Defence Science Journal, Vol. 63, No. 3, May 2013, pp. 254-261, DOI: 10.14429/dsj.63.2414


# Online Trajectory Reshaping for a Launch Vehicle to Minimize the Final Error Caused by Navigation and Guidance

Tessy Thomas[*], D.R. Jahagirdar[#], I.V. Murali Krishna[!], and L. Varaprasad[*]

[*]*Advanced Systems Laboratory, Hyderabad, India*
[#]*Research Center Imarat, Hyderabad, India*
[!]*Jawaharlal Nehru Technological University, Hyderabad. India*
[*]*E-mail: tessy_1963@yahoo.com*

**ABSTRACT**

Autonomous launch vehicles, once lifted off from the launch pad, equipped with an onboard intelligence which aids in achieving the mission objectives with high accuracy. The accuracy of the mission depends basically on navigation and guidance errors caused at burnout condition, after which the vehicle follows an elliptical path upto impact. The paper describes how to handle the final impact and injection error caused by these navigation and guidance errors. In the current work the initial burnout conditions are tuned and corrected such that the terminal impact point is achieved within the desired tolerance bounds. A two point boundary value problem is solved using the gradient method, for determining the impact errors. The algorithm is validated by simulation studies for various burnout conditions.

**Keywords:** Trajectory reshaping, launch vehicle, navigation, guidance errors

## 1. INTRODUCTION

The main aim of any launch vehicle is to deliver the intended subsystem to its target with maximum accuracy (given tolerance bounds). This is of great importance for systems travelling greater distances like the launch vehicles which carry satellites as payload to inject them in the intended orbits, ballistic missiles which carry warheads to other parts of the world, etc. For this purpose, it is very important to design some guidance strategy which will guide the system such that the overall aim of the mission is met.

Navigation plays an important role for launch vehicle guidance. Generally all launch vehicles carry onboard navigation subsystems. The aim of navigation subsystem is to provide the position, velocity and attitude of the launch vehicle which acts as inputs to the guidance subsystem. Guidance subsystem then generates the commands based on the present and target input such that the vehicle is steered to achieve the mission objectives. The navigation subsystem carries an inertial navigation system (INS unit, self-contained) which generates the current position and velocity instantaneously. The main problem with this type of navigation is drift (non-linear in type typically) which may be because of temperature, ageing, vibration, etc. Unless any compensation is done to this drift there will be an error creeping into the navigation output.

The navigation output acts as input to the guidance system causing an overall error in the trajectory which is to be followed by the launch vehicle in order to obtain the desired mission objectives.

Schappel[11], *et al*. describes the guidance laws for a constant thrust, constant mass depletion case. In this author describes linear sine and Q-matrix guidance algorithms. In case of linear sine method aim is to satisfy radial position and velocity, nulling normal position and velocity while satisfying total velocity parallely. In case of Q-matrix guidance the aim is place the vehicle in a free-fall type of trajectory. Sinha and Shrivastava[2] describes an explicit guidance for a satellite launch vehicle where in guidance is valid for a specified form of thrust-time profile. The equations for change in velocity and position due to thrust is solved analytically using the bi-nominal series (recursive) expansion and gravity effects are taken as Ecke's method. Haeussemann[3] and Horn[4] describes the guidance scheme used in Saturn launch vehicle, where the guidance is based on iterative path adaptive mode. The main aim is to get the minimum propellant trajectory for various orbital injection missions. The crust of the method is obtaining a closed-loop solution for the mathematical model which is obtained from the flat-earth model having constant gravitational field obtained from the optimum thrust direction for minimum propellant consumption, assuming a constant mass flow rate and thrust.

Ohlmeya[5], *et al*. describes practical schemes of INS/GPS integrated navigation scheme with kalman filter estimation for an extended range guided munition. Li and Wang[6] described the error correction algorithm for an INS system using celestial navigation parameters. Miller[7] describes the new attitude algorithm which takes three samples of gyro data per update. Lee[8], gives the attitude algorithm including the high-frequencies base motions and takes four samples of gyro data

Received 25 June 2012, revised 19 February 2013, online published 16 May 2013

per update. Wang[9] describes a method where low sampling rate GPS estimations are used to correct the drifting errors of the accelerometer.

To achieve the mission objectives accurately an on-line error analysis need to be done and accordingly trajectory should be corrected. Many direct and indirect computational techniques are described in the literature to obtain trajectories by considering both software and hardware constraints such as minimising bending loads, safe stage separation (in case of multiple stage launch vehicles), etc. Varaprasad and Padhi[10] presented gradient based trajectory optimisation of a hypersonic launch vehicle in which tight initial conditions (final conditions for the carrier launch vehicle) are required for a hypersonic cruise vehicle to operate. For the present study gradient based trajectory computation technique is used to determine the desired trajectory to fulfill the final mission objectives.

## 3. COMPUTATION OF TRAJECTORY

Most launch vehicles carry on-board processors to perform the mathematical computations which are required for the mission. Once the booster phase is over, the load on the processor is minimal compared to it is earlier stages. The full computational power of the processor can be used for performing trajectory computation and error correction. In literature many numerical methods are discussed to solve trajectory computation problems[11]. For present study, widely popular steepest descent method (indirect method) is used to compute the optimum trajectory. Some of the advantages of this method is that it provides guaranteed minimum from iteration to iteration and efficient when the solution is further away from minima (local/global). The disadvantage of this method is the selection of the step size (apart from trapping in local minima) and it converges slower as it approaches the minima.

