Formation Control Strategies for a Separated Spacecraft Interferometer[1]

Andrew Robertson[2], Gokhan Inalhan[3] and Jonathan P. How[4]
Stanford University

Abstract

Formation flying of multiple spacecraft is an enabling technology for future space science missions such as separated spacecraft interferometers. Controllers designed for these multi-vehicle fleets must address many high- and low-level tasks, and will become very complicated for large fleets. As such, this work describes ongoing research to investigate the precise sensing and control of a distributed spacecraft interferometer using a layered approach based on GPS and laser metrology. In particular, it focuses on the design of low-level controllers within several candidate control architectures, and analyzes how well these controllers perform on basic formations. Initial experimental results are presented on a three vehicle formation flying testbed executing station-keeping and rigid body maneuvers.

1 Introduction

Formation flying of multiple spacecraft is an enabling technology for many future space science missions. Of particular interest are the several separated spacecraft interferometer (SSI) missions that have been proposed [1, 2, 3] with the number of vehicles in the fleet ranging from 2 to 20. While the objectives and profiles of these missions are quite diverse, the common theme is that they use a distributed array of highly coordinated satellites to provide a very flexible method of obtaining improved space science observations (longer and reconfigurable baselines). Implementation of this coordinated satellite concept requires tight maintenance and control of the relative distances and phases between the participating satellites. Of course, any ground-based command and control system for relative spacecraft positioning would be very complex and probably would not provide sufficiently rapid corrective commands. Thus, the onboard autonomy of the spacecraft fleet must be developed to support these complex and challenging space science missions.

[1]This work is funded in part under Air Force grant # F49620-99-1-0095, NASA JPL contract # 960941 and NASA GSRP fellowship grant # NGT5-50075

[2]Research Assistant, Dept. of Aeronautics and Astronautics, adr@sun-valley.stanford.edu

[3]Research Assistant, Dept. of Aeronautics and Astronautics, ginalhan@leland.stanford.edu

[4]Assistant Professor, Dept. of Aeronautics and Astronautics, howjo@sun-valley.stanford.edu

Fig. 1: Formation Flying Testbed

Some of the key challenges in this problem are in the design of a fleet control architecture that can perform the high-level (mission management and planning to enable resource allocation across the fleet) and low-level (onboard sensing, autonomous closed-loop relative navigation, and attitude determination) tasks. The primary difficulties are that: 1) with a large fleet, it might not be practical or feasible for any one vehicle to know the entire formation state; 2) the vehicles must work cooperatively to perform the science observations; 3) the differential disturbance environment and nonlinear actuator operations could be uncertain; and 4) the fleet must undergo both resizing and configuration change maneuvers. Although many standard solutions exist for controlling individual spacecraft, the formation flying problem is complicated by the fact that the vehicles must act together to meet the global objectives. Thus it is important to develop a control architecture that provides an efficient control system and that can meet the local and global performance objectives.

This work describes ongoing research to investigate the precise sensing and control of a distributed spacecraft interferometer[4, 5, 6]. In particular, we investigate the control design by developing flexible low-level control algorithms that enable coordination between vehicles (*e.g.,* based on the overall resource state). Our approach is motivated by an early baseline for the New Millennium Interferometer (NMI) mission which required a precisely aligned formation of three spacecraft. Two of the spacecraft would act as collectors that focus the light of a distant star onto the third combiner spacecraft that forms the interference pattern [1]. To form this pattern, the optical paths between the spacecraft

must be controlled to within a fraction of a wavelength of light. A layered control approach has been proposed to achieve this high degree of accuracy using sensors based on the Global Positioning System (GPS) as a coarse sensing layer to measure the relative positions to cm-level so that the more accurate optical metrology systems can be aligned [1, 7]. We have developed a formation flying testbed (FFTB) to analyze the sensing and control design issues for this system. The FFTB is pictured in Fig. 1. In this paper we present some initial results with the FFTB executing a high-level task. In particular, the three vehicle formation executes a complex *rigid body* maneuver that combines both rotation and translation.

2 Formation Control Design

A control system for a SSI typically consists of many elements (see Fig. 2):

Planning algorithm: high-level optimization of maneuvers such as resizing/repointing. A key question is to determine if this calculation is performed centrally or is distributed across the fleet.

Sensing and estimation: an estimation architecture must be chosen to derive the relative position and velocity states from the GPS measurements.

