AIAA-99-4266

SPACECRAFT FORMATION FLYING CONTROL DESIGN FOR THE ORION MISSION*

Andrew Robertson†, Gokhan Inalhan‡, and Jonathan P. How§

Abstract

Formation flying of multiple spacecraft is an enabling technology for many future space science missions. However, the coordination and control of such instruments present many design challenges. This paper addresses the formation flying spacecraft control problem at several levels. We present low-level, multi-vehicle, station keeping algorithms and a control architecture to keep the vehicles aligned in formation. We also present a high-level fleet planner that creates trajectories (e.g. to re-size or re-target the formation) and takes into account the limited fuel onboard each vehicle. A coordinator is introduced at the highest-level to ensure that vehicle resources are expended equally within the fleet. Algorithms are discussed for each level, with simulations to compare performance. The simulation results are then verified on a formation flying testbed. The control design is then discussed with a perspective on the upcoming Orion mission. [1]

1 Introduction

Formation flying of multiple spacecraft is an enabling technology for many future space science missions including enhanced stellar optical interferometers and virtual platforms for earth observing and space science. A distributed array of simple but highly coordinated spacecraft may accomplish these science goals by allowing longer baselines, which translate into improved image resolution for stellar interferometry or earth-mapping synthetic apertures. Strong interest in the formation flying concept has led to several planned and proposed space missions including: Deep Space 3 [2], Terrestrial Planet Finder [3], EO-1 [4], TechSat-21 [5], and Orion [1].

This paper focuses on formation control design with a perspective from Orion, a formation flying mission under development at Stanford [1]. In particular, we investigate several formation control strategies that could be used to coordinate the Orion spacecraft during all mission phases. This paper investigates the high- and low-level control design issues and identifies potential experiments for the Orion mission. Simulation results are presented for several formation controllers and maneuver planners.

Sections 2 and 3 present an overview of the Orion mission and a description of Stanford's formation flying testbed (*FFTB*), which provides a very useful means to investigate control strategies on a high-fidelity hardware simulation (in two-dimensions) of a three-vehicle spacecraft formation.

The goal of formation flying is for several vehicles to act as a single system. The system task is to maintain a desired formation utilizing the limited resources available on each spacecraft. The combined system can then cooperate to accomplish various scientific objectives. We divide this task into 2 parts:

1. *Formation Keeping* initializes the desired formation and maintains it against disturbances.
2. *Formation Planning* creates trajectories for the formation to follow during maneuvers such as re-sizing or re-targeting.

Formation planning is discussed in detail in section 5. The planner develops trajectories for each vehicle in the formation given global knowledge of the vehicle states, the set of mission goals, and a de-

*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

† AIAA student member, Research Asst., Dept. of Aeronautics and Astronautics, Stanford Univ. adr@sun-valley.stanford.edu

‡ AIAA student member, Research Asst., Dept. of Aeronautics and Astronautics, Stanford Univ. ginalhan@leland.stanford.edu

§ Senior Member AIAA, Assistant Professor, Dept. of Aeronautics and Astronautics, Stanford Univ. howjo@sun-valley.stanford.edu

Fig. 1: Orion microsatellite showing the GPS antennas (round) and cold-gas thrusters (at corners).

tailed model of the system (e.g. including a model of the orbital dynamics for a LEO mission). These trajectories are delivered to the vehicles in a feedforward sense, i.e. no account is made for unmodeled disturbances or tracking errors in real-time. A key task for the low-level feedback controllers is to keep each vehicle on the desired trajectory. If the deviations from the trajectories are significant, then a new plan could be developed for the formation. Simulated maneuvers are presented for LEO and deep space. A formation maneuver is also demonstrated on the *FFTB*.

Section 4 describes the low-level formation keeping control algorithms, and the control architecture which serves to distribute the formation keeping task among the vehicles. Performance of the low-level control is evaluated in simulation and on the *FFTB*.

2 Overview of the Orion Mission

The Orion mission will demonstrate formation flying in low earth orbit [1]. The current design calls for 3 micro-satellites (50 cm cube, see Fig. 1) to be launched together. The formation will be initialized and then successively tightened over four mission phases: initialization/checkout, parking, coarse formation flying, and precise formation flying. The Orion vehicles have cold-gas thrusters for attitude and station-keeping maneuvers, Carrier-phase differential GPS sensing for precise relative navigation, and dedicated computation and cross link communication. Thus the Orion mission will demonstrate and validate many key technologies for future formation flying missions. Ref. [1] discusses the mission design in significantly more detail.

Fig. 2: Formation Flying Testbed

3 Formation Flying Testbed

To investigate the guidance, navigation, and control issues associated with formation flying, a testbed has been created at Stanford. The testbed consists of 3 active free-flying vehicles that move on a 12 ft $\times$ 9 ft granite table top (see Fig. 2). These air cushion vehicles simulate the drag-free zero-g dynamics of a spacecraft formation in a horizontal plane. The vehicles are propelled by compressed air thrusters. Each vehicle has onboard computing and batteries, and communicates with the other vehicles via a wireless ethernet, making them self-contained and autonomous. An overhead vision-based sensing system reports vehicle position and attitude. An indoor GPS-based sensing system performs a similar function, as demonstrated in Ref. [6].

4 Formation Keeping

4.1 Multi-Vehicle Station Keeping

One challenge in designing control laws for a formation of spacecraft is the use of on-off thrust actuators for position control. Due to this nonlinearity in the system, linear control techniques (e.g. LQR) are of limited usefulness. This challenge can be addressed by designing pulse-width modulation or thrust-mapping schemes that effectively cause the actuator to behave linearly in a time-averaged sense. Variable-thrust actuators could be used instead of standard thrusters, if feasible; however most conventional missions, including Orion, use on-off type thrusters. Thus, we chose to design control laws that directly account for the on-off nature of conventional thrusters. The control laws described are for position control in one axis, but the results are applicable to three-axis position control for a typical spacecraft with separate thrusters for each axis.

The weighted fuel-time optimal control law for a single vehicle with on-off actuators (e.g. two positive and two negative thrusters per axis) is the well known bang-off-bang trajectory [7]

$$u = \begin{cases} +2 & \dot{\hat{x}} < g_1(\hat{x}) \quad \text{and} \quad \dot{\hat{x}} < g_2(\hat{x}) \\ -2 & \dot{\hat{x}} > g_1(\hat{x}) \quad \text{and} \quad \dot{\hat{x}} > g_2(\hat{x}) \\ 0 & \text{otherwise} \end{cases} \quad (1)$$

$$g_1(\hat{x}) = -\text{sgn}(\hat{x})\sqrt{\left(\frac{4}{1+4\lambda}\right)|\hat{x}|\frac{f_{\text{thruster}}}{m_{\text{vehicle}}}} \quad (2)$$

$$g_2(\hat{x}) = -\text{sgn}(\hat{x})\sqrt{4|\hat{x}|\frac{f_{\text{thruster}}}{m_{\text{vehicle}}}} \quad (3)$$

