Model-based guidance and control for atmospheric guided entry

Published by Proc. AIAA Guidance, Navigation, and Control Conference, Minneapolis, 13 - 16 August 2012.

Model-based guidance and control for atmospheric guided entry

Enrico Canuto[1] and Jose A. Ospina[2]
Politecnico di Torino, Torino, 10129, Italy

and

Marcello Buonocore[3]
Thales Alenia Space, Torino, 10146, Italy

This paper presents a solution of the translational control for a biconic atmospheric entry capsule using the bank angle as a command. The control algorithm is separated into path planning and reference-path tracking. The path-planning algorithm computes the entry trajectory from the navigated state at the Entry Interface Point until the desired Parachute Deployment Point. The algorithm aims to recover the landing site uncertainty caused by Entry Interface Point dispersions. Atmospheric and aerodynamic dispersions are compensated in real-time following the Embedded Model Control methodology in which parametric uncertainty is estimated and rejected as an external disturbance. A hierarchical control structure is designed for facilitating non-linear dynamic inversion of the altitude/density relation and tuning of noise estimators and control laws. Both path planning and reference-path tracking exploit longitudinal and lateral decomposition of the translational dynamics, as well as state equation linearization around a reference trajectory. The main concepts and solutions of the algorithms are presented without formal proofs of convergence, performance and stability. The results of a Monte Carlo simulation campaign conducted on a high fidelity simulator are provided and discussed.

Nomenclature

Δa	=	commanded acceleration variation
c_θ	=	cosine of a generic angle θ
c_D	=	drag coefficient
c_L	=	lift coefficient
d_D	=	drag disturbance
d_L	=	lift disturbance
e_{dr}	=	error of the longitudinal state
e_{cr}	=	error of the lateral state
F_D	=	drag force
F_L	=	lift force
g	=	Mars gravity

[1] Professor, Dipartimento di Automatica e Informatica, Corso Duca degli Abruzzi 24, 10129 Torino, Italy, enrico.canuto@polito.it, Senior Member.
[2] Research fellow, Dipartimento di Automatica e Informatica, Corso Duca degli Abruzzi 24, 10129 Torino, Italy, jose.ospina@polito.it. Now with Deimos Space, Ronda de Poniente 19, 28760 Tres Cantos, Madrid, Spain.
[3] GNC engineer. Business Segment Optical Observation and Science, Strada Antica di Collegno 253, 10146 Torino, Italy, marcello.buonocore@thalesaleniaspace.com.

a_{jk} = partial derivative of j with respect to k times the discrete time unit T.
h = altitude above the reference ellipsoid
H_{atm} = scale height of the atmospheric density exponential function
i = discrete-time step (integer)
k = iteration step (integer)
m = number of discrete steps of the entry phase
m = vehicle mass
r = distance to the planet center of mass
R_M = local radius of the planet surface
s = down-range
S = reference aerodynamic surface
s_θ = sine of a generic angle θ
S_{cr} = sensitivity of the lateral state to displacements of the bank reversal starting time
T = discrete equations time unit
t_{kn} = starting time of the n-th bank reversal at the k-th iteration
u_{dr} = cosine of the bank angle
u_{cr} = sign of the bank-angle sine
v = magnitude of the co-rotating velocity
x = translational state $\begin{bmatrix} r & \Lambda & \lambda & v & \gamma & \chi \end{bmatrix}^T$
$x_{p,k}$ = path-planning trajectory at each step k
x_{dr} = longitudinal state $\begin{bmatrix} s & v & h & \gamma \end{bmatrix}^T$
x_{cr} = longitudinal state $\begin{bmatrix} y & \Delta\chi \end{bmatrix}^T$
y = cross-range
χ = co-rotating heading angle
γ = co-rotating flight path angle
Λ = longitude
λ = latitude
μ = bank angle
ω_M = planet angular rate
ρ = atmospheric density
ρ_s = density at zero altitude

I. Introduction

Atmospheric entry, descent and landing on Mars is a challenging problem. From a registered number of 16 landing attempts, only seven of them have been successful. Of the successful missions, the landing dispersion ellipses varied from 80 to 200 km long (semi-major axis), and low altitude landing sites were selected to dispose of a longer atmospheric entry. Landing close to sites with higher scientific interest requires drastically smaller (from tens of km to a pin point target) dispersion ellipses and the capability of reaching higher altitude sites.

Currently, the key solution to such problems is offered by the technology of lifted entry vehicles (Wolf [1]). Lift allows for the parachute to be deployed at higher altitudes and for an active guidance/control of the vehicle trajectory during the atmospheric entry phase. The latter feature is referred to as Guided Entry.

Biconic capsules become controllable by offsetting their center of mass from the geometric axis, what results into a nonzero equilibrium angle of attack and, in turn, into a lifting aerodynamic force. By controlling the bank angle of the capsule, the direction of the lift can be adjusted and the vehicle motion can be (slightly) driven. According to Braun [2], a lifted capsule should be capable of following a dynamic-pressure-Mach-number profile, which is particularly suitable for reaching safe parachute conditions at higher altitudes; further, the capsule could be designed to have higher ballistic coefficients, and consequently to allow heavier payloads.

Lifted biconic entry capsule with a guided entry algorithm have been tested on Earth, first by NASA on the Apollo missions [3] and then by ESA with an Atmospheric Reentry Demonstrator [4]. In both cases the same "Apollo-derived" guided entry was adopted: a drag-range profile is tracked using a linear feedback law to adjust the lift vertical component, while bank reversal maneuvers (BRM) are commanded each time the capsule leaves a predefined cross-range corridor. Apollo derived guided entry has been used on the US Space Shuttle[5]. The first

guided entry on Mars will be attempted by the NASA Mars Science Laboratory (MSL) [6], which was launched on November 2011, and is planned to reach Mars in the early August 2012.
Several techniques [7,8,9,10,11,12,13,14,15,16,17,18,19] have been proposed for improving the performance of the guided entry control. According to Kluever [7], the approaches can be broadly separated into two methodologies: 1) reference-path tracking and 2) path-planning algorithms. Path planning usually computes a reference path from the current state to the desired parachute deployment state in real-time, using a nominal and simplified model of the entry dynamics. Path-tracking algorithms employ an error feedback law in which the deviation of the capsule trajectory, relative to the reference one, is used to compute the command in real-time. To our knowledge, the Apollo-derived of Carman [3] and profile tracking methods proposed by Hormigo [8], Bharadwaj [9] and Andres [10] exploit feedback linearization techniques (see Chapter 6 of Slotine [20]) and belong to the category of reference-path tracking. The methodologies proposed by Kluever [7], Tu [11] and de Lafontaine [12] fall into the category of path planning.

The approach followed here exploits the principles of the Embedded Model Control (see Canuto [21,22,23,24]), and combines path-planning with a reference-path tracking algorithm. Specifically the path-planning algorithm is not computed in real time, but only at the entry point interface (EIP), the scope being confined to recover initial dispersions. The approach exploits a linearized version of the longitudinal dynamics coupled with the output of the navigation algorithm, which provides the so called pseudo-measurements. The main goal and achievement is estimating the dispersion causes (the combined effect of atmospheric density variation, wind and aerodynamic coefficients errors) in real-time and adjusting the vertical component of the lift in order to compensate (cancel) their effect. The control algorithm is split into the measurement law (combination of embedded model and noise estimation), including the real time estimation of the external/internal disturbances, and the control law, aiming to cancel the estimated disturbances and to stabilize the dynamics around the reference trajectory. The rationale of the approach is outlined in the paper.

The paper provides a description of the selected solution for the path planning and reference-path tracking algorithms, and the output of a simulation campaign using a high-fidelity 6 degrees-of-freedom simulator. The attitude control, in charge of actuating a reaction control system for regulating the vehicle bank angle is described by the same authors in [25]. A preliminary description of the selected solution can be found in [26]. The paper is subdivided into five sections.

1) The first section provides a short discussion of the selected solution for the translational control.
2) The second section describes the 3 DoF translational dynamics, decoupling into the longitudinal and the lateral dynamics, and linearization of the longitudinal dynamics.
3) The third describes the path-planning algorithm.
4) The fourth section the reference-path-tracking algorithm.
5) The last section shows the time profile of the relevant signals of the translational control systems and summarizes the results of a Monte Carlo campaign.

II. The reasons of the proposed solution

The proposed solution comes out by considering the causes of the landing site dispersion for a typical Mars atmospheric entry. According to Wolf [1] they can be summarized as: (i) initial state dispersion, i.e. position and velocity errors at the EIP, (ii) uncertainty and variability of the atmospheric conditions, mainly of the atmospheric density and wind, and (iii) incomplete knowledge of the entry vehicle aerodynamics, as the latter is derived from an uncertain aerodynamic database.

The Embedded Model Control (EMC) methodology subdivides control analysis and design into three sub-problems:

1) Reference generator: the control problem is solved in open loop as a baseline. Based on the initial state and on a model of the system dynamics, reference trajectory and command are computed such that terminal (or intermediate) requirements are met and command/state constraints are respected.
2) Measurement law: the model used to build the reference trajectory is extended to include disturbance dynamics (the complete model is referred to as embedded model). The difference between the sensor measurements and the embedded model output (model error) becomes the source for estimating, in real-time, the causes (external signals and parametric errors) drifting the trajectory from the reference one.
3) Control law: the command is adjusted for compensating the estimated disturbance, and a state-feedback law is included to stabilize the dynamics around the reference trajectory.

As a consequence, the reference trajectory should be re-computed only when one or more of the following conditions occur: the estimated initial state largely differs from the reference state, the knowledge of the model parameters greatly improves, the desired (final) state changes, command/state saturation occurs. Such conditions can be rephrased in terms of the following guided-entry circumstances:

1) The navigated state at the EIP largely differs from the nominal one.
2) Atmospheric and/or aerodynamic conditions change, and their variation can be estimated and predicted for the rest of the trajectory, so as to update the reference model.
3) The final state, here referred to as the parachute deployment point (PDP), changes during the entry phase.
4) The command saturates (the bank angle cosine reaches its maximum) in which case re-computing the trajectory helps in recovering linear conditions.

The first circumstance occurs at the EIP, when the known vehicle state differs from the nominal one because of trajectory deviations during cruise and coasting phases. The second circumstance may occur when a real time estimation of atmospheric and/or aerodynamic parameters are made available. Actually, in order to improve the reference model, a mere estimation of the current deviation of the parameters (density, lift and drag coefficients) from their nominal values would be insufficient. Instead, a prediction of their average value or of a mean profile valid for the rest of the trajectory should be the rather challenging goal. The third circumstance will not be treated here. Adjusting the PDP during the entry would mean that new information about the landing site was gathered by the descent module prior to the PDP. The fourth circumstance suggests a way to recover from command saturation. The problem, not yet been tackled by the authors, has not been found in the literature either.

In summary, the path-planning algorithm is only run when the capsule reaches the EIP. As such, it re-computes an entry trajectory from the EIP navigation state to the desired PDP state. The result is a complete recovery of the landing dispersion as long as the EIP errors are known. If they are unknown, because they are dominated by navigation errors, they cannot be in principle reduced by an active control system which is only fed by navigation data.