Optical alignment: additional sensing and control levels are needed to meet the extremely precise optical alignment requirements. The optical system's range will most likely be very limited, so the coarse control levels must align the optics. Several optical levels are necessary for the NMI, but only two will be included on the FFTB.

Control architecture: distributes and assigns tasks to each vehicle to achieve the desired global behavior, which changes depending on the mission phase.

Low-level controllers: implements the global tasks at the vehicle level. The primary low-level tasks are to realign the vehicles in the formation between periods of data collection and to maintain the formation during reconfiguration maneuvers. Note that thruster firings would disrupt science data collection, so the station keeping controller must be de-activated while the SSI is functioning. Thus the vehicles must be very precisely aligned to ensure long periods of data collection.

This paper focuses on various choices for the last two items and how they interface with the control system. The interface to the other parts of the system are critical: the control architecture is dependent on the estimation architecture; the planner must interface with the low-level algorithm; and the low-level algorithm must align the vehicles with sufficient accuracy for the optical alignment system to function. These interactions become very challenging as the number of spacecraft increases. So a logical starting point is to consider the station keeping algorithm for a single vehicle, which is examined in section 3. Section 5 then investigates the extension to the case of N vehicles.

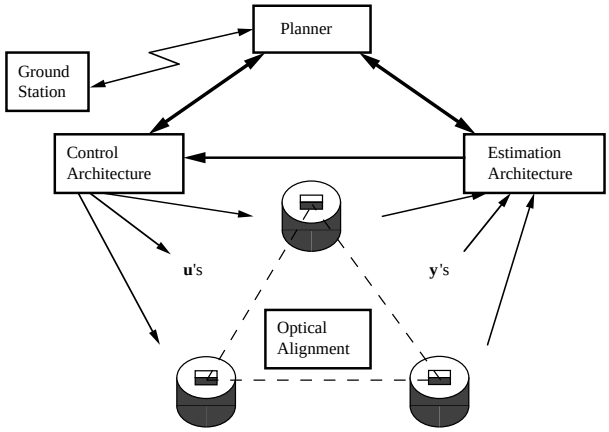

Fig. 2: Formation control system

3 Single Vehicle Station Keeping

The objective of a single vehicle station keeping controller is to maintain the vehicle state at a desired setpoint. We have considered the problem based on the vehicles in the FFTB. They have fixed thrust levels and an operating rate of 60 Hz. In addition, the GPS-based state estimates are relatively noisy. The goal is to regulate the vehicle position to within 2 cm of the desired point.

3.1 PD plus thrust mapper

Our vehicles have four thrusters (two positive and two negative) per axis. Each generates a force level f_{thruster}. Thus (for each axis), the possible thrust commands are:

$$u = \begin{cases} +2 & \text{both positive thrusters} \\ +1 & \text{one positive thruster} \\ +0 & \text{no thrusters} \\ -1 & \text{one negative thruster} \\ -2 & \text{both negative thrusters} \end{cases}$$

We generate this command from the estimated position and velocity ($\hat{x}$, $\dot{\hat{x}}$)

$$u = g(K_p \hat{x} + K_d \dot{\hat{x}}) \tag{1}$$

$$g(y) = \begin{cases} +2 & y > 1.5 f_{\text{thruster}} \\ +1 & y > 0.75 f_{\text{thruster}} \\ -1 & y < -0.75 f_{\text{thruster}} \\ -2 & y < -1.5 f_{\text{thruster}} \\ 0 & \text{otherwise} \end{cases} \tag{2}$$

The PD gains were designed using a linear system model, and then tuned to provide good performance on the vehicles.

3.2 Bang-bang controller

The time optimal trajectory from some initial condition to the origin ($x = \dot{x} = 0$) with constrained actuation is the well known bang-bang solution ([8]). Knowledge of the available thrust and of the plant is necessary to determine when to switch from acceleration to deceleration. This control can be described as:

$$u = \begin{cases} +2 & \dot{\hat{x}} < -\text{sgn}(\hat{x}) \sqrt{4 |\hat{x}| \frac{f_{\text{thruster}}}{m_{\text{vehicle}}}} \\ -2 & \text{otherwise} \end{cases} \tag{3}$$

