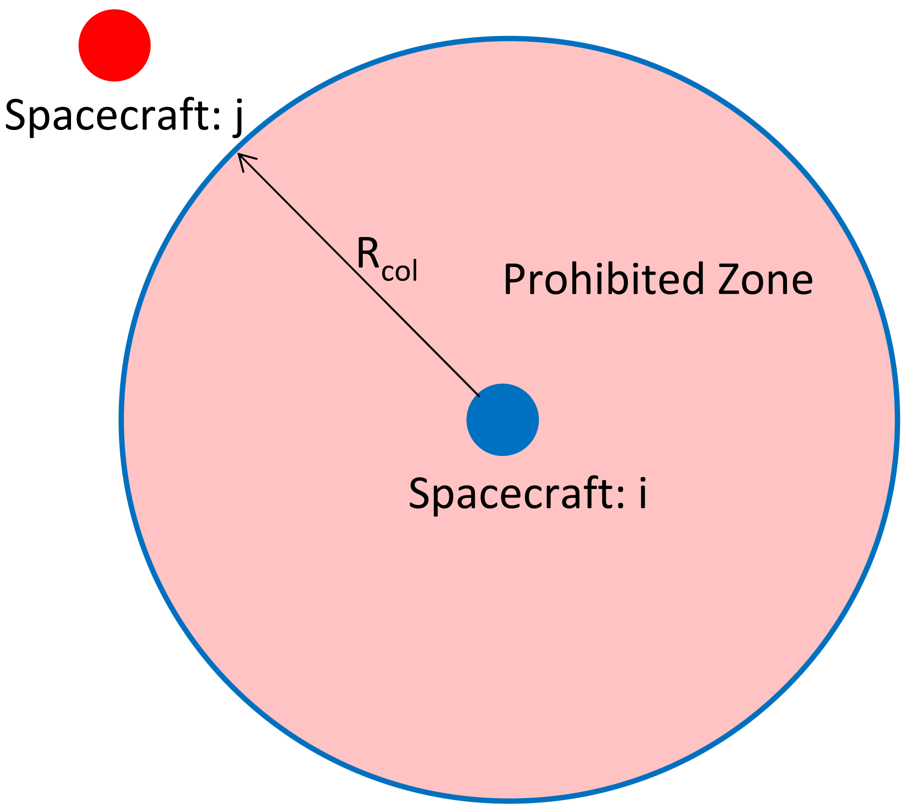
- Develop an affine approximation for the convex function in the inequality

$$||C(\mathbf{x}_j[k] - \mathbf{x}_i[k])|| \geq R_{\text{col}}$$

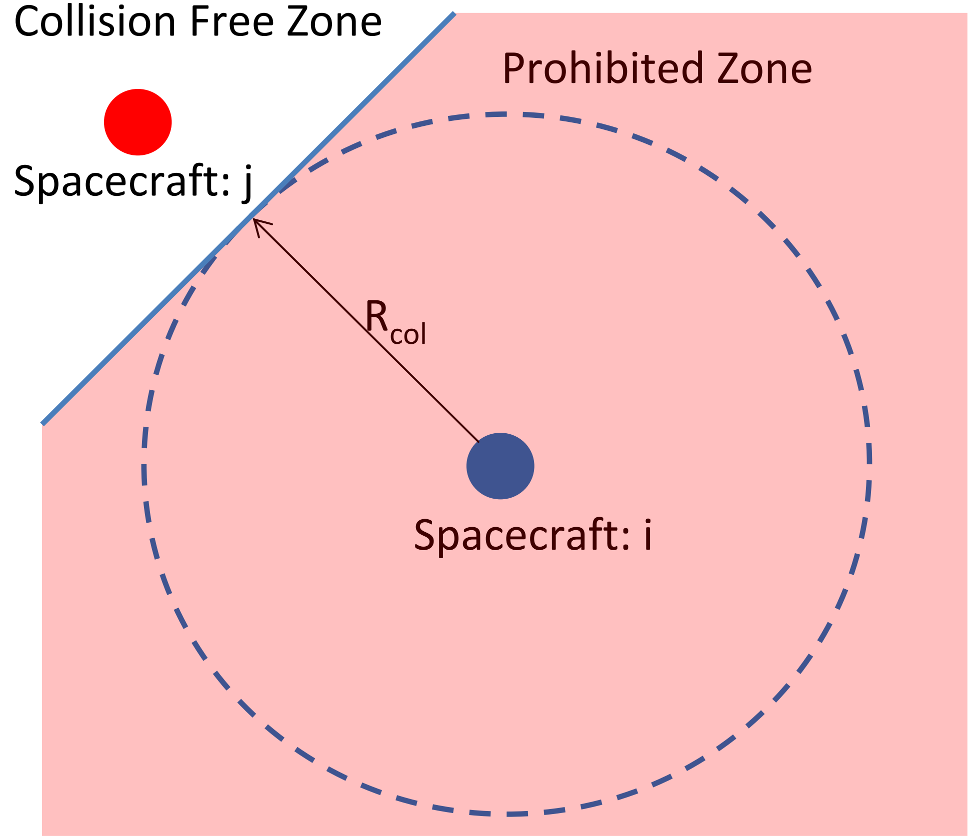
- Must be a sufficient condition for the original, nonconvex constraint
- The resulting condition is

