Robust Reentry Guidance of a Reusable Launch Vehicle using Model Predictive Static Programming

Omkar Halbe[1], Ramsingh G. Raja[2] and Radhakant Padhi[3]
Indian Institute of Science, Bangalore, INDIA

A robust suboptimal reentry guidance scheme is presented for a reusable launch vehicle using the recently-developed computationally efficient model predictive static programming. The formulation uses the nonlinear vehicle dynamics with spherical and rotating earth, hard constraints for desired terminal conditions and an innovative cost function having several components with associated weighting factors that can account for path and control constraints in a soft constraint manner, thereby leading to smooth solutions of the guidance parameters. The proposed guidance essentially shapes the trajectory of the vehicle by computing the necessary angle of attack and bank angle that the vehicle should execute. The path constraints are the structural load constraint, thermal load constraint, bounds on the angle of attack and bounds on the bank angle. In addition, the terminal constraints are the three-dimensional position and velocity vector components at the end of the reentry. Where as the angle of attack command is generated directly, the bank angle command is generated by first generating the required heading angle history and then using it in a dynamic inversion loop considering the heading angle dynamics. Such a two-loop synthesis of bank angle leads to better management of the vehicle trajectory and avoids mathematical complexity as well. Moreover, all bank angle maneuvers have been confined to the middle of the trajectory and the vehicle ends the reentry segment with near zero bank angle, which is quite desirable. It has also been demonstrated that the proposed guidance has sufficient robustness for state perturbations as well as parametric uncertainties in the model.

---

[1] Former Project Associate, Department of Aerospace Engineering
[2] Project Associate, Department of Aerospace Engineering
[3] Associate Professor (Associate Fellow, AIAA), Dept. of Aerospace Engineering. *padhi@aero.iisc.ernet.in*.
An earlier version of this paper was presented in 2010 AIAA GNC conference (Paper No. AIAA-2010-8311).

NOMENCLATURE

α	Angle of attack, *rad*
σ	Bank angle, *rad*
h	Height from earth's surface, *m*
r	Radial distance from earth's center, *m*
V	Velocity, m/sec
V_{CO}	Circular orbital velocity, m/sec
γ	Flight path angle, *rad*
ϕ	Latitude angle, *rad*
θ	Longitude angle, *rad*
ψ	Heading angle, *rad*
t	Time, *sec*
e	Energy height, *m*
Ω_e	Earth rotational velocity, rad/sec
M	Mach number
ρ	Atmospheric density, kg/m^3
ρ_{SL}	Atmospheric density at sea level, kg/m^3
m	Mass of vehicle, *kg*
R_N	Nose radius of vehicle, *m*
q	Dynamic pressure, N/m^2
L	Lift, *N*
D	Drag, *N*
g	Gravity at earth's surface, m/sec^2
S_{ref}	Reference surface area, *m*
Re	Earth radius, *m*
N_z	Normal load, *g*
C_L, C_D	Lift and drag force coefficients respectively

Acronyms

RLV	Reusable Launch Vehicle
TAEM	Terminal Area Energy Management
LQR	Linear Quadratic Regulator
PID	Proportional-Integral-Derivative
VSS	Variable Structure System
L/D	Lift Drag Ratio
MPC	Model Predictive Control
ADP	Approximate Dynamic Programming
MPSP	Model Predictive Static Programming
DOF	Degree of Freedom
DI	Dynamic Inversion

I. Introduction

There is a renewed interest across various space agencies around the world to design economically viable Reusable Launch Vehicles (RLVs) for future space missions to bring down the cost of accessing the space. Where as the philosophy of repeated launch by the same vehicle sounds interesting and attracting, a major challenge posed in such missions is that of atmospheric reentry. During this phase, severe constraints structural load limit and like heat flux constraint come into action, which must be explicitly accounted for and managed well for the vehicle safety. During this phase the vehicle also needs to fly within upper and lower bounds of angle of attack to maintain its controllability as well as to manage dissipation of its energy. Moreover, even if the vehicle is unmanned (which is true for most of the next generation RLVs), it is desirable to fly within the specified bank angle bounds in order not to excite too much the aerodynamic coupling between longitudinal and lateral dynamics as well as to avoid sharp turnings. In addition to the path constraints, at the end of the reentry segment the vehicle also needs to meet the desired terminal conditions in terms of desired position and velocity vector components (which include latitude, longitude, altitude, velocity magnitude, flight path angle and heading angle) so that the vehicle can be successfully recovered. Moreover, the reentry

vehicles are usually unpowered, and the vehicle gets only one chance to land safely. Hence the guidance and control logic during the reentry must be designed with extreme care so as to have sufficient robustness for both state perturbations as well as modeling inaccuracies.

In many missions, the ultimate aim is to recover the vehicle in a runway (e.g. like the space shuttle mission) through an appropriate 'terminal area energy management' (TAEM) guidance [1, 2], followed by appropriate landing logic through typical glideslope and flare. If implemented successfully, this ensures minimal refurbishing requirement for the vehicle and the turn around time can be short, which can play a critical role in bringing down the overall cost. However, before initiating the TAEM and automatic landing logic, it is very crucial to bring the vehicle safely through the reentry corridor to a specified basket in state space, which is the main aim of this paper. Hence, the problem of TAEM and automatic landing are not explored here. Moreover, note that the TAEM and automatic landing is not a must in all missions. In simpler missions, the vehicle is rather required to reach close to a specified final coordinates with sufficiently reduced velocity from where it can glide to the sea, possibly with the help of a parachute [3]. Such missions are typically common in initial flight trials to demonstrate the soundness of vehicle design and reentry technology and/or if the land mass for constructing a runway is not available at a feasible location after the mission of the launch vehicle is over.

A major breakthrough of the guidance design of reentry vehicles can be attributed to the space shuttle entry guidance [1]. In this design philosophy, first a reference drag profile is computed in an off-line trajectory optimization algorithm. This reference trajectory (which is critical for controlled energy dissipation) is then tracked during the actual flight by incorporating a gain-scheduled PID control design logic. Later it has been proposed to eliminate the requirement of tedious gain scheduling by substituting it with a dynamic inversion design [4]. Subsequently, a number of ideas appeared following this basic philosophy of tracking a reference drag profile using various tracking control design methods. For example, philosophies such as the gain scheduled LQR design [5, 6] and receding horizon control [7] have been proposed in the literature.

Irrespective of the control design method used, a major drawback of a drag tracking approach, however, is the over-dependence on the reference profile. For any perturbed flight condition, such a logic forces the flight vehicle to come back to the reference profile and then keep tracking it. Any reentry guidance design that typically generates an optimal trajectory offline and then relies on the philosophy of 'neighboring

optimal control' [8] by forcing to merge the actual trajectory with it is typically not good because of several reasons. First, these techniques lack the operational flexibility as the onboard trajectory redesign is restricted to the vicinity of reference profiles computed offline. To bring in operational flexibility (such as choosing a different runway in case of bad weather), many such reference profiles need to be pre-computed and stored onboard. Where as unlike earlier days storage space is no more a restriction, one should be careful in selecting an appropriate reference profile and/or switching between them, which becomes quite a tedious task. Also, it becomes a vehicle specific logic and hence looses generality. Moreover, in such an approach the overall guidance that acts on the vehicle in a typical mission is not truly optimal (at the best it can only be suboptimal). To avoid such drawbacks, the ideal approach would be to carry out the trajectory optimization process online. Depending on the actual flight condition, ideally a new reference profile itself should be obtained for making the guidance truly optimal. However, this is in general impossible to carry out online since dynamic optimization problems are computationally intensive, which are traditionally impossible to solve in real time using classical techniques and their variants (e.g. using gradient method [9]). Hence, development of efficient algorithms is a must to solve trajectory optimization problems online.

There have been some attempts in the recent literature to generate feasible reentry trajectories online. Roenneke [10] has proposed an adaptive entry guidance algorithm based on autonomous onboard trajectory planning and nonlinear trajectory tracking. This approach essentially eliminates the need for a reference drag profile and the commanded trajectory to the target is computed by maximizing the vehicle's range capability. Cavallo and Ferrara [5] have used a combination of a linear quadratic regulator (LQR) and a variable structure system (VSS) approach. The LQR design minimizes the deviations from the expected trajectory and the VSS approach essentially strives to point the velocity vector towards the target thereby minimizing the heading angle error. Lu [2] presented an algorithm for onboard orbital entry trajectory generation using a quasi-equilibrium glide condition to reduce the dimensionality of the problem for meeting inequality trajectory constraints. The longitudinal and lateral profiles were established through a one-parameter search problem and bank angle reversals at appropriate points in the trajectory were also found. Shen [11] has provided a dynamic onboard logic for bank angle maneuvers based on the crossrange profile. Joshi et al. [12] have employed a predictor corrector approach for the Reusable Launch Vehicle guidance problem where terminal errors are predicted numerically and then control variables are updated to correct the errors.

Nonlinear optimal control theory [8] is the right tool to address a number of challenging trajectory optimization problems in general, including the reentry guidance. This is because it can naturally handle the path and terminal constraints, while simultaneously optimizing a meaningful performance index. However, many of the reentry guidance techniques mentioned above are not based on nonlinear optimal control theory. This is because the theory, if viewed from a calculus of variations approach, essentially leads to a two-point boundary value problem and lands up in the issue of 'curse of complexity', which in turn requires iterative solutions leading to the concern of computing time and convergence risk. On the other hand, if viewed from a dynamic programming formulation, it again leads to the computational bottleneck, known as 'curse of dimensionality' [8]. Recently some fast computational algorithms have been proposed in the literature, but many of them are not fast enough for aerospace applications, where the available computational time window is very small (a few milliseconds).

However, combining the philosophies of model predictive control [13] and approximate dynamic programming [14], an innovative computationally efficient technique has been proposed recently to solve a class of finite horizon optimal control problems with terminal constraints. In addition to its similarity with MPC and ADP designs, since this new technique is essentially formulated in the framework of static (parametric) optimization, it has been named as "model predictive static programming" (MPSP) [15]. Innovations of the MPSP technique can be attributed to the following facts: (i) in contrast to typical two-point boundary value problems in optimal control formulations, it rather demands only a static costate vector (of the same dimension as the output vector) for the control history update, (ii) the costate vector (and hence the control history update) has a symbolic solution and (iii) the sensitivity matrices that are necessary for obtaining this symbolic solution can be computed recursively. Ideas like 'iteration unfolding' [16] can also be incorporated to enhance the computational efficiency further (at the cost of minor compromise on the optimality of the solution). The technique essentially brings in the philosophy of trajectory optimization into the framework of guidance design, which in turn results in very effective guidance logic. Recently, the MPSP technique has been applied to various guidance problems in aerospace engineering with promising results [15, 17, 18].

An alternate promising technique that leads to computationally efficient solutions of optimal control problems is the 'pseudospectral method' [19, 20], which has also been used for reentry guidance problems. However, it has many tuning issues, including careful selection of basis functions and collocation points

(which need to be non-uniform). For successful implementation, it also demands that the user understands complex mathematical concepts like 'co-vector mapping principle'. On the other hand, the MPSP technique presented in this paper is rather simpler and straightforward. For example, there is no need of getting restricted to a non-uniform grid (which brings in additional difficulties for mechanization) and it doe not demand complex mathematical concepts. Moreover, unlike Pseudospectral methods, MPSP is a method in itself and does not rely on any numerical techniques for parametric optimization. As a compromise, however, the technique is still under development and at this moment is not as matured as the Pseudospectral methods. However, as mentioned above, it is capable enough to address many complex real-life problems [15, 17, 18].

Using the MPSP technique, a suboptimal reentry guidance technique is presented in this paper for a Reusable Launch Vehicle (RLV). It is worth mentioning that in an earlier related work [17], a reentry guidance problem only in pitch plane was successfully solved by the third author of this paper along with his other coworkers using the MPSP technique. That problem, however, assumed no bank angle maneuvers, i.e. bank angle was assumed to be maintained at zero throughout the flight corridor. This was possible as there was no restriction on the final position of the vehicle after reentry. However, for operational flexibility (i.e. with better management of vehicle position after reentry) as well as to allow slow dissipation of energy by giving more flight time, having a strategy for bank angle manipulation (with bank angle reversal) is always preferable. Note that this brings in an order of magnitude complexity into the problem formulation, which is successfully addressed in this paper. Another noticeable difference is the 'specific energy based formulation' as opposed to a 'time based formulation' in the earlier work, [17] which makes the new guidance logic operate in true feedback sense.