This control law produces optimal trajectories according to the cost function

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

where the maneuver occurs from t_0 to t_f and λ is the penalty applied to fuel use. This technique has been extended to the two vehicle weighted fuel-time optimal case in [8] for the cost function

$$J = \int_{t_0}^{t_f} [1 + \lambda_1|u_1(t)| + \lambda_2|u_2(t)|]dt \quad (5)$$

where λ_1 and λ_2 provide weights that penalize fuel use for each vehicle. For a given initial offset in relative position, the optimal trajectory is bang-off-bang (with different timing) for each vehicle. The switch times for these events are determined by the control law

$$u_1 = \begin{cases} +2 & \dot{\hat{x}} < g_1(\hat{x}) \quad \text{and} \quad \dot{\hat{x}} < g_2(\hat{x}) \\ -2 & \dot{\hat{x}} > g_1(\hat{x}) \quad \text{and} \quad \dot{\hat{x}} > g_2(\hat{x}) \\ 0 & \text{otherwise} \end{cases} \quad (6)$$

$$u_2 = \begin{cases} +2 & \dot{\hat{x}} < g_3(\hat{x}) \quad \text{and} \quad \dot{\hat{x}} < g_4(\hat{x}) \\ -2 & \dot{\hat{x}} > g_3(\hat{x}) \quad \text{and} \quad \dot{\hat{x}} > g_4(\hat{x}) \\ 0 & \text{otherwise} \end{cases} \quad (7)$$

$$g_i(\hat{x}) = -\text{sgn}(\hat{x})\sqrt{G_i|\hat{x}|} \quad (8)$$

$$G_1 = |\frac{1}{A}| \quad (9)$$

$$G_2 = |\frac{1}{A - \frac{2\lambda_1}{K_1}}| \quad (10)$$

$$G_3 = 2(K_1 + K_2) \quad (11)$$

$$G_4 = |\frac{1}{(A - \frac{2\lambda_1}{K_1})\frac{1}{C^2} - \frac{B}{C} - \frac{K_1 B^2}{2C^2}}| \quad (12)$$

$$B = \frac{\lambda_2}{K_2} - \frac{\lambda_1}{K_1} \quad (13)$$

$$A = -\frac{(1 - K_1 B)^2}{2(K_1 + K_2)} - B + \frac{B^2 K_1}{2} \quad (14)$$

$$C = 1 + \lambda_1 - \frac{\lambda_2 K_1}{K_2} \quad (15)$$

Some initial simulation results were presented in [8]. Simulations for three new cases are presented here to highlight the benefit of using this more complex control law instead of linear control with thrust mapping, or the simpler single vehicle fuel-time optimal controller. The simulation results are then verified experimentally on the formation flying testbed. First, we describe the cases to be simulated.

Proportional plus derivative (PD) control with thrust-mapping is presented as a "baseline" case. The gains were designed using a linear system model, and then tuned to provide good performance on the vehicles. Another "baseline" case is each vehicle using its own fuel-time optimal control (as described in Eqs. 1–3). In this case, each vehicle is operating independently to reduce the relative state error, but note that they each must have knowledge of the relative state. Finally, the two vehicle fuel-time optimal control is also presented. In each case, the control design variables (gains or weights) are chosen to attempt to force vehicle 1 to expend 50% more fuel than vehicle 2 for a maneuver of about 15.6 seconds. By fixing the duration and fuel ratio, the total fuel use and cost (according to Eq. 5) can be compared for each controller. However, the fuel-balance depends strongly on the initial conditions for the first two controllers. The results from the simulation are presented in Table 1.

The simulation is based on the formation flying testbed and includes the GPS 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. Each of the three control cases mentioned were run on a two vehicle system with an initial relative

Table 1: Simulation of 2 vehicle position offset response (normalized fuel, cost $\lambda_1 = 1.86, \lambda_2 = 1.99$).

Controller	Fuel_1	Fuel_2	Time (sec)	*Cost*
PD + map	1.00	0.66	15.63	40.04
F-t Opt (2×1-v)	0.45	0.31	15.60	27.07
F-t Opt (2-veh.)	0.44	0.28	15.63	26.21

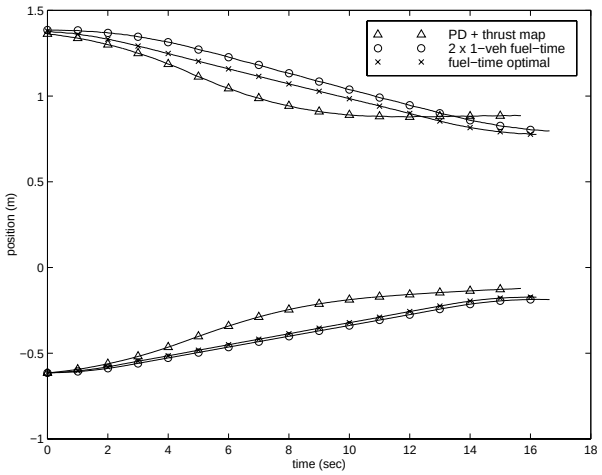

Fig. 3: Experiment: 2 veh. position offset response

position error of one meter. The fuel expended by each vehicle, the time to bring the error to zero, and the resulting cost (according to Eq. 5) are recorded.

From this table, it is clear that the PD control expends a large amount of extra fuel compared to the other controllers. The two vehicles running independent single-vehicle controllers perform much better than the PD, but this is not as fuel efficient as the 2-vehicle fuel-time optimal control. This intuitive result presents a possible trade-off for the control designer: the extra fuel cost of the single vehicle controller may be acceptable in exchange for the simplified controller. This trade-off is explored in Section 4.3. More work is needed to determine how large this difference is for realistic operating conditions (e.g. during a formation maneuver).

The three cases were then tested experimentally on the formation flying testbed. Note that the overhead vision system (which has different noise and accuracy characteristics than the GPS system simulated) was used for this experiment. The vehicles' positions, velocities and fuel use were recorded. Fig. 3 shows the state histories for the three cases. Table 2 shows the performance of each controller on a real system.

The experimental results confirm the performance predicted by the simulation. An interesting source of error is that the vehicle acceleration used in the simulation was optimistic. All vehicles took more fuel to accomplish the maneuver in experiment than they did in simulation. The normalized fuel value in Table 2 is about 50% higher than in Table 1. The two vehicle fuel-time optimal controller is most sensitive

Table 2: Experiment of 2 vehicle position offset response (normalized fuel, cost $\lambda_1 = 1.86, \lambda_2 = 1.99$).

Controller	Fuel_1	Fuel_2	Time (sec)	*Cost*
PD + map	1.00	0.79	15.67	56.78
F-t Opt (2×1-v)	0.62	0.48	16.63	41.90
F-t Opt (2-veh.)	0.53	0.44	16.20	38.43

to this; it shows an overshoot and finishes 3 cm away from the goal of a 1 m relative separation. This highlights the importance of having an accurate thruster model for bang-off-bang type controllers.

4.2 Numerical Calculation of Control Laws

The previous section illustrates the benefit of using a control law designed for a particular case (e.g. two vehicles) and a particular cost function (e.g. Eq. 5) compared to linear control or to a combination of simpler controllers. However, the analytic form of the control law that provides the optimal behavior for the weighted fuel-time cost function is very difficult to derive for the case of three or more vehicles. It is not clear that simple analytic expressions for the optimal control laws even exist.