Beside navigation errors, to be recovered by appropriate sensors during the terminal descent phase, uncertainty and variability of aerodynamic and atmospheric data become the residual cause of landing dispersion, to be compensated by a reference-path-tracking algorithm. The proposed approach follows the EMC methodology by extending the entry model with a disturbance dynamics capable of capturing the combined causes of the residual dispersion in the form of disturbance signals, and by continuously adjusting the bank angle command to cancel their effect on the vehicle trajectory. Besides being at the core of EMC, disturbance modeling and estimation is well known in the literature (see for instance Han [27]) as active disturbance cancellation.

As a final remark, the development of path-planning and reference-path-tracking algorithms assumes that a nominal entry profile, characterized by EIP state, bank angle profile, and PDP state, is available. It is also assumed that trajectory requirements in terms of load factor, heat flux, etc., are guaranteed by the nominal trajectory with a sufficient margin for recovering EIP dispersions. The assumption has been verified a posteriori for typical entry conditions and dispersions. Extension of the algorithms to guarantee load factor and heat flux limits will be a further development.

III. Entry translational equations

In this section, the generic 3 degrees-of-freedom (DoF) nonlinear equations describing the center-of-mass (CoM) motion of an entry capsule are presented (see Chapter 4 in Vinh [28]). Under assumptions to be clarified below, the whole dynamics can be arranged into the series of longitudinal and lateral dynamics, driven by cosine and sine of the bank angle, respectively. The control of the longitudinal and lateral trajectories can be thus decomposed if the longitudinal trajectory is assumed to be a known input to the lateral one, and a factor of the bank angle sine (the sign) which is independent of the cosine is employed as the lateral command. Moreover, a nominal trajectory being available, only the relevant trajectory variations need to be computed, which suggests to linearize the longitudinal dynamics around a reference path.

The original 3 DoF dynamics provides the best on-board tool for integrating the reference entry trajectory, given the command solved by the path-planning algorithm. Decomposed longitudinal and lateral dynamics, together with their linearization around the reference trajectory, are intermediate design models for solving both path-planning and reference-path tracking. Interconnection of models and control algorithms is sketched in Fig 6.

A. Set of the CoM motion equations

The following assumptions will be used in the derivation of the entry equations.

1) The planet is spherical and rotating.
2) The gravity is radial and constant.
3) The aerodynamic lift and drag coefficients (c_L and c_D, respectively) are assumed constant.
4) The density is an exponential function of the altitude above the surface.
5) Notations of the trigonometric functions are simplified to $c_\theta = \cos\theta$, $s_\theta = \sin\theta$ and $t_\theta = \tan\theta$, θ being a generic angle.

These assumptions are coherent with literature [28]. The resulting equations are

$$\dot{r} = v\,\mathrm{s}_\gamma \qquad \dot{v} = -\frac{F_D}{m} - g\,\mathrm{s}_\gamma + r\omega_M^2\,\mathrm{c}_\lambda\left(\mathrm{s}_\gamma\mathrm{c}_\lambda - \mathrm{s}_\lambda\mathrm{c}_\chi\mathrm{c}_\gamma\right)$$

$$\dot{\Lambda} = \frac{v\,\mathrm{c}_\gamma\mathrm{s}_\chi}{r\,\mathrm{c}_\lambda}\text{, and } \dot{\gamma} = \frac{F_L}{mv}c_\mu + \left(\frac{v}{r} - \frac{g}{v}\right)\mathrm{c}_\gamma + 2\omega_M\,\mathrm{s}_\chi\mathrm{c}_\lambda + \frac{1}{v}r\omega_M^2\,\mathrm{c}_\lambda\left(\mathrm{c}_\gamma\mathrm{c}_\lambda + \mathrm{s}_\lambda\mathrm{c}_\chi\mathrm{s}_\gamma\right). \tag{1}$$

$$\dot{\lambda} = \frac{v\,\mathrm{c}_\gamma\mathrm{c}_\chi}{r} \qquad \dot{\chi} = \frac{v}{r}\mathrm{c}_\gamma\mathrm{s}_\chi\mathrm{t}_\lambda + \frac{F_L}{v\,\mathrm{c}_\gamma}\mathrm{s}_\mu + 2\omega_M\,\mathrm{s}_\lambda - \mathrm{c}_\chi\mathrm{c}_\lambda\,t_\gamma + r\omega_M^2\,\frac{\mathrm{s}_\chi\mathrm{c}_\lambda\mathrm{s}_\lambda}{v\,\mathrm{c}_\gamma}$$

The notations that have been employed in (1) are as follows. The radius r is the distance from the planet center of mass, Λ is the longitude, λ is the latitude, v is the magnitude of the co-rotating velocity, γ is the co-rotating flight path angle (FPA), χ is the co-rotating heading, ω_M is the angular rotation rate of the planet, g is the planet gravity obtained as a function of the radius r, and m is the mass of the capsule. The bank angle μ plays the command role.

The aerodynamic forces are written as

$$F_D = \frac{1}{2}\rho(r)v^2c_DS$$
$$F_L = \frac{1}{2}\rho(r)v^2c_LS \tag{2}$$

The atmosphere density is written as a function of the radius r

$$\rho(r) \cong \rho_s e^{-(r-R_M)/H_{atm}}, \tag{3}$$

where R_M is the mean radius of Mars (in general of the planet), and H_{atm} is the atmosphere scale height.

B. Simplified set of longitudinal and lateral equations

Equation (1) can be simplified by arranging longitudinal and lateral dynamics so that they are coupled in series. Series arrangement is made possible by assuming that (i) the altitude h above the planet surface is small compared to the planet radius R_M, (ii) the planet rotation ω_M is neglected, and (iii) the heading angle is written as the variation $\Delta\chi = \chi_{eip} - \chi$ of the current heading angle χ with respect to the same angle χ_{eip} at the EIP.

The longitudinal dynamics is found to be

$$\dot{s} = v\cos\gamma \qquad \dot{v} = -\frac{F_D}{m} - g\,\mathrm{s}_\gamma$$
$$\text{, and} \qquad \text{,} \tag{4}$$
$$\dot{h} = v\sin\gamma \qquad \dot{\gamma} = \frac{F_L}{mv}u_{dr} + \left(\frac{v}{R_M + h} - \frac{g}{v}\right)\mathrm{c}_\gamma$$

where $u_{dr} = \cos\mu$ is the command and s is the downrange (arc-length along the direction of the velocity at the EIP).

Lateral dynamics becomes

$$\begin{aligned} \dot{y} &= v\cos\gamma\sin\Delta\chi \\ \Delta\dot{\chi} &= -\frac{F_L}{v\,\mathrm{c}_\gamma}\left|s_\mu\right|u_{cr} \end{aligned}, \tag{5}$$

where y is the cross-range (arc-length perpendicular to the direction of the velocity at the EIP).

The set of variables $x_{dr} = \begin{bmatrix} s & v & h & \gamma \end{bmatrix}^T$ is referred to as the longitudinal state, whereas $x_{cr} = \begin{bmatrix} y & \Delta\chi \end{bmatrix}^T$ denotes the lateral state. Though equations (4) and (5) are forced by the cosine and sine of the same variable μ (the bank angle), they can be converted into two independent command variables, by making explicit the sign u_χ of the sine as follows

$$\begin{aligned} u_{dr} &= \cos\mu \\ u_{cr} &= \mathrm{sign}\left(\sin\mu\right) \end{aligned}. \tag{6}$$

As a result, equations (4) and (5) become connected in series, since the longitudinal dynamics (4) is independent of the lateral dynamics (5), whereas the latter equation depends through the variables v, γ, $\rho(h)$ and μ on (4), and can be controlled by toggling u_{cr} from 1 to -1 and vice versa.

Command separation in (6) provides the basis for the design of the bank reversal maneuver. Each BRM performs a jump of the bank angle, say from μ_0 to μ_1, such that

$$\begin{aligned} \cos\mu_0 &= \cos\mu_1 \\ \mathrm{sign}\left(\sin\mu_0\right) &= -\mathrm{sign}\left(\sin\mu_0\right) \end{aligned}. \tag{7}$$

C. Linearization of the longitudinal dynamics

To fully exploit a reference entry trajectory, the longitudinal motion can be expressed as a perturbation with respect to the latter. The perturbations Δx_{dr} and Δu_{dr} of the longitudinal state x_{dr} and of the command u_{dr}, respectively, are defined as the difference between the state variables and the command in (4), and the reference variables (underline distinguishes reference from state variables) as follows

$$\Delta x_{dr} = x_{dr} - \underline{x}_{dr} = \begin{bmatrix} s \\ v \\ h \\ \gamma \end{bmatrix} - \begin{bmatrix} \underline{s} \\ \underline{v} \\ \underline{h} \\ \underline{\gamma} \end{bmatrix}, \; \Delta u_{dr} = u_{dr} - \underline{u}_{dr}. \tag{8}$$

Linearization of (4) around a reference trajectory together with Euler discretization provides the following fourth order discrete state equations, written in the state vector Δx_{dr} defined in (8) and in the integer step i as follows

$$\begin{aligned} &\Delta x_{dr}(i+1) = A_{dr}(i)\Delta x_{dr}(i) + B_{dr}(i)\Delta u_{dr}(i) \\ &x_{dr} = \begin{bmatrix} \Delta s \\ \Delta v \\ \Delta h \\ \Delta\gamma \end{bmatrix}, \; A_{dr}(i) = \begin{bmatrix} 1 & a_{vs}(i) & 0 & a_{\gamma s}(i) \\ 0 & 1+a_{vv}(i) & a_{hv}(i) & a_{\gamma v}(i) \\ 0 & a_{vh}(i) & 1 & a_{\gamma h}(i) \\ 0 & a_{v\gamma}(i) & a_{h\gamma}(i) & 1+a_{\gamma\gamma}(i) \end{bmatrix}, \; B_{dr}(i) = \begin{bmatrix} 0 \\ 0 \\ 0 \\ b_\gamma(i) \end{bmatrix}. \end{aligned} \tag{9}$$

Given a time unit T (sampling time of the Inertial Measurement Unit) the entries a_{jk} in (9) (j and k indicate generic elements of x_{dr}) are the partial derivatives of (4) along the reference trajectory, i.e.

$$
\begin{aligned}
&a_{sv} = T\cos\underline{\gamma},\ a_{s\gamma} = -T\underline{v}\sin\underline{\gamma} \\
&a_{vv} = -\underline{v}T\rho c_D S/m \\
&a_{vh} = 0.5T\rho\underline{v}^2 c_D S/\left(mH_{atm}\right),\ a_{v\gamma} = -Tg\cos\underline{\gamma} \\
&a_{hv} = T\sin\underline{\gamma},\ a_{h\gamma} = -T\underline{v}\cos\underline{\gamma} \\
&a_{\gamma v} \cong 0.5T\rho c_L S\underline{u}_{\Delta r}/m + Tg\cos\underline{\gamma}/\underline{v}^2 \\
&a_{\gamma h} = -0.5T\rho\underline{v}\,c_L S\underline{u}_{dr}/\left(mH_{atm}\right),\ a_{\gamma\gamma} \cong Tg\sin\underline{\gamma}/\underline{v}
\end{aligned}
\tag{10}
$$

Finally

$$b_\gamma = 0.5T\rho_s e^{-\underline{h}/H_{atm}} c_L S/m\,. \tag{11}$$

Using a typical atmospheric entry profile, it can be proved that $|a_{vv}|<1$ and $|a_{\gamma\gamma}|<1$, and consequently they can be dropped without degrading the approximation degree of (9). A block diagram of the perturbed longitudinal dynamics is in Fig 1.Blocks with the symbol Σ stand for discrete time integration (dimensionless). The block diagram shows the guided entry controllability principle (see The block diagram shows the guided entry controllability principle (see Buonocore et al. [26] and Benito et al. [29]). In order to (ideally) force the downrange deviation Δs to zero, the velocity perturbation Δv must be driven by an altitude variation Δh which in turn demands a FPA variation $\Delta\gamma$ to be commanded by the lift vertical component Δu_{dr}.