3.3 Weighted fuel-time optimal control

The bang-bang controller is time-optimal, but can be very expensive in terms of fuel. A standard solution is to add fuel usage to the cost function

$$J = \int_{t_0}^{t_f} [1 + \lambda |u(t)|] dt \tag{4}$$

The control law that produces optimal trajectories according to this cost function is also well known ([8]):

$$u = \begin{cases} +2 & \dot{\hat{x}} < g_1(\hat{x}) \quad \text{and} \quad \dot{\hat{x}} < g_2(\hat{x}) \\ -2 & \dot{\hat{x}} > g_1(\hat{x}) \quad \text{and} \quad \dot{\hat{x}} > g_2(\hat{x}) \\ 0 & \text{otherwise} \end{cases} \tag{5}$$

$$g_1(x) = -\text{sgn}(x) \sqrt{\left(\frac{4}{1+\lambda}\right) |x| \frac{f_{\text{thruster}}}{m_{\text{vehicle}}}} \tag{6}$$

$$g_2(x) = -\text{sgn}(x) \sqrt{4|x| \frac{f_{\text{thruster}}}{m_{\text{vehicle}}}} \tag{7}$$

This controller produces trajectories that accelerate, coast, and then decelerate. In Eq. 4, λ is the penalty associated with fuel use (larger λ yields longer coasts).

3.4 Modified fuel-time controller

Optimal analytic solutions are only available for a few problems of this type. To handle more general cost functions that might be of interest, we have developed a heuristic technique that tries to improve the cost function in real-time by splitting the problem into two parts. The first part considers the control input design over a short time horizon. The second part assumes that the control inputs after this horizon will be determined using one of the previously quoted analytic techniques. Only the inputs in the short time horizon are actually implemented, and the entire optimization is repeated at each time step. This makes the optimization problem tractable and means that we can handle quite complex cost functions, but, of course, the approximation removes any guarantee of optimality from the resulting control.

Consider an example based on the objective in section 3.3

$$\min_u J(\hat{x}, \dot{\hat{x}}) = \int_t^{t_f} [1 + \lambda |u(t)|] dt \tag{8}$$

although more general cost functions could be considered. Without knowledge of the analytic solution in section 3.3, it would be very difficult to optimize this cost function in real time. Thus we solve the problem discretely and split the cost into the contribution from the current time step and the time interval from $t+1$ to t_f

$$\tilde{J} = J_{[t,t+1)}(u) + J_{[t+1,t_f]}(u) \tag{9}$$

$J_{[t,t+1)}$ is the cost function evaluated over the short horizon window (one time step in this case). $J_{[t+1,t_f]}(u)$ is the cost function evaluated from the end of the horizon window until t_f. For simplicity, we will assume that bang-bang control will be used from $t+1$ to t_f, which is clearly an approximation. We then optimize the approximate problem by choosing the sequence of inputs u that minimize $\tilde{J}$. For the cost in Eq. 8

$$J_{[t,t+1)} = (1 + \lambda |u|) \Delta t \tag{10}$$

$$J_{[t+1,t_f]} = (1 + \lambda |u_{\text{bb}}|) t_{\text{bb}}(\hat{x}, \dot{\hat{x}}, u) \tag{11}$$

where t_{bb} is a known function of $\hat{x}(t+1)$ and $\dot{\hat{x}}(t+1)$ which are computed from $\hat{x}(t)$, $\dot{\hat{x}}(t)$, and u. Note $|u_{\text{bb}}| = u_{max}$ which is 2 for this case.

3.5 Deadband

Because we have set a goal that the vehicle is to stay within 2 cm of its desired position, we add a deadband to each controller. This avoids unnecessary thruster action while the 2 cm goal is met. The deadband logic is:

if ($|x| > x_{\text{b}}$ or $|\dot{x}| > \dot{x}_{\text{b}}$)
then compute u as usual and let $m = -\text{sgn}(\dot{x})$
else
if $\text{sgn}(\dot{x}) = m$ then let $m = 0$

A zero value of m then disables the thrusters. x_{b} and $\dot{x}_{\text{b}}$ create an error box around the origin in position/velocity space.

3.6 Simulation

Each of the controllers was tested in simulation. The response to an initial offset (1 m) is shown in Fig. 3. The time and fuel to reach the origin are recorded in Table 1. The controllers are also compared for 10 min operating around the origin. The fuel, percent of time within the position bound (% Bnd) and the standard deviation of the position error (1-σ Err) are listed in Table 2. The simulation is based on the FFTB and includes the GPS based estimation system with measurement noise based on observed experimental values. GPS measurements are available at 10 Hz, and the estimator and plant are run at 60 Hz.

