DECENTRALIZED, LOW-COMMUNICATION STATE ESTIMATION AND OPTIMAL GUIDANCE OF FORMATION FLYING SPACECRAFT

Dan Dumitriu*, Sónia Marques[§], Pedro U. Lima[†], Bogdan Udrea[‡]

This paper presents an integrated approach to GNC of formation flying spacecraft. The Navigation algorithm estimating the full relative state of all the spacecraft is a full-order decentralized filter, based on an Extended Kalman Filter for local measurements, and on Covariance Intersection for the fusion between local state estimates and estimates communicated by other spacecraft, eliminating EKF divergence problems. For Guidance and Control, an algebraic closed-loop algorithm, based on Pontryagin's maximum principle, is proposed, minimizing the propellant consumption and ensuring collision avoidance. This algorithm is regularly recomputed. Simulation results for a GTO 3-spacecraft formation are presented.

INTRODUCTION

This paper presents an integrated approach to Guidance, Navigation and Control (GNC) of formation flying spacecraft in geostationary transfer orbit (GTO). The Formation Acquisition Manoeuvre (FAM) part of the mission is covered in this paper. FAM is centred around apogee, and its goal is to bring the 3 spacecraft from an initial randomly dispersed disposition within a large sphere, to the desired final disposition, which is a tight formation. In terms of gather-formation distances to be achieved and of propellant consumption, the task volume of the FAM mode is the most important.

The GNC approach is divided in two parts: Navigation, providing the formation full state estimates (i.e., the 3 relative position components and the 3 relative linear velocity components), and Guidance and Control (GC), providing the control inputs to achieve the goal while minimizing the propellant consumption and avoiding collisions.

* Post-doc student at the Institute for Systems and Robotics, Instituto Superior Técnico, Av. Rovisco Pais 1, 149-001 Lisbon, Portugal. Email: ddumitriu@isr.ist.utl.pt. Since December 2005, researcher at the Solid Mechanics Institute, Str. Constantin Mille, nr. 15, Sector 1, Bucharest, Romania. Email: dumitri04@yahoo.com

§ Ph.D Student at Instituto Superior Técnico, Av. Rovisco Pais 1, 149-001 Lisbon, Portugal. Email: sm3@isr.ist.utl.pt

† Assistant Professor at Instituto Superior Técnico, Technical University of Lisbon, Av. Rovisco Pais 1, 149-001 Lisbon, Portugal. Email: pal@isr.ist.utl.pt

‡ ESA/ESTEC – Keplerlaan 1, Postbus 299, 220 AG Noordwijk, Netherlands. Email:Bogdan.Udrea@esa.int

The Navigation algorithm that we have worked out, implemented and tested in order to estimate the full state of all s/c in formation flying, is a full-order decentralized filter. The communications network between s/c is not fully connected, thus reducing the communications between s/c, effectively decentralizing the information handled during the estimation process. With the objective of computing state estimates with high accuracy, in a decentralized multi-systems architecture the main problems are the computational load, communication bottleneck and the risk to fail. In order to estimate the full state decentralized filter of all s/c in formation flying, the algorithm presented is based, for filtering, on an EKF for local measurements (when the observations are from the relative R/F sensors) and on a CI algorithm for the measurements communicated by a predecessor s/c (when no measurements from the relative R/F sensors are available).

For Guidance and Control, we are using an optimal trajectory planning algorithm that minimizes the propellant consumption. Control inputs limitations and collision avoidance are ensured a posteriori. This algorithm computes the spacecraft trajectories from the knowledge of the formation linearized dynamics and full state, and is similar to Tillerson's algorithm[1], although the optimal solutions are obtained using Pontryagin's maximum principle formulation, providing some advantages *w.r.t.* Tillerson's linear programming based method. By re-computing the trajectory at regular time intervals, formation Control is also accomplished, obtaining continuous control solutions.

RELATIVE FORMATION DYNAMICS

The developed algorithms are model-based, therefore the orbital dynamics equations must be determined, as well as the reference frames used and the transformation between the frames.

Reference frames

Concerning the dynamics, the Local Vertical Local Horizon frame (LVLH) and the Inertial Planet Frame (IPQ) are considered, as follows:

1. The Inertial Planet Frame is the reference inertial coordinate system, defined by:
   - Origin: Earth mass center;
   - $\vec{x}_{IPQ}$ axis: in the equator plane, parallel to the Earth vernal equinox direction;
   - $\vec{y}_{IPQ}$ axis: completes the frame;
   - $\vec{z}_{IPQ}$ axis: from the Earth mass center towards North.

2. The Local Vertical Local Horizon frame (see Figure 1) is used to locate the three s/c with respect to the reference orbit:
   - Origin: located on the reference orbit;
   - $\vec{x}_{LVLH}$: completes the right-hand frame;
   - $\vec{y}_{LVLH}$: is normal to the orbital plane, opposite the angular momentum vector of the orbit;
   - $\vec{z}_{LVLH}$: points in the nadir direction.

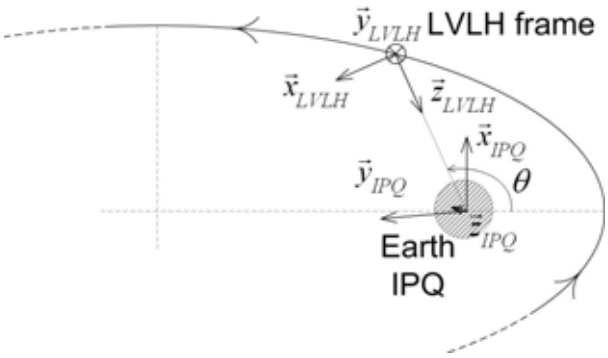

Figure 1 LVLH, view from above the orbital plane

The transformation matrix between LVLH and the IPQ, $\mathbf{R}_{LVLH}^{IPQ}$, is defined using the orbital parameters right ascension of the ascending node Ω, inclination i, argument of perigee ω and true anomaly θ [8], as follows

$$\mathbf{R}_{LVLH}^{IPQ} = [\mathbf{R}_z(\omega+\theta)\mathbf{R}_x(i)\mathbf{R}_z(\Omega)]\begin{bmatrix}0 & 0 & -1\\ 1 & 0 & 0\\ 0 & -1 & 0\end{bmatrix} \tag{1}$$

The transformation between representations of a position vector $\vec{x}$ from LVLH to IPQ is: $\vec{x}^{IPQ} = \mathbf{R}_{LVLH}^{IPQ}\,\vec{x}^{LVLH}$. For velocity vectors, we have: $\dot{\vec{x}}^{IPQ} = \dot{\mathbf{R}}_{LVLH}^{IPQ}\,\vec{x}^{LVLH} + \mathbf{R}_{LVLH}^{IPQ}\,\dot{\vec{x}}^{LVLH}$.