This guidance strategy presented here essentially shapes the trajectory of the RLV by predicting the necessary angle of attack and bank angle that the vehicle should simultaneously execute. The formulation uses the nonlinear vehicle dynamics with spherical and rotating earth, hard constraints for desired terminal conditions and an innovative cost function having several components with associated weighting factors that can account for path and control constraints in a soft constraint manner, thereby leading to smooth solutions of the guidance parameters. However, the angle of attack solution comes out of the MPSP guidance directly. The bank angle command generation is done in two steps. First, the reference heading angle profile is chosen from the converged solution of the MPSP guidance. Next, using the desired heading angle

profile as a command tracking problem, the corresponding bank angle profile is obtained through a dynamic inversion [21] formulation. Such a two-loop synthesis of bank angle leads to better management of vehicle trajectory and avoids mathematical complexity as well. Note that the initial guess history for heading angle in MPSP design is calculated using spherical trigonometry [22] considering earth as a sphere. By choosing appropriate weights for the heading angle profile update, a good bank reversal strategy is also obtained. Overall, the bank angle profiles obtained are smooth. Moreover, all bank angle maneuvers have been confined to the middle of the trajectory and the bank angle is ensured to be near zero at the end of the reentry, which is very much desirable. The normal load path constraint is minimized by manipulating the angle of attack profile in those parts of the trajectory where the normal load is close to its boundary. This is achieved by selecting an appropriate weighting factor (an exponential function of the load profile) in the cost function.

In this paper, the proposed technique has been validated using the nonlinear point mass dynamics of a realistic reusable launch vehicle with spherical and rotating earth. In addition to nominal case results, it has also been demonstrated that the proposed guidance has sufficient robustness both for state perturbations (which may arise from noise input) as well as parametric uncertainties in the model (which can arise from inaccurate aerodynamic and inertia models). It has been found that the proposed guidance algorithm could successfully generate feasible trajectories satisfying all constraints. Moreover, the algorithm has been found to converge very fast with very limited number of iterations. Owing to its computational efficiency and good robustness, the authors sincerely believe that the MPSP reentry guidance technique presented in this paper is quite promising in general. Moreover, with the advancement of the computing technology with fast processors, it holds promise for implementation in onboard processors in the near future.

II. A Brief Summary of MPSP Design

Even though the MPSP technique has been presented recently in other publications [15, 17, 18], a brief summary of the salient steps are included in this section for completeness of the paper. One may notice slight variations of the following algebra in different literature, which may arise due to the selection of different cost function depending on the necessity of the problem.

To begin with, a discrete (or discretized) form of the system dynamics is considered in the MPSP algorithm. Let $X^i \in \Re^n$, $U^i \in \Re^m$, $Y^i \in \Re^p$ denote the state, control and output variables respectively in the i^{th}

iteration, where $i = 1,2,\ldots$ represent the iteration index. Let $X_k^i,\ k = 1,2,\ldots,N$ and $U_k^i,\ k = 1,2,\ldots,N-1$ be the state and control solution respectively at the i^{th} iteration, where X_1^i represents the given initial condition (which remains same for all i) and $U_k^1,\ k = 1,2,\ldots,N-1$ represents the 'guess control history'. The discretized state and output equations in the i^{th} iteration can be represented as

$$X_{k+1}^i = F_k\left(X_k^i,\, U_k^i\right) \tag{1}$$

$$Y_k^i = H\left(X_k^i\right) \tag{2}$$

Like a typical optimal control solution approach, the idea is to predict the system behavior with the most update control history (starting from an initial guess solution) and then to quickly update it with the error information available at the final time. Note that like other algorithms, a fairly good guess history is also recommended to begin the iteration process so that the algorithm converges and converges quickly. However, the method to obtain a good guess control history is obviously problem specific and for the reentry problem discussed in this paper, it has been addressed in detail in Section III E. The primary objective is to obtain a suitable control history $U_k^i,\ k = 1,2,\ldots,N-1$ at (with as less number of iterations as possible), so that the output at the final time step Y_N^i goes to a desired value Y_N^d, i.e. $Y_N^i \to Y_N^d$ for some i.

The error in the final output at iteration step i is defined as $\Delta Y_N^i \triangleq Y_N^i - Y_N^d$. From Eq.(2), taking Taylor series expansion and introducing small error approximation (thereby neglecting higher order terms) yields

$$\triangle Y_N^i \approx dY_N^i = \left[\frac{\partial Y_N}{\partial X_N}\right]_{\left(X_N^i\right)} dX_N^i \tag{3}$$

However from Eq.(1), again using the Taylor series expansion and introducing small error approximation, the error in the state at time step $(k+1)$ at iteration step i can be expressed as

$$dX_{k+1}^i = \left[\frac{\partial F_k}{\partial X_k}\right]_{\left(X_k^i, U_k^i\right)} dX_k^i + \left[\frac{\partial F_k}{\partial U_k}\right]_{\left(X_k^i, U_k^i\right)} dU_k^i \tag{4}$$

where dX_k^i is the error in the state and dU_k^i is the error in the control solution at time step k and iteration i.

Expanding dX_N^i as in Eq.(4) (for $k = N-1$) and substituting it in Eq.(3) leads to

$$dY_N^i = \left[\frac{\partial Y_N}{\partial X_N}\right]_{\left(X_N^i\right)} \left(\left[\frac{\partial F_{N-1}}{\partial X_{N-1}}\right]_{\left(X_{N-1}^i, U_{N-1}^i\right)} dX_{N-1}^i + \left[\frac{\partial F_{N-1}}{\partial U_{N-1}}\right]_{\left(X_{N-1}^i, U_{N-1}^i\right)} dU_{N-1}^i \right) \tag{5}$$

Similarly the error in state at time step $(N-1)$, dX_{N-1}^i can be expanded in terms of the errors in state dX_{N-2}^i and control dU_{N-2}^i at time step $(N-2)$, and so on. Continuing the process until $k = 1$, one obtains

$$dY_N^i \;=\; A\, dX_1^i + B_1 dU_1^i + B_2 dU_2^i + \ldots + B_{N-1} dU_{N-1}^i \tag{6}$$

$$\begin{aligned} \text{where } A \;&\triangleq\; \left[\frac{\partial Y_N}{\partial X_N}\right]_{\left(X_N^i\right)} \left[\frac{\partial F_{N-1}}{\partial X_{N-1}}\right]_{\left(X_{N-1}^i, U_{N-1}^i\right)} \cdots \left[\frac{\partial F_1}{\partial X_1}\right]_{\left(X_1^i, U_1^i\right)} \\ B_k \;&\triangleq\; \left[\frac{\partial Y_N}{\partial X_N}\right]_{\left(X_N^i\right)} \left[\frac{\partial F_{N-1}}{\partial X_{N-1}}\right]_{\left(X_{N-1}^i, U_{N-1}^i\right)} \cdots \left[\frac{\partial F_{k+1}}{\partial X_{k+1}}\right]_{\left(X_{k+1}^i, U_{k+1}^i\right)} \left[\frac{\partial F_k}{\partial U_k}\right]_{\left(X_k^i, U_k^i\right)} \end{aligned} \tag{7}$$

where $k = 1, \ldots, (N-1)$. Since the initial condition is specified, there is no error in the first term, which means $dX_1^i = 0$. With this Eq.(6) reduces to

$$dY_N^i \;=\; B_1 dU_1^i + B_2 dU_2^i + \cdots + B_{N-1} dU_{N-1}^i \tag{8}$$

It can be pointed out here that the sensitivity matrices B_k, $k = 1, \ldots, (N-1)$ in Eq.(7) can be computed recursively (see [15] for details), which saves a substantial amount of computational time and makes the technique very efficient.

Note that Eq. (8) has $(N-1)m$ unknowns and p equations. Since usually $p \ll (N-1)m$, Eq.(8) is an under-constrained system of equations. This paves the way for meeting additional objectives. This situation is exploited by formulating cost functions that can be minimized (or maximized) in addition to satisfying the constraint in Eq. (8). An assumption is made here that the guess control history is fairly good and close to optimal. In practice, a problem-dependent wise selection of control guess history should be done (note that a control guess history specific to the atmospheric reentry problem discussed in this paper has been discussed with sufficient detail in Subsection III E). With this assumption, the updated control history should remain

close to the previous history. Hence, the performance index to be minimized can be chosen as

$$J = \frac{1}{2}\sum_{k=1}^{N-1}(dU_k^i)^T R_k (dU_k^i) \tag{9}$$

where dU_k^i is the corresponding 'error' in the control at iteration i that needs to be subtracted from the previous control value to obtain the new updated control value. $R_k > 0$ is a time-varying weighting matrix in general, which needs to be chosen judiciously by the control designer. Note that the performance index in Eq.(9) is used mainly to demonstrate the MPSP algorithm. The actual cost function with the associated algebra for reentry guidance problem is included in Section III.

With the above discussion, it is obvious that the cost function in Eq.(9) needs to be minimized subject to the constraint in Eq.(8). Note that Eq.(8) and Eq.(9) form an appropriate constrained static optimization problem, which can then be solved in closed form using static optimization theory [8]. Using the very basic conditions of optimality followed by necessary algebraic manipulations, it then leads to (see [15] for details)

$$dU_k^i = -R_k^{-1} B_k^T A_\lambda^{-1}\, dY_N^i \tag{10}$$

$$\text{where} \quad A_\lambda \triangleq \left[-\sum_{k=1}^{N-1} B_k R_k^{-1} B_k^T\right] \tag{11}$$

The updated control at time step $k = 1, 2, \ldots, (N-1)$ is given by

$$U_k^{i+1} \;=\; U_k^i - dU_k^i \tag{12}$$

where U_k^i, $k = 1, \ldots, (N-1)$ is the previous control history solution. The new updated control solution U_k^{i+1}, $k = 1, \ldots, (N-1)$ is then used to propagate the system dynamics in Eq.(1) and output dynamics in Eq.(2). The iterations are carried out until the objectives are met.

Innovations of the MPSP technique can be attributed to the following facts: (i) in contrast to typical two-point boundary value problems in optimal control formulations, it rather demands only a static costate vector (that too of the same dimension as the output vector) for the entire control history update (ii) the

costate vector (and hence the control history update) has a symbolic solution and (iii) the sensitivity matrices that are necessary for this solution can be computed recursively. Because of the above facts, this technique is computationally very efficient, and hence holds promise for online implementation. Ideas like 'output convergence' to terminate the algorithm and 'iteration unfolding' [16] (where the control history is updated only a finite number of times in a particular time step) can also be incorporated to enhance the computational efficiency further at the cost of a minor compromise about optimality of the solution. As it turns out and have been demonstrated in a variety of challenging problems [15, 17, 18], the convergence is usually very rapid and one needs only a few iterations before the control history converges to the optimal control history.

It needs to be emphasized here that even though the basic idea of the MPSP technique is outlined above for completeness, the actual form of the cost function in a given problem need not conform to Eq.(9). Depending on the application problem, one can choose an appropriate cost function and this may lead to necessary modifications in the algebra and the final control expression. The reader is referred to Section III H) for a detail expression of the selected cost function for the reentry problem discussed in this paper and subsequent sections for the associated algebra.

III. RLV Reentry Guidance Design Using MPSP

The Reusable Launch Vehicle (RLV) considered here is a technology demonstrator to demonstrate various critical technologies, including new guidance and control algorithms, from flight experiments [3]. It does not have sufficient energy to launch satellites in their orbit. Instead, it is launched for a suborbital mission, where it is supposed to lift off and travel outside the atmosphere and, after the separation of the booster, is supposed to reenter through the atmosphere safely so that it can be recovered. During the reentry, the point where the dynamic pressure builds up to 1.5 kPa has been assumed to be the initial point of reentry. Owing to its lesser energy because of the suborbital flight, this dynamic pressure is typically experienced at an altitude of about 51 km with the reentry velocity of approximately 1800 m/s (with flight path angle of approximately -15^0). However, the vehicle can actually start its reentry from a ball of initial conditions around these values in the state space (including position coordinates and heading angle) and guidance scheme should be capable of guiding the vehicle from any point within this ball. In other words, irrespective of its initial condition from within this ball, all along the reentry segment the vehicle has to dissipate its associated potential and

kinetic energy in a careful manner without violating the path constraints, namely normal load constraint and the heat flux constraint. At the end of reentry, it has to meet a desired set of final conditions as well. Because of these constraints the problem is quite challenging. The way it has been successfully solved is discussed in this section with all necessary mathematical details.

