JOURNAL OF SPACECRAFT AND ROCKETS
Vol. 40, No. 1, January–February 2003

Reentry Flight Controller Design Using Nonlinear Dynamic Inversion

R. R. da Costa,* Q. P. Chu,† and J. A. Mulder‡
Delft University of Technology, 2600 GB Delft, The Netherlands

The application of nonlinear dynamic inversion to the design of a flight controller for atmospheric reentry is discussed. Nonlinear dynamic inversion is used due to the large flight envelope that characterizes the reentry of the small wingless lifting body vehicle. Moreover, the increased computational capability of modern flight computers, as well as the mission requirements of crew return vehicles, supports the application of a nonlinear control law. The control design is based on nonlinear/linear inversion of the vehicle model and is divided in three phases: nonlinear inversion of the vehicle's dynamics, linearization and linear inversion of the aerodynamic database, and actuator allocation. The vehicle's dynamics were inverted assuming timescale separation between attitude and attitude rates. The assignment of the vehicle's actuators was scheduled against dynamic pressure, accounting for the efficiency of each actuator on the different phases of flight. Input–output linearization was performed for the phase of flight where only two actuators were available. The correctness of the controller and its performance are evaluated with numerical simulations of two different entry trajectories.

Nomenclature

B	=	bandwidth matrix
C_x	=	generic aerodynamic coefficient
$\boldsymbol{F}_V$	=	external force in the vertical frame, N
I	=	inertia matrix, kg · m^2
K_{Ii}	=	inner-loop integral gain
K_{Io}	=	outer-loop integral gain
K_{Pi}	=	inner-loop proportional gain
K_{Po}	=	outer-loop proportional gain
K_β	=	sideslip angle gain
K_σ	=	bank angle gain
$K_{\dot\sigma}$	=	bank angle rate gain
$\boldsymbol{M}_{\text{aero}}$	=	aerodynamic moment vector, N · m
$\boldsymbol{M}^B$	=	external moment vector in the body frame, N · m
$\boldsymbol{M}_T$	=	thruster moment vector, N · m
m	=	mass of the vehicle, kg
p, q, r	=	angular rates in body frame, rad/s
Q_i	=	ith quaternion
$\bar{q}$	=	dynamic pressure, Pa
R	=	distance toward the center of Earth, m
$T_{V\leftarrow W}$	=	transformation from wind to vertical frame
$\boldsymbol{u}$	=	input vector
V	=	velocity vector, m/s
v_R	=	velocity in down direction, m/s
v_δ	=	velocity in north direction, m/s
v_τ	=	velocity in east direction, m/s
$\boldsymbol{x}$	=	state vector
$\boldsymbol{y}$	=	output vector
α	=	angle of attack, deg
β	=	sideslip angle, deg
δ	=	latitude, °
δ_a	=	aileron deflection, deg
δ_b	=	speed brake deflection, deg
δ_e	=	elevator deflection, deg
δ_r	=	rudder deflection, deg
σ	=	bank angle, deg
τ	=	longitude, °
Ω_E	=	Earth's angular velocity, rad/s
ω	=	angular velocity, rad/s

Subscripts

red	=	redefined
trim	=	aerodynamic trim condition

Superscripts

cmd	=	commanded
des	=	desired
err	=	error

Introduction

FUTURE cargo and crew return vehicles will have increased flexibility to perform a large set of different reentry missions. Guidance, navigation, and control (GNC) has been identified as one of the main key areas where new technology may decrease operational costs and increase safety. Several authors have investigated the possibility of increasing the set of feasible missions by introducing nonlinear control laws.[1,2] New flight controller algorithms based on more precise knowledge of the vehicle's dynamics showed improvements in performance and robustness against stochastic parameter variation.[3,4]

This paper discusses the application of nonlinear dynamic inversion (NDI) to a lifting body reentry vehicle (LBRV). NDI is an approach to nonlinear control design that has attracted research interest in recent years. The main idea of the approach is to transform the dynamics of the nonlinear system into a linear one by feedback of the system states or outputs. The main difference between this approach and the conventional linearization is that the linearized system is achieved by exact state transformations and feedback, rather than by linear approximations.[5,6] At the present time, NDI is being more commonly used in the design of GNC systems for fighter aircraft.[7–9] The applicability of NDI in spacecraft control, and in particular in reentry guidance and control, has been showed in recent studies.[10,11]

The flight controller design for a reentry vehicle must account for the dynamic range on which the spacecraft operates. The drag

Received 3 August 2001; revision received 3 September 2002; accepted for publication 3 September 2002.

*Graduate Student, Faculty of Aerospace Engineering; currently Guidance, Navigation, and Control System Engineer, Flight Dynamics and Flight Software Systems Department, Mail Code IO64, Astrium Space Infrastructure, P.O. Box 286156, 28361 Bremen, Germany; Rodrigo.daCosta@astrium-space.com. Member AIAA.

†Assistant Professor, Control and Simulation Division, Faculty of Aerospace Engineering, P.O. Box 5058; Q.P.Chu@LR.TUDelft.NL.

‡Professor, Chairman of the Control and Simulation Division, Faculty of Aerospace Engineering, P.O. Box 5058; J.A.Mulder@LR.TUDelft.NL.

vs velocity reference profile is mainly tracked by controlling the direction of the lift vector[12]: To increase drag, the lift vector is directed sideward, whereas decrease in drag is achieved by pointing the lift vector upward. A bang–bang controller is also included to drive the vehicle to the landing site. This path control technique results in demanding maneuvers in terms of bank angle. Typically, bank angle can change from 70 to −70 deg in short time intervals (about 30 s). The Mach number variation during reentry ranges from Mach 28 to Mach 1.5, whereas the angle of attack profile is approximately constant during the hypersonic phase. Because of heat flux restrictions, the sideslip angle is maintained as close as possible to zero. The reentry flight envelope is larger than the one of a fighter airplane executing demanding maneuvers. The fast variation of the flight states and the poor aerodynamic maneuverability of lifting bodies make the design task challenging. Also, the required tracking accuracy is critical for successful mission accomplishment.

Entry flight controller algorithms operational today[13] were designed based on classical techniques with gain scheduling. The reference trajectory and the controller gains are tuned to specific nominal missions. Future entry vehicles that autonomously perform a set of missions will require flight control laws that enlarge the set of feasible missions, while using the increasing computational power of flight computers available nowadays.

The NDI controller presented here was designed to control the reentry flight of the LBRV. The LBRV is a conceptual small reentry vehicle based on a real-life example.[14,15] The vehicle creates lift by flying at high angles of attack, and it is able to carry a maximum of seven passengers. The model of the system, published in an earlier paper,[16] is a nonlinear six-degree-of-freedom model. Accurate models of sensors and actuators are included, as well as an extended nonlinear aerodynamic database.

The design of NDI entry flight controller is based on the timescale separation concept.[9,17,18] From the nonlinear inversion of the attitude dynamics, the commanded torque is obtained. The inversion of the attitude kinematics is performed using quaternion. Although quaternion are not commonly used on the design of NDI controllers,[4,11,19] singularities on the equations of motion are avoided. The static linear inversion of the aerodynamic database was based on a linearized version of the nonlinear database. The aerodynamics of the vehicle are in the form of lookup tables, which were trimmed and linearized for the entire flight envelope. Analysis of the problem indicated that the selection of which actuator is activated to generate a commanded moment plays an important role on the overall performance. The reaction of the vehicle to activation of thrusters, aerodynamic surfaces, and coupled thrusters and aerodynamic surfaces proved to be different. Thus, actuator assignment is one of the central points of the research presented. The numerical simulations were conducted to verify and validate the designed algorithm. The reference trajectories consisted on two precomputed trajectories obtained from an optimization tool.[20] The two missions studied are high-lift entry trajectories from the International Space Station to two different landing sites. The trajectories were given in terms of angle of attack, bank angle, and sideslip angle time history.