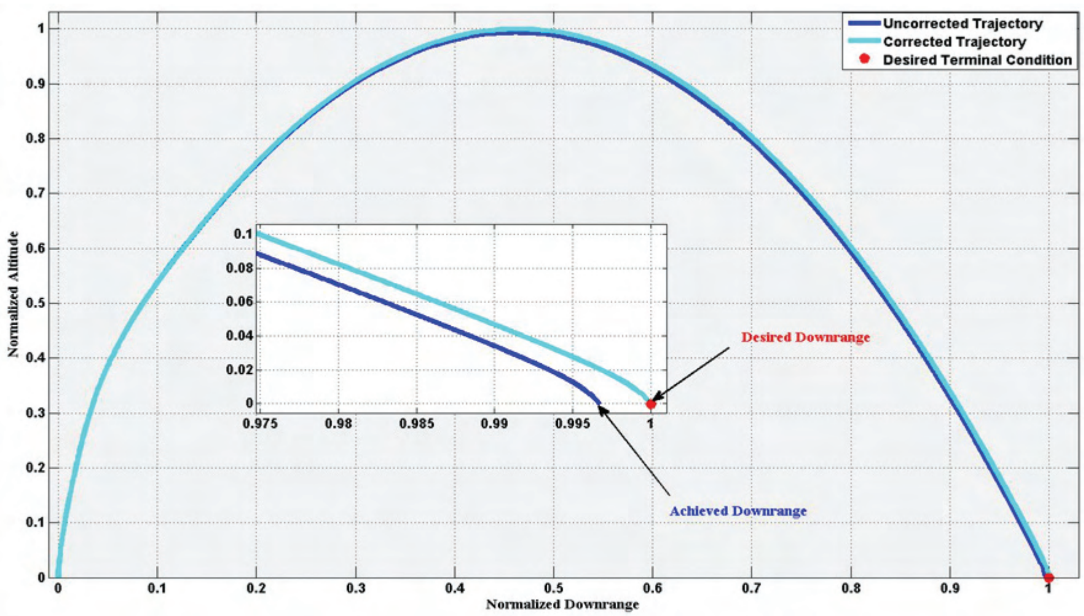


**Figure 1. Normalised downrange vs normalised altitude.**

## 2. MATERIAL AND METHOD

In the present day scenario there is an ever growing demand for the launch vehicles in order to place the desired subsystem at the desired location by minimising cost function (which can be cost included with the mission, final error minimisation, operational cost, etc.).

Navigation model is used to estimate the navigation errors from the inertial navigation system (INS). If INS and guidance systems are performing error free, then the payload is accurately placed in the free flight trajectory and should reach the intended target. The present study deals with the errors associated with INS and explicit guidance. In order to minimise these errors an on-line trajectory is computed by gradient method to reach the desired target within the specified error bounds. A pictorial representation of the desired and actual normalised trajectory under burnout condition (injection conditions) variations is shown in Fig. 1. Finally the application of the algorithm to the real-time systems has been demonstrated by considering various test cases.

## 4. MATHEMATICAL MODELLING OF THE SYSTEM

A nonlinear point mass model with 3-DoF, two translational and one rotational motion is considered for the current work. The model represents nonlinear point mass equations of motion with non-rotating spherical earth. The vehicle considered for the current work is not a lifting vehicle. The equations of motion in the spherical earth coordinate system is given as follows:

$$\begin{bmatrix} \dot{r} \\ \dot{V} \\ \dot{\gamma} \end{bmatrix} = \begin{bmatrix} V\sin(\gamma) \\ \dfrac{F\cos(\alpha)-D}{m} - g\sin(\gamma) \\ \dfrac{F\sin(\alpha)+L}{mV} - \dfrac{g\cos(\gamma)}{V} + \dfrac{V}{R_0+h}\cos(\gamma) \end{bmatrix} \tag{1}$$

D = Drag force = $\frac{1}{2}\rho V^2 S_{ref} C_D$

L = Lift force = $\frac{1}{2}\rho V^2 S_{ref} C_L$

m = mass of the vehicle

$\gamma$ = flight path angle (with respect to horizontal)

V = velocity of the launch vehicle

h = height (from the surface of the earth to the launch vehicle CG)

Equation (1) can be written in generic form:

$$\dot{X} = f(X, U) \quad (2)$$

where the state and control vectors are:

$$X \triangleq (X_1, X_2, X_3) = (r, V, \gamma) \text{ and } U \triangleq \alpha$$

$$f \triangleq (f_1, f_2, f_3)$$

These equations of motion are integrated with respect to time by using Runge-Kutta method[12].