A. Mathematical Model with Spherical and Rotating Earth

It is assumed that the reentry vehicle is unpowered (which is typically true), mass variation is negligible and the atmosphere is stationary. The only forces acting on the vehicle are gravity and aerodynamic lift and drag. Under such circumstance, assuming the vehicle to be a point mass, the equations of motion of the RLV in three-dimensional space over a spherical, rotating earth are described by the following kinematic and dynamic equations of motion [23]:

$$\dot{h} = V\,sin\gamma \tag{13}$$

$$\dot{V} = -\frac{D}{m} - g\,sin\gamma + \Omega_e^2\,r\,cos\phi\,(sin\gamma\,cos\phi - cos\gamma\,sin\phi\,sin\psi) \tag{14}$$

$$\dot{\gamma} = \frac{1}{m\,V}\,L\,cos\sigma - \frac{g}{V}\,cos\gamma + \frac{V}{r}\,cos\gamma + 2\,\Omega_e\,cos\phi\,cos\psi + \frac{\Omega_e^2\,r\,cos\phi}{V}\,(cos\gamma\,cos\phi + sin\gamma\,sin\phi\,sin\psi) \tag{15}$$

$$\dot{\phi} = \frac{V\,cos\gamma\,sin\psi}{r} \tag{16}$$

$$\dot{\theta} = \frac{V\,cos\gamma\,cos\psi}{r\,cos\phi} \tag{17}$$

$$\dot{\psi} = \frac{1}{m\,V}\,\frac{L\,sin\sigma}{cos\gamma} - \frac{V}{r}\,cos\gamma\,cos\psi\,tan\phi + 2\,\Omega_e\,(tan\gamma\,cos\phi\,sin\psi - sin\phi) - \frac{\Omega_e^2\,r}{V\,cos\gamma}\,sin\phi\,cos\phi\,cos\psi \tag{18}$$

where the aerodynamic lift L and drag D are given by

$$L = \frac{1}{2}\,\rho\,V^2\,C_L\,S_{ref} \tag{19}$$

$$D = \frac{1}{2}\,\rho\,V^2\,C_D\,S_{ref} \tag{20}$$

The aerodynamic coefficients C_L and C_D are functions of the angle of attack α and can be written as

$$\begin{aligned} C_L &= C_{L_0} + C_{L_\alpha}\,\alpha \\ C_D &= C_{D_0} + C_{D_\alpha}\,\alpha \end{aligned} \tag{21}$$

Note that the coefficients $C_{L_0}, C_{D_0}, C_{L_\alpha}$ and C_{D_α} are also functions of angle of attack α and Mach number M. These are available in the form of look-up tables at various values of angle of attack and Mach number, which are generated using computational fluid dynamics by a separate team of experts. Linear interpolation technique is used to compute their values for any given values of the Mach number and angle of attack. In fact, the lift and drag forces also depend on other additional variables such as control surface deflections, which are small and hence typically not considered in the guidance design. However, the incremental lift and drag, as well as the moment components generated due to control surface deflections, are typically part of the detail six degree-of-freedom model, which is beyond the scope of this paper. Indian Standard Atmosphere data [24] is used to compute atmospheric properties such as air density and temperature, which are interpolated at a particular height from a tabulated data. Note that atmosphere density ρ used to compute the dynamic pressure, where as atmosphere temperature T used to compute the speed of sound, which is in turn needed to compute the Mach number.

B. Objectives of Guidance Design

The objectives of attaining the terminal conditions and satisfying the path constraints can be done through appropriate manipulations of the guidance parameters, namely the angle of attack and the bank angle. In addition, the angle of attack profile is constrained to lie within a minimum and a maximum bound that are functions of mach number. Similarly, it should not build up high bank angles to avoid problems related to complicated aerodynamics and difficulty in reversal maneuver. Moreover, the bank angle profile must have reversals preferably be at the middle part of the trajectory, whereas towards the end of the trajectory the bank angle should be as close to zero as possible (at the beginning it should also be close to zero, since the vehicle is expected to enter the atmosphere with near-zero bank angles as well). Note that the path constraints, namely the normal load constraint and control bounds, constitute a small entry corridor for the vehicle in the reentry phase through which the vehicle must travel to meet the desired terminal constraints.

1. Path Constraints

Heat Flux Constraint [3, 25]

$$\frac{11030}{\sqrt{R_n}}\left(\frac{\rho}{\rho_{sl}}\right)^{0.5}\left(\frac{V_r}{V_{co}}\right)^{3.15} \leq Q_{max} \tag{22}$$

Normal Load Constraint [3, 25]

$$\frac{L\,cos\alpha + D\,sin\alpha}{m} < N_{max} \tag{23}$$

where the limiting value of the heat flux Q_{max} is 60 W/cm^2 and normal load factor N_{max} is $3g$. It turns out that the heat flux constraint is not an active constraint for this mission as it is a suborbital mission (since the initial energy build up is not as high as an orbital mission, the discipation requirement is not quite high).

In order to maintain sufficient controllability as well as to avoid the stall condition, the angle of attack α is constrained by the following relationship

$$\alpha_{min}(M) \leq \alpha(M) \leq \alpha_{max}(M) \tag{24}$$

where $\alpha_{min}(M)$ and $\alpha_{max}(M)$ are the minimum and maximum values of the angle of attack, which are functions of Mach number M. Note that the α profile should remain around the middle angle of attack boundary at a given mach number so that there is enough margin on both sides for the control action in case of unexpected disturbances as well as to cater for transient effects.

The bank angle history is also desired to be limited by appropriate bounds as dictated by the following constraint

$$\sigma_{min} \leq \sigma \leq \sigma_{max} \tag{25}$$

Note that even if the vehicle is unmanned (which is true for most of the next generation RLVs), it is desirable to fly within the specified bank angle bounds in order not to excite too much the aerodynamic coupling between longitudinal and lateral dynamics as well as to avoid sharp turnings. In this paper, the bank angle is

constrained to lie within $\pm 45^0$.

2. Terminal Constraints

The vehicle has to meet the terminal constraints defined in terms of the final height h, final velocity V and final flight path angle γ at the end of reentry phase. The vehicle must also reach a desired range of terminal coordinates at the end of the reentry phase.

$$h_f = h^d, \quad V_f = V^d, \quad \gamma_f = \gamma^d, \quad \phi_f = \phi^d, \quad \theta_f = \theta^d \tag{26}$$

where the desired values of final height (h^d), final velocity (V^d), final flight path angle (γ^d), final latitude (ϕ^d) and final longitude (θ^d) are taken from the reference trajectory that was designed earlier using a gradient based classical algorithm. Note that even though the formulation has sufficient generality, the final position constraint is somewhat relaxed here (with larger tolerance bounds) as the task of guiding the vehicle precisely to the exact final location is that of the terminal area energy management phase, which is not considered here.

3. Smoothness of Guidance Parameter Profiles

In addition to this, the converged solution of the guidance parameters α and σ that satisfies the aforementioned objectives should preferably be continuous and smooth. This is ensured of by including additional terms in the cost function that ensure sufficient smoothness and minimum deviation in the updated control during the update process. Moreover, at the end of reentry, the bank angle profile should ideally take the form $\sigma_{t=t_f} = 0$ and $\dot{\sigma}_{t=t_f} = 0$ which ensures that there is sufficient lateral stability at the end of the trajectory. This makes the tasks of the terminal area energy management phase simpler facilitating a smooth flight from the end of reentry to the landing or splashdown.

C. System Dynamics with Specific Energy as Independent Variable

The specific energy (i.e. total mechanical energy of the vehicle per unit weight) is chosen as the independent variable in this guidance design mainly because of two reasons: (i) it eliminates the need to select a final time (which is a difficult task) and hence also avoids the additional task of optimizing its selection and (ii) it makes the design operate on a true feedback sense and guidance command generation depends on the current energy

and not on the current time. Note that such a change of variable is possible as energy is a continuous and monotonic variable with respect to time. This is because, due to absence of thrust, only energy dissipation can take place and hence it can only decrease with time. Theoretically speaking, the kinetic energy of the vehicle has two components, one due to the body's center of mass translational kinetic energy and the second due to the energy of rotation around the center of mass. In this case, the kinetic energy component due to the rotation of the body around its center of mass is assumed to be negligible compared to the component due to translational kinetic energy. Therefore the specific energy can be expressed as

$$e = \frac{V^2}{2g} + h \tag{27}$$

The initial value of specific energy is known from the initial conditions of the vehicle and the final value of specific energy is defined from the desired terminal velocity and terminal height. Taking time derivative of Eq. (27) and substituting for $\dot{V}$ and $\dot{h}$, yields the expression for $\dot{e}$. Subsequently, $\dot{e}$ has been used to obtain the set of dynamic equations with energy as the independent variable, such as $dh/de = \dot{h}/\dot{e}$, $dV/de = \dot{V}/\dot{e}$ etc. Note that for rest of the paper 'energy' means 'specific energy'.

Note that with this approach, the time t becomes a state variable and $t' = dt/de$ is integrated along with other states to have an idea about the final time needed, which differs with each initial condition. The problem therefore transforms into a finite energy problem, eliminating the need to predict a final time apriori. Unlike a finite time based design, it does not unnecessarily put an extra constraint on the problem. One can also notice that the height $\dot{h}$ equation defined in Eq. (13) is eliminated in the energy domain equations and knowing e and V naturally yields the information about h from Eq.(27).

D. Guidance Design Philosophy

One can notice that the longitudinal dynamics is strongly coupled with the angle of attack, the manipulation of which can cause the error in the terminal velocity and terminal flight path angle to go to the desired values. Hence, angle of attack is a natural choice as a control variable. Similarly, a proper heading angle profile will take the vehicle in the direction of the desired terminal position and hence one is tempted to select it as another control variable. However, a heading angle profile cannot be controlled by manipulating the body rate equations in the flight control design which is based on the six-DOF dynamics of the vehicle. To avoid

this difficulty, even though the heading angle profile is generated from the MPSP guidance, an appropriate bank angle history is generated to track this heading angle history (a bank angle profile can be controlled by manipulating the roll rate of the vehicle in the flight control design). Note that care is taken to design the lateral profile in such a way that the lateral profile is smooth and bank angle maneuvers (including bank angle reversals) occur in the middle of the trajectory. As evident from the above discussion, the choice of this design structure has strong physical and mathematical foundation and hence leads to a robust design. The block diagram of this guidance design philosophy is illustrated in Figure 1. Note that in this subsection as well as in rest of the paper the variables with no superscript means the superscript *i* and the variables with superscript *p* means the superscript $(i-1)$ in the context of the MPSP algorithm.

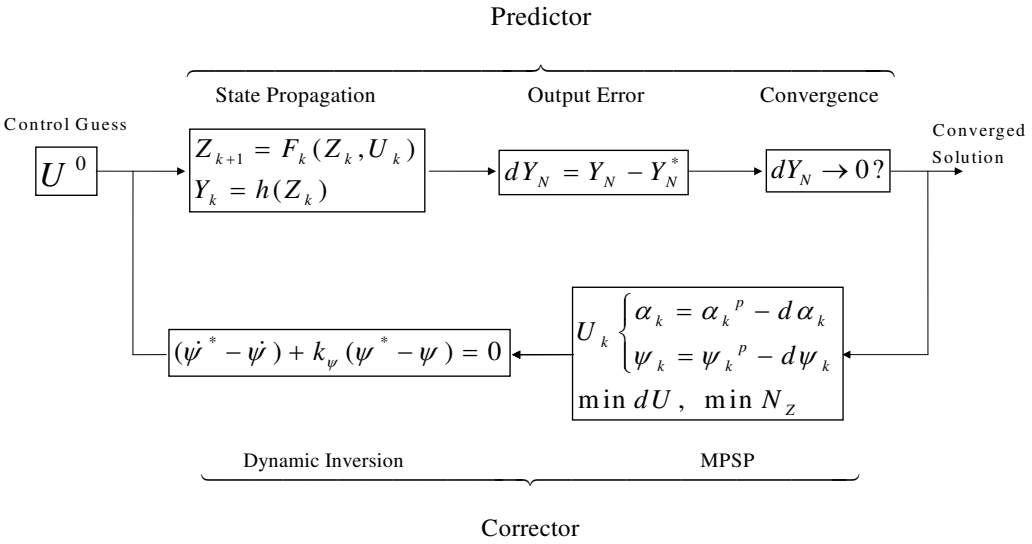

Fig. 1 Guidance Design Philosophy

E. Selection of Guess Histories of Control Variables in MPSP

After selecting the control variables as the angle of attack and bank angle, like any optimal control solution approaches, MPSP also needs guess histories of the control variables to start with (which is then

rapidly updated online). The middle of the angle of attack bounds is chosen as the guess value for angle of attack, where as spherical trigonometry is used to determine the nominal heading angle to move towards the desired location. Details of this selection are discussed next.