Reentry Vehicle

The NDI controller performance is closely related with the accuracy of the model. In the case of the LRBV, its model was available from previous research.[16] The model comprises three submodels: the motion dynamics, the aerodynamics, and the onboard sensors and actuators.

The vehicle dynamics during the planetary entry is modeled as a six-degree-of-freedom motion. In the modeling of this motion, uncoupled translational and angular motions were considered. The uncoupling between angular and translational motion is widely accepted.[21] The motion is described by means of the equations of motion. The translational motion represents the point-mass trajectory motion, whereas the angular motion represents the rigid-body attitude maneuvers. The body flexibility is neglected, which simplifies the modeling problem.

The equations that describe the translational motion are obtained from Newton's second law (see Refs. 21 and 22). This motion can be modeled by the generic equations of dynamics and kinematics:

$$\begin{bmatrix}\dot{v}_\delta\\ \dot{v}_\tau\\ \dot{v}_r\end{bmatrix} = \frac{1}{m}\boldsymbol{F}_V + \begin{bmatrix} -2\Omega_E v_\tau \sin\delta - \Omega_E^2 R \sin\delta\cos\tau - \dfrac{v_\tau^2\tan\delta + v_\delta v_r}{R}\\ -2\Omega_E(v_r\cos\delta - v_\delta\sin\delta) + \dfrac{v_\tau}{R}(v_\delta\tan\delta - v_r)\\ +2\Omega_E v_\tau\cos\delta + \Omega_E^2 R\cos^2\delta + \dfrac{v_\tau^2 + v_\delta^2}{R}\end{bmatrix} \tag{1}$$

$$\begin{bmatrix}\dot{\delta}\\ \dot{\tau}\\ \dot{R}\end{bmatrix} = \begin{bmatrix}\dfrac{1}{R} & 0 & 0\\ 0 & \dfrac{1}{R\cos\delta} & 0\\ 0 & 0 & 1\end{bmatrix}\begin{bmatrix}v_\delta\\ v_\tau\\ v_r\end{bmatrix} \tag{2}$$

Equation (1) represents the dynamics of the system, whereas Eq. (2) is the kinematics equation of translational motion. In the reentry flight, the total external force acting on the vehicle $\boldsymbol{F}_V$ is the sum of the aerodynamic forces (lift, drag, and side force), gravitational force, and thrust force.

The dynamics equation of angular motion in the body frame (as commonly defined for aircraft) is obtained from Euler's law (see Ref. 21):

$$\begin{bmatrix}\dot{p}\\ \dot{q}\\ \dot{r}\end{bmatrix} = I^{-1}\boldsymbol{M}_B - I^{-1}\left(\begin{bmatrix}p\\ q\\ r\end{bmatrix}\times I\begin{bmatrix}p\\ q\\ r\end{bmatrix}\right) \tag{3}$$

The total external moment acting on the vehicle is composed of the aerodynamic torque and the thrust torques.

The attitude position of the vehicle can be expressed by using the quaternion vector or rotation angles. The angles that express the vehicle's attitude are the attitude angles ϕ, θ, and ψ that define the rotation between the vertical and the body frame, or the aerodynamic angles α, β, and σ that define the rotation between the wind and the body frame. If the attitude position is expressed using angles, the kinematics equation of angular motion will have a singularity. The singularity can appear at several points, depending on the order of rotation.

Quaternions were selected to express the attitude position of the vehicle, and although an extra state is inserted in the system, their use is justified by the absence of singularities in the rotations.[16] The kinematics equation of angular motion using the quaternion vector yields

$$\begin{bmatrix}\dot{Q}_1\\ \dot{Q}_2\\ \dot{Q}_3\\ \dot{Q}_4\end{bmatrix} = \frac{1}{2}\begin{bmatrix}Q_4 & -Q_3 & Q_2\\ Q_3 & Q_4 & -Q_1\\ -Q_2 & Q_1 & Q_4\\ -Q_1 & -Q_2 & -Q_3\end{bmatrix}\begin{bmatrix}p\\ q\\ r\end{bmatrix} \tag{4}$$

For user interface purposes, the quaternion vector is transformed to Euler attitude angles. The transformation is performed by well-known methods.[21]

The available nonlinear aerodynamic database is presented in the form of lookup tables, ranging from Mach 0.5 to Mach 28. These tables were generated merging results from computational fluid dynamics with results from wind-tunnel tests.[23] The aerodynamic computations are divided in two speed regions: subsonic/supersonic (Mach number ≤ 4.6) and hypersonic (Mach number ≥ 4). The aerodynamic coefficients for Mach numbers between 4 and 4.6 are computed using a "bridging" function. The resulting aerodynamic force and moment coefficients are obtained as a function of the control surface deflections, aerodynamic angles, angular rates, Mach number, Knudsen number, air density, and air temperature.

The LBRV has two actuators used during reentry: the reaction control system (RCS) and the aerodynamic surfaces. The RCS is

mostly used in the orbital phase of the spacecraft. However, thrusters are also used in the early phase of the reentry flight, whereas in the end they are almost ineffective. This system consists of several thrusters located in groups in the nose and rear portions of the spacecraft. It is considered that the thrusters produce pure moments, being the total force caused by the RCS neglected.

There are four aerodynamic surfaces in the LBRV: two rudders and two flaps, each one positioned in one of the sides of the spacecraft, as shown in Fig. 1. These surfaces can be driven independently, which means that the flaps and the rudders can turn simultaneously or differentially. The deflection of the surfaces can be coupled, to simulate the traditionally defined control deflections (elevator, rudder, flap, and speed brake).

The LBRV was simulated by a simulation tool for atmospheric reentry dynamics.[16] An accurate atmospheric model was used, and the simulation of the aerodynamic surfaces included rate and acceleration limitations. GESARED, implemented in MATLAB®/SIMULINK, is a sophisticated and efficient tool developed for the design and validation of GNC systems.[3,24–26]

Attitude Controller Concept

The concept of NDI is simple and straightforward. In the ideal case, the purpose is to obtain a closed-loop system that behaves as a linear one. Therefore, the controller is designed such that all of the nonlinearities of the plant are canceled. Thus, the nonlinear controller contains the inverted nonlinearities of the plant. The output of the controller is computed by using the inverse model of the open-loop system, and the desired linear closed-loop system is achieved.[5]

The principal consequence of the working principle of NDI is the dependence between the performance of the controller and the quality of the available model of the system. If the model of the system is accurate, and if its inversion is possible considering no approximations, then the controller will have a satisfactory performance. In other words, the NDI controller shows satisfactory performance if the system considered is simple or well known. When the model of the system is not accurate, or the inversion of the model is not possible without simplifications, the performance of the controller is degraded.

The theory behind NDI is mathematically formal and elaborated. The theory, developed a few years ago, still contains drawbacks: NDI can not be used in all nonlinear systems, and it has no ensured robustness to perturbations. However, current studies consider the application of robust control techniques to closed-loop systems linearized by a NDI controller.[8,4,24]

The NDI attitude controller discussed in this paper is shown in Fig. 2. Its design is divided in two parts: the inversion of the equations of motion and the inversion of the behavior of the actuators (thrusters and aerodynamic surfaces).

The inversion of the equations of motion consists of the inversion of the dynamics motion's model presented in the preceding section. When it is assumed that the dynamics are 10 times faster than the kinematics, the timescale separation principle can be invoked. This principle renders into the separation of the inversion of the equations of angular motion in two loops: 1) the inner (or fast) loop, which consists in the inversion of the dynamics equation of angular motion, and 2) the outer (or slow) loop, which performs the inversion of the kinematics equation of angular motion.