$$(\bar{\mathbf{x}}_j[k] - \bar{\mathbf{x}}_i[k])^T C^T C(\mathbf{x}_j[k] - \mathbf{x}_i[k]) \geq R_{\text{col}} ||C(\bar{\mathbf{x}}_j[k] - \bar{\mathbf{x}}_i[k])||$$

Convex Collision Avoidance Constraints



Original constraint



Convexified constraint

Decoupling the Collision Avoidance Constraints

- Current collision avoidance constraints involve the positions of multiple spacecraft

$$(\bar{\mathbf{x}}_j[k] - \bar{\mathbf{x}}_i[k])^T C^T C(\mathbf{x}_j[k] - \mathbf{x}_i[k]) \geq R_{\text{col}} \| C(\bar{\mathbf{x}}_j[k] - \bar{\mathbf{x}}_i[k]) \|$$

- Approximate other spacecraft's location using the preceding iteration's trajectories

$$(\bar{\mathbf{x}}_j[k] - \bar{\mathbf{x}}_i[k])^T C^T C(\mathbf{x}_j[k] - \bar{\mathbf{x}}_i[k]) \geq R_{\text{col}} \| C(\bar{\mathbf{x}}_j[k] - \bar{\mathbf{x}}_i[k]) \|$$

- The constraints are now decoupled and each spacecraft can solve its own convex program

JPL

ILLINOIS
UNIVERSITY OF ILLINOIS AT URBANA-CHAMPAIGN

Convex Programming Formulation

- Objective: Minimize the fuel usage

$$\min_{\mathbf{u}_j} \sum_{k=0}^{T-1} \|\mathbf{u}_j[k]\| \Delta t$$

- Constraints
  - Dynamics:

$$\mathbf{x}_j[k+1] = A_d(\bar{\mathbf{x}}_j, \text{\oe})\mathbf{x}_j[k] + B_d\mathbf{u}_j[k] + c_d(\bar{\mathbf{x}}_j, \text{\oe})$$

  - Maximum allowable acceleration: $\|\mathbf{u}_j[k]\| \leq U_{\max}$
  - Collision avoidance:

$$(\bar{\mathbf{x}}_j[k] - \bar{\mathbf{x}}_i[k])^T C^T C(\mathbf{x}_j[k] - \bar{\mathbf{x}}_i[k]) \geq R_{\text{col}} \|C(\bar{\mathbf{x}}_j[k] - \bar{\mathbf{x}}_i[k])\|$$

  - Initial and terminal states: $\mathbf{x}_j[0] = \mathbf{x}_{j,0} \quad \mathbf{x}_j[T] = \mathbf{x}_{j,f}$

JPL

ILLINOIS
1867
UNIVERSITY OF ILLINOIS AT URBANA-CHAMPAIGN

SCP Method

- Method for iteratively solving a nonconvex problem using convex programming
  - Calculate initial trajectories
  - Repeat until the process converges
    - Convexify the constraints about these trajectories
    - Solve the convex programming problem
    - Update the initial trajectories using the solution to the convex program