## 5. GRADIENT METHOD

In the day to day life, all systems in the nature use optimisation techniques. For example, ants seeking a path between their colony and a food source, honey bee finding their food, human immune system, evolution of life on the earth, etc., the same concept is adopted in engineering applications to find an optimum and acceptable solution to a given problem within the given domain. The optimisation starts with selection of a cost function (yields a number) which has to be minimised/maximised based on the requirement subjected to constraints (else unconstrained optimisation). The aim is to find the values of the variable which optimises the desired cost function. In this optimisation technique, it is very important to select initial guess. Initial guess places an important role in the convergence of the algorithm (number of iterations for convergence), if the guess is close to the desired solution then the convergence is faster and vice-versa. The algorithm starts with an initial guess of the optimal values of the variables and generate a sequence of improved estimates until they reach a solution. Opitmisation criteria uses the values of the objective function, the constraints and mostly the first and second derivatives of the functions. Most of the algorithms accumulate the information gathered during previous iterations, others use only local information from the current point.

Here, the objective is to generate minimum guidance commands to meet the final terminal conditions accurately. In the computation process, minimisation of control is also considered as an important cost factor. To achieve the objectives, the following cost function is considered, which consists of terminal penalty terms and a dynamic control minimisation term.

$$J = \frac{1}{2} S_D (x^* - x_f)^2 + \frac{1}{2}\int_{t_i}^{t_f} \{\alpha(t)^T S_\alpha \alpha(t)\}\, dt \quad (3)$$

where $t_i$, $t_f$ = start and end time, respectively for the optimisation routine (s), $S_D$ = weightage given to final downrange, $S_\alpha$ = weightage given to control variable from $t_i$ to $t_f$, and $x^*$, $x_f$ = desired & final downrange (m) respectively, $\alpha$ = control variable.

The primary objective is to minimise the downrange error at impact point and the secondary objective is minimisation of the control variable, accordingly the weighting factors $S_D$ and $S_\alpha$ are chosen. From the optimal control theory[13,14] the augmented cost function is given as:

$$\bar{J} = \frac{1}{2} S_D (x^* - x_f)^2 + \frac{1}{2}\int_{t_i}^{t_f} \{\alpha(t)^T S_\alpha \alpha(t) + \lambda^T f - \lambda^T \dot{X}\}\, dt \quad (4)$$

The Hamiltonian is defined as:

$$H = L + \lambda^T f = \frac{1}{2}\{\alpha(t)^T S_\alpha \alpha(t) + \lambda^T f\} \quad (5)$$

where $\lambda = (\lambda_1\ \lambda_2\ \lambda_3)^T$ is the Co-state vector (Adjoint state vector).

From the vector calculus, the necessary conditions of optimality are given as follows:

1. State Eqn (1)
2. Co-state equations:

$$\dot{\lambda} = -\frac{\partial H}{\partial X} = -\left(\frac{\partial L}{\partial X} + \left(\frac{\partial f}{\partial X}\right)^T \lambda\right) \quad (6)$$

where $$\frac{\partial f}{\partial X} = \begin{pmatrix} \frac{\partial f_1}{\partial x_1} & \frac{\partial f_1}{\partial x_2} & \frac{\partial f_1}{\partial x_3} \\ \frac{\partial f_2}{\partial x_1} & \frac{\partial f_2}{\partial x_2} & \frac{\partial f_2}{\partial x_3} \\ \frac{\partial f_3}{\partial x_1} & \frac{\partial f_3}{\partial x_2} & \frac{\partial f_3}{\partial x_3} \end{pmatrix}$$

$$\frac{\partial f}{\partial X} = \begin{pmatrix} 0 & \sin\gamma & V\cos\gamma \\ 0 & -\left(\frac{\rho V S_{ref} C_D}{m}\right) & -g\cos\gamma \\ -\frac{V\cos\gamma}{(R_O + h)^2} & \left(\frac{0.5\rho V^2 S_{ref} C_L - F\sin\alpha}{mV^2} + \frac{g\cos\gamma}{V^2} + \frac{\cos\gamma}{r}\right) & \left(\frac{g}{V} - \frac{V}{r}\right)\sin\gamma \end{pmatrix}$$

3. Optimal control equation: $\frac{\partial H}{\partial \alpha} = 0$

$$\frac{\partial (L + \lambda^T f)}{\partial \alpha} = \frac{\partial L}{\partial \alpha} + \left(\frac{\partial f}{\partial \alpha}\right)^T \lambda = 0 \quad (7)$$

$$S_\alpha \alpha - \lambda_1 (V\cos\gamma) + \lambda_2\left(\frac{-T\sin\alpha}{m} + g\cos\gamma\right) + \lambda_3\left(\frac{T\cos\alpha}{mV} - \frac{g\sin\gamma}{V} + \frac{V\sin\gamma}{r}\right) = 0 \quad (8)$$

4. Boundary conditions for $t = t_f$

$$\lambda_f = \frac{\partial H}{\partial X_f} = \frac{1}{2} S_D \frac{\partial [x^* - \{x_{f-1} + dt(V_f \cos\gamma_f)\}]^2}{\partial X_f} \quad (9)$$

$$\lambda_{1f} = \frac{1}{2} S_D \frac{\partial [x^* - \{x_{f-1} + dt(V_f \cos\gamma_f)\}]^2}{\partial r_f} = 0 \quad (10)$$

$$\lambda_{2f} = \frac{1}{2} S_D \frac{\partial [x^* - \{x_{f-1} + dt(V_f \cos\gamma_f)\}]^2}{\partial V_f} = S_D (x^* - x_f)(-dt\cos\gamma_f) \quad (11)$$

$$\lambda_{3f} = \tfrac{1}{2} s_D \frac{\partial [x^* - \{x_{f-1} + dt\,(V_f \cos\gamma_f)\}]^2}{\partial \gamma_f} = s_D (x^* - x_f)\{dt (V_f \sin\gamma_f)\} \tag{12}$$