The timescale separation assumption is commonly accepted[9,17,18] and simplifies the inversion problem. The inverted equations of motion are responsible for the computation of the total desired moment to achieve a reference angular maneuver. However, it is not possible to feed this desired moment directly into the open-loop system. The moment should be first transformed into thrust commands and deflections of the aerodynamic surfaces, and therefore, the modeling and inversion of the actuator's behavior is required.

In the case of the LBRV, the inversion of the thrust behavior is unnecessary because the desired external moment can be directly fed into the system. For the behavior of the aerodynamic surfaces, the inversion is necessary and consists in the inversion of the nonlinear aerodynamic database. Thus, the commanded external moment is transformed into surface deflections by means of the aerodynamic inversion.

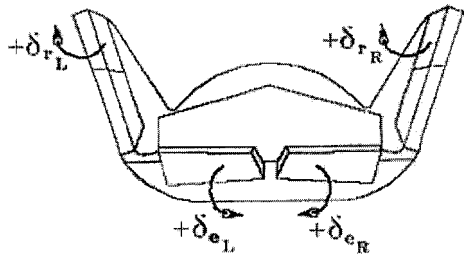

Fig. 1 Aerodynamic surfaces of the LBRV.

Inversion of the Equations of Motion

The first step of the NDI attitude controller implementation is the inversion of the equations of angular motion. Assuming the timescale separation, the NDI attitude controller is composed by two control loops: the inner and the outer loop. In terms of the inversion of the equations of motion, this time division is reflected in the separate inversion of the attitude kinematics and dynamics.

The attitude dynamics, represented in Eq. (3), are here repeated in state-space form:

$$\dot{\boldsymbol{x}}_1 = \boldsymbol{f}_1(\boldsymbol{x}_1, \boldsymbol{u}) \tag{5}$$

$$\boldsymbol{y}_1 = h(\boldsymbol{x}_1) \tag{6}$$

where

$$\boldsymbol{x}_1 = \begin{bmatrix} p \\ q \\ r \end{bmatrix}, \qquad \tilde{\boldsymbol{u}} = \begin{bmatrix} M_x^B \\ M_y^B \\ M_z^B \end{bmatrix} = \begin{bmatrix} g_1(\delta_r, \delta_a, \delta_e, \delta_b) + M_x^T \\ g_2(\delta_r, \delta_a, \delta_e, \delta_b) + M_y^T \\ g_3(\delta_r, \delta_a, \delta_e, \delta_b) + M_z^T \end{bmatrix}$$

$$\boldsymbol{y}_1 = \begin{bmatrix} p \\ q \\ r \end{bmatrix} \tag{7}$$

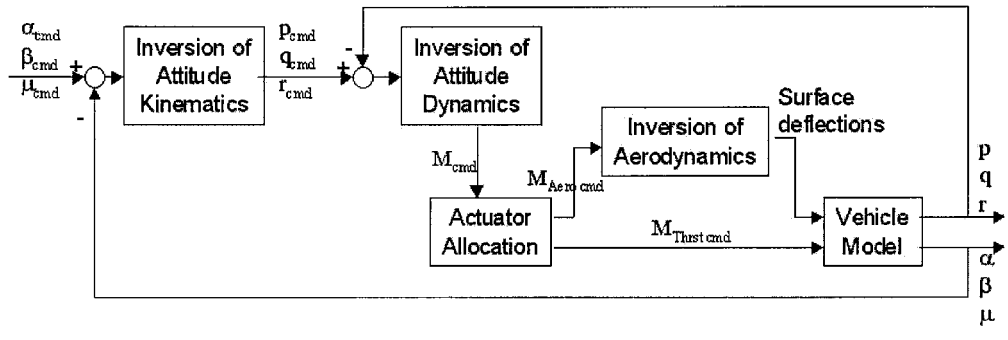

Fig. 2 NDI attitude controller concept.

whereas the state-space representation of the kinematics [Eq. (4)] yields

$$\dot{\boldsymbol{x}}_2 = \boldsymbol{f}_2(\boldsymbol{x}_1, \boldsymbol{x}_2) \tag{8}$$

$$\boldsymbol{y}_2 = h(\boldsymbol{x}_2) \tag{9}$$

where

$$\boldsymbol{x}_2 = \begin{bmatrix} Q_1 \\ Q_2 \\ Q_3 \\ Q_4 \end{bmatrix}, \qquad \boldsymbol{y}_2 = \begin{bmatrix} Q_1 \\ Q_2 \\ Q_3 \\ Q_4 \end{bmatrix} \tag{10}$$

Equations (5) and (8) represent the two state-space nonlinear systems to be inverted. These two systems are characterized by having the states as outputs. Therefore, the navigation state should include the angular rates and the quaternions (or other representation of the attitude that can be transformed to the quaternion vector, e.g., aerodynamic angles).

In accord with the feedback linearization theory,[5] the inversion of the angular dynamics equation is a state-input inversion problem. Equation (6) is not immediately invertible, and thus, the output should be successively time differentiated until inversion is possible, or in other words, the output should be time differentiated $\boldsymbol{r}$ times, where r is the relative degree of the system. After one time differentiation, Eq. (6) yields

$$\dot{\boldsymbol{y}}_1 = \begin{bmatrix} \dot{p} \\ \dot{q} \\ \dot{r} \end{bmatrix} = I^{-1}\boldsymbol{M}_B - I^{-1}\left(\begin{bmatrix} p \\ q \\ r \end{bmatrix} \times I \begin{bmatrix} p \\ q \\ r \end{bmatrix} \right) \tag{11}$$

The analytical inversion of the Eq. (11) yields

$$\begin{bmatrix} M_x \\ M_y \\ M_z \end{bmatrix} = I \begin{bmatrix} \dot{p} \\ \dot{q} \\ \dot{r} \end{bmatrix} + \begin{bmatrix} p \\ q \\ r \end{bmatrix} \times I \begin{bmatrix} p \\ q \\ r \end{bmatrix} \tag{12}$$

The nonlinear control law is obtained by considering that the total desired differentiated output is given by

$$\dot{\boldsymbol{y}}_1^{\text{des}} = \boldsymbol{y}_1^{\text{cmd}^{(r)}} - k_{r-1}\boldsymbol{y}_1^{\text{err}^{(r-1)}} - \cdots - k_0\boldsymbol{y}_1^{\text{err}^{(0)}} \tag{13}$$

where $\boldsymbol{y}_1^{\text{err}^{(n)}}$ is the tracking error $(\boldsymbol{y}_1^{\text{cmd}} - \boldsymbol{y}_1)$ differentiated n times, and r is the relative degree of the system. The relative degree of the system is one, and Eq. (13) simplifies to

$$\dot{\boldsymbol{y}}_1^{\text{des}} = -k_0\boldsymbol{y}_1^{\text{err}} = B_1\left(\boldsymbol{x}_1^{\text{cmd}} - \boldsymbol{x}_1\right) \tag{14}$$

where B_1 is the bandwidth matrix.

When Eq. (14) is substituted into Eq. (12), the nonlinear control law yields

$$\begin{bmatrix} M_x^{\text{cmd}} \\ M_y^{\text{cmd}} \\ M_z^{\text{cmd}} \end{bmatrix} = I \times B_1 \begin{bmatrix} p^{\text{cmd}} - p \\ q^{\text{cmd}} - q \\ r^{\text{cmd}} - r \end{bmatrix} + \begin{bmatrix} p \\ q \\ r \end{bmatrix} \times I \begin{bmatrix} p \\ q \\ r \end{bmatrix} \tag{15}$$

When it is assumed that the system model exactly represents the vehicles, or in other words, that nonlinearities of the system are fully canceled by the inverse control law, the closed-loop system response is given by the following linear relation:

$$\dot{\boldsymbol{x}}_1 + B_1\left(\boldsymbol{x}_1 - \boldsymbol{x}_1^{\text{cmd}}\right) = \boldsymbol{0} \tag{16}$$

Thus, in the ideal case, the open-loop nonlinear system is linearized by the nonlinear control law by means of the exact state feedback. However, the system nonlinearities are not fully inverted by the control law if the inversion or the model are not exact. The dynamics of the system may not be fully canceled and can make the closed-loop system unstable or with undesired transient responses. In this case, the bandwidth matrix plays a significant role: its tuning can stabilize and improve the performance of the closed-loop system.

