

# An Accurate Guidance Algorithm for Implementation Onboard Satellite Launch Vehicles

M. Marrdonny<sup>1</sup> and M. Mobed<sup>1,\*</sup>

**Abstract.** *An algorithm for guiding a launch vehicle carrying a small satellite to a sun synchronous LEO is presented. Before the launch, a nominal path and the corresponding nominal control law for the entire journey are computed. For each sampling instant during the guided flight, a linear equation approximately relating the differences between the actual and nominal values is considered, and a Least-Squares formula using data from on-line state measurements is applied to compute the actual control. The coefficient matrices of the Least-Squares formula can be determined by off-line computations. The method enjoys simplicity of implementation by onboard computers, as well as robust accuracy against strong winds and uncertainties in thrust magnitude.*

**Keywords:** *Disturbance rejection; Guidance systems; Satellites; Optimal trajectory; Launch vehicle.*

## INTRODUCTION

Artificial satellites are placed into Earth orbits using missiles known as “carrier rockets or launch vehicles”. Some of the issues concerning commercial launch vehicles have been addressed in [1,2]. Today, launching satellites has become a vast and multifaceted endeavor with research continuing on its various aspects including rocket guidance.

In a previous work of the authors, a guidance algorithm was proposed in which the so-called path-to-go and the corresponding control were recomputed at the juncture at which the rocket left the atmosphere and entered the exoatmosphere where atmospheric disturbances were supposed to be negligible or nonexistent [3]. The main attribute of the recomputed path and control was that they were used as the nominal trajectory to be followed throughout the rest of the flight, and no further corrections were introduced once the new path-to-go and control became available. The algorithm was shown through simulations to work very well in the absence of exoatmospheric disturbances. However, since disturbances of various kinds do exist during all phases of the flight, it stands to reason

to expect that even during the exoatmospheric flight, corrections to the control could be used to improve precision. The present paper aims at continually providing such corrections in a manner as efficiently as possible from the standpoint of the computational burden imposed upon the onboard computer. The term “computational burden” has been used rather loosely here to refer to any one or more of the concepts of computational effort, computational complexity, computational load etc. Given the nature of the problem under discussion, the number of sampling periods needed to obtain one sample of the control signal has been used as a measure of the computational burden. In the algorithm presented in this article, the control signal to be applied during the  $k$ th sampling period is computed during the  $(k - 1)$ th, according to a myopic policy, and this continues throughout the exoatmospheric phase until the moment of orbit insertion. As such, it takes only one sampling period to obtain each sample of the control signal. This is to be compared with [3] where several sampling periods were needed to complete the updating computations, which were done only once at the beginning of the exoatmospheric phase.

Guiding a satellite launch vehicle traditionally involves an open-loop phase for the endoatmospheric part, and a closed-loop phase for the exoatmospheric part of the flight. Moreover, it is sometimes helpful

1. School of Electrical Engineering, Sharif University of Technology, Tehran, P.O. Box 11155-9363, Iran.

\*. Corresponding author. E-mail: mobed@sharif.edu

Received 4 October 2009; received in revised form 18 April 2010;  
accepted 17 July 2010

to distinguish between the trajectory selection and the control issues of guidance. The trajectory, the control, or both, may be optimal or near-optimal, with respect to appropriate criteria. In this regard, the Linear Tangent Law (LTL) has been shown to have certain desirable properties [4,5].

Guidance computations fall into two categories: Off-line and on-line. "Off-line" refers to computations performed before the take-off of the launch vehicle. Usually, in off-line computations, an optimal or near-optimal trajectory and control law are found for the entire rocket mission before it leaves the launch pad. The trajectory thus determined is treated as the nominal or reference trajectory, and the corresponding control law may be applied to track and keep the vehicle near the reference trajectory during the guided flight. In finding the nominal trajectory and control, off-line computations must also take into account the structural limitations of the rocket's fuselage, which is longitudinally strong and laterally weak. This can be accomplished either by introducing constraints into the underlying thrust-controlled optimization, as in for example [6-9], or by using pitch-over and gravity-turn maneuvers, as in [10]. References [6,11-15] discuss some of the methods of computing the optimal trajectory. Methods of tracking the reference trajectory can be found in [7,8,11,14]. Some of the methods of implementing entirely closed-loop guidance laws have been discussed in [9,15-17]. These methods involve on-line computations.

The term "on-line" refers to computations performed by an onboard computer after the take-off. On-line guidance algorithms tend to be computationally demanding and challenging to implement on small- to medium-powered computers.

An important consideration, especially in the case of on-line guidance, is the availability of computation results in a short, definite time. However, convergence does not always hold in computations involving infinite sequences of iterations. Even in cases where convergence does hold, its rate may be difficult to determine or estimate. Thus, algorithms that employ infinite sequences of iterations for on-line computations seem to be of limited interest.