Relative Dynamics for Eccentric Orbits

The two other orbital parameters are the semi-major axis a and the eccentricity e. The natural frequency n of the reference orbit is defined by:

$$n = \sqrt{\frac{\mu}{a^3}} \quad \text{where } \mu = G\cdot m_{Earth} = 3.986\cdot 10^5\,\frac{\text{km}^3}{\text{s}^2} \tag{2}$$

The orbit's true anomaly θ increases monotonically with time t and provides a natural basis for parameterizing the s/c motion. Thus, the differential dynamics equations will be expressed *w.r.t.* θ, rather than to t. For elliptic orbits, the relation between t and θ is[8]:

$$t - t_p = \frac{1}{n}\left[2\arctan\left(\sqrt{\frac{1-e}{1+e}}\tan\frac{\theta}{2}\right) - \frac{e\sqrt{1-e^2}\sin\theta}{1+e\cos\theta}\right] \tag{3}$$

where t_p is the passage time at the perigee.

The motion of each s/c in the formation is described *w.r.t.* the virtual s/c VSC_4, located in LVLH's origin. There are three s/c in the formation and subscript i=1,2,3 will designate each of them. In the LVLH frame, the set of linearized θ-varying equations which describes the relative motion of the i^{th} spacecraft (denoted s/c$_i$) in an eccentric orbit is[1,2]:

$$\frac{d}{d\theta}\begin{bmatrix} x_i \\ x_i^{'} \\ z_i \\ z_i^{'} \end{bmatrix} = \mathbf{A}_{xz}(\theta)\begin{bmatrix} x_i \\ x_i^{'} \\ z_i \\ z_i^{'} \end{bmatrix} + \frac{(1-e^2)^3}{(1+e\cos\theta)^4 n^2}\begin{bmatrix} 0 & 0 \\ 1 & 0 \\ 0 & 0 \\ 0 & 1 \end{bmatrix}\begin{bmatrix} f_{x,i} \\ f_{z,i} \end{bmatrix} \tag{4}$$

$$\frac{d}{d\theta}\begin{bmatrix} y_i \\ y_i^{'} \end{bmatrix} = \mathbf{A}_{y}(\theta)\begin{bmatrix} y_i \\ y_i^{'} \end{bmatrix} + \frac{(1-e^2)^3}{(1+e\cos\theta)^4 n^2}\begin{bmatrix} 0 \\ 1 \end{bmatrix} f_{y,i} \tag{5}$$

where $\mathbf{A}_{xz}(\theta)$ is the following 4×4 matrix:

$$\mathbf{A}_{xz} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ \frac{e\cos\theta}{1+e\cos\theta} & \frac{2e\sin\theta}{1+e\cos\theta} & \frac{-2e\sin\theta}{1+e\cos\theta} & 2 \\ 0 & 0 & 0 & 1 \\ \frac{2e\sin\theta}{1+e\cos\theta} & -2 & \frac{3+e\cos\theta}{1+e\cos\theta} & \frac{2e\sin\theta}{1+e\cos\theta} \end{bmatrix} \tag{6}$$

and $\mathbf{A}_{y}(\theta)$ is the following 2×2 matrix:

$$\mathbf{A}_{y}(\theta) = \begin{bmatrix} 0 & 1 \\ \frac{-1}{1+e\cos\theta} & \frac{2e\sin\theta}{1+e\cos\theta} \end{bmatrix} \tag{7}$$

Matrices $\mathbf{A}_{xz}(\theta)$ and $\mathbf{A}_{y}(\theta)$ depend only on θ. x_i, y_i and z_i are the coordinates in LVLH frame of the relative position vector of s/c$_i$ (i=1,2,3) *w.r.t. VSC*$_4$. $x_i^{'}$ denotes $\frac{dx_i}{d\theta}$, $y_i^{'} = \frac{dy_i}{d\theta}$ and $z_i^{'} = \frac{dz_i}{d\theta}$. The relative positions x_i, y_i and z_i and the relative velocities $x_i^{'}$, $y_i^{'}$ and $z_i^{'}$ characterize the state of s/c$_i$ *w.r.t. VSC*$_4$. The relative dynamics Eqs. (4) describe the in-plane motion, while Eqs. (5) describe the out-of-plane motion. $f_{x,i}$, $f_{y,i}$ and $f_{z,i}$ are the components in LVLH of the external forces vector $\vec{f}_i$, which includes the control inputs $\vec{u}_i$ acting on s/c$_i$ and the differential perturbations experienced by s/c$_i$:

$$\vec{f}_i = \vec{u}_i + \sum \vec{w}_i \tag{8}$$

The differential perturbations are the relative perturbations experienced by s/c$_i$ *w.r.t.* the perturbations affecting *VSC*$_4$. There are several perturbations: J_2 effect, third-body (Sun, Moon) gravitational perturbations, solar radiation pressure, atmospheric drag, micrometeoroids. In this paper, the relative dynamics equations used by the GNC algorithms neglect the differential perturbations, i.e., $\sum \vec{w}_i = 0$, these differential perturbations being small for the considered relative distances.

NAVIGATION

A decentralized navigation architecture is used. The full relative state estimates are computed at each spacecraft (s/c), avoiding the fully-connected communication network. In GTO, the access to GPS signals is not possible or very limited, and thus absolute positioning sensors are not available onboard. The navigation algorithm obtains its relative measurements from a RF system installed onboard. In order to update the estimates that are not locally estimated through the RF measurements, the predecessor s/c state vector estimates are used as *measurements*. A correlation problem arises when the local states estimates are combined with the new *measurements*, leading the EKF to diverge. This problem can be avoided by using the Covariance Intersection (CI) algorithm[5]. However, the use of CI algorithm leads to reduced accuracy, since the error of the combined estimates is lower bounded by the error of the EKF. Thus, the filtering part of the estimation filter is divided in two steps: the calculation of the local state estimates through the local sensor measurements performed by the Extended Kalman filter (EKF), and the update of the remaining state vector variables, by using the predecessor s/c state navigation knowledge.

Formation Flying State Vector

Relative states are more convenient to represent the formation state than the absolute states since the sensor available is the RF subsystem where the observations are inherently expressed between s/c, and the number of equations and variables to be determined by each s/c will lead to an undetermined system, as there are 6 variables, $x_j, y_j, z_j, x'_j, y'_j, z'_j$, to be determined, with 3 measurements only per s/c.