The bandwidth matrix will contain proportional–integral (PI) elements. The integrators are responsible for the elimination the steady-state error in the computation of the commanded external moments. In other words, the integrator of the inner loop has to deal not only with the model uncertainties but also with the uncertainties inserted by the inversion of the actuators (in particular the inversion of the aerodynamic database). Although the uncertainties inserted by the actuator inversion could also be solved by inserting an integrator in the outer loop, the actuator inversion is closely linked with the inner loop. Because it is assumed that the behavior of p, q, and r is similar, the matrix B_1 is diagonal, and all of the elements of the diagonal are equal. This means that the proportional and integral gains of the three components of the angular velocity vector are equal. The matrix B_1 yields

$$B_1(s) = \begin{bmatrix} K_{Pi} + \dfrac{K_{Ii}}{s} & 0 & 0 \\ 0 & K_{Pi} + \dfrac{K_{Ii}}{s} & 0 \\ 0 & 0 & K_{Pi} + \dfrac{K_{Ii}}{s} \end{bmatrix} \tag{17}$$

and the error between the commanded angular velocity and the vehicle angular velocity yields

$$\dot{\omega}^{\text{des}} = K_{Pi}(\omega^{\text{cmd}} - \omega) + K_{I_i}\int (\omega^{\text{cmd}} - \omega)\,\mathrm{d}t \tag{18}$$

The tuning of the bandwidth elements K_{Pi} and K_{Ii} was performed aiming for best step response performance, while avoiding actuator saturation.

The procedure of designing the outer-loop controller is very similar to the one applied to the inner loop. Once again, the output Eq. (9) is not immediately invertible and should be time differentiated r times (1 in this case):

$$\dot{\boldsymbol{y}}_1 = \begin{bmatrix} \dot{Q}_1 \\ \dot{Q}_2 \\ \dot{Q}_3 \\ \dot{Q}_4 \end{bmatrix} = \frac{1}{2}\begin{bmatrix} Q_4 & -Q_3 & Q_2 \\ Q_3 & Q_4 & -Q_1 \\ -Q_2 & Q_1 & Q_4 \\ -Q_1 & -Q_2 & -Q_3 \end{bmatrix} \begin{bmatrix} p \\ q \\ r \end{bmatrix} \tag{19}$$

The inverse of the Eq. (19) is obtained from the definition of quaternions[27]:

$$\begin{bmatrix} p \\ q \\ r \end{bmatrix} = -2Q_4 \begin{bmatrix} \dot{Q}_1 \\ \dot{Q}_2 \\ \dot{Q}_3 \end{bmatrix} + 2\dot{Q}_4 \begin{bmatrix} Q_1 \\ Q_2 \\ Q_3 \end{bmatrix} - 2\begin{bmatrix} 0 & -Q_3 & Q_2 \\ Q_3 & 0 & -Q_1 \\ -Q_2 & Q_1 & 0 \end{bmatrix} \begin{bmatrix} \dot{Q}_1 \\ \dot{Q}_2 \\ \dot{Q}_3 \end{bmatrix} \tag{20}$$

If the same line of thought applied to the inner loop is applied to the outer loop, the desired differentiated output $\dot{\boldsymbol{y}}^d$ is given by

$$\dot{\boldsymbol{y}}_2^{\text{des}} = -k_0\boldsymbol{y}_2^{\text{err}^0} = B_2\boldsymbol{x}_2^{\text{err}} \tag{21}$$

where $\boldsymbol{x}_2^{\text{err}}$ is given by[27]

$$\begin{bmatrix} Q_1^{\text{err}} \\ Q_2^{\text{err}} \\ Q_3^{\text{err}} \\ Q_4^{\text{err}} \end{bmatrix} = \begin{bmatrix} Q_4^{\text{cmd}} & Q_3^{\text{cmd}} & -Q_2^{\text{cmd}} & -Q_1^{\text{cmd}} \\ -Q_3^{\text{cmd}} & Q_4^{\text{cmd}} & Q_1^{\text{cmd}} & -Q_2^{\text{cmd}} \\ Q_2^{\text{cmd}} & -Q_1^{\text{cmd}} & Q_4^{\text{cmd}} & -Q_3^{\text{cmd}} \\ Q_1^{\text{cmd}} & Q_2^{\text{cmd}} & Q_3^{\text{cmd}} & Q_4^{\text{cmd}} \end{bmatrix} \begin{bmatrix} Q_1 \\ Q_2 \\ Q_3 \\ Q_4 \end{bmatrix} \tag{22}$$

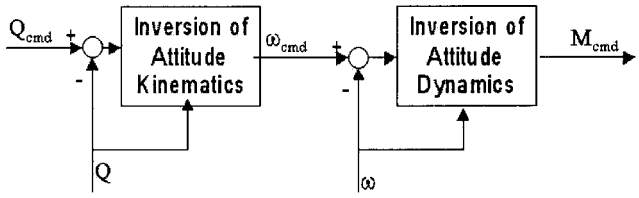

Fig. 3 Inversion of the equations of motion.

and the nonlinear control law is computed by substituting Eqs. (21) and (22) into Eq. (20):

$$\begin{bmatrix} p^{\text{cmd}} \\ q^{\text{cmd}} \\ r^{\text{cmd}} \end{bmatrix} = -2Q_4 \begin{bmatrix} B_{2_{11}} Q_1^{\text{err}} \\ B_{2_{22}} Q_2^{\text{err}} \\ B_{2_{33}} Q_3^{\text{err}} \end{bmatrix} + 2B_{2_{44}} Q_4^{\text{err}} \begin{bmatrix} Q_1 \\ Q_2 \\ Q_3 \end{bmatrix} - 2 \begin{bmatrix} 0 & -Q_3 & Q_2 \\ Q_3 & 0 & -Q_1 \\ -Q_2 & Q_1 & 0 \end{bmatrix} \times \begin{bmatrix} B_{2_{11}} Q_1^{\text{err}} \\ B_{2_{22}} Q_2^{\text{err}} \\ B_{2_{33}} Q_3^{\text{err}} \end{bmatrix} \tag{23}$$

The closed-loop system response is linear, and it is given by

$$\dot{\boldsymbol{x}}_2 + B_2 \boldsymbol{x}_2^{\text{err}} = 0 \tag{24}$$

The dynamic response of the closed-loop is defined by the bandwidth matrix B_2.

Detailed analysis showed that in reality the inversion of the system does not cancel all of the nonlinearities of the plant. In fact, residual steady-state errors in the attitude position tracking are still in the system if no integrator is used in this loop. Therefor, an integrator element in the outer loop is also necessary to eliminate the attitude tracking steady-state error. As well as the inner loop, the outer loop will also have PI elements, to generate the desired quaternion time derivative. As in the case of the matrix B_1, it is also assumed that the bandwidth matrix of the outer loop is diagonal, whose elements are

$$K_{Po} + K_{Io}/s \tag{25}$$

and the error in attitude tracking yields

$$\dot{Q}_i^{\text{des}} = K_{Po}\left(Q_i^{\text{err}}\right) + K_{Io} \int_0^t \left(Q_i^{\text{err}}\right) \mathrm{d}t \tag{26}$$

Once again, the bandwidth matrix was tuned to achieve best step response performance without actuator saturation. The inversion of the equations of motion, obtained by coupling the inner and the outer loop described earlier, is shown in Fig. 3.