MPC Formulation

- Optimization begins at the current time and runs for a finite horizon
- Collision avoidance is only considered for pairs of spacecraft that are near each other
- Terminal constraint is replaced by a terminal cost function

$$\min_{\mathbf{u}_j} \sum_{k=k_0}^{k_0+T_H-1} \|\mathbf{u}_j[k]\|\Delta t + h_j(\mathbf{x}_j[k_0+T_H], k_0+T_H)$$

$$\mathbf{x}_j[k+1] = A_d(\bar{\mathbf{x}}_j, œ)\mathbf{x}_j[k] + B_d\mathbf{u}_j[k] + c_d(\bar{\mathbf{x}}_j, œ), \quad k = k_0, \ldots, k_0+T_H-1$$

$$(\bar{\mathbf{x}}_j[k] - \bar{\mathbf{x}}_i[k])^T C^T C(\mathbf{x}_j[k] - \bar{\mathbf{x}}_i[k]) \geq R_{\mathrm{col}}\|C(\bar{\mathbf{x}}_j[k] - \bar{\mathbf{x}}_i[k])\|$$

$$k = k_0, \ldots, k_0+T_H, \quad \{i,j\} : i \in \mathcal{N}_j[k_0], \quad \mathcal{N}_j[k_0] = \{i | i > j, \|\mathbf{x}_j[k_0] - \mathbf{x}_i[k_0]\| \leq R_{\mathrm{comm}}\}$$