Note that the selection of the weighting factors $S_D$ and $S_\alpha$ are selected appropriately based on the requirement and initial conditions are obtained from the trajectory. In Gradient method, one starts from an initial point where the function value is calculated and then takes a step in a downward direction such that the function value (cost function) is minimised. Algorithm uses local information and explores the immediate vicinity of the current point i.e. local exploration. A descent step is considered for each iteration and if the iterative scheme converges, the process will end at a stationary point where improvement is negligible. The sequence of steps involved in the process are summarised below:

1. Start with an initial guess control $\alpha°(t)$ where $t_0 \le t \le t_f$.
2. Propagate the states from $t_0$ to $t_f$ using $\alpha°(t)$ with initial conditions $X_0$ (Forward Integration of the system dynamics).
3. Obtain $\lambda(t_f)$ by using the boundary conditions (terminal boundary conditions).
4. Propagate the co-state vector from $t_f$ to $t_0$ using step (3) values as the initial values (Back Integration of the co-state equations).
5. Calculate the gradient $\left(\frac{\partial H}{\partial \alpha}\right)$ from $t_0$ to $t_f - \Delta t$.
6. Calculate the control update as:
   $\alpha(t)^{(k+1)} = \alpha(t)^{(k)} - \tau\left(\frac{\partial H}{\partial \alpha}\right)$
7. Repeat from step ( 2) to step (6) until optimality conditions are met within the specified tolerance.

## 6. SIMULATION STUDIES AND RESULTS

In order to validate the algorithm, following simulation studies are carried out with gravity[15], atmosphere[16], wind[17], and acutator[18] models. To start the algorithm an initial guess control history of zero is considered (it is observed from the studies, the convergence is not affected with initial guess value but the number of iterations taken to achieve the objective within tolerance bounds are increased). Here the assumption is that, the error due to the Navigation and Guidance will lead to a maximum of ±4 km error in the final downrange. The input data considered is shown in Table 1.

The computation routine will terminate, as and when the final error (i.e. the difference between the desired and actual downrange when height is zero) is within the tolerance bounds. To simulate the different cases, $\gamma_i$ is perturbed by a maximum $\pm$0.03° in steps of 0.01°. Different cases considered for simulation studies are shown in Table 2.

**Table 1. Data considered at the start of computation routine**

| | |
|---|---|
| Starting time of the computation routine($t_i$) | 221.472 s |
| Downrange ($x_i$) | 377040 m |
| Height (altitude) ($h_i$) | 193258 m |
| Velocity ($V_i$) | 4381.236 m |
| Desired downrange ($x^*$) | 2656024 m |
| Tolerance bound considered for terminating the optimization routine ($\lvert\varepsilon\rvert$) | 100 m |

**Table 2. Cases considered for simulation**

| | |
|---|---|
| Case 1 | (19.367-0.03)° |
| Case 2 | (19.367-0.02)° |
| Case 3 | (19.367-0.01)° |
| Case 4 | (19.367+0.01)° |
| Case 5 | (19.367+0.02)° |
| Case 6 | (19.367-0.03)° |

From the simulation studies it is found that if $\gamma_i$=19.367°, the free flight trajectory will achieve the intended downrange with an error of 0.138457 km. Therefore for all studies $\gamma_i$=19.367° is taken as initial value. As the flight path angle ($\gamma_i$) is varying from case to case, the free flight trajectory (without any correction) undershoots or overshoots the desired downrange. From the model it is clear that the control variable is the angle of attack ($\alpha$). To achieve the desired downrange from the perturbed trajectory correction is done by adding small velocity, which is provided by a velocity augmentation package (VAP). The VAP is a hardware subsystem (liquid velocity augmentation package), which is mathematically modeled for the numerical simulation with ON delay, rise time, fall time and OFF delay. The operating time of this package for the corresponding downrange error is given by an analytical formula. The formula for calculating the VAP operating duration is given as

$$\text{VAP ON DURATION TIME (Sec)} = \lvert \text{Err} \rvert * 0.25 \tag{13}$$

where Err is the error between the desired and achieved downrange (km)

The injection flight path angle ($\gamma_i$) is perturbed by 0.03° (19367-0.03)°. If the vehicle follows the free flight trajectory the vehicle will not reach the target within acceptable error. For case 1, from the simulation studies (without correction) it is seen that the vehicle achieves a downrange of 2652.272 km as against the desired downrange of 2656.024 km which indicates an error of 3.751 km undershoot. In order to correct the trajectory the gradient based computation technique is used and trajectory is corrected such that the vehicle reaches the target within the specified tolerance bounds. Figure 2 shows the control variable ($\alpha$) wrt time. From the figure it is clear that the VAP has been fired for 0.9378 s. During this period the vehicle is steered to follow a free flight trajectory such that at the end it will reach the target accurately within the specified tolerance bounds. From the Fig. 3 it is evident that the algorithm converges such that the error is driven to zero within few iterations itself.