Table 1 shows that the bang-bang controller out performs the PD/map controller in both fuel and time for recovery from an initial position error. The fuel-time optimal controller allows a trade-off between fuel and time (through λ). Thus for $\lambda = 5.8$, the fuel-time controller recovers nearly as fast as the PD/map controller yet uses less than 20% of the fuel. The modified fuel-time control deviates from the optimal performance substantially; but note, however, that λ can be tuned to match a desired behavior (*e.g.* with $\lambda = 0.05$ the solution essentially duplicates the fuel-time optimal behavior for Eq. 8 with $\lambda = 5.8$). The results of the 10 min station-keeping run (see Table 2) show that all controllers accomplish the task (maintain position to within 2 cm), but the bang-bang controller is quite expensive in terms of fuel, while the fuel-time controller (and modified fuel-time controller with $\lambda = 0.05$) outperforms the PD/map controller in terms of fuel.

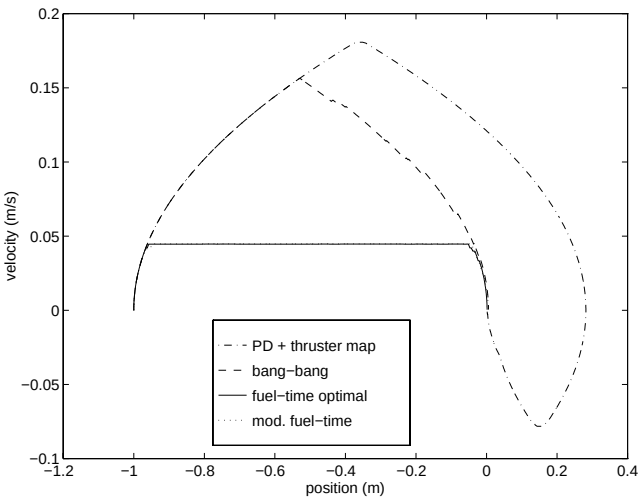

Fig. 3: Single vehicle response to initial offset (fuel-time and mod. fuel-time paths are indistinguishable)

Table 1: One veh. position offset response

Controller	Fuel	Time (sec)
PD + thrust map	1.00	22.9
Bang-bang	0.64	13.7
Fuel-time opt. ($\lambda = 5.8$)	0.17	25.2
Mod. fuel-time ($\lambda = 5.8$)	0.11	52.6
Mod. fuel-time ($\lambda = .05$)	0.17	25.3

Table 2: One veh station keeping performance

Controller	Fuel	% Bnd.	1-σ Err (m)
PD + thrust map	0.54	99.92	0.0093
Bang-bang	1.00	100	0.0094
Fuel-time ($\lambda = 5.8$)	0.39	100	0.0090
Mod. f-t ($\lambda = .05$)	0.47	100	0.0095

4 Station Keeping With Two Vehicles

For the two vehicle case, the relative state is to be controlled, but the system now has two inputs since each vehicle can fire the thrusters independently. The PD/thrust map and modified fuel-time controllers were extended to the two vehicle case. In addition, a weighted fuel-time solution was derived for the two vehicle case. This derivation is not presented here due to space constraints.

4.1 Controllers

The two vehicle PD/map control law is

$$u_1 = g_1(K_{p1}\hat{x} + K_{d1}\dot{\hat{x}}) \ , \ u_2 = g_2(K_{p2}\hat{x} + K_{d2}\dot{\hat{x}}) \quad (12)$$

The weighted fuel-time controller minimizes

$$J = \int_{t_0}^{t_f} [1 + \lambda_1|u_1(t)| + \lambda_2|u_2(t)|]dt \qquad (13)$$

The modified fuel-time controller also uses this cost function (again using the bang-bang approximation discussed in section 3.4. Each controller has a mechanism for adjusting the control effort produced by each vehicle: $K_{p1,2}$ and $K_{d1,2}$ for the PD/map controller, and $\lambda_{1,2}$ for the other controllers. Yet, the λ's provide the more natural approach as they are penalties on the fuel expenditure of each vehicle.