1. Angle of Attack Guess

The angle of attack has upper and lower boundary constraints given by Eq. (24). The operating region for the angle of attack is therefore constrained to this region for a given mach number. This fact therefore motivates the use the middle values of the boundary as the α guess history to allow good flexibility during the control update as the probability of hitting one of the bounds becomes lesser. Note that because of the fact that a uniform grid point in time does not necessarily lead to same uniform grid points in energy variable, the cubic spline based interpolation approach is adopted with the available data points to select an appropriate value. In other words, the available middle values of the angle of attack boundary are considered and a cubic spline is fitted with these data points. Selection of spline interpolation makes the α guess history smooth as well.

2. Heading Angle Guess

Heading angle is primarily used for pointing the vehicle in the direction where it ideally should fly to reach the target. This objective is achieved through bank angle maneuvers. Since the heading angle is directly coupled with the latitude-longitude dynamics, the heading angle is considered for manipulation to attain the desired coordinates. Subsequently, a corresponding bank angle profile that leads to the heading angle maneuver is generated by using the heading angle dynamics in a dynamic inversion sense. Spherical trigonometry laws over a spherical earth are used to obtain the nominal heading angle profile [5, 26]. The desired heading angle depends on the current coordinates (ϕ, θ) and the desired final coordinates (ϕ_f, θ_f) coordinates. Figure 2 shows a spherical earth with the start and end of the reentry phase represented by points *A* and *B* respectively. The dashed line between points *A* and *B* represents the curvilinear abscissa or the ground trace of the trajectory followed by the vehicle. Using the law of cosines [22] in the spherical triangle *NACB*, the desired heading angle $\tilde{\psi}$ is obtained as a function of ϕ and θ, details of which are omitted here for brevity. An interested reader can see the details in [5].

Note that the computations involved in the selection of guess histories for both α and σ are very minimal

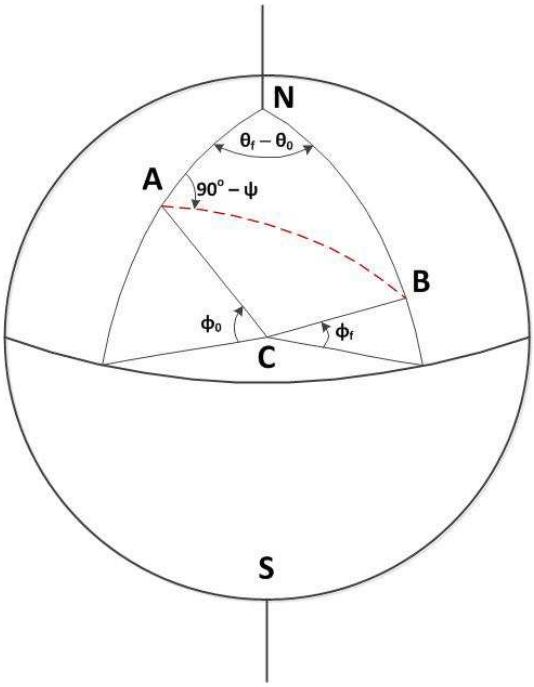

Fig. 2 Heading Angle Guess using Spherical Geometry

to start the solution approach proposed in this paper.

F. System Dynamics in MPSP Formulation

The state vector considered for the MPSP guidance design is $Z \triangleq [V \;\; \gamma \;\; \phi \;\; \theta]^T$. The remaining variables are considered as energy-varying parameters (like time-varying parameters). The dynamics of h is ignored since once velocity V is known as a function of energy e, height h automatically gets constrained as per the relationship $e = h + V^2/(2g)$. Moreover ψ is considered as a 'control variable' in MPSP and its dynamic equation is kept aside for σ computation. Moreover, even though time t is considered as a state and dt/de equation is introduced to compute the evolution of time as a function of energy (and hence all variables can be plotted against the corresponding time), t does not explicitly appear in any other equation and hence the dt/de equation has been ignored.

The selected state variables are normalized next by defining a set of 'normalized states' given by $Z_n \triangleq [V_n \;\; \gamma_n \;\; \phi_n \;\; \theta_n]^T$ where $V_n \triangleq \frac{V}{V^*}$, $\gamma_n \triangleq \frac{\gamma}{\gamma^*}$, $\phi_n \triangleq \frac{\phi}{\phi^*}$, $\theta_n \triangleq \frac{\theta}{\theta^*}$. Here V^*, γ^*, ϕ^*, $\theta^*$are the normalizing values, taken as the corresponding desired terminal values. The control vector is given by $U \triangleq [\alpha \;\; \psi]^T$. Note that the control vector here is not normalised. The normalized system dynamics are represented as $Z_n' \triangleq f(Z_n, U)$, where the superscript $'$ stands for derivatives with respect to energy e. As the

MPSP technique starts with a discretized state equation, using the Euler integration approach the discretized state equations are written as

$$Z_{n_{k+1}} \;=\; F_k\left(Z_{n_k},\, U_k\right) \;=\; Z_{n_k} + \Delta e\; f\left(Z_{n_k},\, U_k\right) \tag{28}$$

G. Output Vector in MPSP Formulation

Since the objective is to drive the error in the final velocity, final flight path angle and final coordinates, the normalized output vector is chosen as

$$Y_n \;\triangleq\; \left[\, V_n \quad \gamma_n \quad \phi_n \quad \theta_n \,\right]^T \tag{29}$$

Note that since by definition $e_f = h_f + V_f^2/(2g)$, if V_f goes to the desired value at e_f, then h_f must go to its desired value as well. Hence there is no need to include it as a component of the output vector.

H. Cost Function Selection

Selection of a suitable cost function to optimize is one of the key features of any optimal control formulation and should be done with utmost care. While the terminal constraints can be directly inserted into the MPSP formulation for optimal guidance design, the path constraints are dealt indirectly through appropriate selection of the cost function, along with associated weighting factors. In addition, continuity and smoothness in guidance parameters should be maintained as much as possible as these will eventually be tracked by the inner autopilot loop (which is not within the scope of this paper). Keeping these objectives in mind, the following cost function was selected to be minimized

$$J \;=\; J_1 \;+\; J_2 \;+\; J_3 \tag{30}$$

where,

$J_1 = \frac{1}{2}\;\sum_{k=1}^{N-1}\; dU_k^T\; R_d\; dU_k$, for $R_d > 0$, minimizes the deviation of the updated control from its previous value. It ensures that the updated control profile remains in the vicinity of the previous control profile. This is done with the assumption that the initial guess histories of angle of attack and

bank angle (which are chosen carefully) are fairly close to the optimal solution.

$J_2 = \frac{1}{2} \sum_{k=2}^{N-1} (U_k - U_{k-1}^p)^T R_S (U_k - U_{k-1}^p)$, for $R_S > 0$, is used for additional smoothness of the updated control profile, where superscript 'p' stands for the previous value. Note that smoothness is achieved primarily when the difference between U_k and U_{k-1} is minimized. However, if one chooses a term $(U_k - U_{k-1})$ in the cost function, it leads to complicated algebra which should be avoided as the aim here is to obtain a closed form control update to retain computational efficiency. From Figure 3, $(U_k - U_{k-1})$ is given by length *CD*. However, *CD* can also be minimized by simultaneously minimizing *AB*, *AD* and *BD*. As the previous control profile is assumed to be smooth (which is true for guess histories as well), *AB* minimization is ensured. Moreover, J_1 essentially minimizes segment *BD*. Hence, minimization of segment *AD* by J_2 (in conjunction with minimization of J_1) is sufficient to ensure smooth control profiles.

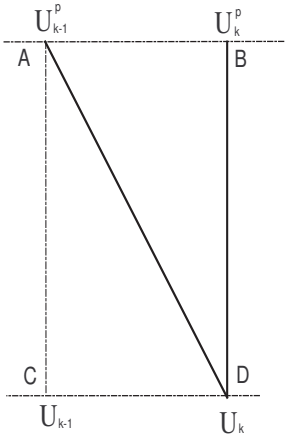

Fig. 3 The geometry of control smoothness.

$J_3 = \frac{1}{2} \sum_{k=1}^{N-1} A_{NL}\, e^{B_{NL}\, N_{Z_k}^p}\, N_{Z_k}$, for $A_{NL}, B_{NL} > 0$, minimizes the normal load along the path of the vehicle. Note that the exponential weight is a function of the previous value of the normal load profile. Although the path constraints specify maintaining the normal load values below N_{max}, simulation studies for this problem showed that the normal load lies in the vicinity of N_{max} only for a small duration of the flight. Thus, the exponential weight activates the minimizing function only in specific areas of the trajectory. It also ensures that the smoothness of the control profile is not compromised at all points on the trajectory and J_1 and J_2 dominate the update process when the normal load is significantly lower than its upper bound.

The cost function from Eq. (30) now becomes

$$J \;=\; \frac{1}{2}\sum_{k=1}^{N-1} dU_k^T\, R_d\, dU_k \;+\; \frac{1}{2}\sum_{k=2}^{N-1} (U_k - U_{k-1}^p)^T\, R_S\, (U_k - U_{k-1}^p) \;+\; \frac{1}{2}\sum_{k=1}^{N-1} A_{NL}\, e^{B\,N_{Z_k}^p}\, N_{Z_k} \tag{31}$$

Using $U_k = U_k^p - dU_k$ and the terminal constraint given by Eq. (6), the augmented cost function is given by

$$\bar{J} \;=\; \frac{1}{2}\sum_{k=1}^{N-1} dU_k^T\, R_d\, dU_k \;+\; \frac{1}{2}\sum_{k=2}^{N-1} (-dU_k + U_k^p - U_{k-1}^p)^T\, R_S\, (-dU_k + U_k^p - U_{k-1}^p) \tag{32}$$

$$+\; \frac{1}{2}\sum_{k=1}^{N-1} A_{NL}\, e^{B\,N_{Z_k}^p}\, N_{Z_k} \;+\; \lambda^T\, (dY_{n_N} \;-\; \sum_{k=1}^{N-1} B_k\, dU_k) \tag{33}$$

From the basic condition of optimality concerning the current problem objectives,

$$\frac{\partial \bar{J}}{\partial dU_k} = \begin{cases} \begin{bmatrix} R_{d_\alpha} & 0 \\ 0 & R_{d_\psi} \end{bmatrix} \begin{bmatrix} d\alpha_k \\ d\psi_k \end{bmatrix} + \begin{bmatrix} R_{S_\alpha} & 0 \\ 0 & R_{S_\psi} \end{bmatrix} \begin{bmatrix} d\alpha_k \\ d\psi_k \end{bmatrix} \\ - \begin{bmatrix} R_{S_\alpha} & 0 \\ 0 & R_{S_\psi} \end{bmatrix} \begin{bmatrix} \alpha_k^p \\ \psi_k^p \end{bmatrix} - \left[B_k^T\right]_{2\times 4} [\lambda]_{4\times 1} \qquad\qquad ....k > 1 \\ + \begin{bmatrix} R_{S_\alpha} & 0 \\ 0 & R_{S_\psi} \end{bmatrix} \begin{bmatrix} \alpha_{k-1}^p \\ \psi_{k-1}^p \end{bmatrix} + A_{NL} e^{BN_{Z_k}^p} \begin{bmatrix} \partial N_{Z_k}/\partial d\alpha_k \\ \partial N_{Z_k}/\partial d\psi_k \end{bmatrix} \end{cases} \tag{34}$$

$$\frac{\partial \bar{J}}{\partial dU_k} = \begin{cases} \begin{bmatrix} R_{d_\alpha} & 0 \\ 0 & R_{d_\psi} \end{bmatrix} \begin{bmatrix} d\alpha_k \\ d\psi_k \end{bmatrix} + A_{NL} e^{BN_{Z_k}^p} \begin{bmatrix} \partial N_{Z_k}/\partial d\alpha_k \\ \partial N_{Z_k}/\partial d\psi_k \end{bmatrix} \quad ....k = 1 \\ - \left[B_k^T\right]_{2\times 4} [\lambda]_{4\times 1} \end{cases}$$

Further algebra is explained using the notations $R_d \triangleq \begin{bmatrix} R_{d_\alpha} & 0 \\ 0 & R_{d_\psi} \end{bmatrix}$, $R_S \triangleq \begin{bmatrix} R_{S_\alpha} & 0 \\ 0 & R_{S_\psi} \end{bmatrix}$ and $R_{NL} \triangleq A_{NL} e^{BN_{Z_k}^p}$.