Ref. [8] presented one possible solution to this problem by using a real-time optimization routine that solves an approximate form of the problem. This technique was shown to replicate the performance of the fuel-time optimal controller for the two vehicle case. However, this technique is not guaranteed to be optimal and requires significant computational resources to work in real-time. A control law based on switching conditions (such as Eqs. 1–3) would demand less computation; however the difficulty lies in finding a control law for a given complex cost function for a large number of vehicles.

This section presents an alternate method for finding control laws based on a numerical optimization technique called *Inverse Dynamic Optimization* (see [9], pages 362–363). In this technique, points on state variable histories are determined so that the path is optimal according to some cost function. The control histories that produce the optimal path are then found through numerical differentiation of the state variable histories. In other words, a general cost function is specified, and then an optimal

trajectory along with the control history that produces it are found for some initial condition. In this way, a family of optimal trajectories can be determined. Given the on-off nature of the actuators in our system, it is feasible to create a sufficient number of these optimal trajectories to determine the appropriate control behaviors and especially the *switching times* for a wide range of cases.

This technique is first applied to the single vehicle case to illustrate the steps. We start from Eq. 4 without knowledge of Eqs. 1–3, and solve for the optimal trajectories from a family of initial conditions from x_0 = -2.0 to -0.25. The resulting trajectories (with fairly large time steps) are shown in Fig. 4 in the position-velocity phase plane. Note that all of the responses follow bang-off-bang trajectories, and two switching points are apparent on each trajectory. By tracing a line through each set of switching points, we find the switching lines (which separate the initial acceleration region from the coast region and the coast region from the final deceleration region). These lines are compared to the true optimal switching lines in Fig. 5.

The limitation on this technique is that even a small number of trajectory segments requires a large amount of computation. Thus the switching lines found using the inverse dynamic optimization will be approximate. However, the switching lines from this technique can then be used as an initial guess for the optimal control law. The lines could be varied and tested in a conventional forward dynamic simulation to find the true lines to within a desired tolerance. Using this technique, the value for the coefficient in Eq. 3, which should equal $\left(\frac{4}{1+4\lambda}\right)$, was determined to within 2% by the initial curve fit, and then within 0.0001% using the forward dynamic simulation.

Note the inverse dynamic optimization finds the optimal *form* of the control, i.e. the bang-off-bang nature of the solution is clear from the optimal trajectories. Once this is known, refining the exact shape and location of the switching lines (for each λ) is straightforward, even for a many-vehicle system.

The inverse dynamic optimization has the additional benefit that if the correct form of the control law is known, (i.e. if we knew the form of the coefficient is $\left(\frac{a}{b+c\lambda}\right)$ in Eq. 2), it can be used to find the constants. This produces a general control law (i.e.

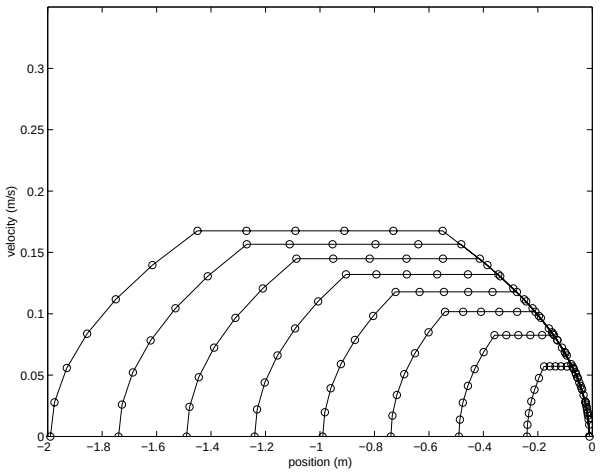

Fig. 4: Family of optimal trajectories from inverse dynamic optimization

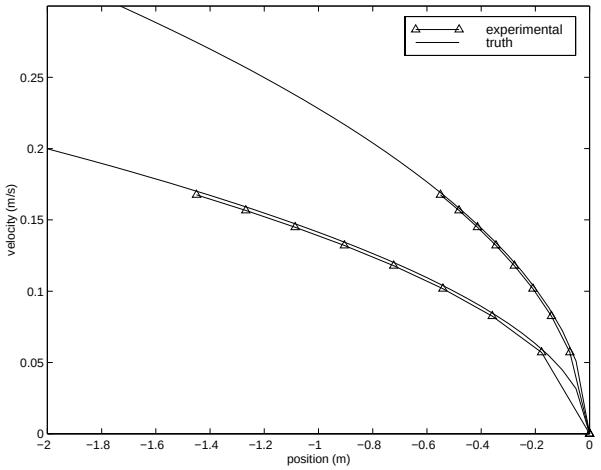

Fig. 5: Numerical switching line compared to truth

for any λ value).

As another example, consider the three vehicle case with the cost function

$$J = \int_{t_0}^{t_f} [1 + \lambda_1 |u_1(t)| + \lambda_2 |u_2(t)| + \lambda_3 |u_3(t)|] dt \quad (16)$$

An analytic solution for the optimal control law is not known to be currently available [10]. A trajectory was found for this 3-vehicle cost function using inverse dynamic optimization. Fig. 6 shows the time histories of the relative positions and velocities for the vehicle pairs: 1 & 2, and 2 & 3. Fig. 7 shows the optimal control history for each vehicle. It is immediately obvious that each vehicle follows a bang-off-bang trajectory. Each vehicle coasts for a different time period, and they leave coast mode in the reverse order from which they entered (i.e. vehicle 2 starts coasting first and is the last to stop coasting). The state histories (Fig. 6) show that *pos12* is

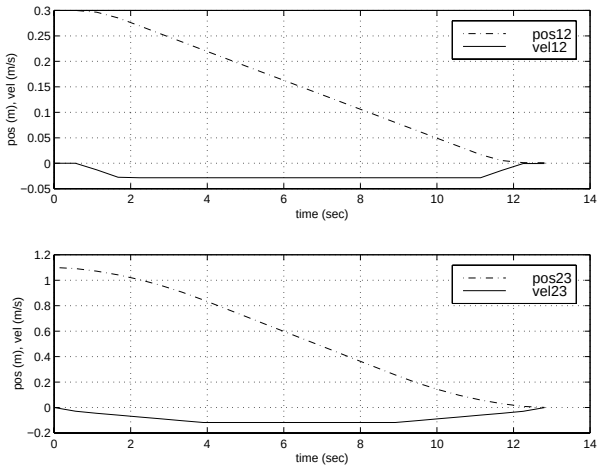

Fig. 6: Relative state histories for the 3 vehicle case

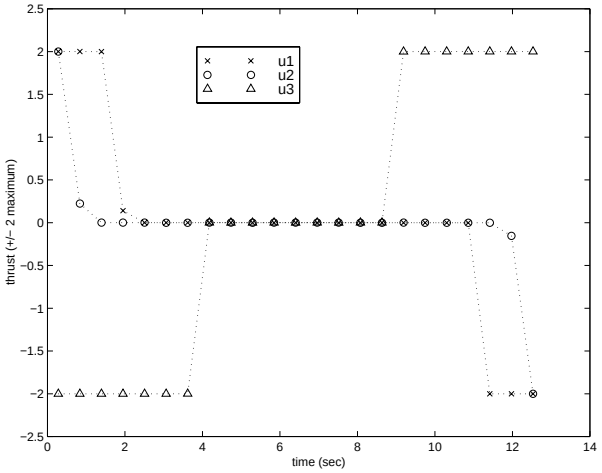

Fig. 7: Control histories for the 3 vehicle case

brought to zero before *pos23*. And u_2 (which affects both relative states) is used the least of the three vehicles.