From the simulation studies, for Case 1 to Case 6,

**For Case 1**

Achieved downrange = 2652.272 km
Error = 3.752 km
VAP operation time = 0.9378 s

**For Case 2**

Achieved downrange = 2652.986 km
Error = 3.0372 km
VAP operation time = 0.7593 s

## Case 1

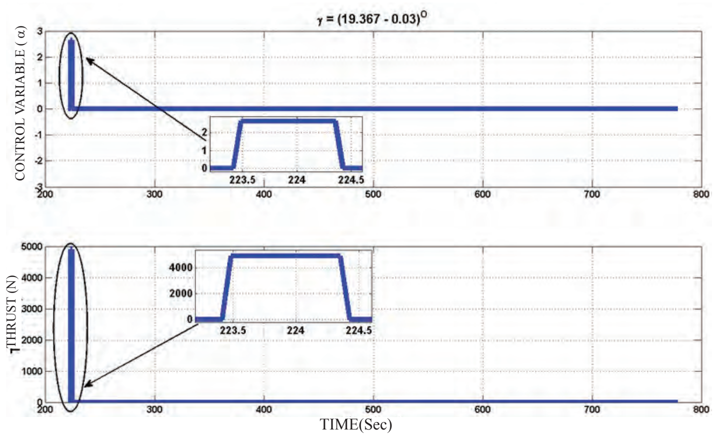


**Figure 2. Time *vs* control variable thrust.**

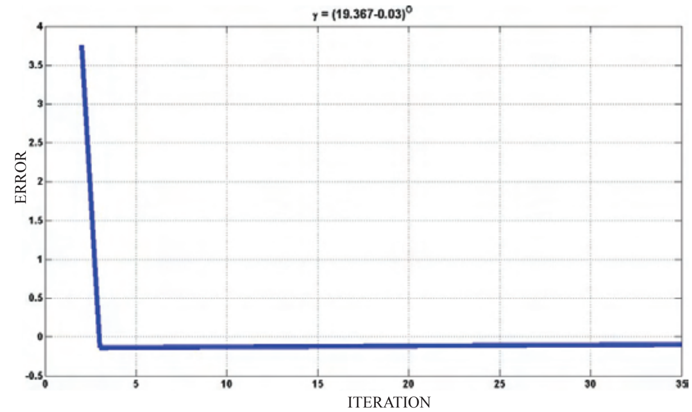


**Figure 3. Iteration *vs* error (km).**

## Case 2

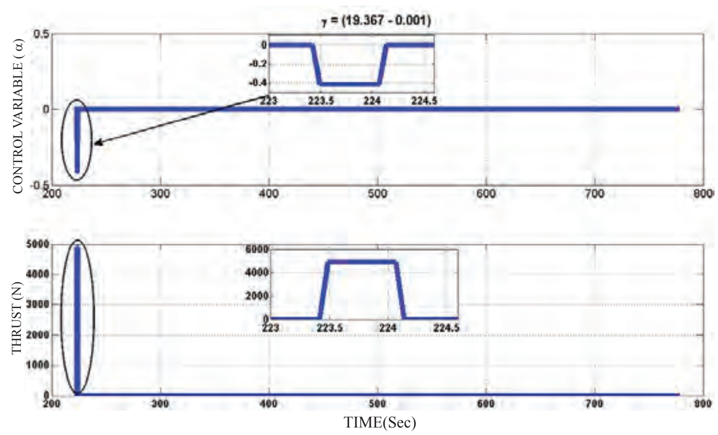


**Figure 4. Time *vs* control variable thrust .**

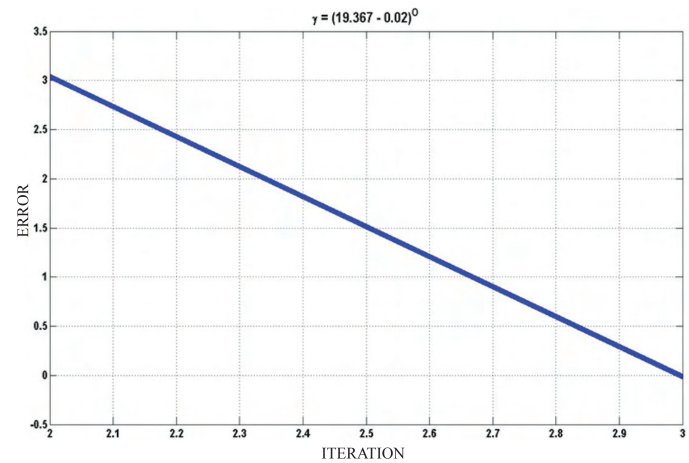


**Figure 5. Iteration *vs* error (km).**

## Case 3

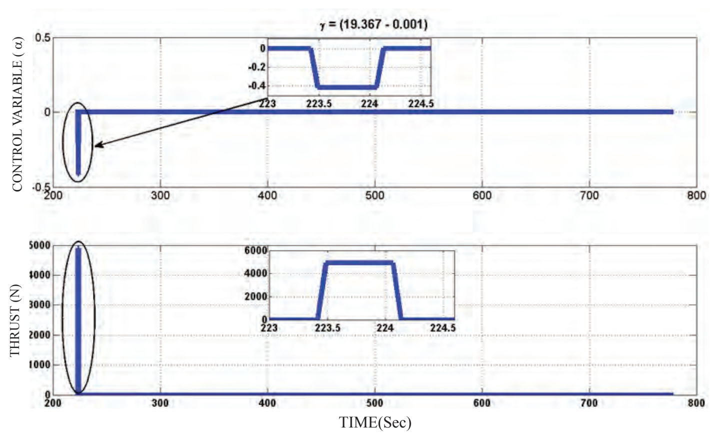


**Figure 6. Time *vs* control variable thrust.**

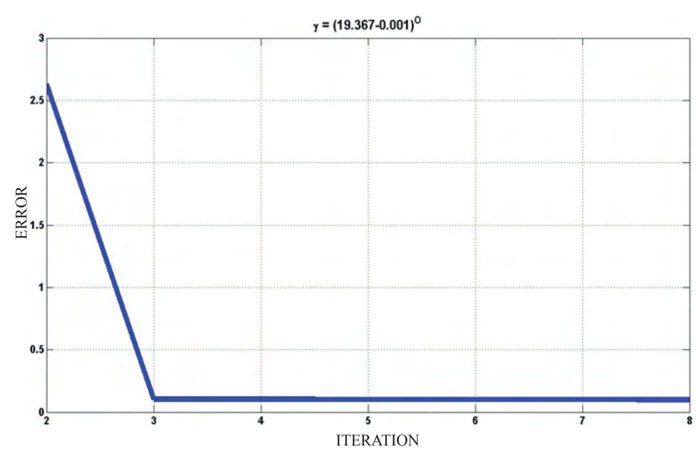


**Figure 7. Iteration *vs* error (km).**

## Case 4

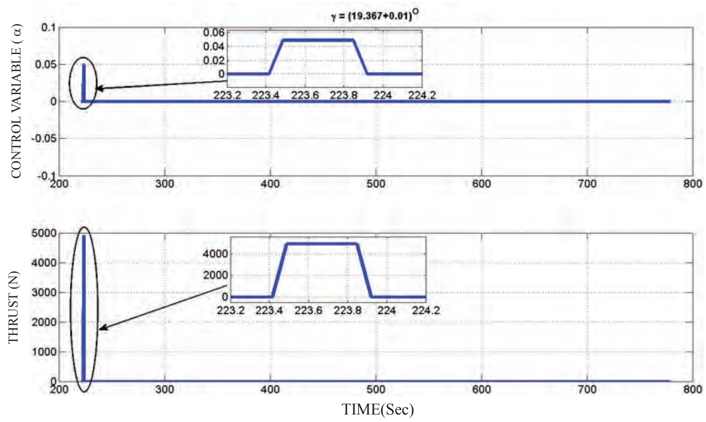


**Figure 8. Time *vs* control variable thrust.**

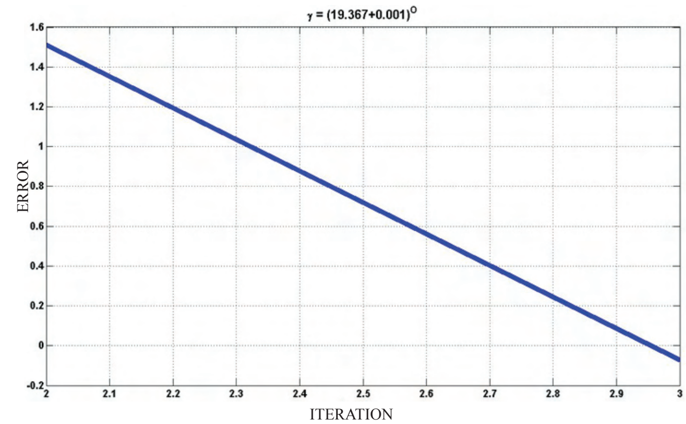