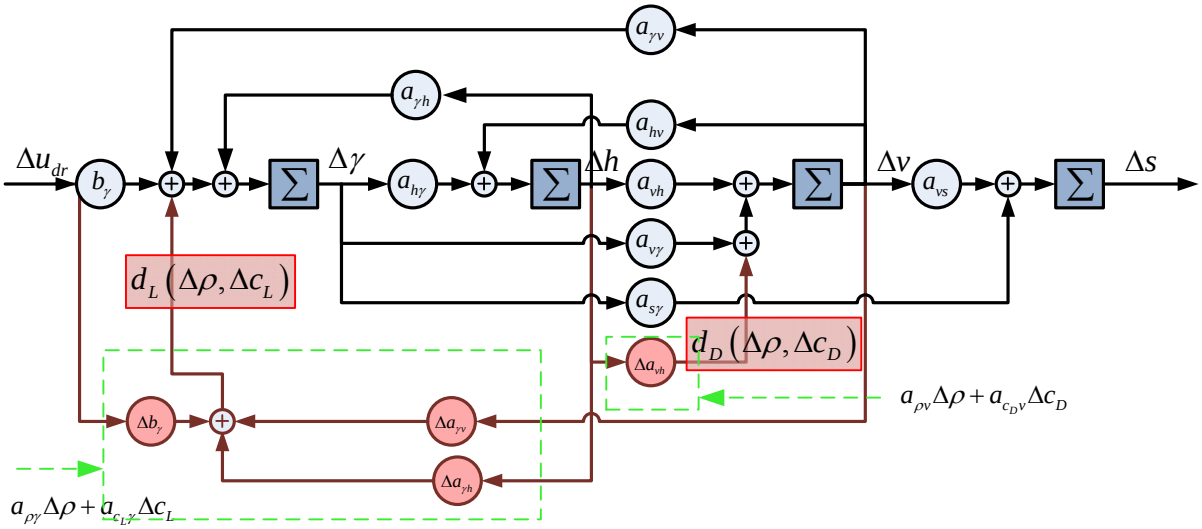

Fig 1. Block diagram of the perturbed longitudinal dynamics including the effects of the atmospheric and aerodynamic dispersions.

A longitudinal path-planning algorithm must find the command (the bank angle cosine) which drives longitudinal dynamics to the desired PDP state. Being a two-dimensional (2D) motion problem driven by a single command, controllability needs to be verified. Non-linear controllability analysis using Lie derivatives (see [30] for theory and procedure) have been used to investigate the local accessibility (a weak form of controllability for nonlinear systems defined in Chapter 3 of [31]) by the command u_{dr}. Only some results are summarized below.

1) Neglecting gravity and the density dependence on the altitude in (3), longitudinal dynamics becomes uncontrollable, implying that deceleration cannot be modulated by a variable bank angle.
2) Making explicit gravity, but neglecting the density dependence on h, longitudinal dynamics becomes controllable.
3) Making explicit the density dependence on h, and neglecting gravity, longitudinal dynamics becomes controllable.

The second case might suggest acting on the FPA for modulating the acceleration component due to gravity. The strategy seems inappropriate here, since deceleration is mostly caused by aerodynamic drag and not by gravitation. On the contrary, the third case is coherent with the principle (mentioned by de Lafontaine [12]) that a guided entry can be pursued if the capsule is made capable of flying at higher or lower altitudes in order to modulate deceleration. Recovering longitudinal controllability through density correlation with h is at the core of the proposed reference-path-tracking algorithm. As a main controllability result, all the four state variables of the longitudinal dynamics (4) can be moved to any arbitrary value at the PDP by modulating the lift vertical component (the bank angle cosine) u_{dr}.

IV. Path-planning algorithm

The path-planning algorithm is computed at the EIP, with the aim of finding a bank angle profile guiding the capsule from the EIP navigation state to the desired PDP state. The algorithm must solve a two-point boundary value problem for the non-linear dynamics in (4).

Different methods and solutions have been proposed. Drag-based methods like the one by Tu [11] rewrite equations (1) in a simpler form by using the energy as independent variable. Other authors like de Lafontaine [12] and Kluever [7] employ a form of parameterized trajectories, whose parameters are recomputed at each control step in order to guarantee that the target PDP conditions are met. De Lafontaine [12] trajectories either assume a piecewise constant vertical velocity, or a constant FPA. These assumptions simplify the entry equations (1) and facilitate the finding of a solution. Polynomial trajectories connecting EIP and PDP have been proposed by Kluever [7]. An advantage of such methods is that they do not need a linearized dynamics, and can provide closed-form solutions of the path-planning algorithm. A drawback comes by considering a trajectory that has been computed in the previous time steps as a new nominal trajectory, that must be corrected in the next steps because of parameter uncertainty. Since such a nominal trajectory imposes a variable bank angle in order to track the specific profile of the adopted trajectories, the bank angle cosine may progressively approach saturation thus leaving a smaller margin for future corrections.

On the contrary, a linearized dynamics allows exploiting a nominal path driven by a bank angle profile which is as simple as possible (usually constant, $u_{dr} = u_{dr,0}$), thus leaving a wider range to the command variations in charge of recovering EIP dispersions and of compensating atmospheric and aerodynamic uncertainty. Iterations of the algorithm are only needed to approach target conditions, because no exact closed-form solution is available. The main differences of the proposed path-planning algorithm with respect to literature can be summarized as follows:

1) Path-planning algorithm is computed only once at EIP.
2) A preliminary (quasi) optimal flight time is computed so as to reduce command authority and attenuate the saturation risk.
3) Longitudinal path-planning is combined with lateral path-planning into a full 3D guidance.

The need of reversing bank angle (BRM), and longitudinal/lateral decomposition suggest the splitting of the algorithm into a sequence of three sub-problems.

1) To compute the optimal flight time from the navigated EIP to the nominal PDP.
2) To compute the longitudinal trajectory by finding the time profile of the lift vertical component capable of reaching the desired altitude, downrange, velocity and FPA at PDP.
3) To compute the lateral trajectory by finding the bank reversal instants capable of reaching the desired cross-range and heading at PDP.

A. Longitudinal path planning

The algorithm aims to find the k-th, $k = 0,1,...,p-1$, time profile $\mathbf{u}_{dr,k}$ (bold symbols denote the whole sequence of discrete-time samples) bringing the longitudinal state x_{dr} from the initial EIP point $x_{dr}(0)$ to the desired PDP point $\bar{x}_{dr}(m-1)$ where $\bar{x}_{dr}$ denotes the target PDP longitudinal state. Discrete times at EIP and PDP have ben denoted with $i = 0$ and $i = m-1$, respectively. Subscript k denotes the current iteration, p the total number of iterations and m the duration, in discrete steps, of the entry phase. A point-to-point control problem for the non-linear equations (4) is in general rather difficult to solve. The solution can be approximated (as it is done by Stengel [32] in Chapter 4) using linear and discrete-time equations of the perturbations around a (variable) reference trajectory as in (9).

The algorithm uses the initial navigation state $x_{dr,nav}(0)$ and the current bank angle profile (from $i=0$ to $i=m-1$) to numerically integrate equations (4) for the state trajectory $\mathbf{x}_{dr,k}$. Integration allows to compute the final longitudinal error $e_{dr,k}(m-1)=x_{dr,k}(m-1)-\overline{x}_{dr}$ at the iteration step k. Equations (9) are then used to solve for a command variation $\Delta\mathbf{u}_{dr,k}$ capable of making $\Delta x_{dr,k}(m-1)$ equal to $-e_{dr,k}(m-1)$, and of forcing $e_{dr,k+1}(m-1)=0$ through the iteration $\mathbf{u}_{dr,k+1}=\mathbf{u}_{dr,k}+\Delta\mathbf{u}_{dr,k}$ in (4).

Computing $\Delta\mathbf{u}_{dr,k}$ is a point-to-point control problem with fixed time, but under linear and time-varying state equations. Linearity allows to express the final state as a function of the command profile (the initial perturbation is zero)

$$\Delta x_{dr,k}(m-1)=C_{dr,m}(k)\Delta\mathbf{u}_{dr,k}, \tag{12}$$

where

$$\Delta\mathbf{u}_{dr,k}=\left[\Delta u_{dr,k}(0)\quad \Delta u_{dr,k}(1)\quad \ldots\quad \Delta u_{dr,k}(m-1)\right]^T, \tag{13}$$

and $C_{dr,m}(k)$ is a controllability matrix of dimension $4\times m$ built from (9) using the current reference trajectory $\mathbf{x}_{dr,k}$. Expression (12) is the linear set of four equations to be solved for $m>>4$ unknowns. A solution exists because of controllability, and an optimal solution must be looked for because of the large number of unknowns. The least (weighted) Euclidean norm solution has been chosen, that is

$$\Delta\mathbf{u}_{dr,k}=\left(Q_{dr}^{-1}C_{dr,m}^{T}(k)\right)\left(C_{dr,m}(k)Q_{dr}^{-1}C_{dr,m}^{T}(k)\right)^{-1}\left(-e_{dr,k}(m-1)\right), \tag{14}$$

where Q_{dr} is a weight matrix for exploiting the available degrees-of-freedom and constraining the solution along specific portions of the entry phase (for instance for reducing command perturbations at the end of the trajectory).

The solution $\Delta\mathbf{u}_{dr,k}$ is combined with $\mathbf{u}_{dr,k}$ and replaced in (4) to yield $\mathbf{u}_{dr,k+1}$. The optimal solution (14) can be thus iterated to further reduce the residual terminal error. It has been proved experimentally - a formal prove is not reported here – that less than five iterations are sufficient for bringing the residual terminal error close to tens of meters. Thus it does not contribute to a target PDP dispersion of the order of 1 km.

B. Optimization of the entry phase duration

A least-squares perturbation of the nominal bank angle profile has been obtained in the previous section assuming a fixed maneuver duration mT. Actually, m is an additional degree of freedom, which becomes essential because of the hard limit [-1,1] of the bank angle cosine. It has been experimentally observed that the shape of $\mathbf{u}_{dr,k}$, and specifically the peak values (min and max) strongly depend on m.