The control designer can run this optimization for a grid of initial conditions to generate these trajectories over the state region of interest. This could be used to construct a family of switching conditions. These conditions can then be used directly for control, or they can serve as the initial guess for standard optimization techniques with conventional dynamic simulations. This technique may be applied for any cost function and any number of vehicles.

4.3 Control Architectures

A single control law for the entire formation (such as if the three vehicle control law in Section 4.2 were implemented on a 3-vehicle formation) offers the best performance in terms of a cost such as in Eq. 16, however such *centralized* control is quite demanding in terms of inter-vehicle communication and knowledge of the system states.[2] For a formation flying mission with limited computation resources (such as Orion), a distributed control approach might be necessary.

Several candidate *control architectures* – from centralized to fully distributed – could be used for the fleet control system. The architecture selection would be based on determining which design meets the minimum performance requirements with the least cost (in terms of communication and computation). Of course, factors such as robustness and flexibility play an important role in the design decision as well. The recent interest in coordinated motion in robotics, aerial and land vehicles, as well as spacecraft formation flying has generated numerous approaches, which are briefly summarized below.

Ref. [12] discusses cluster geometries for satellites in geosynchronous orbit. There are constraints on the spacing of the satellites and on the location and size of the cluster. Formation keeping control is not discussed in the paper, but it is clear that any formation control scheme would have to focus on the absolute (Earth referenced) satellite motions because the orbital dynamics dominate the motions of the satellites. We refer to the method where formation geometry is controlled by carefully controlling the orbit of each satellite as *absolute* control.

Ref. [13] discusses control of aerial vehicles flying in formation to take advantage of reduced total aerodynamic drag. A centralized approach is compared to a "leader follower" technique where each vehicle maintains a fixed offset relative to the vehicle ahead of it in the formation. The vehicle at the lead of the formation maintains its absolute position and the rest maintain formation with respect to the leader (directly or indirectly), thus we refer to this as a *leader-referenced* architecture. Refs. [14, 15] discuss coordination strategies for groups of mobile robots. Strategies based on combined leader-referenced and "nearest neighbor" (where each non-lead vehicle moves based on its position relative to one or more other vehicles in the formation) architectures are applied for troop maneuvers, formation initialization, and target capture.

[2] Note that it has already been shown [11] that the GPS-based estimation problem grows rapidly in complexity as the number of relative estimation states increases.

Refs. [16, 17] consider several control architectures for fleets of vehicles. Ref. [16] discusses absolute, nearest neighbor, and multiple nearest neighbor strategies. For the nearest neighbor strategy, the leader vehicle ($i = 1$) moves absolutely and the rest of the vehicles follow in a chain (i.e. vehicle i follows vehicle $i - 1$). The multiple nearest neighbor strategy designates two or more leaders, then each non-leader vehicle moves based on its position relative to the two or more nearest neighbor vehicles. Ref. [17] compares centralized, leader-referenced, and nearest neighbor strategies for controlling a platoon of four ground vehicles. In general the performance of the centralized strategy is better than the others, however limited system resources such as sensing and communications are given as reasons why a leader-referenced strategy may be prefered for their application. System trade studies such as these must also be performed for upcoming spacecraft formation flying missions.

In summary, we have the following control architectures candidates:

1. Centralized - Each vehicle moves based on all relative states in the formation.
2. Leader-referenced - One vehicle controls its absolute position. Others either all move relative to the leader, or move as a chain (i.e. vehicle i follows vehicle $i-1$; the leader is vehicle $i = 1$). Ref. [16] combines the two, suggesting multiple chains following a single leader.
3. Absolute - Each vehicle moves based on its own absolute state; relative positions are not directly sensed nor controlled.

These three architectures are illustrated in Fig. 8 for a three vehicle case such as the *FFTB*. A solid arrow from vehicle 1 to vehicle 2 indicates that vehicle 1 is flying relative to vehicle 2. Dashed arrows indicate a vehicle is attempting to maintain absolute position. Thus in the centralized case (D), vehicle 2 must maintain absolute position and relative position. If centralized relative control (e.g. based on Eq. 16) is implemented, this means absolute control commands must be combined with relative control commands. A simple method is to form the weighted sum

$$u = \alpha u_{abs} + (1 - \alpha) u_{rel} \tag{17}$$

Any of the architectures in Fig. 8 could be tested on a formation flying mission such as Orion. However for a formation of more than 3 vehicles, there are other alternatives between the fully centralized and the leader-referenced cases. In particular, we propose a hierarchical control architecture that divides a formation of vehicles into smaller sub-formations. Each sub-formation has an *internal* control architecture as well as external inputs that allow the sub-formations to fly in formation with each other and to follow absolute commands.

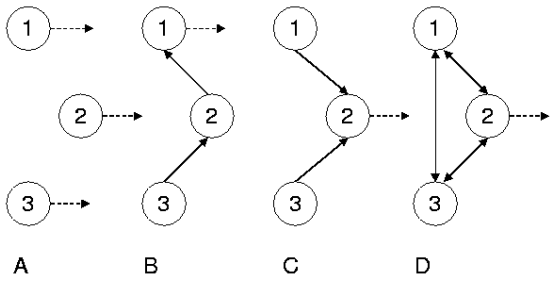

Fig. 8: Control architectures: absolute (A), leader-referenced (B & C), and centralized (D).

Fig. 9 shows two possible hierarchical control architectures. Case A shows a hierarchy of two 2-vehicle sub-formations. Vehicles in the sub-formation maintain relative station (e.g. using Eqs. 1–3) as well as following an external input which could be either an absolute control or the relative location of another sub-formation. Thus each vehicle will have to combine two sets of command inputs (e.g. using Eq. 17). Case B illustrates two 3-vehicle sub-formations; each sub-formation is the same as the centralized formation in Fig. 8D.

The control architecture plays a key role in the performance of the formation. This is illustrated by the experiment in Section 5. Control architectures such as those in Fig. 8 (and Fig. 9 for missions with 4 or more vehicles) should be compared by the mission designer based on factors including: control performance, fuel cost, fuel balance, flexibil-

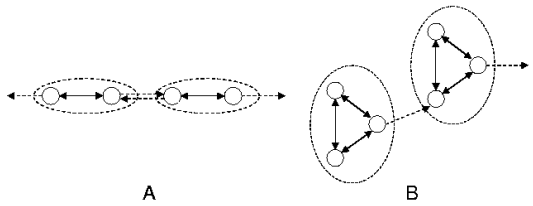

Fig. 9: Control architectures: 2-vehicle sub-formations (A), and 3-vehicle sub-formations (B).

ity, robustness, and sensing requirements. Different architectures may be used during different mission phases (e.g. Orion may use an absolute architecture for its initial phase and a centralized architecture for its precise formation phase). Note that the work in Ref. [6, 11] clearly showed that the choice of an *estimation architecture* must be strongly linked to the choice of a control architecture. Ref. [6] discusses estimation architecture choices and computation demands for GPS-based relative sensing. Orion will serve as an excellent testbed for comparing control architectures on-orbit.