The expression for the normal load factor $\frac{\partial N_{Z_k}}{\partial d\alpha_k}$ becomes a linear function in $d\alpha_k$. This allows the update process to use angle of attack manipulations to reduce the normal load along the path. The expression for the

discretized normal load (N_z) on expansion yields

$$\begin{aligned} N_{z_k} &\triangleq \frac{L_k}{m} cos\alpha_k + \frac{D_k}{m} \sin\alpha_k \\ &= \frac{q_k S_{ref}}{m} \left[(C_{L_{0_k}} + C_{L_{\alpha_k}} \alpha_k) \, cos\alpha_k + (C_{D_{0_k}} + C_{D_{\alpha_k}} \alpha_k) \, \sin\alpha_k \right] \\ &= \frac{q_k S_{ref}}{m} \left[\left(C_{L_{0_k}} + C_{L_{\alpha_k}} (\alpha_k^p - d\alpha_k) \right) \left(\cos\alpha_k^p \cos(d\alpha_k) + \sin\alpha_k^p \sin(d\alpha_k) \right) \right. \\ &\quad \left. + \left(C_{D_{0_k}} + C_{D_{\alpha_k}} (\alpha_k^p - d\alpha_k) \right) \left(\sin\alpha_k^p \cos(d\alpha_k) - \cos\alpha_k^p \sin(d\alpha_k) \right) \right] \end{aligned} \tag{35}$$

This expression is partially differentiated with respect to $d\alpha_k$. Using $\cos(d\alpha_k) \cong 1$ and $\sin(d\alpha_k) \cong d\alpha_k$ since the value of $d\alpha_k$ is assumed sufficiently small, Eq. (35) then becomes

$$\frac{\partial N_{Z_k}}{\partial d\alpha_k} = T_k \, d\alpha_k + W_k \tag{36}$$

where

$$\begin{aligned} T_k(\alpha_k^p) &= \frac{q_k S_{ref}}{m} [-2C_{L_\alpha} sin(\alpha_k^p) + 2C_{D_\alpha} cos(\alpha_k^p) - C_{L_0} cos(\alpha_k^p) - C_{D_0} sin(\alpha_k^p) \\ &\qquad - C_{L_\alpha} \alpha_k^p cos(\alpha_k^p) - C_{D_\alpha} \alpha_k^p sin(\alpha_k^p)] \end{aligned} \tag{37}$$

$$\begin{aligned} W_k(\alpha_k^p) &= \frac{q_k S_{ref}}{m} [C_{L_0} sin(\alpha_k^p) - C_{D_0} cos(\alpha_k^p) - C_{L_\alpha} cos(\alpha_k^p) - C_{D_\alpha} sin(\alpha_k^p) \\ &\qquad + C_{L_\alpha} \alpha_k^p sin(\alpha_k^p) - C_{D_\alpha} \alpha_k^p cos(\alpha_k^p)] \end{aligned} \tag{38}$$

The normal load is not a function of heading angle, and so $\frac{\partial N_{Z_k}}{\partial d\psi_k} = 0$. Representing $\frac{\partial N_{Z_k}}{\partial d\psi_k} = T_{2_k} d\psi_k + W_{2_k}$, Eq. (34) now becomes

$$\frac{\partial \bar{J}}{\partial dU_k} = \begin{cases} R_d \begin{bmatrix} d\alpha_k \\ d\psi_k \end{bmatrix} + R_S \begin{bmatrix} d\alpha_k \\ d\psi_k \end{bmatrix} - R_S \begin{bmatrix} \alpha_k^p \\ \psi_k^p \end{bmatrix} - \left[B_k^T \right]_{2\times 4} [\lambda]_{4\times 1} \\ + R_S \begin{bmatrix} \alpha_{k-1}^p \\ \psi_{k-1}^p \end{bmatrix} + R_{NL} \left(\begin{bmatrix} T_{1_k}(\alpha_k^p) & 0 \\ 0 & T_{2_k}(\psi_k^p) \end{bmatrix} \begin{bmatrix} d\alpha_k \\ d\psi_k \end{bmatrix} + \begin{bmatrix} W_{1_k}(\alpha_k^p) \\ W_{2_k}(\psi_k^p) \end{bmatrix} \right) \end{cases} \quad \ldots \quad k > 1$$

which yields

$$\frac{\partial \bar{J}}{\partial dU_k} = \begin{cases} R_k\, dU_k - R_S\, U_k^p + R_S\, U_{k-1}^p + R_{NL}\, W_k - B_k^T\, \lambda = 0 & \ldots \quad k > 1 \\ R_k\, dU_k + R_{NL}\, W_k - B_k^T\, \lambda = 0 & \ldots \quad k = 1 \end{cases} \tag{39}$$

$$\text{where } R_k \triangleq \begin{cases} R_d + R_S + R_{NL}\, T_k & \ldots \quad k > 1 \\ R_d + R_{NL}\, T_k & \ldots \quad k = 1 \end{cases}$$

From the condition of optimality, $\frac{\partial \bar{J}}{\partial dU_k} = 0$. Solving for dU_k from Eq. (39),

$$dU_k = \begin{cases} R_k^{-1}\,(R_S\, U_k^p - R_S\, U_{k-1}^p - R_{NL}\, W_k + B_k^T \lambda) & \ldots \quad k > 1 \\ R_k^{-1}\,(-R_{NL}\, W_k + B_k^T \lambda) & \ldots \quad k = 1 \end{cases} \tag{40}$$

Again, from the condition of optimality $\frac{\partial \bar{J}}{\partial \lambda} = 0$. This gives

$$dY_{n_N} = \sum_{k=2}^{N-1} B_k\, dU_k + B_1\, dU_1 \tag{41}$$

Substituting dU_k from Eq. (40) into Eq. (41), one obtains

$$dY_{n_N} = \begin{cases} \sum_{k=2}^{N-1} B_k\, R_k^{-1}\,(R_S\, U_k^p - R_S\, U_{k-1}^p - R_{NL}\, W_k + B_k^T \lambda) \\ \qquad + B_1\, R_1^{-1}\,(-R_{NL}\, W_1 + B_1^T \lambda) & \ldots \quad k > 1 \\ B_k\, R_k^{-1}\,(-R_{NL}\, W_k + B_k^T \lambda) & \ldots \quad k = 1 \end{cases} \tag{42}$$

dY_{n_N} in Eq. (42) can also be written as

$$dY_{n_N} = \begin{cases} A_\lambda\, \lambda + b_{\lambda 1} - b_{\lambda 2} - b_{\lambda 3} & \ldots \quad k > 1 \\ A_\lambda\, \lambda - b_{\lambda 3} & \ldots \quad k = 1 \end{cases} \tag{43}$$

where

$$
\begin{aligned}
A_\lambda &\triangleq \sum_{k=1}^{N-1} B_k\, R_k^{-1}\, B_k^T \\
b_{\lambda 1} &\triangleq \sum_{k=2}^{N-1} B_k\, R_k^{-1}\, R_S\, U_k^p \\
b_{\lambda 2} &\triangleq \sum_{k=2}^{N-1} B_k\, R_k^{-1}\, R_S\, U_{k-1}^p \\
b_{\lambda 3} &\triangleq \sum_{k=1}^{N-1} B_k\, R_k^{-1}\, R_{NL}\, W_k
\end{aligned}
$$

Hence, λ can be computed from Eq. (43) as

$$
\lambda = \begin{cases} A_\lambda^{-1}\,(dY_{n_N} - b_{\lambda 1} + b_{\lambda 2} + b_{\lambda 3}) & \ldots \quad k > 1 \\ A_\lambda^{-1}\,(dY_{n_N} + b_{\lambda 3}) & \ldots \quad k = 1 \end{cases} \tag{44}
$$

Substituting λ into Eq. (40), dU_k is obtained as

$$
dU_k = \begin{cases} R_k^{-1}\,[R_S\, U_k^p - R_S\, U_{k-1}^p - R_{NL}\, W_k + \\ \quad B_k^T\, A_\lambda^{-1}\,(dY_{n_N} - b_{\lambda 1} + b_{\lambda 2} + b_{\lambda 3})] & \ldots \quad k > 1 \\ R_k^{-1}\,[-R_{NL}\, W_k + B_k^T\, A_\lambda^{-1}\,(dY_{n_N} + b_{\lambda 3})] & \ldots \quad k = 1 \end{cases} \tag{45}
$$

Thus, the guidance command can now be updated as

$$
U_k = U_k^p - dU_k, \qquad (k = 1, \ldots, N-1) \tag{46}
$$

Note that in order to update the guidance command, it is necessary to evaluate the B_k matrices from Eq. (7) in Section II. The terms comprising the B_k matrix are $\left(\frac{\partial F_k}{\partial Z_{n_k}}\right)$, $\left(\frac{\partial F_k}{\partial U_k}\right)$ and $\left(\frac{\partial Y_{n_k}}{\partial Z_{n_k}}\right)$. The values of the state variables at the N^{th} step are taken as the outputs so that $\left(\frac{\partial Y_{n_k}}{\partial Z_{n_k}}\right) \triangleq I_4$. $\left(\frac{\partial F_k}{\partial Z_{n_k}}\right)$ and $\left(\frac{\partial F_k}{\partial U_k}\right)$ are matrices containing the partial derivatives of F_k with respect to Z_{n_k} and U_k respectively.

As discussed earlier, the B_k matrices are computed recursively which saves a significant amount of computational effort. Once again, this feature of the MPSP technique is instrumental in enabling the reentry guidance scheme to be computationally very efficient.

I. Bank Angle Computation

In this subsection, a method to obtain the bank angle (σ) profile from the updated heading angle (ψ) is given. The technique of dynamic inversion (DI) [21] is used to accomplish this task. In the present case, the real control variable is the bank angle, where as the heading angle is an intermediate control variable. Hence, using dynamic inversion, the necessary bank angle history is found that will result in the vehicle having the heading given by the heading angle history, as predicted by the MPSP guidance loop. This is done by enforcing the following first-order error dynamics

$$(\psi^{*\prime} - \psi^{\prime}) + k_{\psi}(\psi^{*} - \psi) = 0 \tag{47}$$

where, superscript $^{\prime}$ denotes derivative with respect to the energy variable e. The value of ψ^{*} in Eq. (47) is known from the converged solution of the MPSP formulation.

Introducing quasi-steady approximation, $\psi^{*\prime}$ is assumed to be zero in each guidance energy interval window, even though its value gets updated at each grid point. Along with this assumption, substituting the value of $\psi^{\prime}$ from the system dynamics and carrying out the necessary algebra, one can arrive at

$$\begin{aligned}\sigma = \sin^{-1}\Bigg(\frac{m V \cos\gamma}{L} \Big[\dot{e}\, k_{\psi}(\psi^{*} - \psi) + \frac{V}{r}\cos\gamma\cos\psi\tan\phi \\ - 2\,\Omega_e(\tan\gamma\cos\phi\,\sin\psi - \sin\phi) + \frac{\Omega_e^2 r}{V\cos\gamma}\sin\phi\,\cos\phi\,\cos\psi \Big]\Bigg)\end{aligned} \tag{48}$$

In addition to obtaining a closed form solution for the bank angle, an additional advantage of using Eq. (48) is that the $sin^{-1}(.)$ term will always yield values in the interval $[-90^0, 90^0]$ based on the sign of its argument (which largely depends on $k_{\psi}(\psi^{*} - \psi)$, by appropriate selection of k_{ψ}). The design is based on energy domain, so one can use the information of both initial and final energy values for selecting the time constant τ_{ψ}, and hence the $k_{\psi} = 1/\tau_{\psi}$ value. The time constant τ_{ψ} is set in such a manner that the tracking is neither too aggressive nor very sluggish and the bank angle is always maintained within a bound of $\pm 45^0$.

The dynamic inversion technique, along with the heading angle as an intermediate control variable, also incorporate the means to determine bank reversals. First, note that it is desirable to have bank reversals in the middle segment of the trajectory. This is because in the beginning the vehicle is not expected to have large dynamic pressure and hence it is difficult to do large bank angle maneuvers (it is not advisable to

activate the reaction control system for this). More important, at the end of the reentry the vehicle should preferably be in the no roll attitude to facilitate good terminal area energy management. Hence, the bank angle should be close to zero at the end of the reentry phase. In practice, this is done by setting the weight for $d\psi$ minimization in the control update process large at the start and end of the trajectory and small in the middle of the trajectory. The weight chosen for $d\psi$, $R_{d\psi}$ is illustrated in Figure 4. A hyperbolic tangent function represented in Eq. (49) is used to generate this weight. Clearly, $d\psi$ will have little room to vary at the start and near the end of the trajectory, but will have good flexibility in the middle part.