Let us define the formation relative state vector for three s/c (see Figure 2) as:

$$X = \left[(\vec{\rho}_{12})^T \quad (\vec{\rho}_{32})^T \quad (\vec{\rho}_{13})^T \quad (\vec{\rho}'_{12})^T \quad (\vec{\rho}'_{32})^T \quad (\vec{\rho}'_{13})^T\right]^T \tag{9}$$

where:

- ρ' denotes the derivative w.r.t. the true anomaly θ, $\rho' = d\rho / d\theta$
- $\vec{\rho}_{ij} = \left[x_j - x_i \quad y_j - y_i \quad z_j - z_i\right]^T$ is the relative vector between s/c *i* and s/c *j*, where *i,j*=1,2,3. Also, $\vec{\rho}_{ij} = \vec{\rho}_j - \vec{\rho}_i = -\vec{\rho}_{ji}$ with $\vec{\rho}_i = \left[x_j \quad y_j \quad z_j\right]^T$

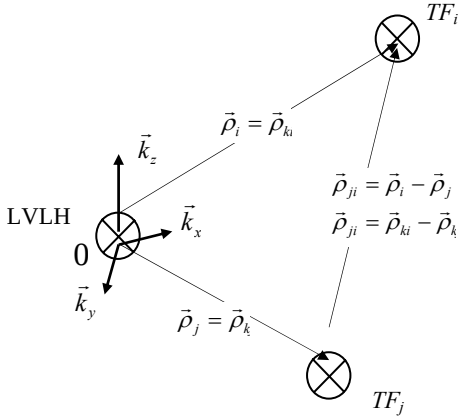

Figure 2 Representation of the relative states between s/c *i, j, k*, considering LVLH frame placed in s/c *k*.

RF Measurements

In the case of the pseudo-range signals (which are the only ones available in the simulator) the mathematical representation is as follows:

$$\begin{aligned}\rho_i^j &= \left\|\vec{\rho}_i - \vec{\rho}_j\right\| + c(T^j - T^i) + E_{multipath}^{i,j} + \varepsilon_\rho^{i,j} \\ &= \vec{\rho}_{ji} + c\Delta T^{ji} + \nu_i\end{aligned} \tag{10}$$

where:

- ρ_i^j is the code phase between transmitting spacecraft *i* and receiver spacecraft *j*, $\left\|\vec{\rho}_i - \vec{\rho}_j\right\| = \sqrt{(x_j - x_i)^2 + (y_j - y_i)^2 + (z_j - z_i)^2}$ is the equation that relates the true distance, between s/c *i* at time of signal transmission and s/c *j* at measurement time, with the formation state, in body reference frame
- T^i and T^j are the time bias of the receiving and transmitting satellite clock,
- c is the speed of light (3×10^8 m/s),
- $\varepsilon_\rho^{i,j}$ is the pseudorange measurement noise due to the receiver thermal noise,
- $E_{multipath}^{i,j}$ represents the multipath error.

When s/c *j* is receiving signals from s/c *i*, one must take into account the disposition of the three receiving antennas in the s/c (Figure 3):

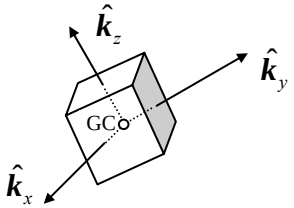

Figure 3 The three antennas, R_1, R_2, R_3, are placed in the sides of the s/c in the positions [*ap* 0 0], [0 *ap* 0], [0 0 *ap*] meters respectively w.r.t. the body frame.

- for receiving antenna 1: $\rho_i^{j,R_1} = \left\|\vec{\rho}_i - \vec{\rho}_j\right\|^{R_1} + c(T^j - T^i) + E_{multipath}^{i,j} + \varepsilon_\rho^{i,j}$
- for receiving antenna 2: $\rho_i^{j,R_2} = \left\|\vec{\rho}_i - \vec{\rho}_j\right\|^{R_2} + c(T^j - T^i) + E_{multipath}^{i,j} + \varepsilon_\rho^{i,j}$
- for receiving antenna 3: $\rho_i^{j,R_3} = \left\|\vec{\rho}_i - \vec{\rho}_j\right\|^{R_3} + c(T^j - T^i) + E_{multipath}^{i,j} + \varepsilon_\rho^{i,j}$

as shown in Figure 4.

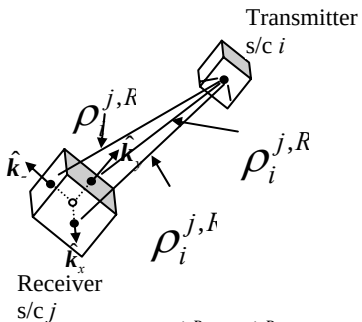

Figure 4 Three measurements, ρ_i^{j,R_1}, ρ_i^{j,R_2}, ρ_i^{j,R_3}, transmitted from s/c *i* are received by each antenna, R_1, R_2, R_3, on s/c *j*.

However the receiver antennas are not placed in the centre of the s/c, but placed *ap* meter ahead in each side of the s/c as shown in Figure 4. Antenna 1 is placed *ap* meter ahead from the gravity centre in *x* direction of the body frame $[ap \quad 0 \quad 0]$, antenna 2 is placed *ap* meter ahead from the gravity centre in *y* direction of the body frame $[0 \quad ap \quad 0]$, and antenna 3 is placed *ap* meter ahead from the gravity centre in *z* direction of the body frame $[0 \quad 0 \quad ap]$.

Thus, there are three measurements instead of just one, as shown in Eq. (10). Therefore, the relations between measurements received in the three antennas, $\rho_i^{j,R_1}, \rho_i^{j,R_2}, \rho_i^{j,R_3}$ and states are:

$$\begin{aligned} \left\|\vec{\rho}_i - \vec{\rho}_j\right\|^{R_1} &= \sqrt{(x_{ij}^b - ap)^2 + (y_{ij}^b)^2 + (z_{ij}^b)^2} \\ \left\|\vec{\rho}_i - \vec{\rho}_j\right\|^{R_2} &= \sqrt{(x_{ij}^b)^2 + (y_{ij}^b - ap)^2 + (z_{ij}^b)^2} \\ \left\|\vec{\rho}_i - \vec{\rho}_j\right\|^{R_3} &= \sqrt{(x_{ij}^b)^2 + (y_{ij}^b)^2 + (z_{ij}^b - ap)^2} \end{aligned} \tag{11}$$

everything expressed in body reference frame.

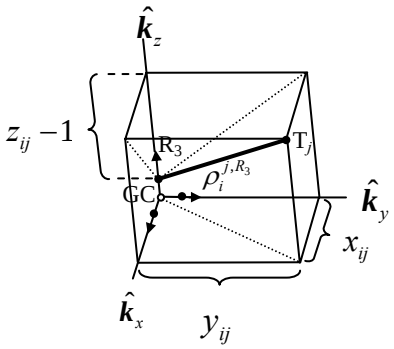

Figure 5 Geometric representation of the measurement received by antenna 3 in the s/c *i* and transmitted by the s/c *j*.