It is now possible to summarize the overall scheme as follows: Off-line computations determine a nominal trajectory-and-control pair for the entire launch mission by using gravity-turn and LTL, respectively, for the endoatmospheric and exoatmospheric phases. During the vertical launch and the pitch-over maneuver, control is open-loop and it is as found previously by the off-line computations. The nominal trajectory remains in force for the entire flight period. However, control becomes closed-loop during the gravity-turn maneuver, and it stays closed-loop throughout the rest of the flight. The closed-loop control at each sampling

instant is realized via a gain matrix calculated by Least-Squares methods devised to minimize error at the next sampling instant.

The paper has been organized in seven sections. First three dimensional model of the launch vehicle is described. Then, the nominal flight path and the proposed guidance algorithm for closed-loop operation are introduced and its convergence is studied. Following that, simulation results of the proposed algorithm for a two-stage launch vehicle are discussed and, finally, concluding remarks are given.

## PHYSICAL MODEL

Models of the launch vehicle, gravitational field and atmosphere are exactly the same as those used in [3] from which this section has been taken almost verbatim and presented here for ease of reference.

A two-stage launch vehicle is considered. Each stage is powered by a liquid-propellant engine. The engines cannot be throttled. However, Thrust Vector Control (TVC) is possible. By design, the first stage separates when its fuel is depleted and the second stage ignites. The objective is to insert a research satellite with a mass of 10-kg into a sun synchronous low Earth orbit at a height of 200 km. The three-degree-of-freedom equations of motion relative to the Earth are [18]:

$$\dot{\lambda} = \frac{V \cos \gamma \cos \psi}{r \cos \tau}, \quad (1)$$

$$\dot{\tau} = \frac{V \cos \gamma \sin \psi}{r}, \quad (2)$$

$$\dot{h} = V \sin \gamma, \quad (3)$$

$$\begin{aligned} \dot{V} = & \frac{1}{m}(T \cos \alpha - D - mg \sin \gamma \\ & + w_E \cos \gamma \cos \psi + w_N \cos \gamma \sin \psi + w_U \sin \gamma) \\ & + r\omega^2 \cos \tau (\cos \tau \sin \gamma - \sin \tau \cos \gamma \sin \psi), \end{aligned} \quad (4)$$

$$\begin{aligned} \dot{\gamma} = & \frac{1}{mV}[(T \sin \alpha + L) \cos \mu - mg \cos \gamma \\ & - w_N \sin \gamma \sin \psi - w_E \sin \gamma \cos \psi + w_U \cos \gamma] \\ & + \frac{V \cos \gamma}{r} + 2\omega \cos \tau \cos \psi \\ & + \frac{r\omega^2}{V} \cos \tau (\cos \tau \cos \gamma + \sin \tau \sin \gamma \sin \psi), \end{aligned} \quad (5)$$

$$\begin{aligned} \dot{\psi} = & -\frac{1}{mV \cos \gamma} [(T \sin \alpha + L) \sin \mu + w_E \sin \psi \\ & - w_N \cos \psi] - \frac{V}{r} \tan \tau \cos \gamma \cos \psi \\ & + 2\omega(\cos \tau \tan \gamma \sin \psi - \sin \tau) \\ & - \frac{r\omega^2}{V \cos \gamma} \cos \tau \sin \tau \cos \psi, \end{aligned} \quad (6)$$

$$\dot{m} = -c_i. \quad (7)$$

In Equations 1-7,  $\lambda$  is the longitude,  $\tau$  is the latitude,  $h$  is the altitude above mean sea level,  $V$  is the velocity,  $\gamma$  is the flight path angle,  $\psi$  is the heading angle,  $m$  is the sum of the masses of the launch vehicle and the payload,  $r$  is the distance between the launch vehicle and the Earth centre,  $g$  is the local gravitational acceleration,  $\omega$  is the rotational velocity of Earth,  $\alpha$  is the angle of attack,  $\mu$  is the velocity roll angle, and  $T$ ,  $D$ ,  $L$  are the thrust, drag and lift magnitudes, respectively. Furthermore,  $w_E$ ,  $w_N$  and  $w_U$  are wind force components in the east, north and up directions. Thus, wind force components have been introduced explicitly in the equations and not implicitly through aerodynamic forces. This allows for a more direct evaluation of the algorithm's performance against wind disturbances. Figure 1 depicts the coordinate frame in which the equations have been written.