To better clarify the point consider Fig 2, left, in which the longitudinal path-planning aims to recover an initial FPA which is 0.5 degree steeper than the nominal one. Were the path correction computed along the nominal duration $mT=300$ s, the adjustment $\Delta\mathbf{u}_{dr,k}$ would make $\mathbf{u}_{dr,k+1}$ to overcome the interval $[-1,\ 1]$. The corresponding profile in Fig 2, dotted line, shows that the cosine of the bank angle almost reaches the value of -1.5, much lower than -1, at about 70 s. Were the duration extended to 325 s, then $\mathbf{u}_{\gamma,k+1}$ would reenter inside $[-1,\ 1]$ as shown by the continuous line in Fig 2. Gray lines are intermediate solutions in the case of intermediate durations between 300 s and 325 s.

Instead of exploiting an analytical relation between a norm of the bank angle profile and m, a gradient descent method has been implemented which minimizes the l_∞ norm of $\mathbf{u}_{dr,k}$, i.e. $\max_i\left|\cos\left(\mu_k(i)\right)\right|$. Once a local minimum is found, the solution is selected as optimal. The profile of the l_∞ norm (normalized) compared with l_1 and l_2 norms for different variations of the time duration is shown in Fig 2, right.

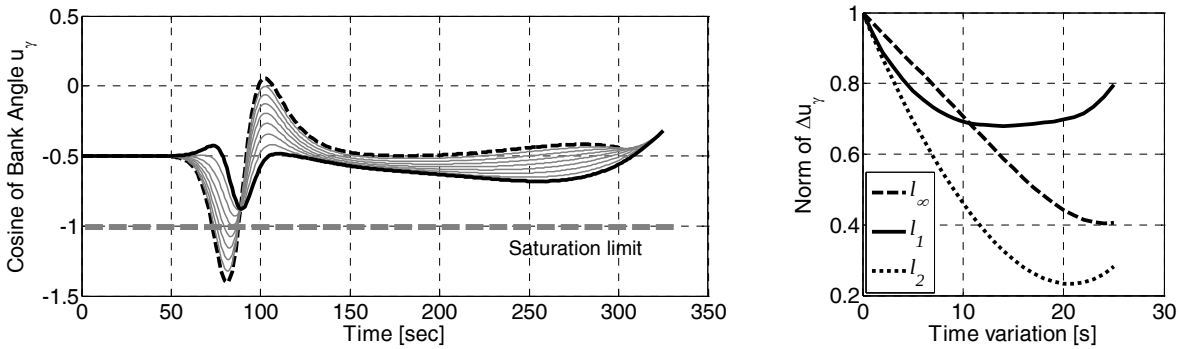

Fig 2. Left: optimal bank angle cosine for different entry durations, from the nominal value of 300 s (dotted line) to the optimal value of 325 s (continuous line). Gray lines are intermediate solutions. Right: profile of the l_∞ norm compared together to l_1 and l_2 norms.

The optimal-time search algorithm greatly extends the algorithm capability of recovering larger EIP dispersions. Some performance variables of the longitudinal path planning featuring the optimal-time algorithm are shown in Fig 4 and Fig 4. The altitude h is plotted versus the downrange s and the velocity magnitude v. The load factor and the bank angle cosine $\cos\mu$ are plotted versus time. Five different cases of EIP radial dispersion, namely -10, 5, 0, 5 and 10 km, are considered. One may observe that the bank angle cosine is always feasible.

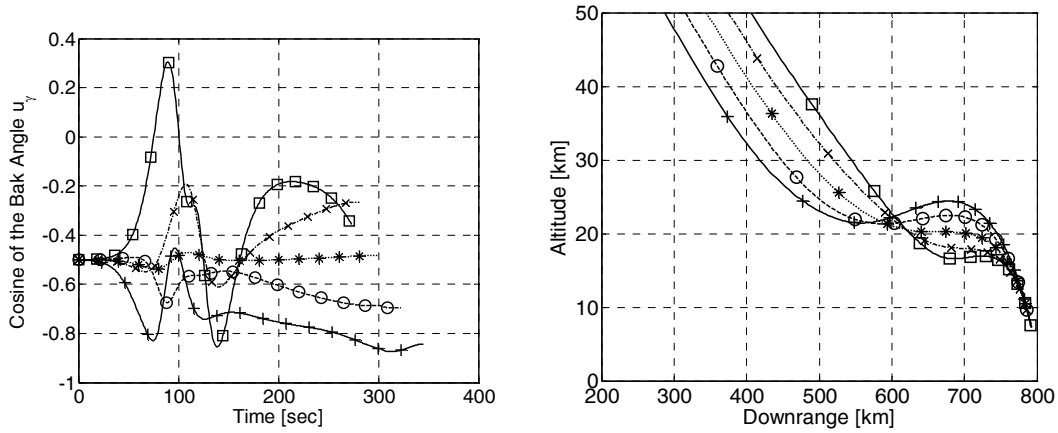

Fig 3. Bank angle cosine and altitude versus downrange for EIP radial dispersions of -10, 5, 0, 5 and 10 km. The legend in Fig 4, right, applies to subfigures.

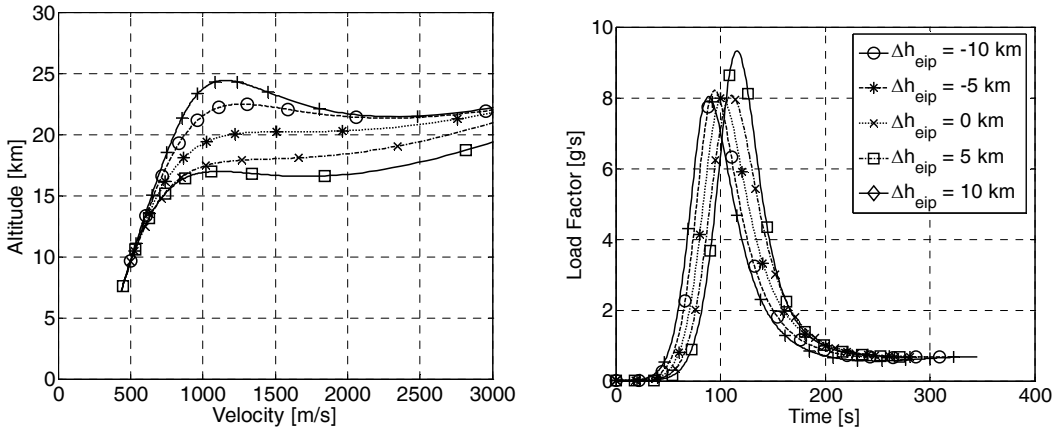

Fig 4. Altitude versus velocity and load factor for EIP radial dispersions of -10, 5, 0, 5 and 10 km. The right legend applies to subfigures.

C. The lateral path-planning algorithm

The second-order linearized lateral dynamics of (5), at the iteration step k, is obtained through the approximation $\sin\Delta\chi \approx \Delta\chi$ as follows

$$\dot{x}_{cr,k}(t) = \begin{bmatrix} 0 & v_k(t)\cos\gamma_k(t) \\ 1 & 0 \end{bmatrix} x_{cr,k}(t) + \begin{bmatrix} 0 \\ -\dfrac{F_{L,k}}{v_k(t)\cos\gamma_k(t)}\left|\sin\mu_k(t)\right| \end{bmatrix} u_{cr,k}(t) \quad . \tag{15}$$

$$x_{cr,k}(t) = \begin{bmatrix} y_k \\ \Delta\chi_k \end{bmatrix}(t)$$

The lateral command $u_{cr,k}$ in (15) toggles between 1 and -1 at a finite number $n = 1,...,N$ of times t_{kn} (N is the number of bank reversal maneuvers, as a baseline equal to $N = 4$). That is, t_{kn} denotes the starting time of the n-th bank reversal at the k-th iteration. The lateral path-planning algorithm exploits the sensitivity of the terminal error (at the PDP) to a variation Δt_{kn} of t_{kn} (delay or anticipation). A command $u_{cr,k}$ where t_{kn} has been modified into $t_{kn} + \Delta t_{kn}$ is denoted with $u_{cr,kn}$, and the corresponding state variable with $x_{cr,kn}$. The differential command and state $\Delta u_{cr,kn} = u_{cr,k} - u_{cr,kn}$ and $\Delta x_{cr,kn} = x_{cr,k} - x_{cr,kn}$ satisfy (15), and $\Delta u_{cr,kn} = u_{cr,k} - u_{cr,kn}$ is zero everywhere except during the open interval $\left[t_{kn}, t_{kn} + \Delta t_{kn}\right)$. Passing to discrete time, a linear relationship can be found from (15) between the command perturbation $\Delta\mathbf{u}_{cr,kn}$ and the final lateral state variation at the PDP $\Delta x_{cr,kn}(m-1)$, namely

$$\Delta x_{cr,kn}(m-1) = C_{cr,m}(k)\,\Delta\mathbf{u}_{cr,kn}\,. \tag{16}$$

Since $\Delta\mathbf{u}_{cr,kn}$ only depends on the interval $\left[t_{kn}, t_{kn} + \Delta t_{kn}\right)$, (16) provides the sensitivity bi-dimensional vector c_{kn} of the PDP lateral state to a shift Δt_{kn} of the n-th bank reversal time, namely

$$c_{kn} = \frac{\Delta x_{cr,kn}(m-1)}{\Delta t_{kn}}\,. \tag{17}$$

By collecting the sensitivity coefficient in the sensitivity matrix $2 \times N$

$$S_{cr}(k) = \begin{bmatrix} c_{k1} & c_{k2} & \cdots & c_{kN} \end{bmatrix}, \tag{18}$$

and the variations of the bank reversal times in the vector

$$\Delta\mathbf{t}_k = \begin{bmatrix} \Delta t_{k1} & \Delta t_{k2} & \cdots & \Delta t_{kN} \end{bmatrix}^T, \tag{19}$$

and by defining the terminal lateral error as $e_{cr,k}(m-1) = x_{cr,k}(m-1) - \bar{x}_{cr}$, where $\bar{x}_{cr}$ denotes the target PDP lateral state, the linear relation is obtained

$$-e_{cr,k}(m-1) = S_{cr}(k)\Delta\mathbf{t}_k. \tag{20}$$

Equation (20) can be solved for the unknown $\Delta\mathbf{t}_k$ using the weighed least squares as follows

$$\Delta\mathbf{t}_k = \left(Q_{cr}^{-1}S_{cr}^T(k)\right)\left(S_{cr}(k)Q_{cr}^{-1}S_{cr}^T(k)\right)^{-1}\left(-e_{cr,k}(m-1)\right). \tag{21}$$

As in the case of the longitudinal path planning, an iterative procedure is pursued so as to progressively reduce the residual errors. A typical evolution of the lateral guidance solution for an initial cross-range dispersion of 10 km and a heading error equal to -1 degree is shown in Fig 5. One may observe that keeping the nominal starting times of the bank reversal in the presence of a lateral error at the EIP (see the 'NotControlled' profile in Fig 5, right) generates a lateral error of more than 20 km at the PDP, whereas at the third iteration of (21) it has been reduced to less than 200 m. The heading angle is also recovered.