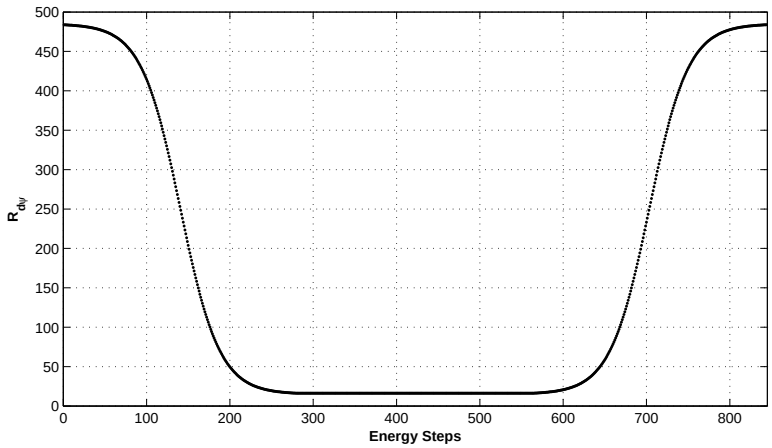

Fig. 4 Weighting function for $d\psi$ Minimization

$$R_{d\psi_k} = \begin{cases} A - B \tanh \dfrac{m\left(k - \frac{N}{6}\right)}{\left(\frac{N}{6}\right)} & k = 1, \ldots, N/3 \\ A - B \tanh m & k = N/3, \ldots, 2N/3 \\ A - B \tanh \dfrac{m\left(\frac{5N}{6} - k\right)}{\left(\frac{N}{6}\right)} & k = 2N/3, \ldots, N \end{cases} \tag{49}$$

where A and B are constants that define the maximum and minimum value of the weight and the factor m determines how fast the transition from the maximum to minimum values occurs.

IV. Numerical Results

The reentry guidance technique presented in this paper is simulated using realistic vehicle data as generated from the computational fluid dynamics analysis. The mass of the vehicle is assumed constant for the entire flight since (i) there is no thrust in the vehicles (ii) heat flux being not high ablative cooling mechanism is not there in the vehicle and (iii) reaction control jet fuel depletion is very small compared to the mass of the vehicle and can safely be neglected especially in the guidance computation. The nominal initial condition for reentry as well as the final desired conditions after the reentry phase are tabulated below.

Table 1 Initial Nominal Conditions for Reentry

Variables	Value
Height (h)	51000 m
Velocity (V)	1796 m/s
Flight path angle (γ)	-15.32 deg
Latitude angle (ϕ)	16.0 deg
Longitude angle (θ)	84.0 deg

Table 2 Final Desired Conditions for Reentry

Variables	Value
Height (h)	20042 m
Velocity (V)	556.59 m/s
Flight path angle (γ)	- 20.44 deg
Latitude angle (ϕ)	14.03 deg
Longitude angle (θ)	85.93 deg

Note that whereas the desired final condition remains the same, the initial condition can vary and results are included for a number of initial conditions around these nominal values. Moreover, perturbation studies with respect to aerodynamic data perturbation is also carried out. Details of this study as well as the results obtained are included in this section.

A. Selection of Energy Step Size

Since energy is considered as the independent variable in the formulation presented in this paper, its step size for the state propagation must be chosen wisely. In fact, guidance cycle update for stable vehicles such as the RLV under consideration typically takes place at 100 *ms* interval in time domain. Hence, there are two options (i) to select a variable energy interval that will correspond to 100 *ms* interval in time domain or (ii) to select a constant energy interval which will be representative of this situation. Note that option (i) is difficult to mechanize since without closing in the guidance and control loops a variable energy step size to represent a constant step size in time is impossible to obtain as it depends on the actual trajectory followed (which is unknown before designing the guidance loop). Hence option (ii) was selected to be followed in this paper. With this in mind, open loop simulation were carried out to determine the largest value of the energy step that has a corresponding time step approximately equal to 100 *ms*. Note that because energy

is a continuous and monotonic function of time (see Fig. 8 for representative plots of energy vs time), such a mechanization will ensure that the actual time interval remains equal to or higher than 100 *ms* in reality and hence computational time delay will not be much of a concern from time delay margin point of view. On the other hand, since energy is a slowly changing variable (and hence approximately a linear function of time), the maximum time interval is not substantially higher than 100 *ms* and hence guidance objective is not compromised either. From a few open loop simulations, Δe was finally fixed at $-248\,m$.

B. Convergence condition in MPSP

The convergence bound for terminating the MPSP interation is defined such that the algorithm recognizes whether the solution is good enough or not. Even though a number of possibilities exists for defining a good convergence condition, it is assumed here that the convergence of the algorithm takes place when all of the following conditions hold: (i) the absolute error in altitude is less than $25m$, (ii) absolute error in velocity is less than $0.5m/s$, (iii) absolute error in flight path angle is less than 0.2^0, and (iv) absolute error of the position coordinates, i.e. latitude and longitude, are individually less than 0.1^0. One can clearly notice that these are quite small, and hence, the terminal accuracy is enforced to be quite good. Moreover, it was also decided that the solution is 'acceptable' only if all the path constraints are also met.

C. Results with Nominal and Perturbations in Initial Conditions

The proposed guidance technique is tested by assuming the nominal as well as perturbed initial conditions at the reentry point. The initial conditions along with the errors in the outputs at the end of the reentry are given in Table 3, where 'Base case' stands for the 'Nominal case' and other cases stand for perturbations around it. From the results obtained, it can be inferred that the algorithm successfully works not only for the nominal case, but also for a number of cases around it and results in very good final accuracy in each case.

Various trajectory results are plotted in Figs. 5–16. The altitude and velocity trajectories are plotted in Figs. 5 and 6 respectively. It can be observed that whereas the velocity trajectory in monotonically decreasing, the altitude trajectory is not so. In fact, after some time, the vehicle actually climbs a bit before coming down again, thereby allowing itself to get more time to dissipate its energy. This is also the phase of higher lift, as evident from the normal load plots in Fig. 7. The plots in Fig. 7 also show that the normal load acting on the vehicle always remain less than $3g$ for all time in all cases, which is a strong requirement from the

Table 3 Different initial conditions and corresponding errors in the outputs at the end of reentry

Case	$h\,(km)$	$V(m/s)$	γ(deg)	$E_h(km)$ $\lvert h_N - h^d \rvert$	$E_V(m/s)$ $\lvert V_N - V^d \rvert$	$E_\gamma(deg)$ $\lvert \gamma_N - \gamma^d \rvert$	$E_\phi(deg)$ $\lvert \phi_N - \phi^d \rvert$	$E_\theta(deg)$ $\lvert \theta_N - \theta^d \rvert$
Base Case	51.000	1,796	-15.32	0.003	0.059	0.1656	0.0562	0.0540
Case 1	46.920 (-8%)	1,867 (+4%)	-15.63 (+2%)	0.005	0.086	0.0568	0.019	0.018
Case 2	52.530 (+3%)	1,760 (-2%)	-14.86 (-3%)	0.008	0.159	0.002	0.0754	0.0726
Case 3	49.980 (-2%)	1,832 (+2%)	-15.63 (+2%)	0.002	0.033	0.150	0.0272	0.0260
Case 4	53.550 (+5%)	1,743 (-3%)	-14.4 (-6%)	0.005	0.071	0.0474	0.0703	0.0679
Case 5	47.940 (-6%)	1,832 (+2%)	-14.55 (-5%)	0.013	0.223	0.0262	0.0135	0.0128
Case 6	49.980 (-2%))	1,760 (-2%)	-14.4 (-6%)	0.005	0.093	0.1616	0.0915	0.0883
Case 7	47.940 (-6%)	1,849 (+3%)	-15.01 (-2%)	0.015	0.265	0.0515	0.0058	0.0053

vehicle safety consideration. The associated profiles for specific energy (which is a function of altitude and velocity) are shown in Fig. 8, which conforms to the assumption that it remains monotonic throughout the reentry phase owing to the presence of drag and absence of thrust.

The flight path angle and heading angle trajectories are plotted in Fig. 9 and Fig. 10 respectively. From Fig. 9 one can observe how the velocity vector keeps changing its direction on its flight path. Moreover, the change is quite smooth. This essentially due to the smoothness enforcement on α in the cost function formulation, because of which the α trajectory turns out to be smooth. From Fig. 10 it is clear that the initial heading angle guess as done through the spherical trigonometry algebra is quite good and only a small variation about that is necessary to reach the final destination with good accuracy. The variation along the way is not much either. Moreover, all the variations happen in the middle of the trajectory, which is very nice. Note that both ψ and $\dot{\psi}$ stabilize towards the end of the trajectory, thereby driving the bank angle to zero in all cases.

The latitude and longitude trajectories are plotted in Fig. 11 and Fig. 12 respectively. Here the change is not as apparent since they all start with the same latitude and longitude (variation of that is studied separately). However, it should be noted that in each case they end up with $\pm 0.1^0$ error at the end of the reentry (the desired coordinates at the end of the reentry are 14.03^0 and 85.93^0 respectively).

The dynamic pressure and heat flux trajectories are plotted in Fig. 13 and Fig. 14 respectively. First, Fig. 14 shows that the heat flux is well within the allowable bound (which is put as $40 W/cm^2$ in this paper), and hence there is no danger to the vehicle. Moreover, the heat flux stabilizes at a very small value of

$2W/cm^2$ at the end in all cases. This corroborates the fact that the reentry is actually over at this point of time. A high dynamic pressure with a low normal load at the end of the reentry also indicate a good control authority as angle of attack α can then be varied freely after the end of the reentry.

Finally the guidance parameter trajectories (i.e. trajectories of α and σ) are plotted in Fig. 15 and Fig. 16 respectively. Figure 15 shows that the angle of attack (α) trajectories are smooth for all cases and remain fairly close to the middle of the upper and lower boundaries defined by $\alpha_{max}(M)$ and $\alpha_{min}(M)$ respectively, leaving quite a bit of gap on both sides. This is very good as it gives margins for the autopilot to correct unexpected disturbances on the way. Figure 16 also shows that the actual values are quite small and are well within the bounds of $\pm 45^0$. However, at this point it makes sense to exercise a bit of caution while inferring this, since all of these simulation cases start with the same initial latitude and longitude coordinates. Once that is perturbed, the required bank angles turn out to be much more, as evident from the results presented in a subsequent subsection.

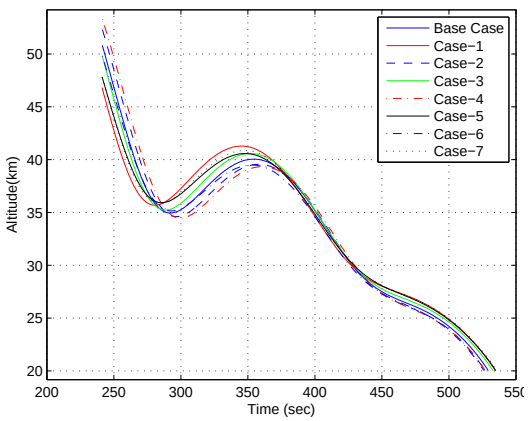

Fig. 5 Altitude trajectories for initial conditions in Table 3

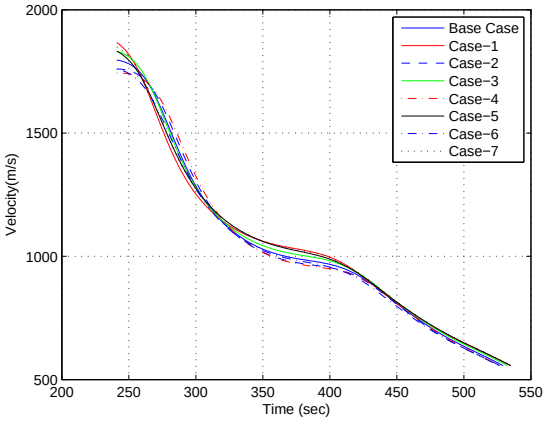

Fig. 6 Velocity trajectories for initial conditions in Table 3

D. Simulation study with large number of random perturbations

Next, in order to adequately demonstrate the robustness of the proposed technique to variations in the initial conditions, a large number of simulation studies (100 to be exact) were carried out with random perturbations in the initial velocity, initial height and initial flight path angle (which turn out to be sensitive parameters). Table 4 gives the range of these perturbations.