The thrust model is given by:

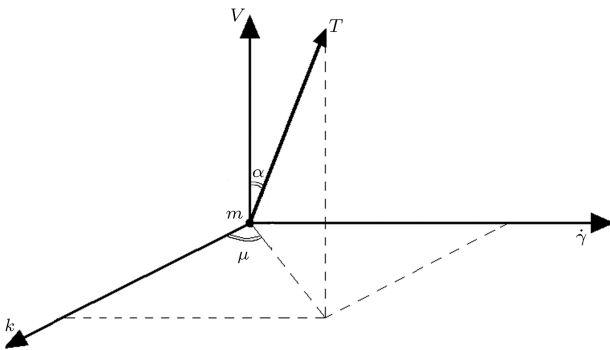
$$T = T_{\text{vac}} - pA_e, \quad (8)$$

where  $T_{\text{vac}}$  is the engine thrust in vacuum,  $p$  is atmospheric pressure, and  $A_e$  is the exit area of the engine.

The air density and pressure are modeled as the following exponential functions of altitude [18]:

$$\rho = \rho_s \exp\left(-\frac{h}{h_1}\right), \quad (9)$$

$$p = p_s \exp\left(-\frac{h}{h_2}\right), \quad (10)$$



**Figure 1.** Coordinate frame.

where  $\rho_s$  and  $p_s$  are the air density and pressure at sea level,  $\rho$  and  $p$  are those at altitude  $h$ , and  $h_1$  and  $h_2$  are positive constants. The Earth is assumed to be a rotating, spherical body with an inverse squared law of gravity field. The speed of sound is given using the perfect gas law:

$$a = \sqrt{\beta \frac{p}{\rho}}, \quad (11)$$

where  $a$  is the speed of sound and  $\beta$  is the ratio of specific heat of air. The dynamic pressure,  $q$ , the drag,  $D$ , and the lift,  $L$ , are given by:

$$q = \frac{1}{2} \rho V^2, \quad (12)$$

$$D = q A_b C_D, \quad (13a)$$

$$L = q A_b C_L. \quad (13b)$$

In the above,  $C_D$  is the drag coefficient,  $C_L$  is the lift coefficient, and  $A_b$  is the aerodynamic reference area of the launch vehicle. The drag and lift coefficients,  $C_D$  and  $C_L$ , are modeled as polynomials in  $\alpha$  during the first stage:

$$\begin{aligned} C_D = & C_{D0}(M) + C_{D1}(M)\alpha + C_{D2}(M)\alpha^2 \\ & + C_{D3}(M)\alpha^3, \end{aligned} \quad (14)$$

$$C_L = C_{L1}(M)\alpha. \quad (15)$$

Wind force components are given by:

$$w_E = \frac{1}{2} \rho \nu_E^2 A_L C_D, \quad (16a)$$

$$w_N = \frac{1}{2} \rho \nu_N^2 A_L C_D, \quad (16b)$$

$$w_U = \frac{1}{2} \rho \nu_U^2 A_L C_D, \quad (16c)$$

where  $\nu_E$ ,  $\nu_N$  and  $\nu_U$  are the east, north and up components of the wind speed, and  $A_L$  is the lateral cross section area of the launcher.

After staging, the vehicle will fly at hypersonic velocities and aerodynamic force coefficients will be independent of the Mach number. The aerodynamic models for the second stage are:

$$C_D = C_{D0} + C_{D2}\alpha^2, \quad (17)$$

$$C_L = C_{L1}\alpha + C_{L2}\alpha|\alpha|. \quad (18)$$

The drag and lift coefficients in Equations 14, 15, 17 and 18 are obtained by applying interpolation techniques to tabular data [19]. In the dense part of the

**Table 1.** Numerical value of the launch parameters.

Parameter	Symbol	Value	Unit
Mass burning rate in the first stage	$c_i$	120	kg/s
Mass burning rate in the second stage	$c_i$	12	kg/s
Thrust magnitude in the first stage	$T_{vac}$	352800	N
Thrust magnitude in the second stage	$T_{vac}$	35280	N
Initial mass of launcher and payload	$m(t_0)$	28174	kg
Mass of launcher and payload after separation of the first stage		504	kg
Mass of fuel of the first stage		23880	kg
Mass of fuel of the second stage		420	kg
Payload (satellite) mass		10	kg
Exit velocity of gases from launcher motor		2940	m/s
Air density constant	$h_1$	8240	m
Air pressure constant	$h_2$	8240	m
Orbit altitude (nominal value)	$h(T)$	200	km
Earth's radius		6377940	m
Earth's rotation rate	$\omega$	$7.2722 \times 10^{-5}$	rad/s
Staging time		199	s
Engine exit area	$A_e$	2.7	m <sup>2</sup>
Lateral cross sectional area of launcher	$A_L$	15.0	m <sup>2</sup>
Cross sectional area of launcher	$A_b$	5.0	m <sup>2</sup>
Air density at sea level	$\rho_s$	1.29	kg/m <sup>3</sup>
Air pressure at sea level	$p_s$	101326	Pa
Ratio of specific heats of air	$\beta$	1.4	-
Constant in Equation 26	$\omega_p$	0.05	Rad/s
End of vertical launch time	$t_v$	20	s
Burnout time		234	s
End of pitch-over time		23	s
End of gravity-turn maneuver		165	s
Gravitational acceleration at sea level		9.78	m/s <sup>2</sup>

atmosphere, the thrust and velocity vectors are almost aligned; therefore, the pitching moments are assumed negligible for both stages. Numerical values for the launch vehicle parameters are listed in Table 1.