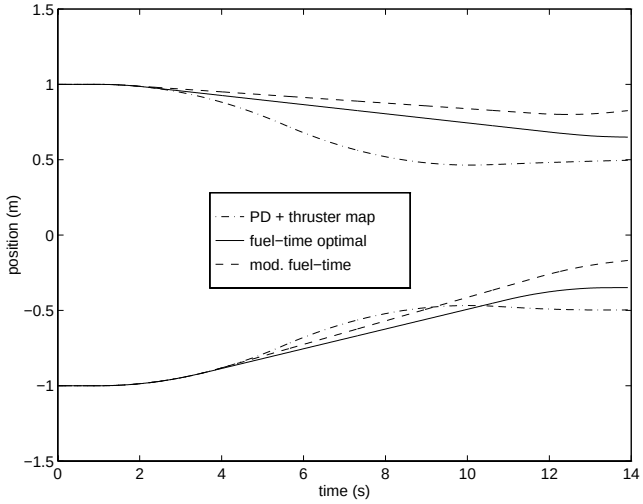

Fig. 4: Two vehicle response to initial offset

Table 3: Two veh. position offset response

Controller	Fuel_1	Fuel_2	Time (sec)
PD + map $\frac{K_{p1,d1}}{K_{p2,d2}}$=2	1.00	0.91	13.9
F-t Opt. $\frac{\lambda_2}{\lambda_1}$ =1.5	0.51	0.22	13.8
Mod. f-t $\frac{\lambda_2}{\lambda_1}$=1.5	0.46	0.24	13.6

4.2 Simulation

The 2 vehicle case was simulated for each controller for an initial offset in position. The vehicle positions are plotted vs. time in Fig. 4. The absolute velocity was not explicitly controlled in this simulation. As before, the PD/map control uses more fuel than the other controllers, yet takes about the same amount of time to align the vehicles (see Table 3). Also, the fuel use is nearly even between vehicles even though the gains for vehicle one are twice those of vehicle two (a function of the saturation nonlinearity in the thruster mapper). As expected, the other two controllers perform better and allow fuel use to be adjusted more directly.

A cost function-based controller provides better performance in general than the PD controller due to the large effect of the non-linearity (which is hard to account for in the design of the PD gains). Also, cost function-based controllers enable more direct control of resource allocation. There is a trade-off, of course, in the complexity of the controller. As the formation size grows, true optimal control becomes prohibitively difficult to implement and heuristic-based approaches (such as the modified fuel-time control) are necessary. Further research is required to develop an N-vehicle version of an optimization-based control.

5 Control Architectures

Controllers will rapidly grow in complexity as the number of vehicles increases since the number of relative states for an N-vehicle formation is $\frac{1}{2}N(N-1)$. The mission designer must assign importance to these relative states according to the mission objectives (*e.g.* for the NMI mission, the collector-combiner arms require ten- to twenty-times tighter control than the collector-collector arm [1]). The high-level planner/mission manager must adjust these weights according to the mission phase and assign fuel/resource costs appropriately. If the control output of each vehicle depends on its state relative to every other vehicle, then the control law could become very complex. This suggests a trade-off: limiting the number of relative states in the control calculation for each vehicle reduces the control complexity (and estimation complexity) but will increase the formation error. The resolution of these design conflicts constitutes a key step in the selection of the formation control architecture. Note that it has been shown [6] that the GPS-based estimation problem also grows rapidly in complexity as the number of relative states to be estimated increases. Thus the estimation architecture design must be considered simultaneously.

To illustrate the control architecture design method, two possible implementations are pictured for a generic 3-vehicle SSI (see Fig. 5). In case A, vehicle 2 is responsible for all estimation and control computations. It generates commands for all vehicles according to the entire formation state. This implementation would typically yield very good performance, but it will require high-rate communication between the vehicles and significant computational capability on vehicle 2. In case B, the communication requirement is reduced (recall the GPS system operates at 10 Hz vs. 60 Hz for the control) and the computational burden is divided between vehicle 1 and 3. However, in case B the formation is not maintained as accurately as in case A. Also note that u_2 is not used to help maintain the formation in this case (it can be used to perform the absolute positioning and orientation of the fleet). In the next section, Case B is implemented on the FFTB for a representative formation maneuver task.