Actuator Inversion

During the reentry flight, both the aerodynamic surfaces and RCS are used to control the vehicle. Therefore, the design of the NDI controller should account for the performance characteristics of the actuators, by means of the static actuator inversion.

The thruster activation system computes the opening of the thrusters' valves necessary to generate the commanded moment. The commanded moments are directly fed to the RCS system, and therefore, the inversion of the thrust system is not necessary to the implementation of the NDI controller.

To compute the deflections of the aerodynamic surfaces necessary to produce the desired moment, the aerodynamic inversion is included. The aerodynamic inversion consists of the inversion of the nonlinear aerodynamic database. To overcome the nonlinearities of the database, the suggested NDI controller approach uses three successive steps: trim, linearization, and inversion. The first step of the aerodynamic inversion is to compute the deflection of the aerodynamic surfaces that makes the external moments equal to zero, along the nominal reentry trajectory. With the trim deflections available, it is possible to compute the delta deflections that generate the commanded moments. To linearize the aerodynamic database, it is assumed that the aerodynamic moments depend only on the disturbed motion variables and their derivatives.[28] The simplified linearized moments coefficients yield

$$\begin{bmatrix} C_l \\ C_m \\ C_n \end{bmatrix} = \begin{bmatrix} 0 & \frac{\partial C_l}{\partial \beta} & \frac{\partial C_l}{\partial p} & 0 & \frac{\partial C_l}{\partial r} \\ \frac{\partial C_m}{\partial \alpha} & 0 & 0 & \frac{\partial C_m}{\partial q} & 0 \\ 0 & \frac{\partial C_n}{\partial \beta} & \frac{\partial C_n}{\partial p} & 0 & \frac{\partial C_n}{\partial r} \end{bmatrix} \times \begin{bmatrix} \Delta\alpha \\ \Delta\beta \\ \Delta p \\ \Delta q \\ \Delta r \end{bmatrix} + \begin{bmatrix} 0 & \frac{\partial C_l}{\partial \delta_a} & \frac{\partial C_l}{\partial \delta_r} \\ \frac{\partial C_m}{\partial \delta_e} & 0 & 0 \\ 0 & \frac{\partial C_n}{\partial \delta_a} & \frac{\partial C_n}{\partial \delta_r} \end{bmatrix} \begin{bmatrix} \Delta\delta_e \\ \Delta\delta_a \\ \Delta\delta_r \end{bmatrix} \tag{27}$$

The moment derivatives were numerically computed from the nonlinear database by means of the central difference method.

Finally, the inversion of the linearized database is equivalent to the inversion of Eq. (27). Therefore, the commanded surface deflections are

$$\Delta\delta_e = \left(C_m - C_{m\alpha}\Delta\alpha - C_{mq}\Delta q\right) \Big/ C_{m\delta_e} \tag{28}$$

$$\begin{bmatrix} \Delta\delta_a \\ \Delta\delta_r \end{bmatrix} = C_\delta^{-1} \left(\begin{bmatrix} C_l \\ C_n \end{bmatrix} - C_x \begin{bmatrix} \Delta\beta \\ \Delta p \\ \Delta r \end{bmatrix} \right) \tag{29}$$

where

$$C_x = \begin{bmatrix} C_{l\beta} & C_{lp} & C_{lr} \\ C_{n\beta} & C_{np} & C_{nr} \end{bmatrix}, \qquad C_\delta = \begin{bmatrix} C_{l\delta_a} & C_{l\delta_r} \\ C_{n\delta_a} & C_{n\delta_r} \end{bmatrix}$$

There are two necessary conditions for the inversion: The derivative $C_{m\delta_e}$ must be different from zero, and the C_δ matrix must be invertible. Equations (28) and (29) compute the commanded deflections necessary to accomplish the desired external moments and are implemented in the aerodynamic inversion block of Fig. 2.

Actuator Assignment

The actuator assignment represents a critical point of the flight controller design. The intrinsic nonlinearity of these devices strongly influences the performance of the control algorithm.

Seven actuators are available during the reentry flight: roll X, pitch Y, and yaw Z thrust moments; elevator; rudder; ailerons; and speed brake. Each actuator is only available depending on its own specifications. The thrusters are only available in the beginning of the reentry, whereas the aerodynamic surfaces can only be used when the air density increases. The efficiency of the thrusters depends inversely on the dynamic pressure. The ailerons and elevators of the vehicle can be used almost during the entire flight. However, the rudder can only be used in lower speed because of the high structural stresses created. Table 1 shows the actuators used vs reentry phase, dynamic pressure, and Mach number.

The three components of the external moment vector (roll, pitch, and yaw moment) have a group of actuators assigned for each phase of flight. The roll moment is mainly generated by the X thrust or by the ailerons, whereas the pitch moment is mainly obtained from the Y thrust or by the elevators. Finally, the yaw moment is mainly generated by the Z thrust or by the rudder. When the correspondence between a moment and an actuator is unique, it is clear which

Table 1 Availability of actuators during reentry

Phase	X thruster	Y thruster	Z thruster	δ_a	δ_e	δ_r
1 $\bar{q} \leq 100$	Yes[a]	Yes	Yes	No[b]	No	No
2 $100 < \bar{q} \leq 500$	Yes	Yes	Yes	Yes	Yes	No
3 $500 < \bar{q} \leq 1500$	No	No	Yes	Yes	Yes	No
4 $1500 < \bar{q}$, $M > 6$	No	No	No	Yes	Yes	No
5 $M \leq 6$	No	No	No	Yes	Yes	Yes

[a]Available. [b]Not available.

actuator will be used to generate the desired moment. The exclusive relation between moment and actuator happens almost during the entire flight. However, there are two situations when this doesnot happen: the transition between thrusters and aerodynamic surfaces and the phase where the thrusters are no longer available and the rudder is not yet used (Table 1).

The transition between thrusters and aerodynamic surfaces is performed by means of a bridging function with respect to dynamic pressure:

$$M_T^{\text{cmd}} = k_X M^{\text{cmd}} \tag{30}$$

$$M_{\text{aero}}^{\text{cmd}} = (1 - k_X) M^{\text{cmd}} \tag{31}$$

$$k_X = (\bar{q}_2 - \bar{q})/(\bar{q}_2 - \bar{q}_1) \tag{32}$$

where $\bar{q}_1$ and $\bar{q}_2$ are the dynamic pressures in the beginning and in the end of the transition phase, respectively.

In the phase where the thrusters are no longer available and the rudder is not yet used (phase 4 of Table 1), there is no actuator directly assigned for the yaw moment. Therefore, the angles of attack, sideslip, and bank must be controlled by means of the ailerons and the elevator.

According to the nonlinear dynamic inversion theory, each output of the system must correspond to a specific actuator or that the actuator is allocated to a specific output. When three actuators are available, the output vector in Eqs. (6) and (9) is the state vector, and the so-called state-input inversion is performed. However, if the rudder is not available, the state-input inversion can no longer be used, and the output function $h(\boldsymbol{x})$ must be redefined to make the inversion possible.

A new output vector with two elements has to be defined as a combination of the traditional outputs (aerodynamic angles). Because $C_{m\delta_a}$ is approximately zero, the angle of attack is controlled by the elevator. This surface is closely related to the pitch motion and, therefore, to the angle-of-attack control. The other output can be an association between sideslip and bank angles.[18] Typically, the trajectory has the commanded β always equal to zero because it is known that the sideslip strongly influences the heat load on the vehicle's surface. Because of the bank reversals, during the reentry the bank angle is always high, and its rate is also high. Thus, the joint bank and sideslip angles' control considers the weighted sum of the sideslip angle, the bank angle, and the bank angle rate of change. The redefined system output yields

$$\boldsymbol{y}_{\text{red}} = \begin{bmatrix} y_{1_{\text{red}}} \\ y_{2_{\text{red}}} \end{bmatrix} = \begin{bmatrix} \alpha \\ K_\beta \beta + K_\sigma \sigma + K_{\dot{\sigma}} \dot{\sigma} \end{bmatrix} \tag{33}$$

The new output vector defined in Eq. (33) takes into account that the sideslip angle should be kept close to zero and that the bank angle should be kept close to the value from the precomputed trajectory. The bank angle rate of change takes into account the dynamics of the roll motion.