The control variables are the angle of attack  $\alpha$  and the velocity roll angle  $\mu$ . Therefore, the guidance system must have two outputs. As a matter of fact, in a model slightly more elaborate than Equations 1 to 7,  $\alpha$  and  $\mu$  would themselves appear as state variables rather than controls. But, even then,  $\alpha$  and  $\mu$  would have relatively fast dynamics and, thus, a good approximation may be obtained by considering them as control variables.

The singularities in Equations 5 and 6, due to

$V = 0$  at the take-off instant and  $\gamma = 90$  deg during the vertical launch, are removed by modifying the equations of motion for the vertical launch segment, as follows:

$$\dot{\lambda} = 0, \quad (19)$$

$$\dot{\tau} = 0, \quad (20)$$

$$\dot{h} = V, \quad (21)$$

$$\dot{V} = \frac{1}{m}(T - D - mg) + r\omega^2, \quad (22)$$

$$\dot{\gamma} = 0, \quad (23)$$

$$\dot{\psi} = 0, \quad (24)$$

$$\dot{m} = -c_i. \quad (25)$$

### NOMINAL PATH

The target orbit is a near-circular sun synchronous LEO at an altitude of 200 km above mean sea-level. The launch site is at latitude of 35 deg. The launch program consists of four segments in chronological order, as follows:

- **Segment 1** is the vertical flight. During this segment, control variables are kept at zero and the vehicle gains both altitude and speed. Take off occurs at a time instant when the launch site reaches the target orbit plane. This yields a planar nominal trajectory. Numerical values for the duration of Segment 1 and the altitude and speed at its end are, respectively, 20 s, 430 m and 44 m/s.
- **Segment 2** is the pitch-over maneuver during which the angle of attack is increased slightly from zero according to the following equation:

$$\alpha_{\text{nom}}(t) = \frac{\pi}{2} - \gamma_{\text{nom}}(t) - \omega_p(t - t_\nu). \quad (26)$$

The flight path angle decreases slightly and by the end of this segment it will be less than, but very close to, 90 deg. to prepare the vehicle for its next maneuver. The pitch-over maneuver lasts for 3 seconds. Numerical values for the time-independent constants,  $\omega_p$  and  $t_\nu$ , are determined by trial-and-error through simulation.

- **Segment 3** is the gravity-turn maneuver during which the control variables,  $\alpha$  and  $\mu$ , are kept at zero to eliminate lateral aerodynamic forces and moments. Flight path angle,  $\gamma$ , decreases, and the vehicle completes its journey across the dense part of the atmosphere [10]. By the end of Segment 3, the vehicle will have gained altitude, velocity and a flight path angle sufficient for the insertion maneuver to begin.
- **Segment 4** begins as soon as the vehicle leaves the atmosphere. Here, the aerodynamic forces and moments are negligible, and control through  $\alpha$  and  $\mu$  becomes possible. Off-line computations corresponding to this segment use, through the following equation, the Linear Tangent Law (LTL) of guidance for its simplicity, effectiveness and near optimality:

$$\tan[\alpha_{\text{nom}}(t) + \gamma_{\text{nom}}(t)] = at + b. \quad (27)$$

Parameters  $a$  and  $b$  are determined by satisfying two final conditions of the powered flight, namely, final

height and final flight path angle:

$$h_{\text{nom}}(T) = 200,000 \text{ m},$$

$$\gamma_{\text{nom}}(T) = 0 \text{ deg.} \quad (28)$$

Final time  $T$  is the fuel depletion time of the second stage engine. The nominal value of  $\mu(t)$  is identically zero throughout the powered flight,  $\mu_{\text{nom}}(t) \equiv 0$ , because the launch site and hence the entire nominal trajectory are both on the target orbit plane. Using Newton's method, the numerical values of  $a$  and  $b$  of Equation 27 have been found to be  $a = -0.022739 \text{ s}^{-1}$  and  $b = 4.657243$ . This concludes the last segment of off-line computations for nominal guidance.