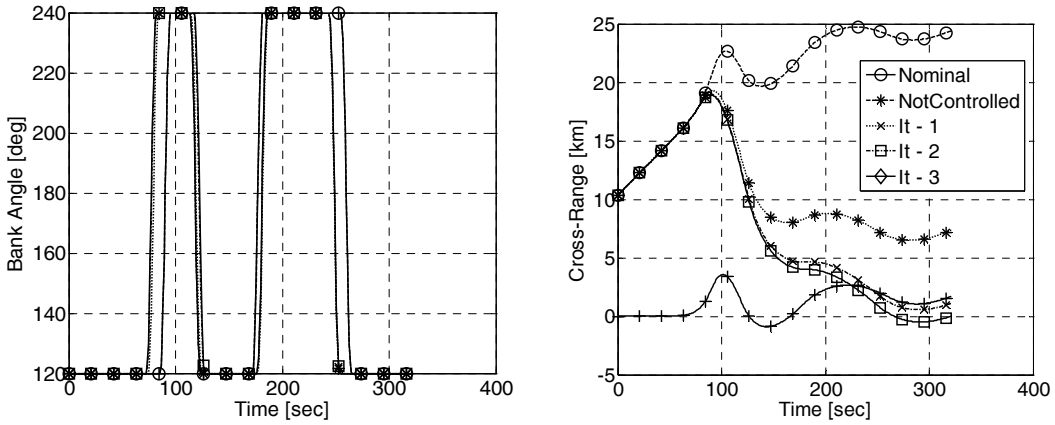

Fig 5. Typical profile of the lateral guidance for an initial cross-range dispersion of 10 km and a heading error of -1 degree.

D. Overall control structure

Longitudinal and lateral path-planning algorithms are not separately solved, but they are combined into a single 3D algorithm exploiting the series connection of equations (4) and (5) until convergence is attained.

The output of the path-planning algorithm to be used by the reference-path-tracking is denoted by $\underline{x}_p$, and corresponds to the last iteration $k = p$. Besides $\underline{x}_p$, the path-planning algorithm provides the initial state $\underline{x}(0) = x_{nav}(0)$, the bank angle profile $\underline{\mu}_p$, the bank angle cosine $\overline{u}_{dr,p}$, the BRM starting times t_{br} and the PDP state $\underline{x}_p(\underline{m}_p - 1)$ where $\underline{m}_p$ provides the entry duration $\underline{t} = (\underline{m}_p - 1)T$. The sensitivity matrix $S_{cr,p}$ is also provided. The lower bar is employed together with subscript p to indicate that the path-planning output becomes the reference trajectory for the reference-path-tracking algorithm.

The outcome of the path-planning algorithm can be accurately tracked by the capsule, except for atmospheric and aerodynamic uncertainty to be handled by reference-path tracking. Interconnection between the 3 degrees-of-freedom entry dynamics, the decomposed longitudinal and lateral dynamics and the path-planning and reference-path-tracking algorithms are shown in Fig 6.

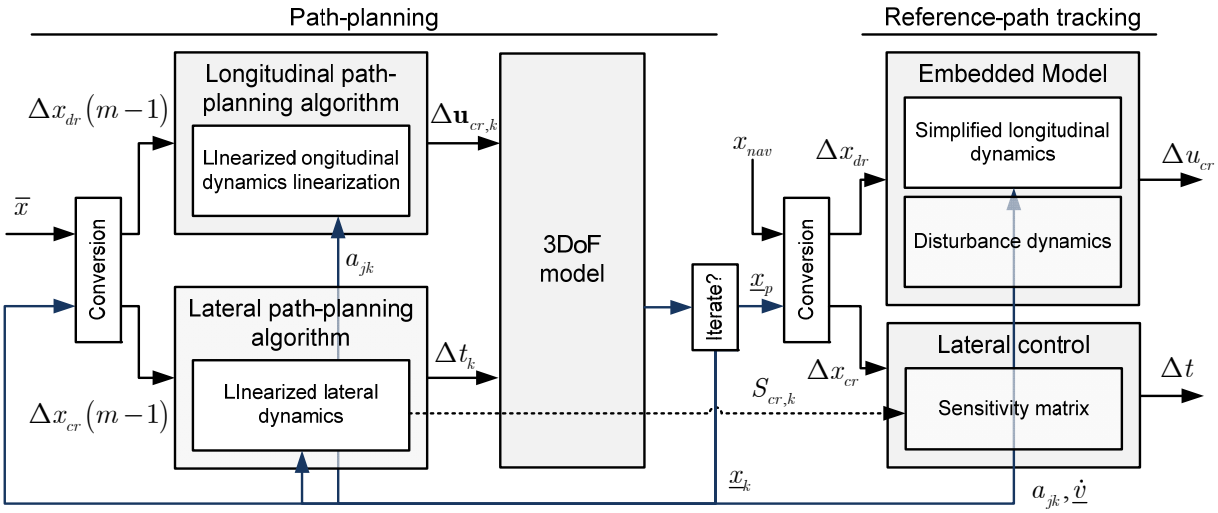

Fig 6. Interconnection between path-planning, reference-path-tracking algorithms and entry dynamic models.

V. The reference-path-tracking algorithm

The reference-path-tracking algorithm has been designed to adjust in real-time the longitudinal command $\bar{u}_{cr,p}$ and the BRM starting times t_{br} so as to keep the capsule as close to the reference trajectory as possible. The algorithm assumes, as for the path-planning algorithm, longitudinal and lateral decomposition of the entry dynamics, but further decomposes the fourth order longitudinal dynamics into a pair of second order dynamics: the altitude/flight-path-angle dynamics and the downrange/velocity dynamics.

The reference trajectory obtained as the output of the path-planning algorithm cannot be exactly tracked by the capsule under realistic conditions, mainly due to deviations of atmospheric conditions and of aerodynamic coefficients from their nominal values. A correction algorithm, the reference-path-tracking algorithm, has been designed to adjust in real-time the longitudinal command $\bar{u}_{\gamma,p}$ and the BRM starting times $\bar{t}_{br}$ so as to keep the capsule close to the reference trajectory as much as possible.

The altitude/FPA dynamics is commanded by the lift vertical component, whereas the down-range/velocity dynamics is commanded by the altitude itself, exploiting the altitude effect on the atmospheric density, since it allows full controllability as outlined in a previous section. Since altitude dynamics becomes the actuator of the velocity magnitude, a hierarchical control organization is suggested, in which the altitude perturbation driving the downrange control (the outer loop) is obtained by driving the vertical component of the lift (the inner loop).

Because inertial navigation in its standard form cannot provide the exogenous and parametric disturbances requested by Embedded Model Control, the errors between the navigated states and the reference trajectory counterparts are used as pseudo measurements of the model error between embedded model and reality. Feeding the model errors back to the embedded model state variables (controllable and disturbance) a form of state observer (actually a state predictor) is obtained capable of estimating in real-time the requested disturbances. Disturbances can be split in two classes: disturbances affecting the downrange/velocity dynamics are referred to as drag disturbances, those affecting altitude/FPA dynamics are referred to as lift disturbances. The embedded model is built around the linearized and discrete-time longitudinal dynamics (9), completed with a stochastic disturbance dynamics driven by noise vectors to be estimated, and by the command perturbation $\Delta \mathbf{u}_{cr}$, defined in (9).

The lateral control is simple. It uses the sensitivity matrix $S_{cr,p}$ to compute deviations of the BRM triggering times t_{br} from the navigated lateral error.

A. Modeling the coefficient uncertainty

The embedded model of the longitudinal dynamics is obtained by including into (4) additional input signals accounting for the effect of atmospheric and aerodynamic uncertainty on the longitudinal state variables. Perturbations of the density ρ and of the aerodynamic coefficients c_L and c_D are defined as follows

$$\begin{aligned} \Delta\rho &= \rho - \underline{\rho}(h) \\ \Delta c_L &= \underline{c}_L - c_L \\ \Delta c_D &= \underline{c}_D - c_D \end{aligned} \quad . \tag{22}$$

The linearized dynamics (9) can be modified to include the expressions in (22). The result is shown in the block diagram of Fig 1, where the signals $d_D(\Delta\rho, \Delta c_D)$ and $d_L(\Delta\rho, \Delta c_L)$ are referred to as drag and lift disturbances, respectively. The parameter Δa_{jk} connecting d_D and d_L to the state variables of (9) stands for the unknown perturbation of the parameter a_{xy} in (9) caused by $\Delta\rho$, Δc_L and Δc_D.
According to EMC, the signals d_D and d_L can be conveniently estimated in real time if they are modeled as external signals that are the output of dynamical systems made of pure integrators (in this case, two cascaded integrators) and forced by arbitrary signals (referred to as noise, see Canuto [21,24]). The relevant dynamics is referred to as disturbance dynamics, and the command to state dynamics is referred to as controllable dynamics. Of course d_D and d_L in Fig 1 cannot be claimed to be actually driven by noise signals, as they are connected to controllable state variables. Therefore closed-loop stability must be guaranteed as in Canuto [21,24]. No analysis of this sort is provided here.

To this end, a drastic assumption is made which consists in considering the perturbations shown in Fig 1 to be described by the disturbance dynamics shown in Fig 7. In addition a pair of noise terms are added to the entrance of the Δh and Δs integrators. A great advantage of the stochastic model of d_D and d_L, shown in Fig 7, is that allows dropping all the parasitic interconnections of Fig 1, leaving only the series interconnections ensuring controllability. The gain a_{hv} has been purposely modified to the non-linear function $f(\Delta h)$ in the double-bordered box, for reasons to be explained below.

Once the longitudinal dynamics have been converted to the embedded model shown in Fig 7, it is possible to use data from the navigation algorithm for building pseudo-measurements of Δx_{dr}. They are referred to as pseudo measurements since they are not obtained from sensors, but as the output of the navigation algorithm. They are reported in Fig 7 as $y_{\Delta\gamma}$, $y_{\Delta h}$, $y_{\Delta v}$ and $y_{\Delta s}$.

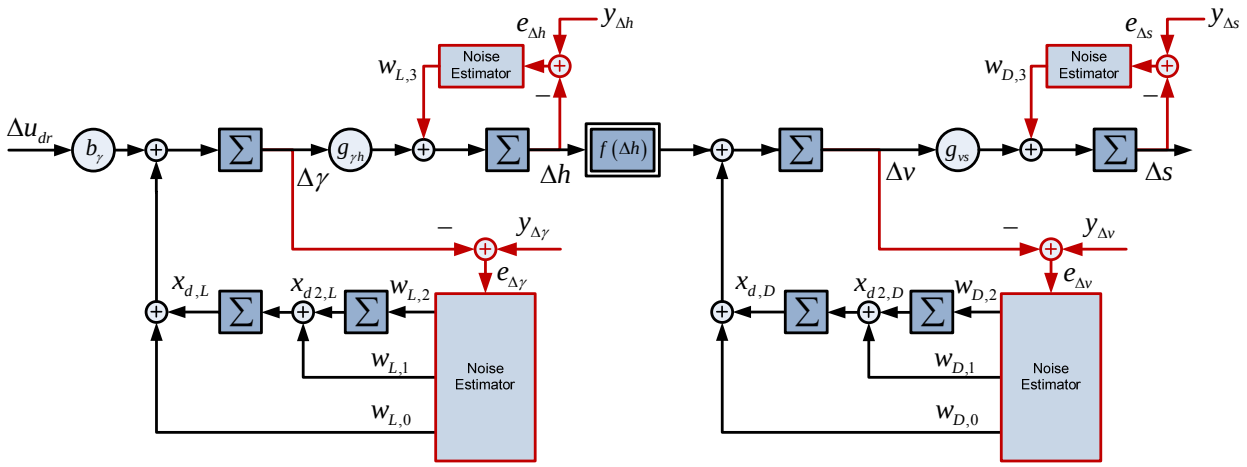

Fig 7. Longitudinal fourth order embedded model with disturbance dynamics and noise estimator blocks.