4.4 Formation Coordinator

The previous sections have discussed the low-level formation keeping control laws and how they are used for formation control in conjuction with a suitable control architecture. However, while the role of the control parameters (e.g. λ's and α's) available to the control designer are clear, it is not clear how to choose them or how they should vary over the course of the mission in response to the formation resource state. A mission controller on the ground could perform this function for a simple, short-term mission (e.g. Orion). But an autonomous agent onboard the spacecraft would allow for higher bandwidth operation as well as reducing controller work load and down-link requirements.

The coordination activity (resource allocation) occurs slower than real-time; the coordinator tracks performance as well as vehicle resource states and periodically adjusts the control parameters in keeping with the goals of the mission. One major goal is *fuel-balancing*: the burden of maintaining the formation should fall on the vehicles with the most fuel remaining. The coordinator is responsible for meeting this goal as well as other mission-specific performance goals. Current work is focused on the design of this autonomous formation coordinator within the control architecture.

5 Formation Planning

One of the key characteristics of the control system is that the spacecraft must be able to perform operations that require coordinated motions of the fleet. This characteristic requires a planning algorithm that can address the problem of trajectory generation, and in the process, develop thrust sequences for individual and groups of vehicles. However, with disturbances, an open-loop control involving only formation planning will not be sufficient. Thus the control scheme in Fig. 10 consisting of both a feedforward planner and a feedback controller is required. The primary objective of the formation planner is to accomplish the desired goal (formation initialization, resizing, or reconfiguration) while minimizing the most important spacecraft resource: fuel.

As shown in the figure, the planner stage can update the formation trajectories in real-time depending on the estimated locations of the vehicles. It would be computationally very intensive to continuously update the formation plan throughout a maneuver to account for disturbances or modeling errors. However, the following section discusses a computationally feasible trajectory planning algorithm that uses the linearized model of the relative dynamics in orbit or deep space. The approach poses the formation planning problem as a *linear program* (LP), which can be solved very quickly and efficiently with standard solvers. For certain planning problems, the LP formulation has clear advantages when compared to nonlinear programming techniques (e.g. computation time grows very slowly with increasing fleet size or maneuver length [19]).

For the formation planning problem, the motion of the vehicles within a fleet can be referenced to each other or to a control point. In either case, the individual maneuvers correspond to the vehicles moving from known initial relative locations to specified final relative locations under a time constraint. Note that the planning must also account for specific limitations and constraints for any of the vehicles in the fleet.

The linearized relative motion of spacecraft in neighboring orbits is described by the well-known Clohessy-Wiltshire equations [20]. The in-plane dy-

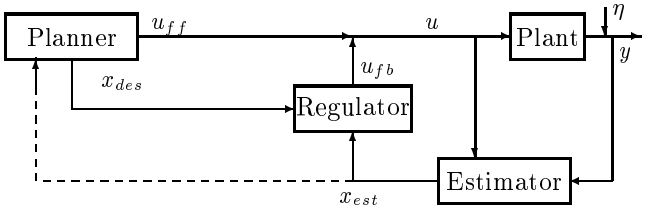

Fig. 10: Overall Control Strategy

namics (Eq. 18) consisting of in-track (x) and radial (z) motion decouples from the out-of-plane dynamics (Eq. 19). Naturally, these two sets of equations can be analyzed separately in the formation planning stage.

$$\frac{d}{dt}\begin{bmatrix}\dot{x}\\ x\\ \dot{z}\\ z\end{bmatrix}=\begin{bmatrix}0 & 0 & 2n & 0\\ 1 & 0 & 0 & 0\\ -2n & 0 & 0 & 3n^2\\ 0 & 0 & 1 & 0\end{bmatrix}\begin{bmatrix}\dot{x}\\ x\\ \dot{z}\\ z\end{bmatrix}+\begin{bmatrix}1 & 0\\ 0 & 0\\ 0 & 1\\ 0 & 0\end{bmatrix}\begin{bmatrix}u_x\\ u_z\end{bmatrix} \quad (18)$$

$$\frac{d}{dt}\begin{bmatrix}\dot{y}\\ y\end{bmatrix}=\begin{bmatrix}0 & -n^2\\ 1 & 0\end{bmatrix}\begin{bmatrix}\dot{y}\\ y\end{bmatrix}+\begin{bmatrix}1\\ 0\end{bmatrix}u_y \quad (19)$$

Note that, in the limit, as the reference orbital frequency n approaches zero, the relative dynamics correspond to motion in deep space (complete decoupling of in-track and radial dynamics). For a three vehicle case in two dimensions this can be written as

$$\frac{d}{dt}\begin{bmatrix}\dot{x}_{12}\\ x_{12}\\ \dot{x}_{13}\\ x_{13}\\ \dot{y}_{12}\\ y_{12}\\ \dot{y}_{13}\\ y_{13}\end{bmatrix}=\begin{bmatrix}0&0&0&0&0&0&0&0\\ 1&0&0&0&0&0&0&0\\ 0&0&0&0&0&0&0&0\\ 0&0&1&0&0&0&0&0\\ 0&0&0&0&0&0&0&0\\ 0&0&0&0&1&0&0&0\\ 0&0&0&0&0&0&0&0\\ 0&0&0&0&0&0&1&0\end{bmatrix}\begin{bmatrix}\dot{x}_{12}\\ x_{12}\\ \dot{x}_{13}\\ x_{13}\\ \dot{y}_{12}\\ y_{12}\\ \dot{y}_{13}\\ y_{13}\end{bmatrix}+\begin{bmatrix}1&-1&0&0&0&0\\ 0&0&0&0&0&0\\ 1&0&-1&0&0&0\\ 0&0&0&0&0&0\\ 0&0&0&1&-1&0\\ 0&0&0&0&0&0\\ 0&0&0&1&0&-1\\ 0&0&0&0&0&0\end{bmatrix}\begin{bmatrix}u_{x1}\\ u_{x2}\\ u_{x3}\\ u_{y1}\\ u_{y2}\\ u_{y3}\end{bmatrix} \quad (20)$$

In Eq. 20, $[\dot{x}_{12}, x_{12}, \dot{x}_{13}, x_{13}, \dot{y}_{12}, y_{12}, \dot{y}_{13}, y_{13}]^T$ correspond to the relative velocity and position in a local reference frame with x and y coordinates. The vector $[u_{x1}, u_{x2}, u_{x3}, u_{y1}, u_{y2}, u_{y3}]^T$ are normalized thrust vector components for each vehicle.

For a given maneuver time $[0,\ t_f]$, one can chose a sampling period $t_s \ll t_{\text{period}}$ such that the discrete description of either set of relative dynamics is

$$\begin{aligned}\underline{x}(k+1) &= \Phi\underline{x}(k)+\mathbf{\Gamma}\underline{u}(k)\\ \underline{y}(k) &= H\underline{x}(k)\end{aligned} \quad (21)$$

where H corresponds to identity matrix for full state knowledge.