### CLOSED-LOOP GUIDANCE ALGORITHM

The nominal path and its computation described in the previous section refer to ideal conditions where no disturbances are present. In reality, disturbances can distract the vehicle and even divert it from the nominal plane, resulting in nonplanar motion. Thus, in order to insert the satellite into its prescribed orbit as precisely as possible, both variables  $\alpha$  and  $\mu$  should be used for control purposes, particularly during the exoatmospheric phase. A control law meeting this requirement will be devised directly in discrete time. However, it would be helpful to first write the original continuous-time state-space equations more compactly in a vector form as follows:

$$\begin{aligned} \dot{\mathbf{x}}(t) &= \phi[\mathbf{x}(t), \mathbf{u}(t), \mathbf{n}(t)], \\ \dot{\mathbf{x}}_{\text{nom}}(t) &= \phi[\mathbf{x}_{\text{nom}}(t), \mathbf{u}_{\text{nom}}(t), \mathbf{n}_{\text{nom}}(t)], \\ \mathbf{y}(t) &= \ell[\mathbf{x}(t)], \\ \mathbf{y}_{\text{nom}}(t) &= \ell[\mathbf{x}_{\text{nom}}(t)], \end{aligned} \quad (29)$$

where  $\mathbf{n}_{\text{nom}}(t) \equiv 0$  and the subscript “nom” everywhere refers to the nominal path. The state vector is  $\mathbf{x} = [\lambda \ \tau \ h \ V \ \gamma \ \psi \ m]'$ , the control vector is  $\mathbf{u} = [\alpha \ \mu]'$ , the disturbance vector is  $\mathbf{n} = [w_E \ w_N \ w_D]'$ ,  $\phi : \mathbb{R}^{12} \rightarrow \mathbb{R}^7$  and  $\ell : \mathbb{R}^7 \rightarrow \mathbb{R}^3$  are fixed known nonlinear functions, and  $\mathbf{y} \triangleq [h \ \gamma \ (\cos \psi)(\cos \tau)]'$  is the output vector in which  $(\cos \psi)(\cos \tau) = \cos i$  and  $i$  is the orbit inclination. Function  $\phi$  is a composition of functions given by Equations 1-7 and several others, such as  $r$ ,  $T$ ,  $C_L$ ,  $C_D$ ,  $p$ ,  $\rho$ ,  $q$  etc. Disturbance-induced perturbations in state, output and control are given by:

$$\begin{aligned} \Delta \mathbf{x}(t) &\triangleq \mathbf{x}(t) - \mathbf{x}_{\text{nom}}(t), \\ \Delta \mathbf{y}(t) &\triangleq \mathbf{y}(t) - \mathbf{y}_{\text{nom}}(t), \\ \Delta \mathbf{u}(t) &\triangleq \mathbf{u}(t) - \mathbf{u}_{\text{nom}}(t). \end{aligned} \quad (30)$$

Because of structural constraints, the angle of attack during Segment 3 is kept near zero, hence, the guidance algorithm is almost open-loop during this segment, which is a substantial part of the path. The impact of disturbances is compensated mainly in Segment 4.

The analysis and design pertaining to Segment 4 are in discrete time. For the continuous-time processes of state  $\mathbf{x}(t)$ , output  $\mathbf{y}(t)$ , control  $\mathbf{u}(t)$  and disturbance  $\mathbf{n}(t)$ , the notations  $\mathbf{x}[k]$ ,  $\mathbf{y}[k]$ ,  $\mathbf{u}[k]$  and  $\mathbf{n}[k]$  will be used to represent  $\mathbf{x}(t_k)$ ,  $\mathbf{y}(t_k)$ ,  $\mathbf{u}(t_k)$  and  $\mathbf{n}(t_k)$  where  $t_k = k\Delta t$ ,  $\Delta t$  is the fixed known sampling period, and  $k$  is an integer. At each sampling time during Segment 4, input vector  $\mathbf{u}$  is determined so as to minimize a metric between  $\mathbf{y}[k+1]$  and  $\mathbf{y}_{\text{nom}}[k+1]$ . To that end,  $\mathbf{y}[k+1]$  is first written as function  $\mathbf{F}$  of state vector  $\mathbf{x}[k]$ , control vector  $\mathbf{u}[k]$  and disturbance vector  $\mathbf{n}[k]$ :

$$\mathbf{y}[k+1] = \mathbf{F}(\mathbf{x}[k], \mathbf{u}[k], \mathbf{n}[k]). \quad (31)$$