Since the state vector is expressed in LVLH, the variables that are referred in the observations should be transformed into LVLH reference frame. Given the matrix that transforms a position vector from the body frame to IPQ frame, R_{body}^{IPQ}, the matrix that transforms a position vector from LVLH to the s/c body frame is:

$$R_{LVLH}^{body} = (R_{body}^{IPQ})^T R_{LVLH}^{IPQ} \tag{12}$$

Thus, any position vector $\vec{x}$ can be transformed from the body reference frame to LVLH frame by:

$$\vec{x}^{LVLH} = R_{body}^{LVLH} \vec{x}^{body} \tag{13}$$

In particular:

$$\begin{bmatrix} x_{ij}^b & y_{ij}^b & z_{ij}^b \end{bmatrix}^T = R_{LVLH}^{body} \begin{bmatrix} x_{ij}^{LVLH} & y_{ij}^{LVLH} & z_{ij}^{LVLH} \end{bmatrix}^T \tag{14}$$

or

$$
\begin{aligned}
x_{ij}^{b} &= a_{11}x_{ij}^{LVLH} + a_{12}y_{ij}^{LVLH} + a_{13}z_{ij}^{LVLH} \\
y_{ij}^{b} &= a_{21}x_{ij}^{LVLH} + a_{22}y_{ij}^{LVLH} + a_{23}z_{ij}^{LVLH} \\
z_{ij}^{b} &= a_{311}x_{ij}^{LVLH} + a_{32}y_{ij}^{LVLH} + a_{33}z_{ij}^{LVLH}
\end{aligned} \tag{15}
$$

Applying the previous equation to the measurements given by Eq. (11), the observations obtained by each receiving antenna are then expressed in LVLH frame as follows:

$$
\left\| \vec{\rho}_i - \vec{\rho}_j \right\|^{R_1} = \sqrt{\begin{aligned}
&(a_{11}\, x_{ij}^{LVLH} + a_{12}\, y_{ij}^{LVLH} + a_{13}\, z_{ij}^{LVLH} - ap)^2 + \\
&(a_{21}\, x_{ij}^{LVLH} + a_{22}\, y_{ij}^{LVLH} + a_{23}\, z_{ij}^{LVLH})^2 + \\
&(a_{31}\, x_{ij}^{LVLH} + a_{32}\, y_{ij}^{LVLH} + a_{33}\, z_{ij}^{LVLH})^2
\end{aligned}} \tag{16}
$$

The same for $\left\|\vec{\rho}_i - \vec{\rho}_j\right\|^{R_2}$ and $\left\|\vec{\rho}_i - \vec{\rho}_j\right\|^{R_3}$.

Full-Order Decentralized Filter

The Navigation algorithm is based on an EKF for local measurements, and on a CI algorithm (plus the EKF prediction part) for the measurements communicated by a predecessor s/c. The CI algorithm avoids the possible divergence of the EKF at the receiving s/c, due to correlation between measurements of the s/c in the fleet. For the prediction part the equations remain unchanged.

Prediction

1. $\hat{\mathbf{X}}^i(k+1/k) = \mathbf{F}(k, \hat{\mathbf{X}}^i(k/k))$, where $\mathbf{F}(k, \hat{\mathbf{X}}^i(k \mid k))$ is approximated by a 4th-order four-stage Runge-Kutta method.
2. $\mathbf{P}^i(k+1/k) = \mathbf{\Phi}_k \mathbf{P}^i(k/k)(\mathbf{\Phi}_k)^T + \mathbf{Q}(k/k+1)$, where $\mathbf{Q}$ is the covariance matrix of the process noise, mainly due to unmodeled dynamics, and $\mathbf{\Phi}_k$ is the transition matrix $\mathbf{\Phi}_k \approx \mathbf{I} + \mathbf{F}(\hat{\mathbf{X}}^i(t), t)T_s + \frac{(\mathbf{F}(\hat{\mathbf{X}}^i(t), t)T_s)^2}{2!}$, where T_s is the sampling period used to propagate the estimate and i stands for the i^{th} s/c.

Filtering

For i=1,2,…, N-1, circularly (i.e., 1 comes after N-1)

Sensor Observation, $\mathbf{y}^i(k)$

1. Compute the local observation matrix: $\mathbf{H}^i(k, \hat{\mathbf{X}}^i(k \mid k-1))$ (the linearization of the original observation function, denoted by $\mathbf{H}^i(k)$ for simplification).
2. Compute the local innovation covariance matrix: $\mathbf{S}^i(k) = \mathbf{H}^i(k)\mathbf{P}^i(k \mid k-1)(\mathbf{H}^i(k))^T + \mathbf{R}(k)$
3. Compute the local Kalman Gain: $\mathbf{K}^i(k) = \mathbf{P}^i(k \mid k-1)\mathbf{H}^i(k)(\mathbf{S}^i(k))^{-1}$
4. Update local state estimate: $\hat{\mathbf{X}}^i(k/k) = \hat{\mathbf{X}}^i(k \mid k-1) + \delta\hat{\mathbf{X}}^i(k)$
5. Re-compute the local observation matrix: $\mathbf{H}^i(k, \hat{\mathbf{X}}^i(k \mid k))$
6. Compute the error covariance matrix: $\mathbf{P}^i(k \mid k) = (\mathbf{I} - \mathbf{K}^i(k)\mathbf{H}^i(k))\mathbf{P}^i(k \mid k-1)(\mathbf{I} - \mathbf{K}^i(k)\mathbf{H}^i(k))^T + \mathbf{K}^i(k)\mathbf{R}^i(k)(\mathbf{K}^i(k))^T$

State estimate from predecessor s/c, $\mathbf{z}^i(k) = \mathbf{X}(k \mid k-1) + \mathbf{v}^{i-1}(k)$

1. Compute the error covariance matrix:
   $(\mathbf{P}^i(k \mid k))^{-1} = \omega(\mathbf{P}^i(k \mid k-1))^{-1} + (1-\omega)(\mathbf{P}^{i-1}(k \mid k-1))^{-1}$
2. Update local state estimate:

$$\hat{\mathbf{X}}^i(k \mid k) = \mathbf{P}^i(k \mid k)(\omega(\mathbf{P}^i(k \mid k-1))^{-1}\hat{\mathbf{X}}(k \mid k-1) + (1-\omega)(\mathbf{P}^{i-1}(k \mid k-1))^{-1}\mathbf{z}^i(k)),$$

where the parameter ω is chosen at every step such that the trace of the matrix $\mathbf{P}^+_{k,i}$ is minimized.