The longitudinal dynamics of Fig 7 is implemented as the core of the control algorithms in the form of a (closed-loop) state observer. The noise estimator (Canuto et al. [23]) of the longitudinal control employs the pseudo-measurements in Fig 7 to update the state variables $x_{d,L}$ and $x_{d,D}$ of the disturbance dynamics and predict aerodynamic uncertainty. The estimator is designed on the embedded model of Fig 7, having four output signals and

eight state variables to be observed. Actually, a decomposition of the output variables is made as in Canuto et al. [23]. The output $y_{\Delta\gamma}$ (or better the model error $y_{\Delta\gamma} - \Delta\gamma$) estimates the driving noise components of d_L, $y_{\Delta v} - \Delta v$ estimates those driving d_D. Finally $y_{\Delta h}$ and $y_{\Delta s}$ estimates the noise terms driving Δh and Δs respectively.

The resulting decoupled noise estimator can be written as

$$\begin{aligned} x_{d,D} &= d_D\left(\Delta c_D, \Delta\rho\right) + f_D\left(\Delta x_{dr}\right) + o_D\left(x_{dr}, \Delta x_{dr}\right) + e_{d,D} \\ x_{d,L} &= d_L\left(\Delta c_L, \Delta\rho\right) + f_L\left(\Delta x_{dr}\right) + o_L\left(x_{dr}, \Delta x_{dr}\right) + e_{d,L} \end{aligned}, \tag{23}$$

where $f_D\left(\Delta\gamma, \Delta v\right)$ and $f_L\left(\Delta\gamma, \Delta v\right)$ are the neglected interconnections from Fig 1 to Fig 7, d_L and d_D are shown in Fig 1, $o_D\left(x_{dr}, \Delta x_{dr}\right)$ and $o_L\left(x_{dr}, \Delta x_{dr}\right)$ are higher order terms, $e_{d,D}$ and $e_{d,L}$ are estimation errors to be bounded by the sensitivity of the state observer, combination of embedded model and noise estimator.

B. Hierarchical longitudinal control

A fourth-order control law designed around the embedded model in Fig 7 was preliminary tested providing satisfactory performance for small uncertainty, but in some worst cases performance was not satisfactory. The main contributor was identified as the nonlinear parameter a_{vh}, that was treated as a an average parameter in the closed-loop eigenvalue tuning. To cope with such a weakness, the fourth order dynamics was decomposed in two second-order parts, one describing altitude and FPA, and the other downrange and velocity. In this way the above nonlinear relation could be treated in an explicit way. As a consequence, a hierarchical control was implemented.

The higher level control monitors the evolution of Δs, Δv and $x_{d,D}$, and computes the altitude variation Δh_{com}. The lower level monitors the evolution of Δh, $\Delta\gamma$ and $x_{d,L}$, and forces Δh to track Δh_{com}.

The higher level dynamics describes the evolution of Δs and Δv, and is driven by Δh. The evolution of Δv is obtained from (9) and the nonlinear longitudinal dynamics in (4). The aerodynamic force and density profile from (2) and (3) are included as follows

$$\begin{aligned} \Delta\dot{v} = \dot{v} - \dot{\underline{v}} &= \frac{1}{2m}\left(\rho\left(\underline{h} + \Delta h\right) + \Delta\rho\right)\left(\underline{v} + \Delta\underline{v}\right)^2\left(c_D + \Delta c_D\right)S - g\sin\left(\underline{\gamma} + \Delta\gamma\right) - \left(\frac{1}{2m}\rho\left(\underline{h}\right)\underline{v}^2 c_D S - g\sin\underline{\gamma}\right) = \\ &= \Delta a\left(\Delta h\right) + f_D\left(\Delta x_{dr}\right) + d_D\left(\cdot\right) + o_D\left(\cdot\right) \end{aligned}, \tag{24}$$

where

$$\Delta a = \dot{\underline{v}}\left(e^{-\Delta h/H_{atm}} - 1\right) \tag{25}$$

stands for the commanded acceleration variation, and f_D, d_D and o_D coincide with the same notations in (23). The term $\dot{\underline{v}}$ is the nominal deceleration obtained from the path-planning trajectory $\underline{x}_p$. The higher level dynamics as in Fig 7 becomes

$$\begin{bmatrix} \Delta s \\ \Delta v \\ x_{d,D} \\ x_{d2,D} \end{bmatrix}(i+1) = \begin{bmatrix} 1 & a_{vs}(i) & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix}\begin{bmatrix} \Delta s \\ \Delta v \\ x_{d,D} \\ x_{d2,D} \end{bmatrix}(i) + \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \end{bmatrix}\Delta a(i) + \begin{bmatrix} w_{D,3} \\ w_{D,0} \\ w_{D,1} \\ w_{D,2} \end{bmatrix}(i). \tag{26}$$

Equation (26) is the cascade of the controllable dynamics (Δs and Δv) driven by Δa and of the disturbance dynamics ($x_{d,D}$ and $x_{d2,D}$) only driven by noise components. The noise estimator in charge of estimating the components of the four-dimensional noise vector w_D in (26) with components $w_{D,k}$, employs the pseudo-measurements $y_{\Delta v}$ and $y_{\Delta s}$ of Δv and Δs to update the disturbance state $x_{d,D}$. The control law computes Δa to stabilize Δv and Δs, and to cancel the effect of $x_{d,D}$ as follows

$$\Delta a = \begin{bmatrix} k_{0,D} & k_{1,D} \end{bmatrix} \begin{bmatrix} \Delta s \\ \Delta v \end{bmatrix} - x_{d,D} \,. \tag{27}$$

The commanded altitude perturbation is obtained by inverting (25), as

$$\Delta h_{com} = H_{atm} \ln\left(1 - \frac{\Delta a}{\dot{\underline{v}}}\right). \tag{28}$$

The gains $k_{0,D}$ and $k_{1,D}$ are computed by fixing the closed-loop poles of (26) and (27), which makes them dependent on a_{vs}. The resulting control law is a combination of nonlinear dynamics inversion as in Reiner et al. [33], disturbance rejection and state feedback law.

The lower level dynamics describes the evolution of Δh and $\Delta\gamma$, and is forced by the perturbation of the lift vertical component Δu_{dr}. State equations repeat (26) upon a change of coordinates as follows

$$\begin{bmatrix} \Delta h \\ \Delta\gamma \\ x_{d,L} \\ x_{d2,L} \end{bmatrix} (i+1) = \begin{bmatrix} 1 & a_{h\gamma}(i) & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \Delta h \\ \Delta\gamma \\ x_{d,L} \\ x_{d2,L} \end{bmatrix} (i) + \begin{bmatrix} 0 \\ b_{\gamma}(i) \\ 0 \\ 0 \end{bmatrix} \Delta u_{dr}(i) + \begin{bmatrix} w_{L,3} \\ w_{L,0} \\ w_{L,1} \\ w_{L,2} \end{bmatrix} (i). \tag{29}$$

Equation (29) is the cascade of the controllable dynamics (Δh and $\Delta\gamma$) driven by $\Delta u_{dr}(i)$ and of the disturbance dynamics ($x_{d,L}$ and $x_{d2,L}$) only driven by noise components. The noise estimator has the same structure of the higher level, and employs the pseudo-measurements of Δh and $\Delta\gamma$ to update $x_{d,L}$. The control law has the same structure of the higher level law (27), that is,

$$\Delta u_{dr} = \begin{bmatrix} k_{0,L} & k_{1,L} \end{bmatrix} \begin{bmatrix} \Delta h_{com} - \Delta h \\ \Delta\gamma \end{bmatrix} - x_{d,L} \,. \tag{30}$$

The control law (30), besides cancelling $x_{d,L}$, stabilizes Δh around Δh_{com} and $\Delta\gamma$ around zero. The gains $k_{0,L}$ and $k_{1,L}$ are computed by fixing the closed-loop poles of (29) and (30).

The feedback gains of (27) and (30) are computed in real-time from the time-varying parameters given by (10) and (11), in such a way that the closed-loop eigenvalues $\lambda_k = 1 - \gamma_k$, $k = 0,1$ are kept fixed. Eigenvalues are related to the coefficients c_0 and c_1 of the characteristic polynomial written in the variable $\gamma = \lambda - 1$ as

$$c_0 = \gamma_0\gamma_1, \; c_1 = \gamma_0 + \gamma_1 \,. \tag{31}$$

The characteristic polynomial of the closed-loop dynamics holds

$$\det\left((1-\gamma)I - (A_c - B_c K)\right) = 0 \,, \tag{32}$$

where $A_c(i)$ and $B_c(i)$ are the state transition and command matrices of the (controllable) time-varying dynamics in (26) and in (29). The closed form of the elements of $K(i)$ is the following

$$\begin{aligned} &k_{0,v} = c_0 / a_{sv}, \; k_{1,v} = c_1 \\ &k_{0,\gamma} = c_0 / \left(b_{\gamma} a_{h\gamma}\right), \; k_{1,v} = c_1 / b_{\gamma} \end{aligned} \,. \tag{33}$$

Linearization of the longitudinal dynamics has been made around the reference trajectory computed by the path-planning algorithm at the EIP, and it remains valid during the whole atmospheric entry. However, limited controllability at the beginning and at the end of the entry phase, because of small b_{γ} and a_{vh}, makes impractical a closed-loop control of the capsule displacement, as it may lead to command saturation, and instability conditions. The overcoming approach is to employ weighting signals for scaling the location of the closed-loop eigenvalues taking them close to one (infinite time-constant, zero-state feedback) when control authority becomes low, and

shifting them to their nominal location when control authority reaches the maximum value, which occurs at the peak of the dynamic pressure.

A block diagram showing the overall longitudinal reference-path-tracking law is shown in Fig 8.