When once again the state-space representation of the system given by Eqs. (5), (6), (8), and (9) is considered, the state vector, the output vector, and the input vector of the redefined open-loop system are given by

$$\boldsymbol{X}_{\text{red}} = (\alpha, \beta, \sigma, p, q, r)^T, \qquad y_{2_{\text{red}}} = \beta + \sigma + \dot{\sigma}$$

$$\boldsymbol{u}_{\text{red}} = (\delta_e, \delta_a)^T \tag{34}$$

The state-space representation of the redefined system yields

$$\dot{\boldsymbol{x}} = \boldsymbol{f}_{\text{red}}(\boldsymbol{x}) + \boldsymbol{g}_{\text{red}}(\boldsymbol{x})\boldsymbol{u} \tag{35}$$

$$\boldsymbol{y} = h_{\text{red}}(\boldsymbol{x}) \tag{36}$$

The smooth vector fields $\boldsymbol{f}(\boldsymbol{x})$ and $\boldsymbol{g}(\boldsymbol{x})$ and the smooth scalar function $h(\boldsymbol{x})$ can be computed. The output of the system will be differentiated as many times as necessary to make the control output appear in the input equation. When the output equation is differentiated once, it yields

$$\dot{y}_{2_{\text{red}}} = \nabla h_{\text{red}} \boldsymbol{f}_{\text{red}} + \nabla h_{\text{red}} \boldsymbol{g}_{\text{red}} \boldsymbol{u}_{\text{red}} \tag{37}$$

$$\dot{y}_{2_{\text{red}}} = \nabla h_{\text{red}} \left(\boldsymbol{f}_{\text{red}} + g_{1_{\text{red}}} u_{1_{\text{red}}} + g_{2_{\text{red}}} u_{2_{\text{red}}} \right) \tag{38}$$

When it is assumed that $L_g h(\boldsymbol{x}) \neq 0$ for some neighborhood of $\boldsymbol{x}$, the inverse of the Eq. (38) is given by

$$u_{2_{\text{red}}} = \left(\nabla h_{\text{red}} g_{2_{\text{red}}} \right)^{-1} - \left[\dot{y}_{\text{red}}^d - \nabla h_{\text{red}} \left(\boldsymbol{f}_{\text{red}} + g_{1_{\text{red}}} u_{1_{\text{red}}} \right) \right] \tag{39}$$

The desired $\dot{y}_{\text{red}}^d$ yields

$$\dot{y}_{\text{red}}^d = K_P [-K_\beta \beta + K_\sigma (\sigma_{\text{cmd}} - \sigma) - K_{\dot{\sigma}} \dot{\sigma}] + K_I \int [-K_\beta \beta + K_\sigma (\sigma_{\text{cmd}} - \sigma) - K_{\dot{\sigma}} \dot{\sigma}] \, \mathrm{d}t$$

where the constants K_β, K_σ, and $K_{\dot{\sigma}}$ are weights to each component of the tracking error. The use of aerodynamic angles for the controller design of phase 4 (as in Table 1) is justified by the increased simplicity of designing the redefined outputs based on angles. In fact, it is more meaningful to have the redefined outputs expressed as two separate channels, one for the longitudinal motion (angle of attack) and the other for the lateral motion (bank and sideslip angles). Phase 4 is the only exception where aerodynamic angles are used in the flight controller design; the controller for the other phases is based on quaternions, as indicated in Eq. (26).

After the redefinition, the aileron deflection yields

$$\delta_a = \left(\nabla h_{\text{red}} g_{2_{\text{red}}} \right)^{-1} \left[\dot{y}_{\text{red}} - \nabla h_{\text{red}} \left(\boldsymbol{f}_{\text{red}} + g_{1_{\text{red}}} \delta_e \right) \right] \tag{40}$$

As already referred, the angle of attack is controlled using the elevator. The elevator deflection yields

$$\Delta\delta_e = (C_m - C_{m\alpha} \Delta\alpha - C_{mq} \Delta q) \big/ C_{m\delta_e} \tag{41}$$

Results

This section describes the numerical evaluation of the NDI attitude controller designed for the LBRV. The tests consist in tracking two precomputed reentry trajectories with different landing sites, available from the aerospace trajectory optimization software (ASTOS) software tool.[20] The reference trajectories are given in terms of commanded aerodynamic angles time history. The tests were conducted in MATLAB/SIMULINK, and the vehicle's dynamics were simulated using GESARED.

The first trajectory tracking test is presented in Fig. 4. The test results are presented in terms of the aerodynamic angles time history, tracking error, and deflections of the aerodynamic surfaces. The angle-of-attack tracking error is small during the entire trajectory, except at 1397 s, when it grows to −1.2 deg. During the last bank reversal (coincident with phase 4 of Table 1), the bank-angle tracking error reaches its maximum for a short period of time; during the rest of the simulation, the error is always smaller than 3 deg. The sideslip-angle tracking error is always smaller than 0.23 deg. In terms of the deflections of the aerodynamic surfaces, the only remarkable behavior is the saturation of the ailerons and the rudder at 1395 s, when the angle of attack starts decreasing and phase 5 starts. However, the saturation is almost instantaneous, and the vehicle immediately recovers. Around 1395 s, there is a problematic phase in the reentry flight because two factors joined together: the reduction of the angle of attack and the beginning of phase 5 (and consequently the beginning of the use of the rudder). Problematic points of the simulation such as this should be avoided when optimizing the reference trajectory because they generate high tracking errors. The general behavior of the controller for the first reentry trajectory is satisfactory.

Figure 5 presents the results of the second trajectory tracking closed-loop simulation. The results presented consist in the tracking error aerodynamic-angles tracking error and the deflections of the aerodynamic surfaces. In the case of the angle of attack, the maximum tracking error is −1.1 deg. The bank-angle maximum coincides with the decrease in the reference angle of attack. The maximum tracking error in terms of sideslip angle is 0.4 deg. The bank-angle maximum tracking error is high. However, the bank-angle reference maneuvers are also very demanding, and although bank-angle tracking is important in the evaluation of the controller, it is not critical. Because the most critical tracking error is the one of the sideslip angle (due to heat flux problems), we can consider that the controller has a satisfactory performance. Moreover, the NDI controller used in this test was exactly the same used in the first test,