The estimates concern the entire fleet state in a decentralized scheme, but without the disadvantages of communication associated to a fully-connected network since only peer-to-peer communication is required with real-time at each time step.

GUIDANCE AND CONTROL

For GC, an optimal trajectory planning algorithm that minimizes the propellant consumption is used[3]. This closed-loop GC algorithm computes the spacecraft trajectories from the knowledge of the formation linearized relative dynamics and the full state, the optimal solutions being obtained using Pontryagin's maximum principle formulation[4]. Control inputs limitations and collision avoidance are ensured a posteriori.

To take the unmodeled perturbations into account, as well as the state estimation errors, this closed-loop GC algorithm is recomputed periodically, at regularly spaced time instants, and the planned trajectory is updated, as well as the associated continuous control solutions.

Model-based Optimal Trajectory Planning

During the Formation Acquisition Maneuver (FAM), i.e., between θ_1 and θ_2, with $\theta_1=\theta(t_1)$ and $\theta_2=\theta(t_2)$ as provided by Eq. (3), the trajectory of each s/c must minimize the propellant consumption and avoid collisions. The optimal trajectory planning problem during FAM includes:

- the state equations;
- the initial and final conditions;
- the limitations concerning the control inputs;
- the cost function to be minimized.

The state equations gather the relative dynamics equations of all three s/c. The global state vector is:

$$\mathbf{X} = \begin{bmatrix} x_1 & x_1' & z_1 & z_1' & y_1 & y_1' & x_2 & x_2' & z_2 & z_2' & y_2 & y_2' & x_3 & x_3' & z_3 & z_3' & y_3 & y_3' \end{bmatrix}^T \quad (17)$$

or, written by components:

$$X_1 = x_1,\ X_2 = x_1',\ X_3 = z_1,\ X_4 = z_1',\ ...,\ X_{18} = y_3'$$

All control inputs are gathered into vector **U**:

$$\mathbf{U} = \begin{bmatrix} u_{1,x} & u_{1,z} & u_{1,y} & u_{2,x} & u_{2,z} & u_{2,y} & u_{3,x} & u_{3,z} & u_{3,y} \end{bmatrix}^T$$

By putting together the linearized θ-varying relative dynamics Eqs. (4) and (5) for all three s/c, the state equations of the model-based optimal trajectory planning problem are:

$$\frac{d\mathbf{X}}{d\theta} = \mathbf{A}(\theta)\mathbf{X} + \mathbf{B}(\theta)\mathbf{U} = \mathbf{f}(\mathbf{X}, \mathbf{U}, \theta) \tag{18}$$

where:

$$\mathbf{A}(\theta) = \begin{bmatrix} \mathbf{A}_{xz} & \mathbf{0}_{4\times2} & \mathbf{0}_{4\times4} & \mathbf{0}_{4\times2} & \mathbf{0}_{4\times4} & \mathbf{0}_{4\times2} \\ \mathbf{0}_{2\times4} & \mathbf{A}_{y} & \mathbf{0}_{2\times4} & \mathbf{0}_{2\times2} & \mathbf{0}_{2\times4} & \mathbf{0}_{2\times2} \\ \mathbf{0}_{4\times4} & \mathbf{0}_{4\times2} & \mathbf{A}_{xz} & \mathbf{0}_{4\times2} & \mathbf{0}_{4\times4} & \mathbf{0}_{4\times2} \\ \mathbf{0}_{2\times4} & \mathbf{0}_{2\times2} & \mathbf{0}_{2\times4} & \mathbf{A}_{y} & \mathbf{0}_{2\times4} & \mathbf{0}_{2\times2} \\ \mathbf{0}_{4\times4} & \mathbf{0}_{4\times2} & \mathbf{0}_{4\times4} & \mathbf{0}_{4\times2} & \mathbf{A}_{xz} & \mathbf{0}_{4\times2} \\ \mathbf{0}_{2\times4} & \mathbf{0}_{2\times2} & \mathbf{0}_{2\times4} & \mathbf{0}_{2\times2} & \mathbf{0}_{2\times4} & \mathbf{A}_{y} \end{bmatrix} \tag{19}$$

with $\mathbf{A}_{xz}(\theta)$ expressed by Eq. (6) and $\mathbf{A}_{y}(\theta)$ by Eq. (7). From Eq. (4) and Eq. (5), it is obvious to express $\mathbf{B}(\theta)$.

The optimal trajectory planning problem is studied between θ_1 and θ_2. Both the initial and the final state are given:

$$\mathbf{X}(\theta_1) = \mathbf{a} \quad \text{and} \quad \mathbf{X}(\theta_2) = \mathbf{b} \tag{20}$$

The control inputs must satisfy the following constraint inequalities:

$$u_{\min} \le |U_j| \le u_{\max}, \quad \text{for } j = 1,\ldots,9 \tag{21}$$

The collision avoidance is considered afterwards, as explained in subsection *A Posteriori Consideration of Collision Avoidance*. In this case, the cost function must minimize only the propellant consumption. The cost function to be minimized is:

$$J = \int_{\theta_1}^{\theta_2} L(\mathbf{X}, \mathbf{U}, \theta) d\theta = \int_{\theta_1}^{\theta_2} \sum_{j=1}^{9} U_j^2 d\theta \tag{22}$$

where $L(\mathbf{X},\mathbf{U},\theta)$ is the weight function. So, by minimizing J, the overall control inputs are minimized. Since the control inputs are proportional to the propellant consumption, the propellant consumption is minimized.

Application of Pontryagin's Maximum Principle

Optimal trajectory planning means to provide, for $\theta_1 \le \theta \le \theta_2$, the optimal trajectories $\mathbf{X}^{opt}(\theta)$ and the associated optimal control inputs $\mathbf{U}^{opt}(\theta)$, which:

- respect the state Eqs. (18);
- meet the two-boundary conditions (20);
- satisfy the control inputs limitations (21);
- minimize the cost function given by (22).