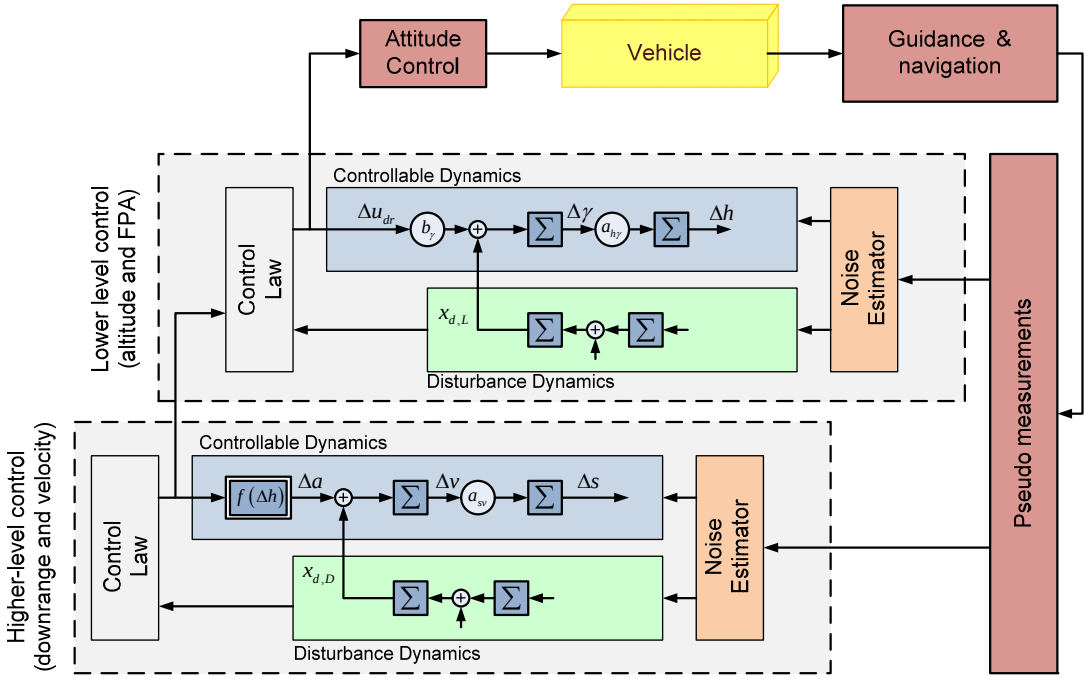

Fig 8. Block-diagram of the longitudinal reference-path-tracking algorithm.

C. Lateral control

Lateral trajectory is computed by the path-planning algorithm. Variations of the bank angle due to longitudinal control may affect lateral dynamics. On the other hand lateral path planning (a sub problem of the path-planning algorithm) computes in (18) the sensitivity matrix S_{cr} of the lateral state (cross-range and heading) at the parachute deployment point with the respect to the time shift of the bank reversal maneuvers. Restricting the sensitivity to cross-range y, equation (17) becomes a scalar equation as follows

$$k_{yn} = \Delta y(m-1) / \Delta t_n, \; n = 1, \ldots, N \tag{34}$$

where $\Delta y(m-1)$ is the cross-range variation at the PDP caused by a time displacement Δt_n of the n-th bank reversal, and N is the number of bank reversals to be performed. The time instants of each bank reversal is thus adjusted to recover the current deviation $\Delta y(i)$ of the cross-range, as follows

$$\Delta t_n(i) = -\Delta y(i) / k_{yn}. \tag{35}$$

VI. Simulated results

The capsule motion has been computed by a high-fidelity simulator which includes the 6-DoF motion of the vehicle, off-the-shelf pulsed thrusters and a state-of-the art inertial measurement unit. The atmospheric parameters are extracted from the European Mars Climate Database (EMCD), the aerodynamics coefficients are interpolated from a 2D database driven by Mach number and total angle of attack. Gravity includes the planet flattening component (J2).

A. 10 % error of the drag coefficient

The time evolution of the tracking law, combined with the path-planning algorithm and the attitude control in Canuto et al. [25] is shown below assuming a 10 % dispersion of the drag aerodynamic coefficient. This should be a worst-case condition since 10 % corresponds to the 3σ dispersion of the capsule aerodynamic database.

Under these conditions, the drag experienced by the capsule becomes greater than the one expected by the guidance. The drag deviation in Fig 9, left, is estimated as a drag disturbance $x_{d,D}$ by the noise estimator of the higher level control (see Fig 7), and is converted by (27) and (28) into a commanded altitude variation Δh_{com} (a higher altitude of about 1.5 km is requested as in Fig 9, right). The lift vertical component $\underline{u}_{dr}$ is then adjusted to Δu_{dr} by the lower level control in (30) to drive the altitude Δh to track the commanded Δh_{com}. Altitude variation adjusts also velocity variation Δv and downrange variation Δs. The cross-range error remains bounded by adjusting the BRM starting times.

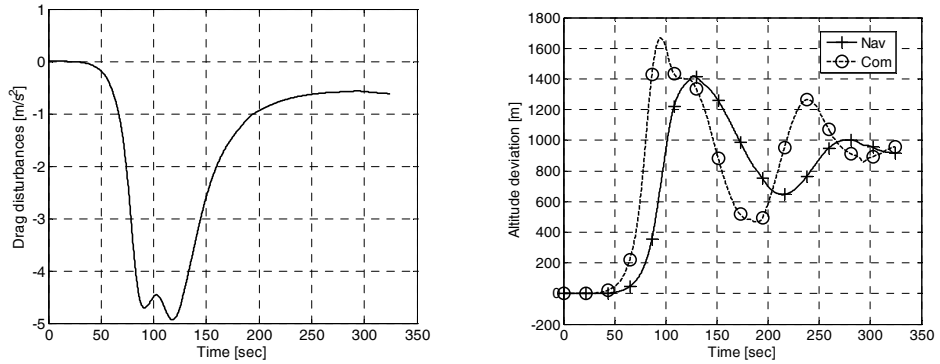

Fig 9. Left: estimated drag acceleration deviation. Right: commanded and navigated altitude deviation.

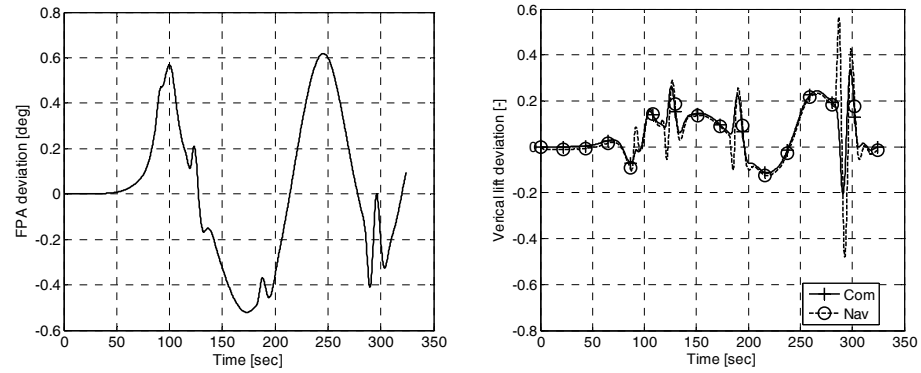

Fig 10. Lower level longitudinal control. Left: FPA deviation. Right: vertical lift deviation (commanded and navigated).

B. Mont Carlo simulations

The outcomes of 500 Monte Carlo runs are outlined. EIP navigation errors are neglected to keep the results aligned with literature.

The initial state (at t_{eip}) and the parametric dispersions are summarized in Table 1 and in Fig 11. Λ and λ denote latitude and longitude, Δx_{CoM} is the center of mass offset providing lift, $u_{t,\max}$ is the maximum thrust of each pulsed thruster, $J_{xx}, J_{yy}, \ldots$ denote inertia matrix entries, c_A and c_N are the axial and normal aerodynamic coefficients. The parachute deployment point is triggered at the time computed by the path-planning trajectory. This means that a time-based trigger (a conservative assumption) is employed to obtain the dispersion plots in Fig 12 and Fig 13. The bias on the Mach error in Fig 12, right, is caused by systematic large-scale deviations of the density compared to the profile used by the path-planning algorithm.

PDP navigated position and velocity dispersions are shown in Fig 12 while the dispersions of the load factor and of the entry phase duration are shown in Fig 13. Taking into account that PDP is triggered on a time-based logic, the dispersion plots in terms of position and velocity are rather encouraging.

Table 1. Parameters and initial state of Monte Carlo runs

Variable	Unit	Value	Dispersion (3σ)	Variable	Unit	Value	Dispersion (3σ)
$r(t_{eip})$	[m]	3516200	5 000	c_A	[–]		10 %
$\Lambda(t_{eip})$	[deg]	0	10 000	c_N	[–]		10 %
$\lambda(t_{eip})$	[deg]	0	10 000	$u_{\gamma,0}$	[–]	-0.5	-
$v(t_{eip})$	[m/s]	5560	20	$u_{T,\max}$	[N]	160	10 %
$\gamma(t_{eip})$	[deg]	-13.4	0,05	J_{xx}	[kg/m^2]	2400	5 %
$\chi(t_{eip})$	[deg]	90	0,05	J_{yy}	[kg/m^2]	2000	5 %
m	[kg]	2200	1 %	J_{zz}	[kg/m^2]	1400	5 %
Δx_{CoM}	[m]	0.1	0.005	$J_{xy-xz-yz}$	[kg/m^2]	0	10
ρ	[kg/m^3]	EMCD	EMCD				

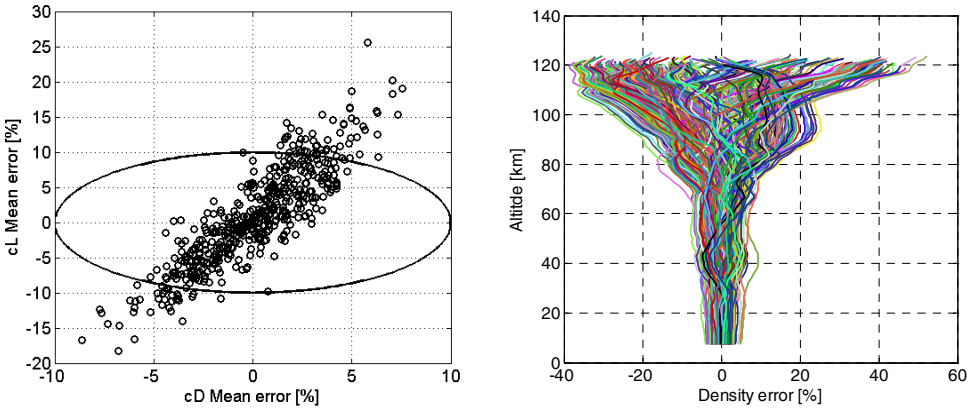

Fig 11. Drag and lift coefficient dispersion and atmospheric density variation.

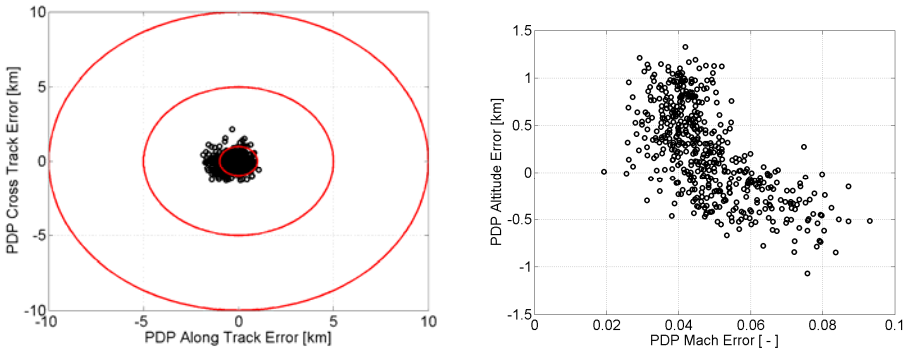

Fig 12. PDP dispersion: horizontal position (left) and altitude versus velocity (right).