**Figure 9. Iteration *vs* error (km).**

## Case 5

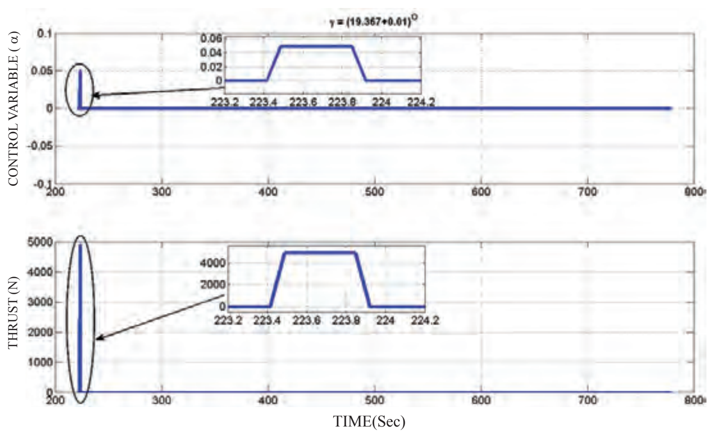


**Figure 10. Time *vs* control variable thrust.**

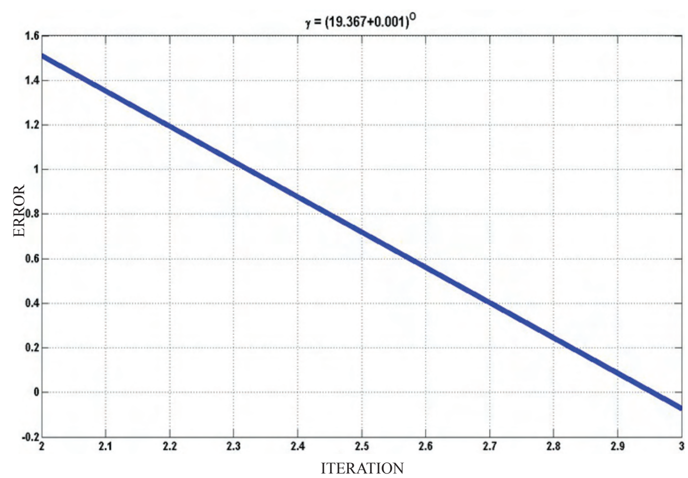


**Figure 11. Iteration *vs* error (km).**

## Case 6

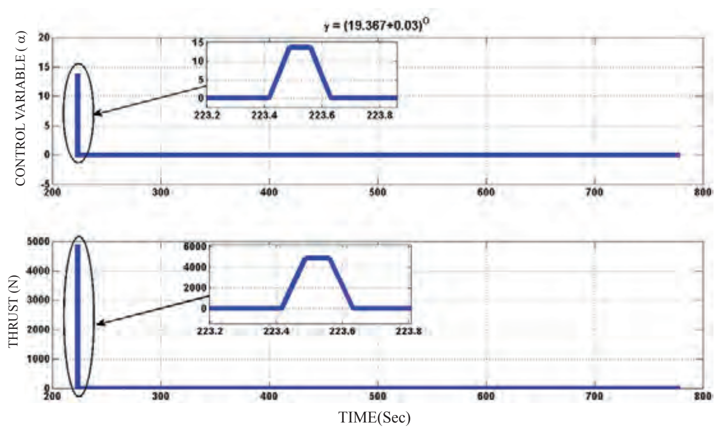


**Figure 12. Time *vs* control variable thrust.**

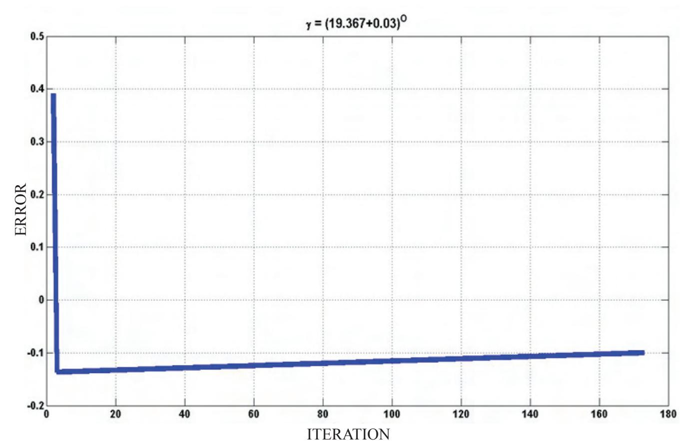


**Figure 13. Iteration *vs* error (km).**

**Table 3. Error in the final achieved downrange**

| | Desired DR (km) | Achieved DR (km) before correction | \|Error\| (km) | Achieved DR (km) after correction | \|Error\| (km) |
|---|---|---|---|---|---|
| Case 1 | 2656.024 | 2652.272 | 3.751 | 2656.123 | -0.099 |
| Case 2 | 2656.024 | 2652.986 | 3.037 | 2656.041 | -0.017 |
| Case 3 | 2656.024 | 2653.393 | 2.630 | 2655.924 | 0.099 |
| Case 4 | 2656.024 | 2654.513 | 1.510 | 2656.096 | -0.072 |
| Case 5 | 2656.024 | 2654.920 | 1.103 | 2655.924 | 0.099 |
| Case 6 | 2656.024 | 2655.633 | 0.390 | 2655.924 | 0.099 |

**For Case 3**
Achieved downrange = 2653.393 km
Error = 2.630 km
VAP operation time = 0.6576 s
**For case 4**
Achieved downrange = 2654.513 km
Error = 1.510 km
VAP operation time = 0.3776 s
**For case 5**
Achieved downrange = 2654.920 km
Error = 1.103 km
VAP operation time = 0.2760 s
**For case 6**
Achieved downrange = 2655.633 km
Error = 0.390 km
VAP operation time = 0.0977s

## 7. CONCLUSIONS

A practically implementable on-line trajectory reshaping algorithm is described here to control the dispersion in the final downrange due to navigation and guidance error at the injection point. The trajectory is corrected on-line using the gradient computation technique, such that the intended payload reaches the target within the specified accuracy (tolerance bound). The robustness of the algorithm is validated by simulation studies with perturbation cases for various burnout conditions (injection conditions).