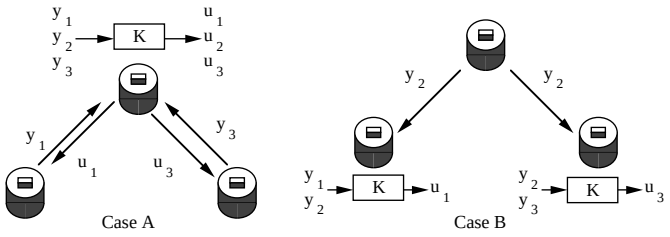

Fig. 5: Two candidate SSI control architectures

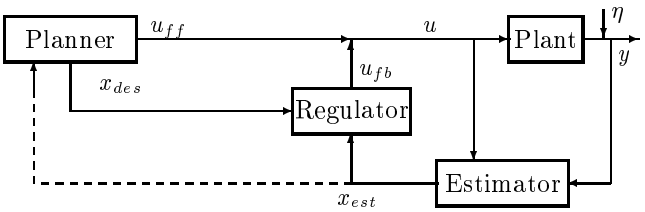

Fig. 6: Overall Control Scheme

6 Formation Planning

Formation planning addresses the problem of trajectory generation and the accompanying thrust programming for individual and groups of vehicles. Characteristic maneuvers such as initialization, retargeting, and resizing require the vehicles to move from known initial locations to specified final locations under a time constraint. However, these motions must be optimized to limit fuel usage. Also the vehicles must avoid collisions, maintain a rigid body formation (if desired), and account for specific limitations of any vehicle in the fleet.

A control scheme involving only open loop formation planning is not feasible for any realistic operating conditions under disturbances. For this reason, a control scheme consisting of a feedforward planner stage and a feedback controller stage is proposed. Fig. 6 shows the three main components of the formation flying control system: the estimator, the planner and the controller.

The planning aspect involves the minimization of fuel for the vehicle fleet over a maneuver time $[0, t_f]$. Using basic kinematics, the accelerations of the follower vehicles can be expressed as functions of the translation and angular motions of the leader vehicle. A conventional optimization approach would then use gradient search techniques to minimize

$$J = \int_0^{t_f} \sum_{i=1}^{N} |u_i| \, dt \tag{14}$$

for the N vehicles under the nonlinear constraints imposed by the kinematics and the actuator bounds. However, as discussed earlier, this typically requires extensive computational time, and thus might not be well suited for an on-line planning scheme that accounts for large deviations from the initial desired trajectory.

A much simpler numerical approach can be formulated by placing restrictions on the leader's thrust and torque levels to ensure that the follower's accelerations can be achieved. This approximate problem can be solved very efficiently using a Linear Program. A simple iteration on the bounds can then be used to achieve the best overall performance. Using a discrete model of the leader vehicle dynamics

$$x(k+1) = \Phi x(k) + \mathbf{\Gamma} u(k) \tag{15}$$

$$y(k) = H x(k) \tag{16}$$

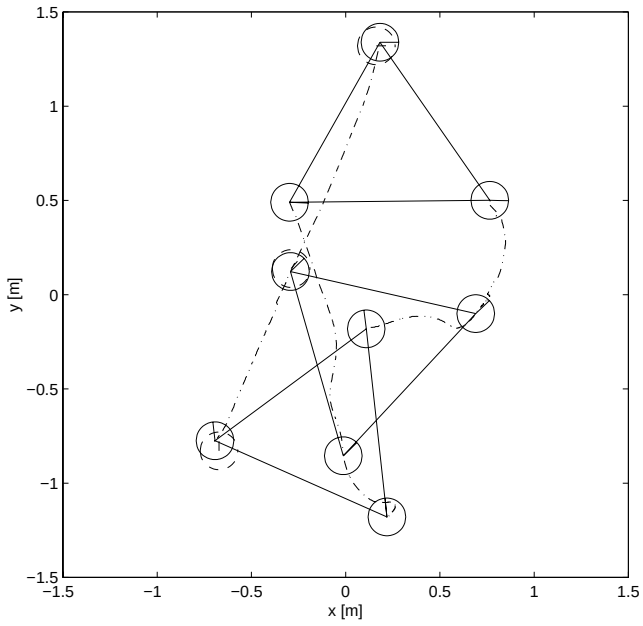

Fig. 7: Rigid body turn and translation maneuver

and a discrete convolution, the fuel optimal problem can be expressed as

$$\min \|u\|_1 \tag{17}$$

subject to

$$\begin{aligned} y(n) &= H\Phi^n x(0) + \sum_{i=0}^{n-1} h(n-i)u(i) \\ |u| \le u_{\max}, &\qquad f(x,u) \le 0, \quad |y_{des} - y(n)| \le \epsilon \end{aligned}$$