$$\|\mathbf{u}_j[k]\| \leq U_{\max}, \quad k = k_0, \ldots, k_0+T_H-1$$

$$\mathbf{x}_j[k_0] = \mathbf{x}_{j,\mathrm{MPC}}$$

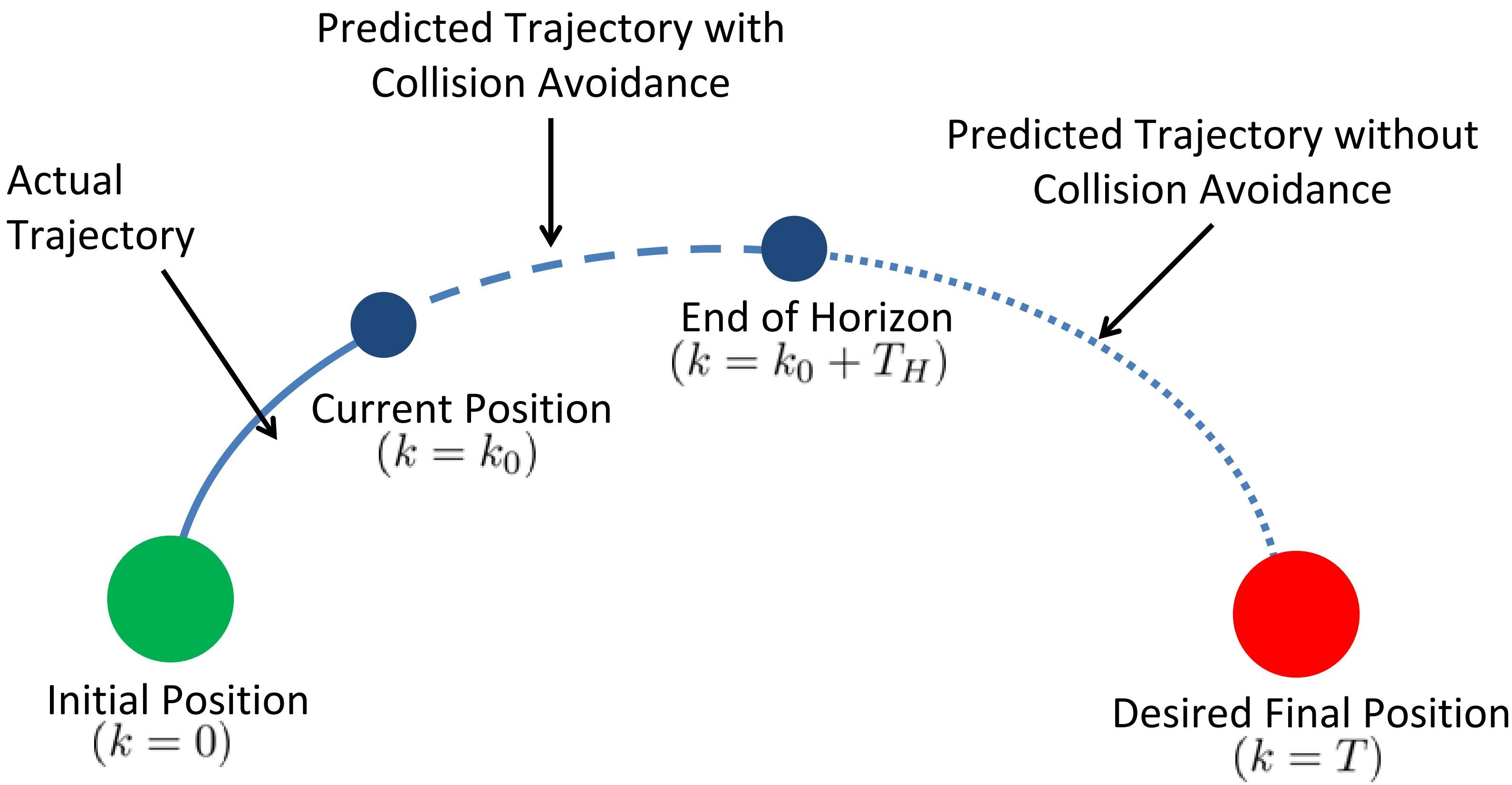
JPL

ILLINOIS
UNIVERSITY OF ILLINOIS AT URBANA-CHAMPAIGN

Terminal Cost Function

- Requirements
  - Accurately represents the fuel required to get from the state at the end of the horizon to the terminal state at the terminal time
  - Easily calculated and convex
  - Ensures convergence of the MPC trajectories
- Resulting terminal cost function
  - Solve the convex program from the horizon to the final time without considering collision avoidance

Stages of the MPC Trajectory



- Note: When the horizon extends beyond the final time, collision avoidance is considered at every time step.

Simulation Parameters

- Chief orbit: [6878 km, 0, 45°, 60°, 0°, 0°]
- Transfer time: 5677 s (one orbit)
- Initial and terminal positions are randomly generated
- Initial and terminal velocities are determined using J_2-invariant initial conditions [Morgan, Chung, 2012]
- Convex optimizations performed using CVX [Grant, Boyd, 2012]

JPL

ILLINOIS
1867
UNIVERSITY OF ILLINOIS AT URBANA-CHAMPAIGN

Open-Loop SCP vs. MPC-SCP

Open-Loop SCP

- Time step: 60s
- Collision radius: 150 m
- All-to-all communication

MPC-SCP

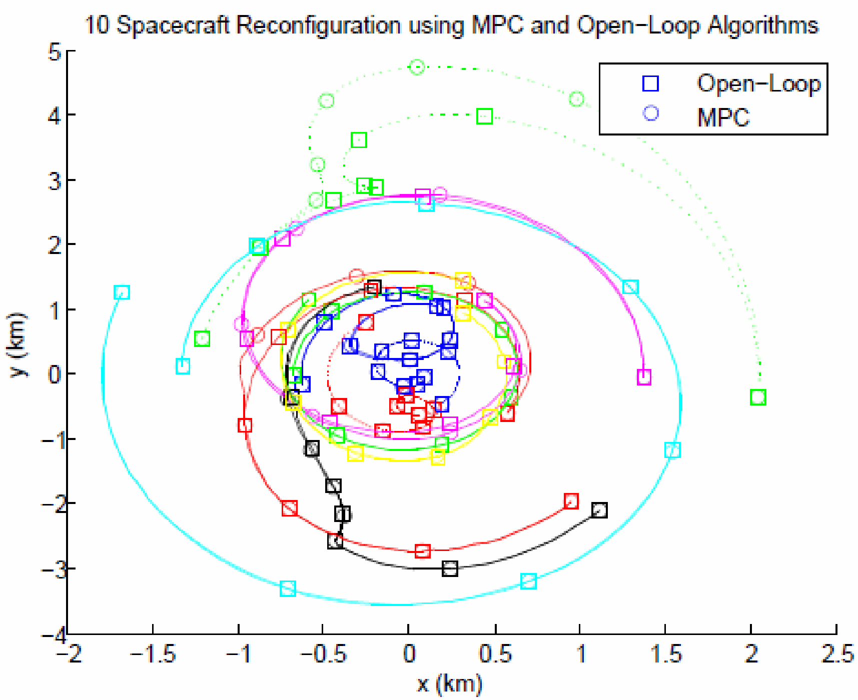
- Time step: 60s
- MPC horizon: 180 s
- Collision radius: 150 m
- Comm. radius: 2 km