## REFERENCES


1. Roger, T. Schappell; Michael, L. Salis; Ray, Mueller; Lloyd, E. Best; Albert, J. Bradt; Robert, Harrison & John, H. Burrell. Improved guidance hardware study for the scout launch vehicle. NASA Technical Report, Technical Report No. NASA CR-2029, 1972.
2. Sinha, S.K. & Shrivastava S.K. Optimal explicit guidance of multi stage launch vehicle along three dimensional trajectory. *J. Guid., Control Dyn.*, 1990, **13**(3), 394–403.
3. Walter, Haeussermann. Description and performance of the Saturn launch vehicle's navigation, guidance and control system. NASA Technical Report, Technical Report No. NASA TN D-5869–1970.
4. Helmut, J. Horn. Application of an iterative guidance mode to a lunar landing. NASA Technical Report, Technical Report No. NASA TN D-2967–1965.
5. Ohlmeya, Ernest J.; Pepliton, Thomas R. & Miller, B. Larry. Assessment of integrated GPS/INS for the Ex-171 Extended range guided munition. AIAA-98-4416, pp.1374-1389.
6. Li, Bang-jie & Wang, Ming-hai. Research on the error correction algorithm of Ballistic missiles CNS/INS integrated navigation. *J. Flight Dyn.*, 2006-01, **24**(1), 41-44.
7. Miller, Robin B. A New strapdown attitude algorithm. *Journal Guidance,* 1983, **6**(4), 287-291.
8. Lee, Jang G. & Yoon, Yong J. Extension of strapdown attitude algorithm for high-frequencies base motion. *Journal Guidance*, 1990, **13**(4), 738-743.
9. Wang, L.S.; Chiang, F.R. & Huang, F.T. Integrated Accelerometer/GPS heading estimator. *In* the Proceedings of the 2002 National Technical Meeting of The Institute of Navigation, San Diego, California, 2002, pp. 294-300.
10. Varaprasad, Lukkana & Padhi, Radhakant. Ascent phase trajectory optimization of a generic launch vehicle. *In* the National Systems Conference, No. 126, 2008, IIT-Roorkee. pp. 178-184
11. Betts, John T. Survey of numerical methods for trajectory optmization. *J. Guid., Control Dyn.*, 1998, **22**(2), 193-207.
12. Kreyszig, E. Advanced Engineering Mathematics, John Wiley, 2006.
13. Arthur, E.; Bryson, Jr. & Ho, Yu-Chi. Applied optimal control: Optimisation, estimation and control. 1975.
14. Shrivastava, Shashi Kant & Reddy, M. Narayan. Determination of optimal trajectory under design constraints for a satellite launch vehicle. *Acta Astronautica,* 1976, **3**, 333-47.
15. Department of Defense World Geodetic System.1984, NSN 7643-01-402-0347.
16. Ananthasayanam, M.R. & Narasimha, R. 1979, Standard atmosphere for aerospace applications in India. Dept. of Aerospace Engg. Indian Institute of Science. Report No. 79 FM 5,
17. Roddam, Narasimha. The wind environment in India. National Aeronautical Laboratory. Technical Memorandum Du 8501.

18. Kadam, N.V. Practical design of flight control systems for launch vehicles and missiles. Allied publishers Pvt. Ltd. 2009.

**Contributors**

**Mrs Tessy Thomas** received her BTech (Electrical) from the Calicut University, in 1985, ME (Guided Missiles) from the Poona University, in 1986 and MBA in Operations Management. She is presently working as Scientist 'G' and Project Director at the Advanced Systems Laboratory, Hyderabad. She has received *Agni Award for Excellence in Self-Reliance - 2001, DRDO Award for Path Breaking Research/Outstanding Technology Development - 2007, DRDO Scientist of the Year Award - 2008, Suman Sharma Award - 2009, DRDO Performance Excellence Award - 2011*. Her research area includes: Missile guidance, mission design and trajectory simulation for long range missile systems.

**Dr D.R. Jahagirdar** received his BE (Electronics Eng.) from Govt. College of Engineering, Amravati University, in 1990; MTech (Microwave Eng.) from IIT, Kharagpur, in 1992 and PhD from Department of Electronics and Computer Science from Southampton, UK in 1997. He is presently working as Scientist in Research Center Imarat, DRDO, Hyderabad. He received Professor SK Mitra memorial award for best research oriented paper from IEEE - 2002, *IETE-IRSI Young Scientist Award - 2005, and DRDO Scientist of the Year Award - 2010*.

**Dr I.V. Murali Krishna** received his MTech from IIT Madras and PhD from IISc, Bangalore in 1977. He is presently working as Director, Institute of Science and Technology and Coordinator of Centre for Atmospheric Sciences and Weather Modification Technologies at Jawaharlal Nehru Technological University, Hyderabad. He worked as a Lecturer, Guest Scientist and Professor in many universities. His areas of interest include: Geospatial technology, data mining, soft computing technologies, disaster management, satellite meteorology and weather informatics.

**Mr L. Varaprasad** received his BTech (Electronics & Instrumentation) from Jawaharlal Nehru Technological University, Hyderabad, in 2002 and ME (Aerospace Eng.) from Indian Institute of Science, Bangalore, in 2009. He is presently working as Scientist in Advanced Systems Laboratory, Hyderabad. He is a member of ASI, INSARM. He has been awarded DRDO Lab Scientist of the year - 2012. His areas of interest includes: Missile guidance, trajectory optimisation.