Given these linearized dynamics, we can now pose the problem to be solved by the formation planner. In particular, the objective is to minimize the amount of fuel used to take the vehicles from an initial set of relative states $x(0)$ to a final desired configuration y_{des} in n time steps. By using discrete convolution, the basic formation planning problem can be posed as

$$\min \|\underline{u}\|_1$$
$$\text{subject to} \begin{cases} \underline{y}(n) = H\Phi^n\underline{x}(0)+\sum_{i=0}^{n-1}\underline{h}(n-i)\underline{u}(i)\\ |\underline{u}(i)| \le \underline{u}_{\max}\\ f(x,u)\le 0\\ \left|\underline{y}_{\text{des}}-\underline{y}(n)\right| \le \underline{\epsilon}\end{cases} \quad (22)$$

where $\underline{h}(i) = H\Phi^{i-1}\mathbf{\Gamma}$ is the impulse response of the system, and $\underline{y}(n)$ and $\underline{y}_{des}$ correspond to the actual and desired final values of the output. $\underline{u}_{max}$ represents the upper bounds on the control input, and $\underline{f}(u,x)$ are linear functions of state and input vectors that describe possible constraints on the vehicles.

This problem statement is affine, and thus convex in $\underline{u}(i)$, which are the optimization variables. By introducing the impulse response matrix P and input matrix $\underline{u}$, the discrete convolution can be expressed as a linear matrix equality constraint

$$\begin{aligned} P &\triangleq \begin{bmatrix}\underline{h}(n) & \underline{h}(n-1) & \dots & \underline{h}(1)\end{bmatrix}\\ \underline{u} &\triangleq \begin{bmatrix}\underline{u}(0)^T & \underline{u}(1)^T & \dots & \underline{u}(n-1)^T\end{bmatrix}^T\\ \underline{b} &\triangleq y_{des}-H\Phi^n\underline{x}(0)\\ \Rightarrow P\underline{u} &= \sum_{i=0}^{n-1}\underline{h}(n-i)\underline{u}(i)=\underline{b}\end{aligned} \quad (23)$$

The cost function in Eq. 22 can also be expressed as a linear function of the input vector by introducing two sets of *slack variables* (see Eq. 24) for its negative and positive parts. Restructuring the problem around these slack variables, we obtain

$$\begin{aligned} \underline{u} &\triangleq \underline{u}_{pos}-\underline{u}_{neg}\\ \underline{u}_{pos}\ge 0, &\quad \underline{u}_{neg}\ge 0\\ \bar{\underline{u}} &= \begin{bmatrix}\underline{u}_{pos}^T & \underline{u}_{neg}^T\end{bmatrix}^T \quad (24)\\ \hat{A} &= \begin{bmatrix}P & -P\end{bmatrix}^T \quad (25)\end{aligned}$$

Similar operations can be carried out for any other linear constraints involving the inputs or states, and this would result in a general inequality constraint,

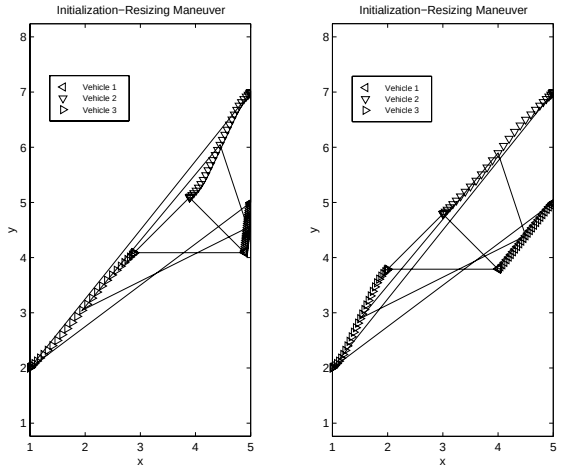

Fig. 11: Resizing Maneuver

$\hat{D}\underline{u} < \underline{e}$. Adding this term, the general linear programming problem can be posed as

$$\begin{array}{rl} \min & c^T \bar{\underline{u}} \\ \text{subject to} & \left\{ \begin{array}{rcl} \hat{A}\underline{u} & = & \underline{b} \\ \hat{D}\underline{u} & < & \underline{e} \\ \bar{\underline{u}} & \leq & \underline{u}_{max} \\ -\bar{\underline{u}} & \leq & 0 \end{array} \right. \end{array} \tag{26}$$

where $\underline{c}^T$ corresponds to weights on the fuel usage of the various vehicles. This general formulation can be used to solve many formation planning problems for distributed satellite systems. For example, separated spacecraft interferometry [21] and space based surveillance of earth [22] are all structured around the following formation flying maneuvers:

- Initialization,
- Resizing, reconfiguration, and retargeting
- Observation maneuvers.

Designs for several of these maneuvers are shown in the following examples.

Formation resizing & reconfiguration: One of the basic requirements for deep space interferometry operations is the ability to reconfigure the formation during aperture filling maneuvers [24, 25]. Fig. 11 shows an example of a reconfiguration maneuver where the vehicles are moved to a second formation while minimizing the total fuel used by the fleet. In this scenario, the collector vehicles, which have just completed an aperture filling maneuver, are moved to the new configuration that is 1 unit (relative distance in both x and y) from the combiner spacecraft.

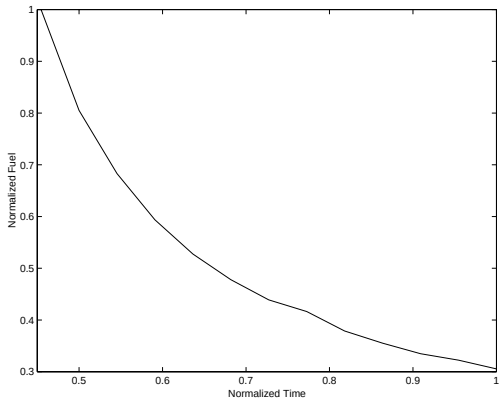

Fig. 12: Fuel-Time tradeoff curve

Aperture filling maneuvers are designed to minimize the maneuver time in order to maximize the number of objects imaged. However, the limited fuel resources put a tight constraint on the speed of the maneuver. The quasi-convex nature of the LP formation planning approach provides a convenient basis for this fuel/time tradeoff analysis. Fig. 12 shows the fuel-time trade-off curve constructed by mapping fuel usage for a set of maneuver time constraints. For an auxiliary cost function describing the fuel-time trade-off,

$$J = \lambda_1 \|\underline{u}\|_1 + \lambda_2 t_f \tag{27}$$

the $J_{\min}$ corresponds to the point with the normal, $N = [-\lambda_2, \ -\lambda_1]^T$. We can use graphs like Fig. 12 to pick any desired maneuver time based on a given trade-off characterized by λ_1 and λ_2.

The fuel balance between vehicles in the fleet can be adjusted using the weights on fuel usage (c in the cost function of Eq. 26). In Fig. 11, the maneuver on the left side is designed with equal weights. However this causes a large difference in the fuel consumption between vehicles 1 and 3. To reduce this effect, the maneuver on the right side is designed with weights of 1, 2, and 6 on the fuel usage of vehicles 1, 2, and 3. Although this results in an increase in the total fuel used by the fleet (14%), it provides a much more evenly distributed allocation of the fleet re-

Table 3: Reconfiguration Maneuvers

Vehicles	Unweighted	Weighted
Vehicle 1	10.2%	20.2%
Vehicle 2	36.2%	53.9%
Vehicle 3	53.6%	25.9%
Normalized Total Fuel Usage	1	1.14

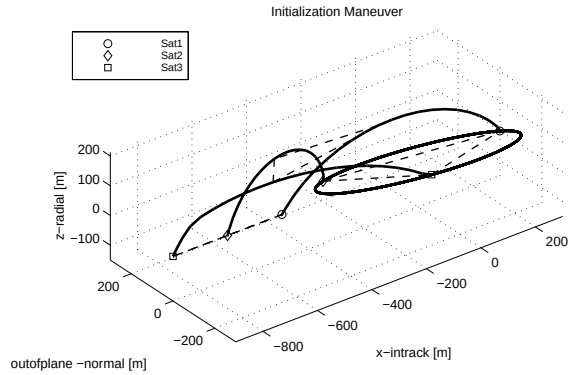

Fig. 13: Initialization to a closed-form solution.

sources (fuel of vehicles 1 and 3 within 6%), thereby increasing the operation life-time (see Table 3).