On linearizing  $\mathbf{F}$  in Equation 31, by using Jacobians evaluated at the nominal path and using discrete-time versions of Equation 30, one arrives at the following expression for  $\Delta\mathbf{y}[k+1]$  as a linear function of  $\Delta\mathbf{x}[k]$ ,  $\Delta\mathbf{u}[k]$  and  $\mathbf{n}[k]$ :

$$\begin{aligned} \Delta\mathbf{y}[k+1] &= \frac{\partial\mathbf{F}}{\partial\mathbf{x}}(\mathbf{Z}(k))\Delta\mathbf{x}[k] + \frac{\partial\mathbf{F}}{\partial\mathbf{u}}(\mathbf{Z}(k))\Delta\mathbf{u}[k] \\ &+ \frac{\partial\mathbf{F}}{\partial\mathbf{n}}(\mathbf{Z}(k))\mathbf{n}[k], \end{aligned} \quad (32)$$

where  $\mathbf{Z}(k) \triangleq (\mathbf{x}_{\text{nom}}[k], \mathbf{u}_{\text{nom}}[k], \mathbf{0})$  denotes the nominal path and controls.

On replacing the zero-mean random quantity  $\mathbf{n}[k]$  in Equation 32 by its mean, that equation is further simplified as follows:

$$\Delta\mathbf{y}[k+1] = \frac{\partial\mathbf{F}}{\partial\mathbf{x}}(\mathbf{Z}(k))\Delta\mathbf{x}[k] + \frac{\partial\mathbf{F}}{\partial\mathbf{u}}(\mathbf{Z}(k))\Delta\mathbf{u}[k]. \quad (33)$$

Designing the control law is accomplished through minimization of errors. To compute  $\mathbf{u}[k]$ , the control to be applied at time  $t$ , one seeks to minimize, by an appropriate choice of  $\Delta\mathbf{u}[k]$ , the quantity

$$\|\Delta\mathbf{y}[k+1]\|^2 = \left\| \frac{\partial\mathbf{F}}{\partial\mathbf{x}}\Delta\mathbf{x}[k] + \frac{\partial\mathbf{F}}{\partial\mathbf{u}}\Delta\mathbf{u}[k] \right\|^2,$$

a measure of error at time  $t$ . However, the matrix  $\frac{\partial\mathbf{F}}{\partial\mathbf{u}}(\mathbf{Z}(k))$  may in some cases be ill-conditioned or singular. There is also the question of differences among the output variables in their orders of magnitude, importance and dimension. Thus, instead of minimizing  $\|\Delta\mathbf{y}[k+1]\|^2$ , one minimizes  $\|\mathbf{Q}\Delta\mathbf{y}[k+1]\|^2 + \|r_u\Delta\mathbf{u}[k]\|^2$  by choosing  $\Delta\mathbf{u}[k]$  where  $r_u$  is a Tikhonov regularization parameter and  $\mathbf{Q}$  a scaling

matrix. Treating this as a Least-Squares problem, the solution is given in terms of  $\Delta\mathbf{x}[k]$  as:

$$\Delta\mathbf{u}[k] = -(\mathbf{A}'[k]\mathbf{A}[k] + r_u^2\mathbf{I})^{-1}\mathbf{A}'[k]\mathbf{B}[k]\Delta\mathbf{x}[k], \quad (34)$$

where:

$$\mathbf{A}[k] = \mathbf{Q}\frac{\partial\mathbf{F}}{\partial\mathbf{u}}(\mathbf{Z}(k)), \quad (35a)$$

$$\mathbf{B}[k] = \mathbf{Q}\frac{\partial\mathbf{F}}{\partial\mathbf{x}}(\mathbf{Z}(k)). \quad (35b)$$

The Jacobians in Equations 35 are computed off-line and numerically for every sampling time. Vector  $\Delta\mathbf{x}[k]$  is computed through  $\Delta\mathbf{x}[k] = \mathbf{x}[k] - \mathbf{x}_{\text{nom}}[k]$  where, of course, nominal state  $\mathbf{x}_{\text{nom}}[k]$  is available from off-line computations and the actual state,  $\mathbf{x}[k]$ , is measured during the flight using various sensors and measurement devices. The  $\Delta\mathbf{x}[k]$  thus found is substituted in Equation 34 and  $\Delta\mathbf{u}[k]$  is, thereby, computed. The control to be applied at time  $k$  is obtained from  $\mathbf{u}[k] = \Delta\mathbf{u}[k] + \mathbf{u}_{\text{nom}}[k]$  where, again, the nominal control is known from off-line computations. In consideration of structural limitations, the control variables are restricted to  $\pm 1$  deg while the launch vehicle is inside the atmosphere. The scaling matrix was taken to be  $\mathbf{Q} = \text{diag}[10^{-5} \ 1 \ 1]$  in simulations.