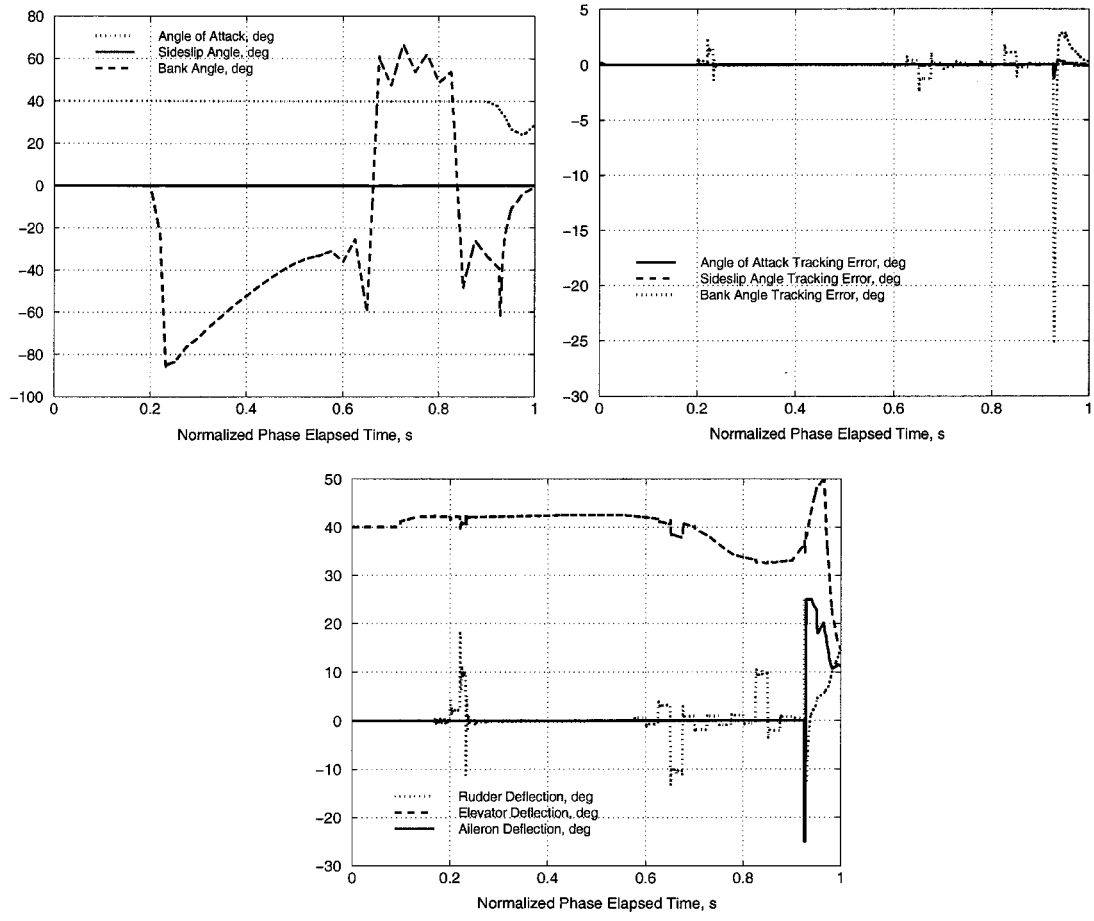

Fig. 4 First trajectory tracking; error attitude angles and surface deflections.

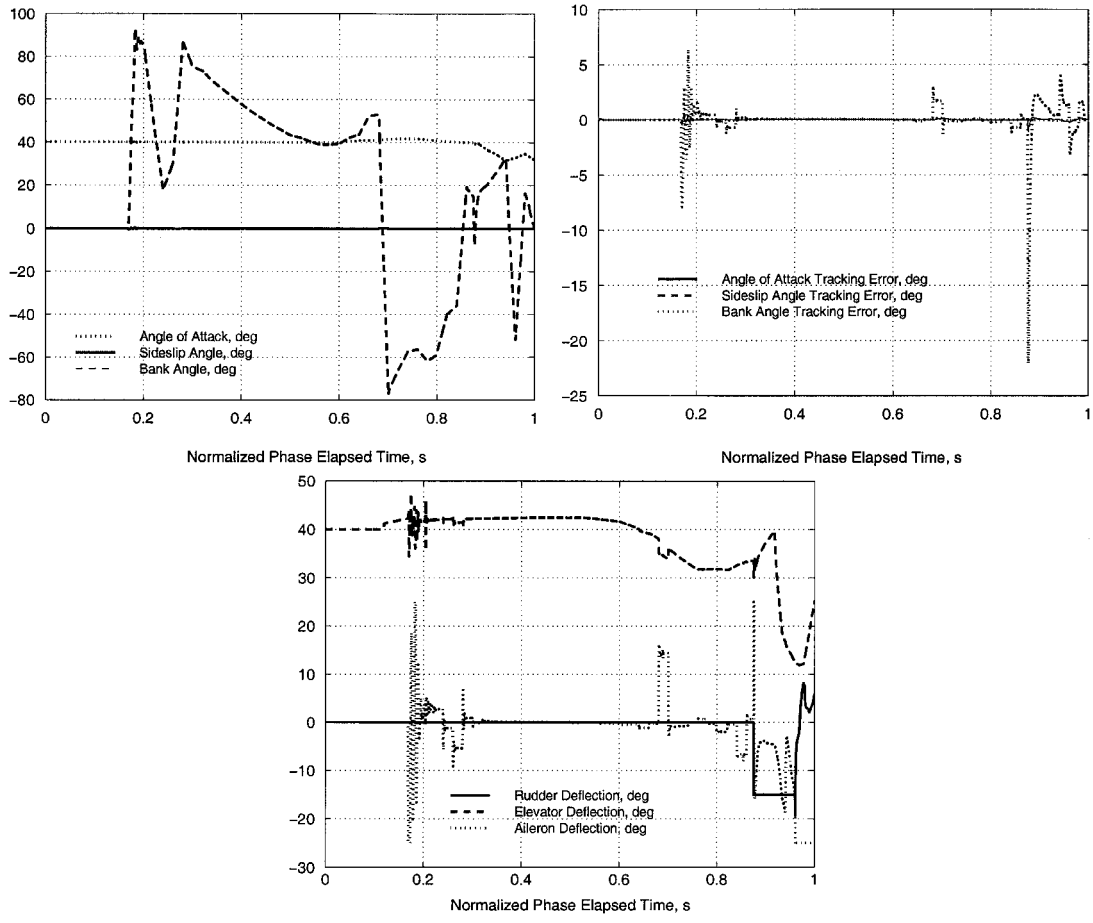

Fig. 5 Second trajectory tracking; error in aerodynamic angles and aerodynamic surfaces deflections.

without further gain tuning or extra computations. Thus, although it is expected that the results can be in some way degraded, they remain satisfactory. Thus, because the NDI controller has a satisfactory behavior in the tracking of both trajectories, it can be considered that there is a considerably wide spectrum of satisfactory performance.

Conclusions

A nonlinear dynamic inversion flight controller design approach for the LBRV is presented. The use of NDI reduces the need of parameter tuning for different missions, contributing to increasing the vehicle's flexibility of tracking different reference trajectories.

Given a nonlinear open-loop system, the concept of the NDI is to create a closed-loop system that behaves as a linear system. In fact, the nonlinearities of the open-loop system are contained within the inversion. Two main advantages of the closed-loop system arise as a direct consequence of the NDI concept: The scheduling of the controller is unnecessary, and the controller is flexible.

On the inversion of the equations of motion, quaternions were used. It was verified that the final results were not degraded, and the singularities that characterize the representation of attitude y angles were avoided. The aerodynamic database was inverted based on a simplified linear model, to decrease CPU time in case of onboard implementation. The simplification inserted undesired errors in the loop. The mismatch in the inversion was solved by appropriate selection of the bandwidth matrices. Inserting integrators on both matrices eliminated steady-state errors.

The models of the actuators used included saturation in the case of the RCS system and saturation, deflection rate, and acceleration in the case of the aerodynamic surfaces. The nonlinear limitations of the actuators inserted limitations on the performance of the controller and required extra effort on the tuning of the proportional gains of the bandwidth matrices.

The actuator assignment proved to be a sensitive point in the entire design strategy. During the phase of flight where only ailerons and elevators are available, the tracking accuracy is satisfactory. Unsatisfactory performance that could be expected was solved by selecting the appropriate outputs. In fact, the output redefinition strategy selected for the design resulted in satisfactory results. The actuator redundancy of phase 2 proved to be difficult to handle and can be a field of future research. The activation of thrusters and aerodynamic surfaces simultaneously can generate contradictory effects on the vehicle. The tracking performance, as well as the actuator response, was degraded during this phase.