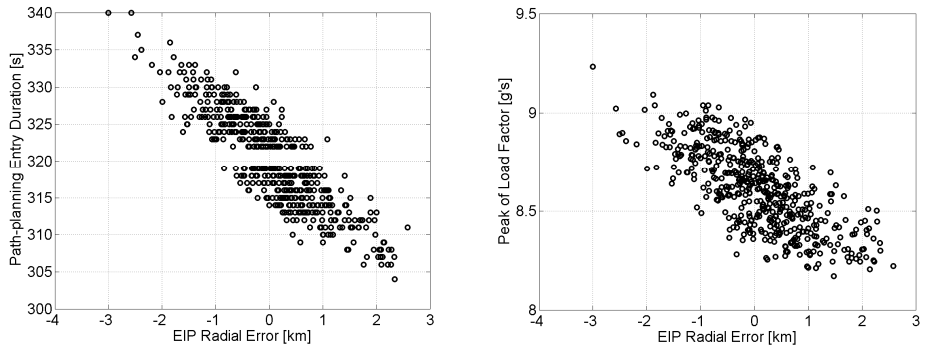

Fig 13. Dispersion of the entry phase duration (left) and of the load factor (right).

VII. Conclusion

A detailed analysis and solution of the guided atmospheric entry problem have been presented without formal proofs of stability, convergence and performance. The case of a low-lift-to-drag biconic capsule has been considered. The solution combines a path-planning algorithm for recovering the Entry-Interface-Point dispersion with a reference-path-tracking algorithm for compensating atmospheric and aerodynamic uncertainty. Standard separation between longitudinal and lateral dynamics has been extended by further decomposing longitudinal dynamics into down-range and altitude dynamics.

The path-planning algorithm is coherent with the cited literature and exploits the linearization of the longitudinal dynamics. The concept of optimal entry duration has been studied and implemented with the scope of reducing the bank-angle variations (with respect to nominal value) that are demanded for recovering the Entry-Interface-Point dispersion.

The core of the paper is devoted to the reference-path-tracking algorithm, that has been developed on the basis of the EMC methodology. The algorithm design starts from the perturbed longitudinal dynamics (obtained as the linearization of the longitudinal dynamics), splits the fourth order dynamics in two second-order parts, one describing downrange and velocity and the other altitude and FPA evolution. Then it designs two control algorithms interconnected in a hierarchical structure. Controlled altitude and FPA become the actuators of the downrange/velocity dynamics. Each control algorithm follows the EMC methodology and as such is separated into state predictor (embedded model plus noise estimator) capable of estimating in real time external disturbances and parametric errors, and control law. The latter cancels the effect of the estimated disturbance and stabilizes the dynamics around the reference trajectory.

The time profiles of the simulated variables are such to validate the designed algorithms. In fact, they show that the effects of the aerodynamic uncertainty is bounded and reduced. Horizontal and velocity dispersions at the Parachute Deployment Point (from Monte Carlo runs) remains within a 2 km horizontal ellipse (3 σ radius) and less than 10 m/s (0.05 Mach). Such values are obtained in the worst case of a time-triggered logic for parachute deployment.

Acknowledgments

One of the authors is grateful to Thales Alenia Space Italia for the financial support.

References

1 Wolf, A. A., Tooley, J., Ploen, S., Ivanov, M., Acikmese, B., and Gromov, K., "Performance Trades for Mars Pinpoint Landing," *2006 IEEE Aerospace Conference*, 2006, pp. 1-16, doi: 10.1109/AERO.2006.1655793.

2 Braun, R. D., and Manning, R. M., "Mars Exploration Entry, Descent and Landing Challenges," *2006 IEEE Aerospace Conference*, Vol. 150, No. 404, 2006, pp. 1-18, doi: 10.1109/AERO.2006.1655790.

3 Carman, G. L., Ives, D. G., and Geller, D. K., "Apollo-Derived Mars Precision Lander Guidance," *Proceedings of the AIAA Atmospheric Flight Mechanics Conference*, AIAA-1998-4570, Boston, MA, 1998.

4 Rolland, J.Y. "Atmospheric Reentry Demonstrator: Flight Control Post-Flight Restitution," *AAAF 2nd Int. Symp. on Atmospheric Reentry Vehicles and Systems*, Arcachon, France, March 26-29, 2001.

5 Harpold, J. C., and Gavert D. E. "Space shuttle entry guidance performance results," *Journal of Guidance, Control, and Dynamics,* Vol. 6, No. 6, 1983, pp. 442-447.

6 Way, D.W., Powell, R.W., Chen, A., Steltzner, A.D., San Martin, A. M., Burkhart, P. D. and Mendeck, G.F. "Mars Science Laboratory: entry, descent and landing system performance," *IEEE Aerospace Conference*, March 3-10, 2006, paper No. 1467.

7 Kluever, C. A. "Entry Guidance Performance for Mars Precision Landing," *Journal of Guidance, Control and Dynamics*, vol. 31, No. 6, 2008, pp. 1537-1544, doi: 10.2514/1.36950.

8 Hormigo, T., Silva, J. A. and Camara, F., "Nonlinear Dynamic Inversion-based Guidance and Control for a Pinpoint Mars Entry, " *AIAA Guidance, Navigation and Control Conference and Exhibit*, Honolulu, 2008, pp. 1-15, AIAA-2008-6817.

9 Bharadwaj, S., Mease, K.D., and Rao, A.V., "Entry Trajectory Tracking Law via Feedback Linearization," *Journal of Guidance, Control, and Dynamics*, Vol. 21, No.5, September 1998, pp. 726-732, doi: 10.2514/2.4318.

10 Andres, M. "Atmospheric re-entry NDI control design for the hopper RLV concept," *17th IFAC Symposium on Automatic Control in Aerospace,* Vol. 3, 2007, pp. 786-791.

11 Tu, K.-Y., Bayard, D. S., Mease, K. D., and Munir, M. S., "Drag-Based Predictive Tracking Guidance for Mars Precision Landing," *Journal of Guidance, Control, and Dynamics*, Vol. 23, No. 4, 2000, pp. 620-628.

12 de Lafontaine, J. and Kron, A. "Robust Guidance and Control Algorithms using Constant Flight Path Angle for Precision Landing on Mars," *AIAA Guidance, Navigation, and Control Conference and Exhibit*, Keystone, August 2006, pp. 1-21, AIAA-2006-6075.

13 Mease, K.D, Chen, D.T., Teufel, P. and Schoenenberger, H. "Reduced-order entry trajectory planning for acceleration guidance," *Journal of Guidance, Control Dynamics*, Vol. 25, No. 2, 2002, pp. 257-266.

14 Bollino, K.P, Ross, I. M. and Doman, D.D. "Optimal nonlinear feedback guidance for reentry vehicles," *AIAA Guidance, Navigation and Control Conference*, Keystone, Colorado, 2006.

15 Braun, R.D., and Powell, R.W. "Predictor-corrector guidance algorithm for use in high-energy aero-braking system studies," *Journal of Guidance, Control Dynamics*, Vol. 15, 1992, pp. 672-678.

16 Joshi, A., Sivan, K. and Amma, S.S. "Predictor-corrector reentry guidance algorithm with path constraints for atmospheric entry vehicles," *Journal of Guidance, Control Dynamics*, Vol. 30, 2007, pp. 1307-1318.

17 Bairstow, S.H., and Barton, G.H. "Orion reentry guidance with extended range capability using Predictive Guidance," *AIAA Guidance, Navigation and Control Conference*, Hilton Head, South Carolina, 2007.

18 Lu, P. "Predictor-corrector entry guidance for low-lift vehicles," *Journal of Guidance, Control Dynamics*, Vol. 31, 2008, pp. 1067-1075.

19 Hamel, J.F., and de Lafontaine J. "Improvement to the analytical predictor-corrector guidance algorithm applied to Mars aerocapture," *Journal of Guidance, Control Dynamics*, Vol. 29, 2006, pp. 1019-1022.

20 Slotine, J.-J., *Applied Nonlinear Control*, London, Prentice-Hall, 1991.

21 Canuto, E., "Embedded Model Control: outline of the theory," *ISA transactions*, Vol. 46, No. 3, June 2007, pp. 363-77, doi: 10.1016/j.isatra.2007.01.006.

22 Canuto, E., "Drag-free and attitude control for the GOCE satellite," *Automatica*, Vol. 44, No. 7, July 2008, pp. 1766-1780, doi: 10.1016/j.automatica.2007.11.023.

23 Canuto, E., Molano, A., and Massotti, L. "Drag-free control of the GOCE satellite: noise and observer design, " *IEEE Trans. Control Systems Technology*, Vol. 18, No. 2, March 2010, pp. 501-509, doi: 10.1109/TCST.2009.2020169.

24 Canuto, E., Acuna-Bravo, W., A. Molano-Jimenez, A., and Perez-Montenegro, C., "Embedded Model Control calls for disturbance modelling and rejection," *ISA Transactions*, Vol. 51, No. 5, 2012, pp. 584-595, doi: 10.1016/j.isatra.2012.04.002.

25 Canuto, E., Ospina, J., and Buonocore, M., "Model-based aerodynamic-angle attitude control of an atmospheric entry capsule," *AIAA Atmospheric Flight Mechanics Conference*, August 13-16, 2012, Minneapolis, MA.

26 Buonocore, M., Calantropio, F., Martelli A., Oddenino D., Ospina J., and Canuto, E. , "Guided Entry : A Necessary Step from Exomars to Precision Landing on Mars," *4th European Conference for Aerospace Science, EUCASS 2011*, Saint Petersburg, 2011.

27 Han, J. "From PID to Active Disturbance Rejection Control," *IEEE Transactions on Industrial Electronics*, Vol. 56, No. 3, March 2009, pp. 900-906, doi: 10.1109/TIE.2008.2011621.

28 Vinh, N. X., *Optimal Trajectories in Atmospheric Flight*, Amsterdam: Elsevier Scientific Software, 1981.

29 Benito, J. & Mease, K.D. "Reachable and Controllable Sets for Planetary Entry and Landing," *Journal of Guidance, Control and Dynamics*, Vol. 33, No. 3, 2010, pp. 641-654.

30 Hermann, R. and Krener, A., "Nonlinear controllability and observability," *IEEE Transactions on Automatic Control*, Vol. 22, No. 5, October 1977, pp. 728-740, doi: 10.1109/TAC.1977.1101601.

31 Eduardo D. Sontag, *Mathematical Control Theory: Deterministic Finite Dimensional Systems*, New York: Springer, 1998.

32 Stengel, R. F., *Flight Dynamics*, New Jersey: Princeton University Press, 2004.

33 Reiner, J., Balas, G. J., and Garrard, W. L., "Flight control design using robust dynamic inversion and time-scale separation," *Automatica*, Vol. 32, No. 11, November 1996, pp. 1493-1504, doi: 10.1016/S0005-1098(96)00101-X.