The results obtained from this study demonstrate that terminal and path constraints have been met with

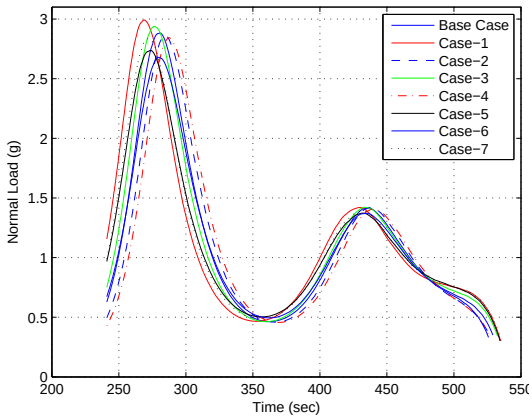

Fig. 7 Normal load trajectories for initial conditions in Table 3

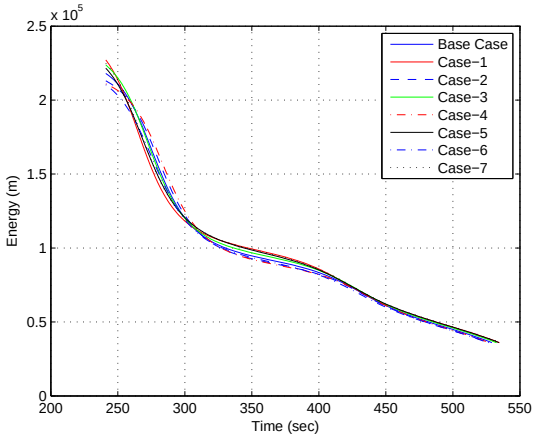

Fig. 8 Specific energy trajectories for initial conditions in Table 3

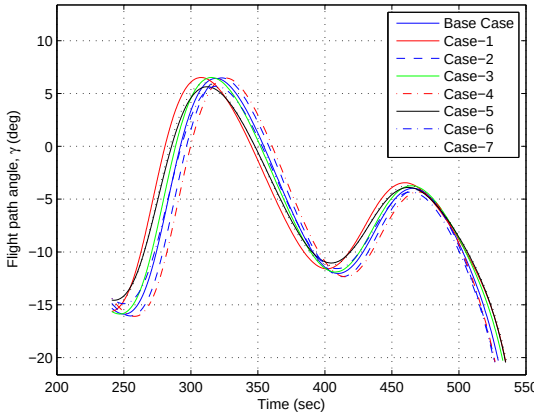

Fig. 9 Flight path angle trajectories for initial conditions in Table 3

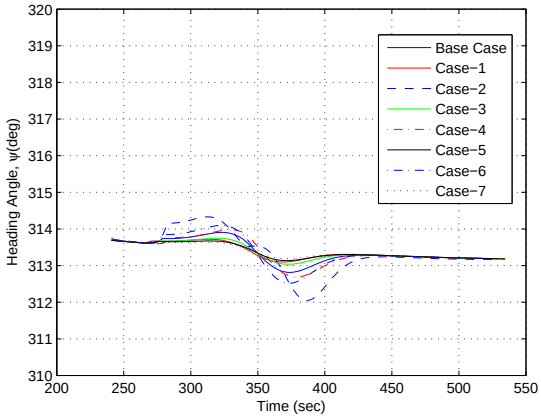

Fig. 10 Heading angle trajectories for initial conditions in Table 3

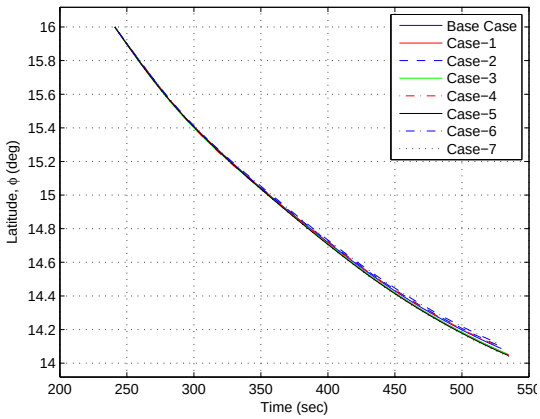

Fig. 11 Latitude trajectories for initial conditions in Table 3

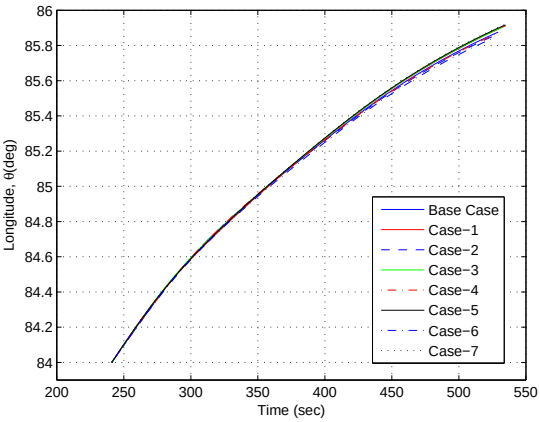

Fig. 12 Longitude trajectories for initial conditions in Table 3

good accuracy as well. Moreover, the convergence rate of the algorithm is 100% and the number of iterations

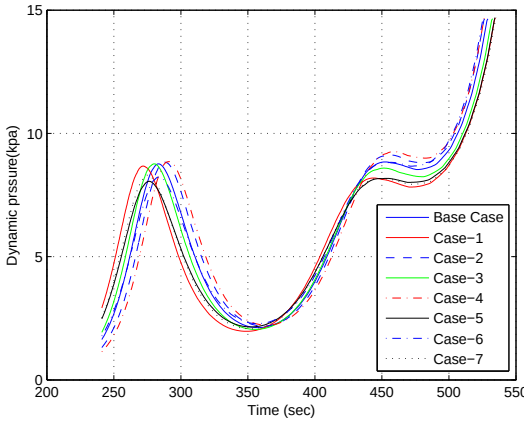

Fig. 13 Dynamic pressure trajectories for initial conditions in Table 3

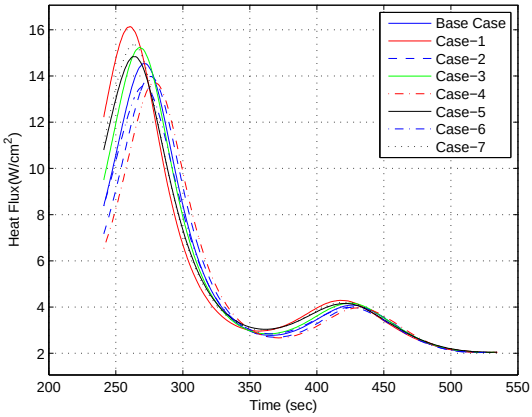

Fig. 14 Heat flux trajectories for initial conditions in Table 3

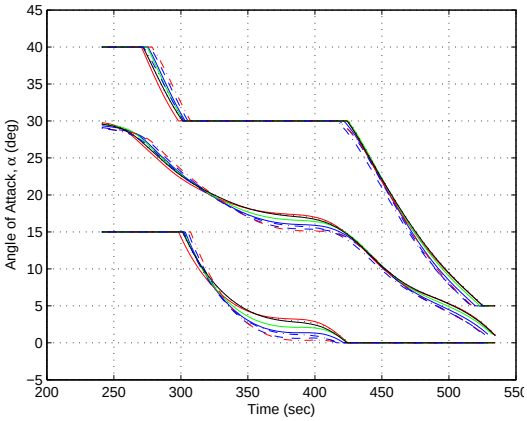

Fig. 15 Angle of attack trajectories for initial conditions in Table 3

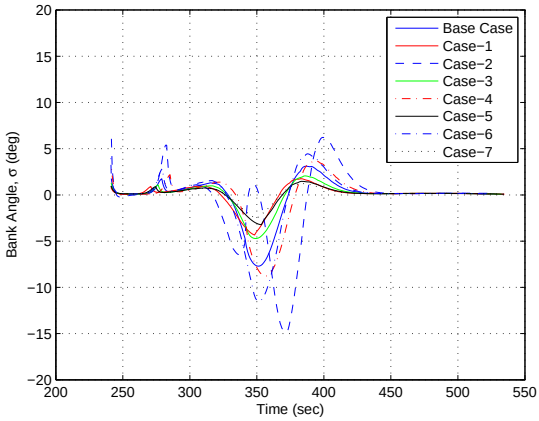

Fig. 16 Bank angle trajectories for initial conditions in Table 3

Table 4 Range of Random Perturbations in Initial Conditions

Variable	Base Initial Value	Range of Perturbations
Height (h), km	51.0	$\pm$ 3.0
Velocity (V), m/s	1796	$\pm$ 100
Flight Path Angle (γ), deg	-15.32	$\pm$ 2

for convergence is found to be generally two (with upper limit being four and lower limit being as small as one). The normal load constraint is maintained below $3g$ for each simulation. Detail plots resulting out of this simulation study are omitted here to contain the length of the paper.

E. Simulation Study with Perturbation in Reentry Coordinates

In this part of the simulation study, perturbations are introduced in in the expected initial position of reentry. Latitude ϕ and Longitude θ are varied by $\pm 1deg$ and altitude is varied by $\pm 2.5km$. Once again, even though a large number of simulation were carried out, eight cases were randomly selected for including representative results in this paper. The numerical values of the actual reentry position as well as the errors obtained for the final desired position for these eight cases are given in Table 5. Note that the desired coordinates at the end of the reentry are $h(t_f) = 20.0km$, $\phi(t_f) = 14.03^0$ and $\theta(t_f) = 85.93^0$ respectively and the actual values obtained are very close to these values.

Table 5 Initial Perturbation in Reentry Coordinates

Case	Initial Altitude $h(km)$	Initial Latitude (ϕ, deg)	Initial Longitude (θ, deg)	$E_h(km)$ $\lvert h_N - h^d \rvert$	$E_\phi(deg)$ $\lvert \phi_N - \phi^d \rvert$	$E_\theta(deg)$ $\lvert \theta_N - \theta^d \rvert$
Base Case	51.000	16	84	0.003	0.0561	0.0542
Case 1	49.980	16.3	84.5	0.013	0.0386	0.0239
Case 2	53.550	15.4	83.5	0.010	0.024	0.0417
Case 3	52.530	16.5	84.7	0.006	0.0789	0.0388
Case 4	48.450	15.3	83.6	0.010	0.0032	0.0060
Case 5	52.530	15.7	83.7	0.009	0.0384	0.0504
Case 6	53.550	16.2	84.3	0.005	0.0061	0.0047
Case 7	49.980	15.5	83.6	0.009	0.0396	0.0617

Only a few selected plots are included form this exercise to contain the length of the paper. Figures 17 and 18 represent the altitude and heading angle trajectories. The nature of the altitude variation is as expected and it is nice to see that the evolution of the altitude variation happens to be quite close to each other in all cases. Even though the variation of the heading angle is not much in any individual case, the value itself depends on the initial position of the vehicle, which is once again conforms to the intuition. The separation between the lowest and highest value happens to be as high as 35^0, which was necessary to guide the vehicle in the desired position direction (which remains the same irrespective of where the reentry starts). The corresponding latitude and longitude trajectories are shown in Figs. 19 and 20 respectively. It is quite evident form these plots how they evolve towards the desired value of $\phi(t_f) = 14.03^0$ and $\theta(t_f) = 85.93^0$ respectively. Note that the vehicle manages to reach the vicinity of the desired location within a $\pm 0.1deg$ accuracy in both latitude and longitude (which is also quite clearly evident in Table 5).