where $h(i) = H\Phi^i\mathbf{\Gamma}$ is the impulse of the system, and $y(n)$ and y_{des} correspond to the actual and desired final values of the output. u_{max} represents the upper bound on the control input, and $f(x,u)$ are linear functions of state and input vectors that describe possible constraints on the leader vehicle.

The experiments conducted use the proposed control scheme in Fig. 6. In these first tests, only an initial plan was developed. The individual locations of the follower vehicles were fixed in the local reference frame of the lead vehicle so that the formation would act as a *rigid body*. Fig. 7 shows the results from a collective 90° turn and translation maneuver. The dashed lines correspond to the actual paths followed by each of the vehicles. The standard deviations are given in Table 4. The lengths correspond to error in arm lengths with respect to the leader vehicle in the rigid body formation. Current research is focused on implementing the planner in real-time so that updates to the overall plan can be made in real-time to account for any anomalies.

Note that while this LP approach is limited in the constraints that can be handled, it can easily be extended to include more complex dynamics such as the linearized Hill's equations [9]. Current work is focused on extending this approach to include constraints on fuel balance within a fleet and collision avoidance.

Table 4: Standard deviations

Elements	1-σ Error[m]
Leader_x	0.040
Leader_y	0.026
Length_{12}	0.010
Length_{13}	0.010
Leader_θ	0.071 rad

7 Conclusions

This paper has identified several challenges associated with the control design for SSI missions. Candidate low-level controllers are compared for the 1- and 2-vehicle cases. A heuristic controller that optimizes a general cost function using an approximate technique is introduced and compared to the more standard controllers. While the results to date are promising, more research is required to validate this technique on a larger formation. In addition, techniques for allocating resources (*e.g.* fuel, computation) among vehicles are outlined. A formation control system is implemented on our FFTB to demonstrate the pieces necessary for a coordinated formation maneuver. Initial experimental results are presented. Future work will refine and extend the low-level controllers to 3 or more vehicles. Also, the high-level planner will be improved and made "on-line" so that trajectory plans can be revised in real-time as necessary.

References

[1] T. Alfery, "Request for proposal for deep space 3 (DS-3) spacecraft system industry partner," Tech. Rep. RFP No. N01-4-9048-213, Jet Propulsion Laboratory, November 1998.

[2] M. Colavita, C. Chu, E. Mettler, M. Milman, D. Royer, S. Shakian, and J. West, "Multiple spacecraft interferometer constellation (MUSIC)," Tech. Rep. JPL D-13369, JPL - Advanced Concepts Program, February 1996.

[3] C. A. Beichman, M. Shao, M. Colavita, and L. A. Lemmerman, "Terrestrial planet finder homepage." http://eis.jpl.nasa.gov/origins/missions/terrplfndr.html.

[4] J. C. Adams, A. Robertson, K. Zimmerman, and J. P. How, "Technologies for spacecraft formation flying," in *Proceedings of the ION GPS-96 Conference*, Sept. 1996.

[5] T. Corazzini, A. Robertson, J. C. Adams, A. Hassibi, and J. P. How, "GPS sensing for spacecraft formation flying," in *Proceedings of the ION GPS-97 Conference*, Sept. 1997.

[6] A. Robertson, T. Corazzini, and J. P. How, "Formation sensing and control technologies for a separated spacecraft interferometer," in *Proceedings of 1998 American Control Conf*, (Philadelphia, PA), June 1998.

[7] G. Purcell, D. Kuang, S. Lichten, S.-C. Wu, and L. Young, "Autonomous formation flyer (AFF) sensor technology development," in *AAS Guidance and Control Conference*, (Breckenridge,Colorado), Feb 1998.

[8] D. E. Kirk, *Optimal Control Theory*. Englewood Cliffs, NJ: Prentice-Hall Inc., 1970.

[9] A. Robertson, G. Inalhan, and J. P. How, "Spacecraft formation flying control for the ORION mission," submitted to the AIAA GNC, (Portland, OR), Aug 1999.