## AN ANALYSIS OF ERROR

The results presented in the next section show how the closed-loop guidance algorithm succeeds in reducing, practically to zero, by the time of orbit insertion, the error between actual and nominal quantities of interest, which consist of altitude,  $h$ , flight path angle,  $\gamma$ , and orbit inclination,  $i$ . This is essentially the question of convergence to zero of the difference between actual and nominal outputs:  $\lim_{t \rightarrow \infty} \Delta\mathbf{y}(t) = 0$ , where  $\Delta\mathbf{y}(t) = \mathbf{y}(t) - \mathbf{y}_{\text{nom}}(t)$ . Clearly, one would like such convergence to hold not just for a few simulation runs corresponding to a few choices of initial state errors, but for all possible initial state errors,  $\Delta\mathbf{x}(0)$ , where  $\Delta\mathbf{x}(0) = \mathbf{x}(0) - \mathbf{x}_{\text{nom}}(0)$ . Thus, one would ideally like to satisfy the following property or condition:

$$\lim_{t \rightarrow \infty} \Delta\mathbf{y}(t) = 0 \quad \forall \Delta\mathbf{x}(0) \in B, \quad (36)$$

where  $B$  is an open ball of nonzero radius centered at the origin of  $\mathbb{R}^7$ . The time  $t = 0$  in  $\Delta\mathbf{x}(0)$  in Equation 36, refers to the time of leaving the atmosphere after which there will be no forces of atmospheric disturbance. An analogy exists between Condition 36 and the definition of global asymptotic stability for a nominal solution of a dynamical system. However, there exists also an important distinction between the two. Condition 36 is concerned with

the convergence of  $\Delta \mathbf{y}(t)$ , whereas the said notion of stability traditionally addresses the convergence of  $\Delta \mathbf{x}(t) = \mathbf{x}(t) - \mathbf{x}_{\text{nom}}(t)$ . Whether Condition 36 is regarded as a condition of convergence or of stability, and proposing such distinctions do not seem to be particularly fruitful anyway, it is hard to achieve an analytical proof of Condition 36, even with strong *ad hoc* assumptions. Therefore, we shall settle for the modest goal of computationally verifying a more tractable, yet important aspect of Condition 36, as explained in the sequel.

To begin, a discrete-time version of Equations 29 is obtained for the actual trajectory by applying Euler's forward approximation,  $\dot{\mathbf{x}} \approx \frac{\mathbf{x}(t+\Delta t) - \mathbf{x}(t)}{\Delta t}$ , at each epoch of discrete time,  $t = t_k = k\Delta t$ :

$$\begin{aligned} \frac{\mathbf{x}[k+1] - \mathbf{x}[k]}{\Delta t} &= \phi(\mathbf{x}[k], \mathbf{u}[k], \mathbf{n}[k]), \\ \mathbf{y}[k] &= \ell(\mathbf{x}[k]). \end{aligned} \quad (37)$$

The approximation sign in Equations 37 may be replaced by an equality sign in order to get the following set of state and output equations in discrete time:

$$\begin{aligned} \mathbf{x}[k+1] &= \mathbf{\Omega}(\mathbf{x}[k], \mathbf{u}[k], \mathbf{n}[k]), \\ \mathbf{y}[k+1] &= \mathbf{F}(\mathbf{x}[k], \mathbf{u}[k], \mathbf{n}[k]), \end{aligned} \quad (38)$$

where:

$$\begin{aligned} \mathbf{\Omega}(\mathbf{x}, \mathbf{u}, \mathbf{n}) &\stackrel{\text{def}}{=} \Delta t \cdot \phi(\mathbf{x}, \mathbf{u}, \mathbf{n}), \\ \mathbf{F}(\mathbf{x}, \mathbf{u}, \mathbf{n}) &\stackrel{\text{def}}{=} \ell(\mathbf{\Omega}(\mathbf{x}, \mathbf{u}, \mathbf{n})), \end{aligned} \quad (39)$$

or  $\mathbf{F} = \ell \circ \mathbf{\Omega}$ . Function  $\mathbf{F}$  in Equations 38 and 39 is the same as the one already encountered in Equation 31. Needless to say,  $\mathbf{x}[k]$  and  $\mathbf{y}[k]$  in Equations 38 and 39 and all subsequent equations only approximately represent  $\mathbf{x}(k\Delta t)$  and  $\mathbf{y}(k\Delta t)$ , respectively. Such approximation is a result of having substituted in Equation 37 the  $\approx$  sign by  $=$  and used, subsequently, in Equation 38,  $\mathbf{x}[k]$ , which is an approximation to  $\mathbf{x}(k\Delta t)$  instead of  $\mathbf{x}(k\Delta t)$  itself. For the discretization approximation to be good,  $\Delta t$  must be sufficiently small. A suitable value for  $\Delta t$  is determined through trial-and-error in simulations. Discrete-time equations of the nominal path are obtained in a like manner from Equations 29 and they correspond to  $\mathbf{n}(t) = \mathbf{n}[k] = 0$ . They are:

$$\begin{aligned} \mathbf{x}_{\text{nom}}[k+1] &= \mathbf{\Omega}(\mathbf{x}_{\text{nom}}[k], \mathbf{u}_{\text{nom}}[k], 0), \\ \mathbf{y}_{\text{nom}}[k+1] &= \mathbf{F}(\mathbf{x}_{\text{nom}}[k], \mathbf{u}_{\text{nom}}[k], 0). \end{aligned} \quad (40)$$