This model-based optimal trajectory planning problem is solved by using Pontryagin's maximum principle (PMP)[4]. This optimization principle introduces 18 co-state (adjoint) variables λ_i, one for each state equation: λ_i corresponds to state equation

$\frac{d\mathbf{X}_i}{d\theta} = \mathbf{f}_i(\mathbf{X},\mathbf{U},\theta)$, for i=1,…,18. These adjoint variables are regrouped into the co-state vector $\mathbf{\Lambda}$:

$$\mathbf{\Lambda} = \begin{bmatrix} \lambda_1 & \lambda_2 & \dots & \lambda_6 & \lambda_7 & \dots & \lambda_{12} & \lambda_{13} & \dots & \lambda_{18} \end{bmatrix}^T \tag{23}$$

By introducing the Hamiltonian:

$$H(\mathbf{X},\mathbf{U},\theta) = L(\mathbf{X},\mathbf{U},\theta) + \mathbf{\Lambda}^T \mathbf{f}(\mathbf{X},\mathbf{U},\theta) \tag{24}$$

and by expressing the co-state equations as:

$$\frac{d\mathbf{\Lambda}}{d\theta} = -\left(\frac{\partial H}{\partial \mathbf{X}}\right)^T = -\left(\frac{\partial L}{\partial \mathbf{X}}\right)^T - \frac{\partial \mathbf{f}^T}{\partial \mathbf{X}}\mathbf{\Lambda}, \tag{25}$$

the PMP states that the control inputs which satisfy, for all $\theta_1 \le \theta \le \theta_2$, the stationarity conditions:

$$0 = \left(\frac{\partial H}{\partial \mathbf{U}}\right)^T = \left(\frac{\partial L}{\partial \mathbf{U}}\right)^T + \frac{\partial \mathbf{f}^T}{\partial \mathbf{U}}\mathbf{\Lambda}, \tag{26}$$

are the optimal control inputs, the corresponding trajectory being optimal as well.

Under the PMP formulation, the stationarity conditions in Eq. (26) provide us with the optimal control inputs $\mathbf{U}_j^{opt}$, as functions of the adjoint variables:

$$\begin{aligned} &0 = \frac{\partial L}{\partial u_{1,x}} + \sum_{k=1}^{18} \frac{\partial f_k}{\partial u_{1,x}}\lambda_k = 2u_{1,x} + \frac{(1-e^2)^3}{(1+e\cos\theta)^4 n^2}\lambda_2 \quad \Rightarrow U_1^{opt} = u_{1,x}^{opt} = -\frac{1}{2}\frac{(1-e^2)^3}{(1+e\cos\theta)^4 n^2}\lambda_2, \\ &U_2^{opt} = u_{1,z}^{opt} = -\frac{1}{2}\frac{(1-e^2)^3}{(1+e\cos\theta)^4 n^2}\lambda_4, \quad (\dots) \end{aligned} \tag{27}$$

So, the linear relation between the optimal control inputs and the adjoint variables is:

$$U_j^{opt} = -\frac{1}{2}\frac{(1-e^2)^3}{(1+e\cos\theta)^4 n^2}\lambda_{2j}, \text{ for } j = 1,\dots,9 \tag{28}$$

By taking into account the stationarity conditions (28), the state Eqs. (18) at θ_k become:

$$\left.\frac{d\mathbf{X}}{d\theta}\right|_{\theta_k} = \mathbf{A}(\theta_k)\mathbf{X}(\theta_k) + \mathbf{B}^{\Lambda}(\theta_k)\mathbf{\Lambda}(\theta_k) \tag{29}$$

where:

$$\mathbf{B}^{\Lambda} = -\frac{1}{2}\left[\frac{(1-e^2)^3}{(1+e\cos\theta)^4 n^2}\right]^2 \begin{bmatrix} \mathbf{I}_{6\times6}^* & \mathbf{0}_{6\times6} & \mathbf{0}_{6\times6} \\ \mathbf{0}_{6\times6} & \mathbf{I}_{6\times6}^* & \mathbf{0}_{6\times6} \\ \mathbf{0}_{6\times6} & \mathbf{0}_{6\times6} & \mathbf{I}_{6\times6}^* \end{bmatrix} \tag{30}$$

with:

$$\mathbf{I}_{6\times6}^* = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \tag{31}$$

Closed-loop GC Algorithm

The differential linear two-boundary equations system to be solved consists of the state Eq. (29) and the co-state Eq. (25). Both initial and final state vectors are known (20), but there is no boundary condition available for the adjoint variables. The differential linear two-boundary equations system is solved by using the purely algebraic algorithm derived below, called closed-loop GC algorithm. To take the unmodeled perturbations into account, as well as the state estimation errors, the closed-loop GC algorithm is recomputed periodically, at regularly spaced time instants, and the planned trajectory is updated.

By using the finite differences expression of the derivative $\left.\frac{d\mathbf{X}}{d\theta}\right|_{\theta_k}$ for a constant step $\delta\theta$ in the true anomaly, and the short notation k instead of θ_k, Eq. (29) becomes:

$$\mathbf{X}(k+1) = [(\delta\theta)\mathbf{A}(k) + \mathbf{I}_{18}]\mathbf{X}(k) + [(\delta\theta)\mathbf{B}^{\Lambda}(k)]\mathbf{\Lambda}(k) \tag{32}$$

where $\mathbf{I}_{18}$ is the identity 18×18 matrix. Finally, the recurrent expression of the state variables is:

$$\mathbf{X}(k+1) = \overline{\mathbf{A}}(k)\mathbf{X}(k) + \overline{\mathbf{B}}(k)\mathbf{\Lambda}(k) \tag{33}$$

where $\overline{\mathbf{A}}(k) = \delta\theta \cdot \mathbf{A}(k) + \mathbf{I}_{18}$ and $\overline{\mathbf{B}}(k) = \delta\theta \cdot \mathbf{B}^{\Lambda}(k)$. Similarly, the co-state Eq. (25) of the simplified form:

$$\left.\frac{d\mathbf{\Lambda}}{d\theta}\right|_{\theta_k} = \mathbf{C}(k)\mathbf{\Lambda}(k) \tag{34}$$

can be transformed into:

$$\mathbf{\Lambda}(k+1) = \overline{\mathbf{C}}(k)\mathbf{\Lambda}(k) \tag{35}$$

where $\overline{\mathbf{C}}(k) = \delta\theta \cdot \mathbf{C}(k) + \mathbf{I}_{18}$.