The path controller and reference trajectory design also has an important influence on the flight controller performance. The ability of the vehicle to track commanded bank maneuvers during early phases of flight is very poor. The thruster efficiency is low due to high dynamic pressure, and the efficiency of the aerodynamic surfaces is also low because dynamic pressure is still low. Therefore, bank maneuvers such as the first one of both trajectories presented are difficult to track and if possible should be avoided.

Although the several phases of flight have different characteristics, the gains of the PID elements in the inner and outer loop were tuned only once for the entire flight. The performance of the controller was still, in general, satisfactory. If the control law had been based on gain scheduling linear control, several gain tunings would be necessary to ensure a satisfactory performance. One of the main characteristics of the LBRV reentry is that the vehicle is not programmed in advance for a specific reentry trajectory. In fact, the reentry trajectory will only be known in the moment of the beginning of reentry. Thus, the advantage of using NDI to design the flight controller is the ability to satisfactorily track different trajectories with the same control system.

Acknowledgment

The first author expresses his thanks to the PRAXIS XXI program of the Portuguese Ministry of Science and Technology for supporting his stay at the Faculty of Aerospace Engineering, in Delft, The Netherlands.

References

[1]Bharadwaj, S., Rao, A., and Mease, K., "Entry Trajectory Tracking Law via Feedback Linearization," *Journal of Guidance, Control, and Dynamics*, Vol. 21, No. 5, 1998, pp. 726–732.

[2]Roenneke, A., "Adaptive Onboard Guidance for Entry Vehicles," *Proceedings of the AIAA Guidance, Navigation, and Control Conference*, AIAA, Reston, VA, 2001, pp. 1–8.

[3]Wu, S., Engelen, C., Babuska, R., Chu, Q., and Mulder, J., "Inteligent Flight Controller Design with Fuzzy Logic for an Atmospheric Reentry Vehicle," *Proceedings of the 38th AIAA Aerospace Science Meeting and Exhibit*, AIAA, Reston, VA, 2000, pp. 1–10.

[4]Shin, J., Balas, G., and Packard, A., "Worst-Case Analyisi of the X-38 Crew Return Vehicle Flight Control System," *Journal of Guidance, Control, and Dynamics*, Vol. 24, No. 2, 2001, pp. 261–269.

[5]Slotine, J., and Li, W., *Applied Nonlinear Control*, 1st ed., Prentice–Hall, Upper Saddle River, NJ, 1987, pp. 207–271.

[6]Isidori, A., *Nonlinear Control Systems*, Springer-Verlag, Berlin, 1989, pp. 178–254.

[7]Bugajski, D., Enns, D., and Elgersma, M., "A Dynamic Inversion Based Control Law with Application to the HARV," *Proceeding of the AIAA Guidance, Navigation, and Control Conference*, AIAA, Washington, DC, 1990, pp. 826–839.

[8]Balas, G., Garrard, W., and Reiner, J., "Robust Dynamic Inversion Control Laws for Aircraft Control," *Proceedings of the AIAA Guidance, Navigation, and Control Conference*, AIAA, Washington, DC, 1992, pp. 192–205.

[9]Adams, R., Buffington, J., and Banda, S., "Design of a Nonlinear Control Laws for High Angle-of-Attack Flight," *Journal of Guidance, Control, and Dynamics*, Vol. 17, No. 4, 1994, pp. 737–746.

[10]Roenneke, A., and Well, K., "Nonlinear Flight Control for a High-Lift Reentry Vehicle," *Proceedings of the AIAA Guidance, Navigation, and Control Conference*, AIAA, Washington, DC, 1995, pp. 1798–1805.

[11]Ito, D., Georgie, J., Valasek, J., and Ward, D., "Reentry Vehicle Flight Control Design Guidelines: Dynamic Inversion," NASA Rept. NAG9-1085, May 2001.

[12]Harpold, J., and Graves, C., "Shuttle Entry Guidance," *Journal of Astronautical Sciences*, Vol. 26, No. 3, 1979, pp. 239–268.

[13]Bennett, D., "Space Shuttle Entry Flight Control Overview," *Journal of Astronautical Sciences*, Vol. 31, No. 4, 1983, pp. 569–578.

[14]"X-38 Technology," NASA Web page, URL: http://www.dfrc.nasa.gov/Projects/X38 [cited 10 Aug. 2000].

[15]"X-38 Back to the Future for a Spacecraft Design," NASA Facts, URL: http://www.dfrc.nasa.gov/PAO/PAIS/HTML/FS-038-DFRC.html [cited 10 Aug. 2000].

[16]Costa, R. R., Silva, J. A., Wu, S. F., Chu, Q. P., and Mulder, J. A., "Atmospheric Reentry Modeling and Simulation," *Journal of Spacecraft and Rockets*, Vol. 39, No. 4, 2002, pp. 636–639.

[17]Bugajski, D., and Enns, D., "Nonlinear Control Law with Application to High Angle-of-Attack Flight," *Journal of Guidance, Control, and Dynamics*, Vol. 15, No. 3, 1992, pp. 761–767.

[18]Wallner, E., Burkhardt, J., Zimmermann, F., Schottle, U. M., and Well, K. H., "A Guidance and Control Concept for the X-38 Re-entry Vehicle," *Proceedings of International Astronautical Federation Congress*, International Aeronautical Federation, Amsterdam, 1999, pp. 1–8.

[19]Wacker, R., Enns, D., Bugajski, D., Munday, S., and Merkle, S., "X-38 Application of Dynamic Inversion Flight Control," *Advances in the Astronautical Sciences—Guidance and Control 2001*, Vol. 107, No. 2, 1983, pp. 233–250.

[20]Wiegand, A., Markl, A., Well, K. H., Mehlem, K., Ortega, G., and Steinkopf, M., "ALTOS—ESA's Trajectory Optimization Tool Applied to Re-Entry Vehicle Trajectory Design," *Proceedings of International Astronautical Federation Congress*, International Astronautical Federation, Amsterdam, 1999, pp. 1–6.

[21]Regan, F. J., and Anandakrishnan, S. M., *Dynamics of Atmospheric Re-Entry*, 1st ed., AIAA Education Series, AIAA, Washington, DC, 1993, pp. 77–144.

[22]Mooij, E., *The Motion of a Vehicle in a Planetary Atmosphere*, Delft Univ. Press, Delft, The Netherlands, 1997, pp. 37–92.

[23]Wu, S. F., and Chu, Q. P., "The Atmospheric Re-Entry Spacecraft CRV/X-38—Aerodynamic Modeling and Analysis," European Space Research and Technology Center, TR 13711/99/NL/MV, Noordwijk, The Netherlands, Dec. 1999, pp. 15–150.

[24]Silva, J., "Simulation and Design of a Robust Nonliear Dynamic Inversion Controller for the Atmospheric Re-Entry Demonstrator," Faculty of Aerospace Engineering, TR LR017153, Delft Univ. of Technology, Delft, The Netherlands, Sept. 2001, pp. 133–228.

[25]Hormigo, T., "Modeling, Re-entry optimization and Guidance for a Precision Landing Mars Mission," Faculty of Aerospace Engineering, TR LR365102, Delft Univ. of Technology, Delft, The Netherlands, May 2001, pp. 58–96.

[26]Guidi, A., Chu, Q., Mulder, J., and Buursink, J., "Implementation of a Re-Entry CFD Toolbox in the GESARED Flight Simulator," *Proceedings of the AIAA Atmospheric Flight Mechanics Conference*, AIAA, Reston, VA, 2001, pp. 1–8.

[27]Wie, B., *Space Vehicles Dynamics and Control*, 1st ed., Vol. 402-404, AIAA Education Series, Washington, DC, 1987, pp. 207–271.

[28]Cook, M. V., *Flight Dynamics Principles*, Arnold, London, 1997, pp. 21–157.

C. A. Kluever
Associate Editor