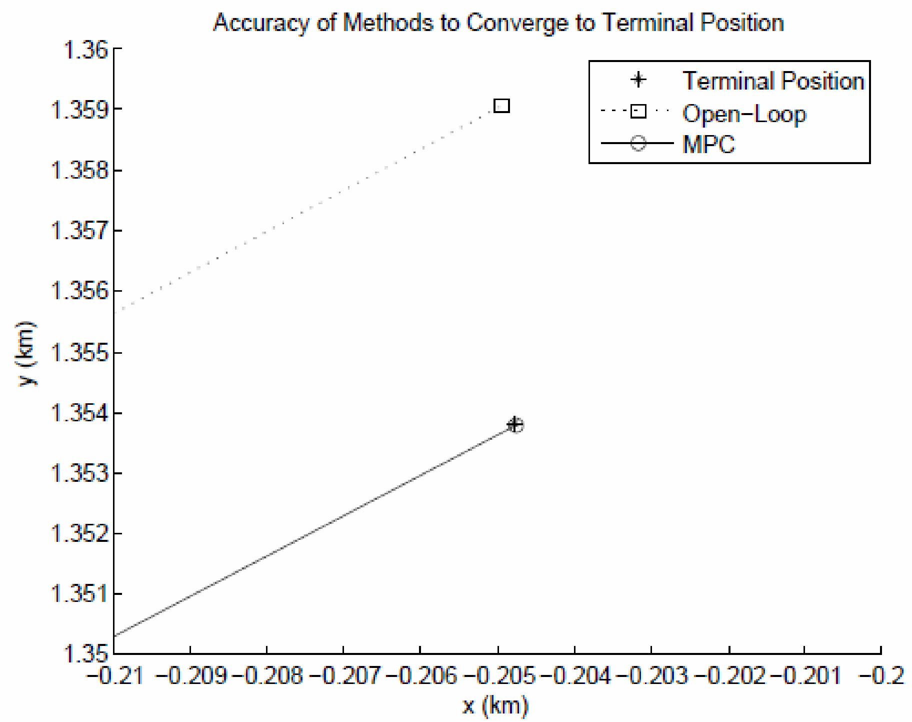
	Average Algorithm Performance	
Algorithm	Fuel Cost (m/s)	Terminal Error (m)
MPC-SCP Algorithm	2.242	0.032
Open-Loop SCP Optimization	2.100	2.293

JPL

ILLINOIS
UNIVERSITY OF ILLINOIS AT URBANA-CHAMPAIGN

Comparison of MPC-SCP and Open-Loop SCP Trajectories





- Terminal error is reduced from 5.2 m to 4.1 cm when MPC-SCP is applied

JPL

ILLINOIS
UNIVERSITY OF ILLINOIS AT URBANA-CHAMPAIGN

Single vs. Multiple Time Step MPC-SCP

Single Time Step MPC-SCP

- Time step: 60s
- MPC horizon: 180 s
- Collision radius: 150 m
- Comm. radius: 2 km

Multiple Time Step MPC-SCP

- Time step before horizon: 15s
- Time step after horizon: 60s
- MPC horizon: 180 s
- Collision radius: 150 m
- Comm. radius: 2 km

	Average Algorithm Performance	
Time Step Size	Fuel Cost (m/s)	Terminal Error (mm)
$\Delta t = 60$ s	2.242	12.974
$\Delta t_1 = 15$ s, $\Delta t_2 = 60$ s	2.178	0.709

Conclusions

- SCP has several important qualities that are useful for real-time, collision avoidance of spacecraft
  - Convex programs can be efficiently solved for a global minimum
  - Using several iterations with SCP decreases the approximation error that occurred when confexifying the problem
- MPC-SCP further improves upon the SCP method
  - Improves convergence to desired terminal position
  - Requires communication between neighbors only
  - Reduces computational run time of the optimizations

Future Work

- Implement MPC-SCP using better software programs to improve computation times
- Verify algorithms using multivehicle testbed
- Determine optimal terminal assignments

Acknowledgements

- The research was carried out in part at the Jet Propulsion Laboratory, California Institute of Technology, under a contract with the National Aeronautics and Space Administration. © 2013 California Institute of Technology. Government sponsorship acknowledged
- This work was supported by a NASA Office of the Chief Technologists Space Technology Research Fellowship

Questions

JPL

ILLINOIS
UNIVERSITY OF ILLINOIS AT URBANA-CHAMPAIGN

References

- G. Xu, D. Wang, Nonlinear dynamic equations of satellite relative motion around an oblate earth, *Journal of Guidance, Control, and Dynamics* 31 (5) (2008) 1521-1524
- D. Morgan, S.-J Chung, L. Blackmore, B. Acikmese, D. Bayard, F. Y. Hadaegh, Swarm Keeping Strategies for Spacecraft under J_2 and Atmospheric Drag Perturbations, *Journal of Guidance, Control, and Dynamics* (2012)
- Grant, M. and Boyd, S., "CVX: Matlab Software for Disciplined Convex Programming (version 1.22)," May 2012, http://cvxr.com/cvx/