Based on the recurrent expressions (33) and (35), $\mathbf{X}(k+1)$ can be expressed directly as function of $\mathbf{X}(0)$ and $\mathbf{\Lambda}(0)$, the same for $\mathbf{\Lambda}(k+1)$:

$$\mathbf{X}(k+1) = \mathbf{P}(k)\mathbf{X}(0) + \mathbf{Q}(k)\mathbf{\Lambda}(0) \tag{36}$$

$$\mathbf{\Lambda}(k+1) = \mathbf{N}(k)\mathbf{\Lambda}(0) \tag{37}$$

where $\mathbf{P}(k)$, $\mathbf{Q}(k)$ and $\mathbf{N}(k)$ are given by the following recurrent sequence:

1. $\mathbf{P}(0) = \overline{\mathbf{A}}(0)$, $\mathbf{Q}(0) = \overline{\mathbf{B}}(0)$, $\mathbf{N}(0) = \overline{\mathbf{C}}(0)$
2. FOR k=1 TO n-1

$$\mathbf{P}(k) = \overline{\mathbf{A}}(k)\mathbf{P}(k-1)$$
$$\mathbf{Q}(k) = \overline{\mathbf{A}}(k)\mathbf{Q}(k-1) + \overline{\mathbf{B}}(k)\mathbf{N}(k-1)$$
$$\mathbf{N}(k) = \overline{\mathbf{C}}(k)\mathbf{N}(k-1)$$

The recurrent sequence above is nothing else than propagating dynamics between $\theta_1 = \theta_{k=0}$ and $\theta_2 = \theta_{k=n}$. The number of steps n is related to the true anomaly step $\delta\theta$ by: $\delta\theta = \frac{\theta_2 - \theta_1}{n}$. The estimated initial state $\hat{\mathbf{X}}(\theta_1)$ corresponds to $\mathbf{X}(0)$, while the desired

final state $\mathbf{X}(\theta_2)$ is the same with $\mathbf{X}(n)$. In this case, expression (36) written for k=n-1 becomes:

$$\mathbf{Q}(n-1)\mathbf{\Lambda}(0) = \mathbf{X}(\theta_2) - \mathbf{P}(n-1)\hat{\mathbf{X}}(\theta_1) \tag{38}$$

where $\mathbf{Q}(n-1)$ and $\mathbf{P}(n-1)$ are provided by the recurrent sequence above.

Expression (38) is an algebraic system of 18 linear equations, with unknowns $\mathbf{\Lambda}(0)$, i.e., the initial adjoint variables at θ_1. This linear system is easily solved by the Gauss elimination method. By using Eq. (37), the knowledge of $\mathbf{\Lambda}(0)$ provides us with the knowledge of all $\mathbf{\Lambda}(\theta)$, for $\theta_1 \le \theta \le \theta_2$. Finally, by means of the stationarity conditions (28), all optimal control inputs $\mathbf{U}^{opt}(\theta)$ are known. The optimal trajectories $\mathbf{X}^{opt}(\theta)$ are known as well, by using Eq. (36).

The control inputs limitations, Eq. (21) are considered only a posteriori. The obtained $\mathbf{U}^{opt}(\theta)$ are just not allowed to exceed the limitations: if component $\mathbf{U}_j^{opt}(\theta) > u_{\max}$, then $\mathbf{U}_j^{opt}(\theta) = u_{\max}$ is imposed.

A Posteriori Consideration of Collision Avoidance

In what concerns collision avoidance, let us consider a possible collision situation between s/c$_1$ (i=1) and s/c$_2$ (i=2). Let us denote by $\vec{\rho}_{12}^{so}$ the relative distance between s/c$_1$ and s/c$_2$, where the evolutions of s/c$_1$ and s/c$_2$ follow the sub-optimal trajectories computed by the closed-loop GC algorithm. At each instant θ, if $\vec{\rho}_{12}^{so} < 40\text{m} = \rho_{\min}$, then we add to the sub-optimal control inputs $\vec{u}_i^{so}$ (provided by the closed-loop GC algorithm) some thrust inputs, i.e., $-\vec{u}_{12}$ and respectively $\vec{u}_{12}$, supposed to move away s/c$_1$ and s/c$_2$. Figure 6 illustrates this modification, $\vec{u}_1 = \vec{u}_1^{so} - \vec{u}_{12}$, $\vec{u}_2 = \vec{u}_2^{so} + \vec{u}_{12}$ and $\vec{u}_3 = \vec{u}_3^{so}$ being the control inputs that will be really applied at instant θ.

Concerning the values of these collision avoidance thrust inputs $\pm\vec{u}_{12}$ added, one can consider they follow a parabolic dependence law between abscissas of 40m and 15m, being null for $\vec{\rho}_{12}^{so} \ge 40\text{m}$ and equal to the maximum thrust when $\vec{\rho}_{12}^{so} \le 15\text{m}$.

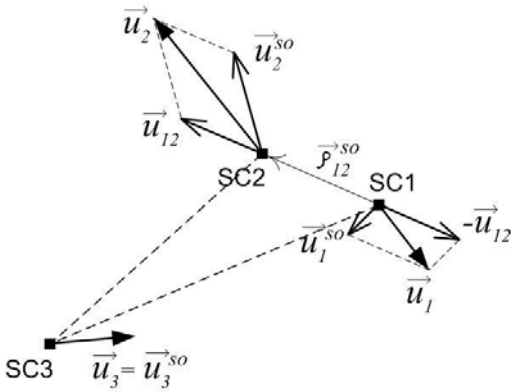

Figure 6 Ensuring collision avoidance a posteriori

GNC RESULTS

The performance of the GNC algorithms was tested in a realistic orbit dynamics simulator. The results below concern a GTO orbit characterized by:

$$a = 26624.1\text{km},\ e = 0.7304,\ \Omega = 0,\ i = 7^\circ,\ \omega = -\frac{\pi}{2}. \tag{39}$$

So far, the GNC simulations are performed with the perturbations disabled. Some further research, especially in the Navigation part, is still needed in order to present satisfactory results with all or part of the perturbations enabled. Note also that s/c_1 is considered coincident with VSC_4, i.e., the origin of LVLH. So, only s/c_2 and s/c_3 are controlled.

The duration of FAM is chosen to be 6h, in order not to saturate the control inputs, which limitations are: $u_{\min} = 0.1\mu\text{N}$ and $u_{\max} = 20\text{mN}$. FAM is centered in duration around apogee, where perturbations are much less significant than close to perigee. More precisely, FAM starts at $t_1 = 10816.94\text{s}$ and ends at $t_2 = 32416.94\text{s}$. The passage time at perigee was considered as the time origin: $t_p = 0$. By using the relation (3), the corresponding true anomalies are: $\theta_1 = 156.5557^\circ$ and $\theta_2 = 203.4442^\circ$.

The closed-loop GC algorithm is recomputed at regularly spaced time instants (every 50s during the last 1h of FAM, and every 400s for the rest of FAM), and the planned optimal trajectory is updated.

The initial state $\mathbf{X}(\theta_1) = \mathbf{a}$ corresponds to a disposition of the three s/c within a sphere of 8km in diameter around the dispenser. The velocities included in $\mathbf{X}(\theta_1)$ have values between ±0.1m/s. In the estimation filter, this initial state has an error of 1m for the positions and 1m/rad for the velocities.