While linear programming for formation planning is very general and can be used to solve many problems with fixed initial and terminal constraints [24], it has its limitations. In particular, extensions such as maintaining fixed relative distances throughout a maneuver involving both translation and rotation, introduce nonlinearities to the problem structure. This effect can be observed from kinematics of relative motion that include both translation and rotation. The following presents another application of LP to design maneuvers for LEO spacecraft. The third example then investigates alternative solution techniques for these nonlinear problems.

Formation initialization, requires that the fleet of satellites be taken from their parking (or initial) orbits to some desired configuration in a fuel optimal sense. Figure 13 shows an example of this maneuver with three satellites. The vehicles are initially located in a parking orbit with a 200m in-track separation. The objective is to locate them on a closed-form solution of the Clohessy-Wiltshire equations [23] with a circular relative ground track (200m radius). The vehicles are initialized to this configuration with respect to a control reference frame, 200m infront of the lead vehicle.

In the final configuration, the satellites are phased at 120°, resulting in constant relative ground track and fixed relative separations on an Earth projection. However, it is expected that differential drag and J_2 effects will cause secular perturbations from this closed form solution and require a feedback control system to track the desired locations. The control scheme shown in Fig. 10 could then be used to maintain the vehicles at the desired relative positions on the orbit.

In the proposed control scheme for formation flying, formation planning presents the high-level control architecture above a low-level control based on tracking. We have shown that linear programming can be used to solve many of the formation planning problems based on distributed satellite systems. Basic maneuvers related to such planning problems include formation initialization, reconfiguration, and resizing. These linear solutions can be used to provide good initial guesses for the nonlinear optimization of more complex maneuvers.

Formation retargeting: For the *rigid body motion* problem, the basic structure of the formation planning algorithm is carried out with the inclusion of constraints that the vehicles keep fixed positions with respect to a local reference frame attached to the lead vehicle. To implement this maneuver on the *FFTB*, the vehicles are initially separated by 1 m. The objective is to complete a collective 90° turn and translation maneuver to a new desired location in 40 seconds while keeping the arm length error to within 5% of the original baseline. With these constraints, the problem is nonlinear, and the formation retargeting maneuver experimental run shown in Fig. 14 was solved by using the *CONSTR* algorithm [9], which is based on a SQP solver. The formation is initially located at the bottom with the leader (vehicle #1) located to the left (moving to the top) and vehicle #2 is below and to the right of the leader.

The primary difficulty with nonlinear optimization for this problem is that it is not particularly efficient and it requires a good initial estimate of the thrust profiles to produce a useful trajectory. Thus the LP solution is used to produce an approximate initial guess (leader fuel-optimal) [8]. We can generate the thrust sequence by solving for the fuel optimal trajectory of the lead vehicle. The thrust profiles of the follower vehicles can then be derived from the relative kinematics assuming *rigid body motion*. By iterating on the thrust limits of the leader vehicle, we can ensure that the follower vehicles do not exceed their available thrust limits. This technique provides a good initial estimate for the CONSTR algorithm. Fig. 15 shows the resulting leader-fuel optimal solution as captured in experimental run on the *FFTB*.

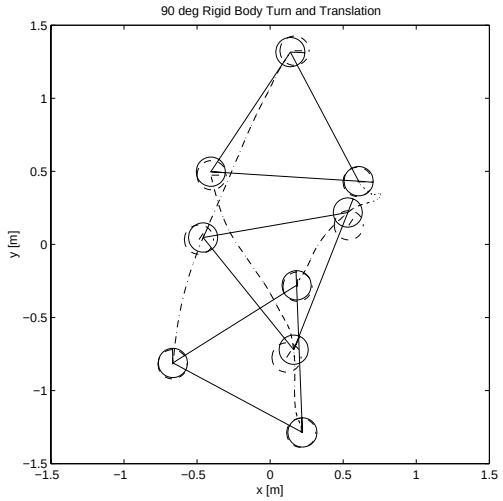

Fig. 14: Collective 90° turn and translation maneuver, Formation-Fuel Optimal – CONSTR.

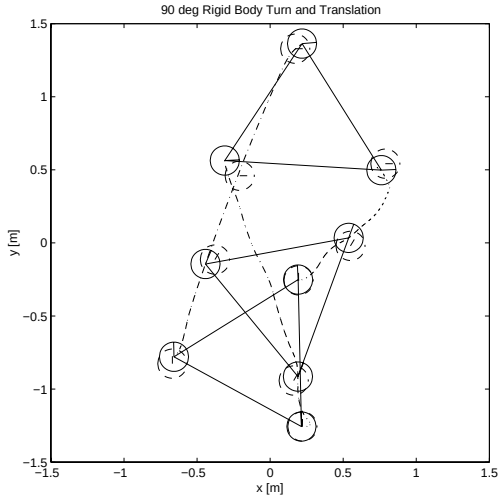

Fig. 15: Collective 90° turn and translation maneuver, Leader-Fuel Optimal – LP

Table 4 summarizes the fuel usage of the vehicles as a result of the feedforward forces generated from the planner for the simulated maneuvers. The fuel amounts are normalized with respect to the fuel used for continuous thrusting in one direction for one second. The formation-fuel optimal solution results in a 14.3% savings in fuel compared to the leader-fuel optimal solution. Also, note that, the fuel usage of the leader vehicle increases by 14.8% where as the fuel usage of the follower vehicles decreases by 7.5% and 38.9% from the initial estimate.

The leader and fleet fuel-optimal maneuvers were both implemented on the formation flying testbed using the control scheme in Fig. 10. The planner supplied a pre-calculated initial plan in the form of

Table 4: Simulated results of feedforward fuel usage.

Vehicles	Leader Fuel optimal	Fleet Fuel optimal
Vehicle 1	6.50	7.46
Vehicle 2	7.00	6.47
Vehicle 3	8.02	4.90
Normalized Total Fuel Usage	21.52	18.83

the desired position, velocity and orientation for the lead vehicle and the forces in x and y directions for all the vehicles. Maneuvers were generated at 2 Hz and sampled at 15 Hz.

The regulation for the vehicles is generated onboard at 60 Hz. For the leader vehicle, the feedback is based on the error in desired position and velocity with respect to a control reference frame. The feedback for the follower vehicles is generated from the relative position and velocity errors with respect to the leader vehicle.

Table 5 presents the 1-σ error in lead vehicles position and orientation, and also the errors in relative distance of the follower vehicles for both experimental runs. The results show that we have met the desired 5% error bounds on the arm lengths.