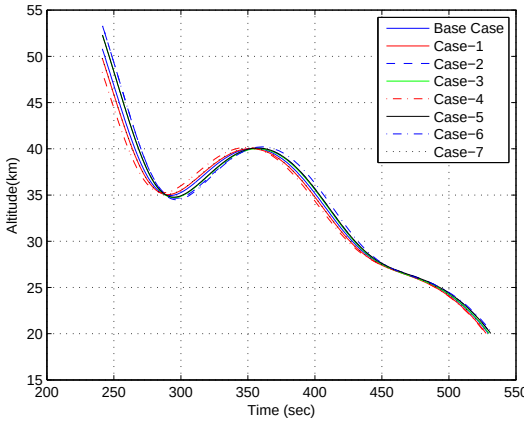

Fig. 17 Altitude trajectories for initial positions in Table 5

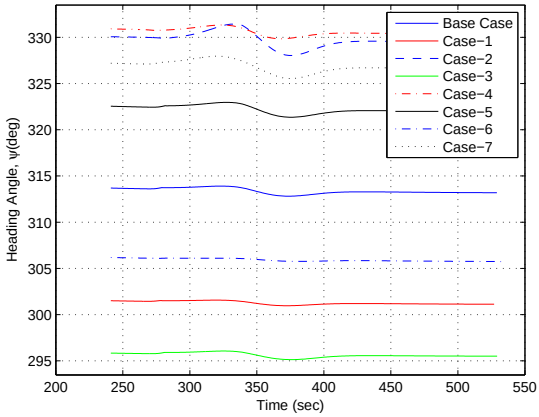

Fig. 18 Heading angle trajectories for initial positions in Table 5

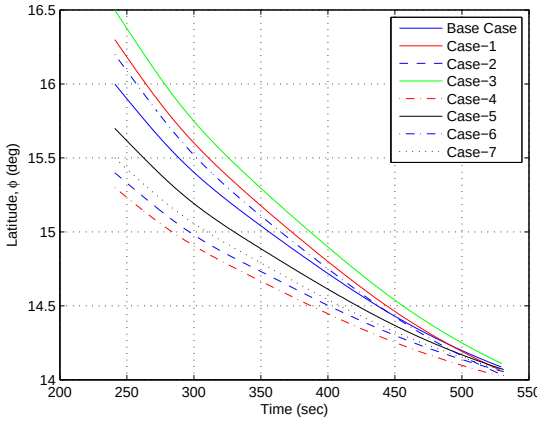

Fig. 19 Latitude trajectories for initial positions in Table 5

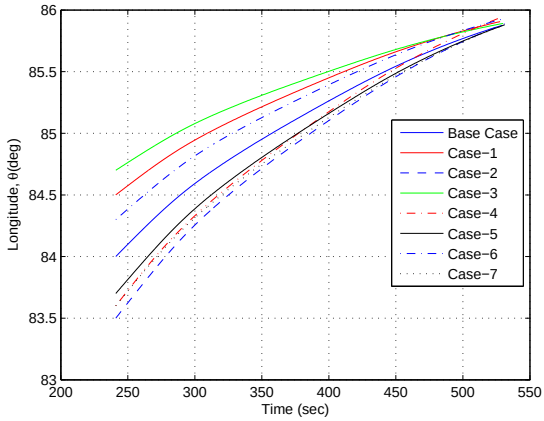

Fig. 20 Longitude trajectories for initial positions in Table 5

The associated guidance parameters, namely the angle of attack α and bank angle σ plots are shown in Figs. 21 and 22 respectively. It is evident that variation in α is not much as it strongly depends on dynamic parameters such as velocity magnitude and flight path angle (which have been kept constant in this exercise). However, the variation of σ is quite a bit here as that plays a role in position correction and hence depends strongly on where the vehicle starts its reentry. Note that in one case (Case - 2), at one point the value of σ is relatively high, the peak value being -30.5^0 (which is still very much within the $\pm 45^0$ bound). This is because of the fact that in that case the initial condition of the longitude is quite far away from the nominal case and hence the vehicle needs to be steered away towards the desired location with a bit of aggressive maneuver. However, it is once again nice to see that the aggressive maneuver takes place in the middle part of the trajectory, which is because of the way the cost function is designed. In each case, the vehicle again

stabilizes with zero bank angle at the end of the reentry.

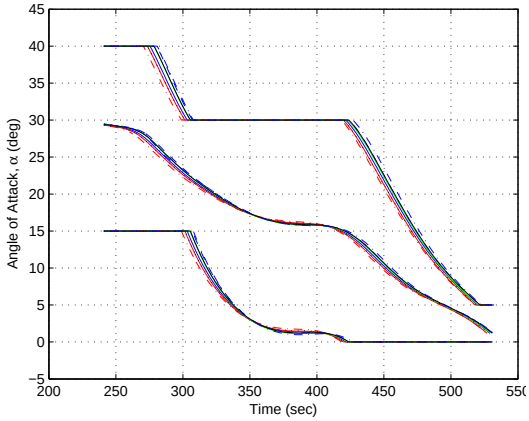

Fig. 21 Angle of attack trajectories for initial positions in Table 5

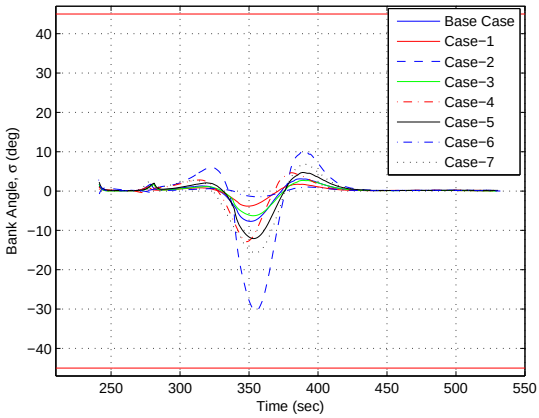

Fig. 22 Bank angle trajectories for initial positions in Table 5

F. Simulation study with Perturbation in Aerodynamic Parameters

This section provides simulation results to test the capability of the proposed guidance technique to recompute the trajectory online, in case of perturbations in the aerodynamic parameters, namely C_L and C_D, which strongly effect the dynamics. In the simulation experiment, a $\pm$ 10% random variations in the aerodynamic parameters C_{L_0}, C_{D_0}, C_{L_α} and C_{D_α} was considered to evaluate the performance of the proposed guidance logic. An assumption made here is that the percentage error of the coefficient values remain same throughout the flight trajectory. However, there was a need to excite the guidance logic frequently along the flight path due to the fact that the actual state was not following the predicted state because of the modelling discrepancy. In total, 100 trajectories were generated in this exercise by randomly selecting a percentage error (within $\pm$ 10% of the base value) for each aerodynamic coefficient in each simulation.

The outcome of this simulation study indicates (i) all trajectories are maintained relatively smooth for the full duration of reentry, (ii) the normal load is always maintained below its upper limit of $3g$ (iii) the angle of attack trajectories remain close to the middle of the upper and lower bounds, leaving aside sufficient margin for the autopilot to handle unwanted disturbances, (iv) bank angles are bounded between $\pm 45^0$ and bank angle reversals happen at the middle of the trajectory. Moreover, it turns out that the algorithm converges within a few iterations (less than five iterations). These results demonstrate the robustness of the guidance technique with respect to perturbations in aerodynamic parameters as well as the rapid convergence of the

MPSP algorithm. Detail plots and table of results are omitted to contain the length of the paper.

V. Conclusions

Taking the help of recently-developed model predictive static programming, a suboptimal reentry guidance logic is presented in this paper for a reusable launch vehicle in a technology demonstration mission. This guidance essentially shapes the trajectory of the vehicle by predicting the necessary angle of attack and bank angle that the vehicle should execute, while satisfying a set of path and terminal constraints. The guidance law is primarily based on nonlinear optimal control theory and hence imbeds effective trajectory optimization concepts into the guidance law. In addition to the promising results for the nominal case, it has also been demonstrated that this guidance also has sufficient robustness for both state perturbations as well as parametric uncertainties in the model.

It can be mentioned here that the guidance commands generated in this paper has also been realized in an inner-loop autopilot design using the full six degree-of-freedom model of the vehicle. While designing the autopilot, extensive attention has been given both for aerodynamic and reaction jet control system designs as well as their fusion. However, the authors believe that the details of the autopilot design, which includes a nominal loop as well as an adaptive loop for enhanced robustness for modeling inaccuracies, merits to be reported in a separate paper.

Acknowledgement

The authors would like to acknowledge the contribution of S. Mathavaraj during the initial phase of this work through technical discussions. He has also contributed significantly in designing a nonlinear and robust inner-loop autopilot accounting for the full six degree-of-freedom vehicle dynamics for realizing the guidance commands.

References

[1] Harpold, J. C. and Graves, C. A., "Shuttle Entry Guidance," Nasa n79–74791, May 1979.

[2] Shen, Z. and Lu, P., "Onboard Generation of Three-Dimensional Constrained Entry Trajectories," *Journal of Guidance, Control, and Dynamics*, Vol. 27, No. 1, 2003, pp. 111–121.

[3] Brinda, V., Arora, R., and Janardhana, E., "Mission Analysis of a Reusable Launch Vehicle Technology Demonstrator," *Proceedings of AIAA/CIRA 13th International Space Planes and Hypersonics Systems and Technologies*, Capua, Italy, May 2005.

[4] Mease, K. D. and Kremer, J. P., "Shuttle Entry Guidance Revisited Using Nonlinear Geometric Methods," *Journal of Guidance, Control, and Dynamics*, Vol. 17, No. 6, 1994, pp. 1350–1356.

[5] Cavallo, A. and Ferrara, F., "Atmospheric Reentry Control for Low Lift/Drag Vehicles," *Journal of Guidance, Control, and Dynamics*, Vol. 19, No. 1, 1996, pp. 47–53.

[6] Dukeman, G. A., "Profile-Following Entry Guidance Using Linear Quadratic Regulator Theory," *Proceedings of AIAA Guidance Navigation and Control Conference and Exhibit*, Monterey, CA, August 2002.

[7] Lu, P., Shen, Z., Dukeman, G. A., and Hanson, J. M., "Entry Guidance by Trajectory Regulation," *Proceedings of AIAA Guidance, Navigation, and Control Conference and Exhibit*, Denver, Colorado, August 2000.

[8] Bryson, A. E. and Ho, Y. C., "Applied Optimal Control," chap. 1–3, Hemisphere Publishing Corporation, 1975.

[9] Kirk, D. E., "Optimal Control Theory: An Introduction," Dover Publications, 1970, pp. 329–413.

[10] Roenneke, A., "Adaptive On-Board Guidance for Entry Vehicles," *Proceeding of AIAA Guidance Navigation and Control Conference and Exhibit*, Montreal, Canada, August 2001.

[11] Shen, Z. and Lu, P., "Dynamic Lateral Entry Guidance Logic," *Journal of Guidance, Control, and Dynamics*, Vol. 27, No. 6, 2004, pp. 949–959.

[12] Joshi, A. and Sivan, K., "Reentry Guidance for Generic RLV Using Optimal Perturbations and Error Weights," *Proceedings of AIAA Guidance Navigation and Control Conference and Exhibit*, San Francisco, August 2005.

[13] Rossiter, J. A., "Model Based Predictive Control: A Practical Approach," chap. 3,4, CRC Press, New York, 1st ed., 2003.

[14] Werbos, P. J., "Approximate Dynamic Programming for Real-time Control and Neural Modeling," *Handbook of Intelligent Control: Neural, Fuzzy & Adaptive Approaches*, Van Nostrand Reinhold, New York, 1992, pp. 65–89.

[15] Padhi, R. and Kothari, M., "Model Predictive Static Programming: A Computationally Efficient Technique for Suboptimal Control Design," *International Journal of Innovative Computing, Information and Control*, Vol. 5, No. 2, 2009, pp. 399–411.

[16] McHenry, R. L., Long, A. D., Cockrell, B. F., Thibodeau, J. R. I., and Brand, T. J., "Space Shuttle Ascent Guidance Navigation and Control," *Journal of the Astronautical Sciences*, Vol. 27, 1979, pp. 1–38.

[17] Chawla, C., Sarmah, P., and Padhi, R., "Suboptimal Reentry Guidance of Reusable Launch Vehicles Using Pitch Plane Maneuver," *Aerospace Science and Technology*, Vol. 14, No. 6, 2010, pp. 377–386.

[18] Dwivedi, P. N., Bhattacharya, A., and Padhi, R., "Suboptimal Midcourse Guidance of Interceptors for High Speed Targets with Alignment Angle Constraint," *Journal of Guidance, Control, and Dynamics*, Vol. 34, No. 3, 2011, pp. 860–877.

[19] Gong, Q., Fahroo, F., and Ross, I. M., "Spectral Algorithm for Pseudospectral Methods in Optimal Control," *Journal of Guidance, Control, and Dynamics*, Vol. 31, No. 3, 2008, pp. 460–471.

[20] Bollino, K. P., Ross, I. M., and Doman, D. D., "Optimal Nonlinear Feedback Guidance for Reentry Vehicles," *Proceedings of AIAA Guidance, Navigation, and Control Conference and Exhibit*, Keystone, CO, August 2006.

[21] Enns, D., Bugajski, D., Hendrick, R., and Stein, G., "Dynamic Inversion: An Evolving Methodology for Flight Control design," *International Journal of Control*, Vol. 59, No. 1, 1994, pp. 71–91.

[22] Donnay, J. D. H., "Spherical Trigonometry," chap. 4, Interscience Publishers Inc., 1945.

[23] Vinh, N. X., Busemann, A., and Culp, R. D., "Hypersonic and Planetary Entry Flight Mechanics," chap. 2, Univ of Michigan Press, 1980.

[24] Ananthasayanam, M. R. and Narasimha, R., "A Proposed International Tropical Reference Atmosphere Upto 1000 km," *Advances in Space Research*, Vol. 7, No. 10, 1987, pp. 117–131.

[25] Gallais, P., "Atmospheric Re-entry Vehicle Mechanics," chap. 4,6,8, Springer, Berlin Heidelberg, 1st ed., 2007.

[26] Young, J., "A method for longitudinal and lateral range control for a high-drag low-lift vehicle entering the atmosphere of a rotating earth," *NASA Technical Note D-954*, 1961.