The desired final state $\mathbf{X}(\theta_2) = \mathbf{b}$ corresponds to a tight formation. The goal is to attain, up to 1h before the next orbit's apogee, an isosceles triangle formation with the equal edges of 250m and with a 120º angle between them. To meet this goal mostly by natural motion, using the periodicity conditions[1] for relative motion in eccentric orbits, the requirement is to obtain $\mathbf{X}(\theta_2) = \mathbf{b}$ at the end of FAM. Table 1 presents both **a** and **b**, but only for s/c_2.

Table 1. Initial and final states, in LVLH

	$\mathbf{X}(\theta_1) = \mathbf{a}$	$\mathbf{X}(\theta_2) = \mathbf{b}$
x_2 [m]	3000	-242.10
$\dot{x}_2$[m/s]	-0.04	0.000559
z_2 [m]	-864	62.35
$\dot{z}_2$[m/s]	-0.04	0.012858
y_2 [m]	300	0.00
$\dot{y}_2$[m/s]	-0.04	0.000155

Figures 7-9 show our GNC algorithms results. Figure 7 presents the evolution of the distances between s/c_2 and s/c_1, as well as between s/c_3 and s/c_1. Figure 8 shows the

control inputs in IPQ obtained for s/c_2. Figure 9 presents the real and estimated relative distances of s/c_1 *w.r.t.* s/c_3 (*y vs x* components).

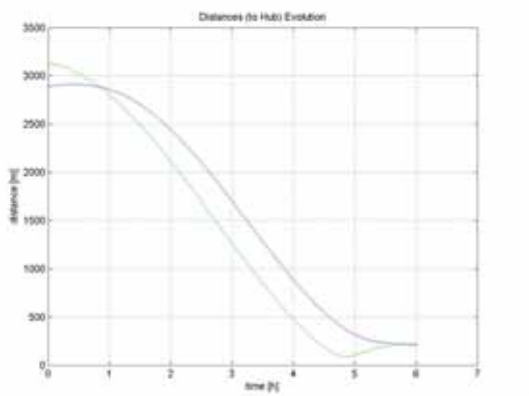

Figure 7 The evolution of the distances between s/c_2 and s/c_1 (green), and between s/c_3 and s/c_1 (blue)

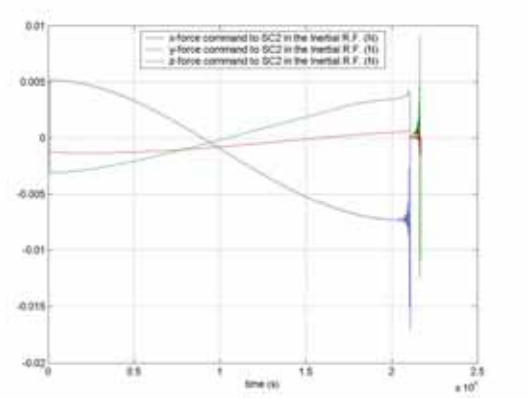

Figure 8 s/c_2 optimal control inputs ($u_{2,x}, u_{2,y}, u_{2,z}$) w.r.t. time *t*, in IPQ

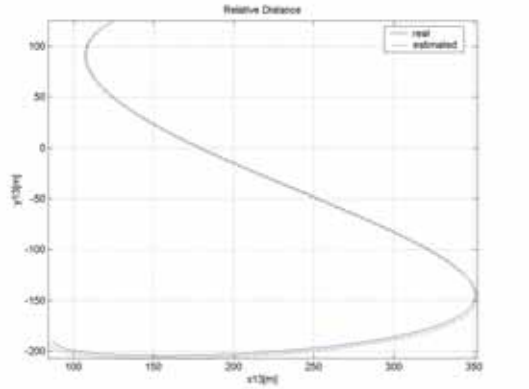

Figure 9 Real and estimated relative distances of s/c1 w.r.t. s/c3 (*y vs x* components).

In terms of performance using closed-loop GNC, the error between the obtained final state $\mathbf{X}(\theta_2)$ and the desired one **b** is maximum 8m for position components and of 0.001m/s for velocities. These results are good, but have still to be improved to better fulfill the mission specifications. Simulations were also done without the estimator in

the loop. In this case only GC is considered and, with all types of perturbations enabled in the simulator, the error is of the order of 0.1m for positions and of 0.0001m/s for velocities, which meets the specifications for the mission.

CONCLUSIONS

This work has been developed under an ESA project where the goal is to obtain simulation results, with all GNC algorithms in the loop, for a 3-spacecraft formation flying in a GTO orbit. For Guidance and Control, this paper presents a model-based optimal trajectory planning algorithm. Our guidance-oriented approach consists in regularly re-computing this algorithm. This (re)planning leads to trajectories that require less control effort during the trajectory tracking phase of the mission. For the Navigation part, the formation state estimation is handled by a full-order decentralized estimator, based on the CI and EKF. The EKF is used for local measurements, and for the measurements communicated by a predecessor spacecraft, in a method with no divergence troubles, i.e., the CI algorithm.

ACKNOWLEDGEMENTS

The authors would like to thank Augusto Caramagno and Juan C. Bastante, from DEIMOS Space, Luis Peñin and João Araújo from DEIMOS Engenharia, for useful insights on formation flying mission analysis and specifications, as well as for making available DEIMOS' FF-FES simulator.

REFERENCES

1. G. Inalhan, M.J. Tillerson and J.P. How, "Relative dynamics and control of spacecraft formations in eccentric orbits", *J. of Guidance, Control, and Dynamics*, Vol. 25, No. 1, January-February 2002, pp. 48-59.
2. M.J. Tillerson, *Coordination and Control of Multiple Spacecraft using Convex Optimization Techniques*, Master thesis, Massachusetts Institute of Technology (MIT), June 2002.
3. D. Dumitriu, P. Lima and B. Udrea, "Optimal trajectory planning of formation flying spacecraft", *In Proceedings of the 16th IFAC World Congress*, July 4-8, 2005, Prague.
4. A.E. Bryson and Y.C. Ho, *Applied optimal control*, Hemisphere Publishing Corporation, NY, 1975.
5. P.O. Arambel, C. Rago and R.K. Mehra, "Covariance Intersection algorithm for Distributed Spacecraft State Estimation", *In Proceedings of the America Control Conference*, Arlington, VA, 2001.
6. C. P. Park, *Precise Relative Navigation Using Augmented CDGPS*, PhD thesis, Stanford University, Dept. of Mechanical Eng, 2001.
7. P. Fergunson, *Distributed Estimation and Control Technologies for Formation Flying Spacecraft*, Master thesis, MIT, 2003.
8. Sidi, M.J. (1997). *Spacecraft Dynamics and Control*. Cambridge University Press.