Table 6 summarizes the fuel usage of the vehicles as a result of the experimental feedforward forces generated from the planner. The difference between the experimental and simulated feedforward forces is a result of the fact that the available minimum impulse bit is 1/60th of a second long and a ceiling approximation was used for the on/off control of the thruster valves. Thus for a maneuver length of 40 seconds, this causes an error of approximately 1.33 units of fuel per direction (worst case approximation).

Finally, Table 7 shows that the fleet fuel-optimal maneuver uses less fuel than the leader fuel-optimal

Table 5: Standard deviations

Elements	Leader Optimal 1-σ Error [m]	Fleet Optimal 1-σ Error [m]
Leader_x	0.033	0.034
Leader_y	0.022	0.028
Length_{12}	0.024	0.014
Length_{13}	0.011	0.008
Leader_θ	0.047 rad	0.059 rad

Table 6: Experimental results of feedforward fuel usage.

Vehicles	Leader Fuel optimal	Fleet Fuel optimal
Vehicle 1	6.53	8.83
Vehicle 2	8.63	8.05
Vehicle 3	9.62	6.45
Normalized Total Fuel Usage	24.78	23.33

Table 7: Experimental results of total (feedforward and feedback) fuel usage.

Vehicles	Leader Fuel optimal	Fleet Fuel optimal
Vehicle 1	7.92	10.05
Vehicle 2	37.48	27.04
Vehicle 3	30.48	24.15
Normalized Total Fuel Used	75.88	61.24

design. The reason for such a difference in the fuel used when compared to the feedforward totals is directly related to the control architecture selection. The current architecture implemented on the *FFTB* is based on a leader-referenced relative control, thus the leader's position and orientation errors (major error source) propagate through the relative control chain. This increases the fuel used by the follower vehicles for feedback control. For the leader vehicle, which is controlled with respect to the table reference frame, the initial plan provides a very fuel efficient path and feedback provides minor corrections due to disturbances and modeling errors. If the plan can be redsigned in real-time, then it might be possible to reduce this overall increased fuel usage for the fleet.

In the experiments conducted we have shown an implementation of a leader-referenced control architecture and identified several key issues regarding formation flying control. In architectures based on relative control, propagation of errors from the lead vehicle results in increased fuel usage of the follower vehicles and probably requires that the planner update the formation plan in real-time. In cases with no plan updates, the formation fuel usage will be dominated by the follower vehicles. This has a direct impact on mission life due to the large fuel imbalance and will require a formation coordinator mechanism to address this resource allocation problem.

6 Conclusions

In this paper we have examined the design of a formation control system including low-level (formation keeping algorithms) and high-level (resource coordination) algorithms. Two planning algorithms have been developed and applied to several typical formation maneuvers. Simulation results show fuel-balancing can be achieved by the planner using weights chosen by a coordinating agent. A retargting maneuver was investigated experimentally on the 2-D *FFTB*. Low-level control algorithms were also compared experimentally for two vehicles. A method for obtaining optimal control laws for three or more vehicles was presented. The control architecture design was investigated, and the need for a high-level coordinator to manage formation resources was highlighted. Thus many aspects of the formation problem have been analyzed and these results will be applied to Orion.

References

[1] J. How, R. Twiggs, D. Weidow, K. Hartman, and F. Bauer, "Orion: A low-cost demonstration of formation flying in space using GPS," in *AIAA Astrodynamics Specialists Conf.*, Aug 1998.

[2] 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, Nov 1998.

[3] C. A. Beichman, "The terrestrial planet finder: The search for life-bearing planets around other stars," in *Proc. of the SPIE Conf. on Astronomical Interferometry*, March 1998.

[4] F. Bauer, J. Bristow, D. Folta, K. Hartman, D. Quinn, and J. P. How, "Satellite formation flying using an innovative autonomous control system (AUTOCON) environment," in *Proc. of the AIAA GNC*, Aug 1997.

[5] AFRL Space Vehicles Directorate, http://www.vs.afrl.af.mil/factsheets/TechSat21.html.

[6] T. Corazzini, A. Robertson, J. C. Adams, A. Hassibi, and J. P. How, "GPS sensing for spacecraft formation flying," *ION GPS-97*, Sep 1997.

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

[8] A. Robertson, G. Inalhan, and J. P. How, "Formation control strategies for a separated spacecraft interferometer," in *Proc. of 1999 ACC*, (San Diego, CA), June 1999.

[9] A. E. Bryson, Jr., *Dynamic Optimization.* Menlo Park, CA: Addison-Wesley, 1999.

[10] A. Bryson, *personal conversation,* April 1999.

[11] A. Robertson, T. Corazzini, and J. P. How, "Formation sensing and control technologies for a separated spacecraft interferometer," in *Proc. of 1998 ACC*, June 1998.

[12] J. G. Walker, "The geometry of satellite clusters,"

Journal of the British Interplanetary Society, vol. 35, pp. 345–354, 1982.

[13] D. F. Chichka and J. L. Speyer, "Solar-powered, formation-enhanced aerial vehicle systems for sustained endurance," in *Proc. of 1998 ACC*, June 1998.

[14] H. Yamaguchi, "A cooperative hunting behavior by mobile robot troops," in *Proc. of the 1998 IEEE Int. Conf. on Robotics and Automation*, IEEE, May 1998.

[15] Q. Chen and J. Y. S. Luh, "Coordination and control of a group of small mobile robots," in *Proc. of the 1998 IEEE Int. Conf. on Robotics and Automation*, 1994.

[16] P. K. C. Wang, "Navigation strategies for multiple autonomous mobile robots moving in formation," *Journal of Robotic Systems*, vol. 8, no. 2, pp. 177–195, 1998.

[17] T. Balch and R. C. Arkin, "Behavior-based formation control for multirobot teams," *IEEE Transactions on Robotics and Automation*, vol. 14, pp. 926–939, Dec 1998.

[18] C. Adams, A. Robertson, K. Zimmerman, and J. P. How, "Technologies for spacecraft formation flying," in *Proc. of the ION GPS-96 Conf.*, (Kansas City, MO), Sep 1996.

[19] S. Boyd and L. Vanderberghe, "Convex optimization." Course reader for EE364 class at Stanford Univ., 1999.

[20] M. H. Kaplan, *Modern Spacecraft Dynamics & Control*. New York, NY: John Wiley & Sons, 1976.

[21] JPL, "DS-3 mission partner visits." Presented at JPL, June 1998.

[22] D. C. Folta, L. K. Newman, and T. Gardner, "Foundations of formation flying for mission to planet earth and new millennium," in *AIAA Guidance, Navigation and Control Conf.*, (San Diego, CA), July 1996.

[23] R. Sedwick, D. Miller, and E. Kong, "Mitigation of differential perturbations in clusters of formation flying satellites," in *The Journal of the Astronautical Sciences, Vol.52, No.2*, April-June 1999.

[24] T. Kia, "DS3: New millenium interferometer formation flying architecture." Presented at JPL, June 1997.

[25] G. Blackwood, S. Dubovitsky, R. Linfield, and P. Gorham, "Interferometer instrument design for new millenium deep space 3," in *SPIE International Symposium on Astronnomical Telescopes and Instrumentation*, March 1998.