The idea is to consider, as state and output errors, the quantities  $\Delta \mathbf{x}[k] = \mathbf{x}[k] - \mathbf{x}_{\text{nom}}[k]$  and  $\Delta \mathbf{y}[k] = \mathbf{y}[k] - \mathbf{y}_{\text{nom}}[k]$  instead of, respectively,  $\Delta \mathbf{x}(t)$  and  $\Delta \mathbf{y}(t)$ , alluded to before:

$$\begin{aligned} \Delta \mathbf{x}[k+1] &= \mathbf{x}[k+1] - \mathbf{x}_{\text{nom}}[k+1] \\ &= \mathbf{\Omega}(\mathbf{x}[k], \mathbf{u}[k], \mathbf{n}[k]) \\ &\quad - \mathbf{\Omega}(\mathbf{x}_{\text{nom}}[k], \mathbf{u}_{\text{nom}}[k], 0), \\ \Delta \mathbf{y}[k+1] &= \mathbf{y}[k+1] - \mathbf{y}_{\text{nom}}[k+1] \\ &= \mathbf{F}(\mathbf{x}[k], \mathbf{u}[k], \mathbf{n}[k]) \\ &\quad - \mathbf{F}(\mathbf{x}_{\text{nom}}[k], \mathbf{u}_{\text{nom}}[k], 0). \end{aligned} \quad (41)$$

Under smoothness conditions on  $\mathbf{\Omega}$  and  $\mathbf{F}$ , Equations 41 may be linearized with respect to the nominal path to yield:

$$\begin{aligned} \Delta \mathbf{x}[t+1] &\approx \mathbf{\Omega}_{\mathbf{x}}(\mathbf{Z}[k])\Delta \mathbf{x}[k] + \mathbf{\Omega}_{\mathbf{u}}(\mathbf{Z}[k])\Delta \mathbf{u}[k] \\ &\quad + \mathbf{\Omega}_{\mathbf{n}}(\mathbf{Z}[k])\mathbf{n}[k], \\ \Delta \mathbf{y}[t+1] &\approx \mathbf{F}_{\mathbf{x}}(\mathbf{Z}[k])\Delta \mathbf{x}[k] + \mathbf{F}_{\mathbf{u}}(\mathbf{Z}[k])\Delta \mathbf{u}[k] \\ &\quad + \mathbf{F}_{\mathbf{n}}(\mathbf{Z}[k])\mathbf{n}[k], \end{aligned} \quad (42)$$

where, of course,  $\mathbf{Z}[k] \triangleq (\mathbf{x}_{\text{nom}}[k], \mathbf{u}_{\text{nom}}[k], \mathbf{0})$ . Notation  $\Lambda$ , instead of  $\Delta$ , will be used henceforth on account of the additional approximation introduced through linearization. This is done in order to emphasize the distinction between  $\Delta \mathbf{x}[k]$  and  $\Lambda \mathbf{x}[k]$ , between  $\Delta \mathbf{y}[k]$  and  $\Lambda \mathbf{y}[k]$ , and also between  $\Delta \mathbf{u}[k]$  and  $\Lambda \mathbf{u}[k]$  where the sequences  $\Lambda \mathbf{x}[k]$ ,  $\Lambda \mathbf{y}[k]$  and  $\Lambda \mathbf{u}[k]$  are defined as being governed by the following sets of equations:

$$\begin{aligned} \Lambda \mathbf{x}[t+1] &\approx \mathbf{\Omega}_{\mathbf{x}}(\mathbf{Z}[k])\Lambda \mathbf{x}[k] + \mathbf{\Omega}_{\mathbf{u}}(\mathbf{Z}[k])\Lambda \mathbf{u}[k] \\ &\quad + \mathbf{\Omega}_{\mathbf{n}}(\mathbf{Z}[k])\mathbf{n}[k], \\ \Lambda \mathbf{y}[t+1] &\approx \mathbf{F}_{\mathbf{x}}(\mathbf{Z}[k])\Lambda \mathbf{x}[k] + \mathbf{F}_{\mathbf{u}}(\mathbf{Z}[k])\Lambda \mathbf{u}[k] \\ &\quad + \mathbf{F}_{\mathbf{n}}(\mathbf{Z}[k])\mathbf{n}[k]. \end{aligned} \quad (43)$$

According to the result obtained previously in Equation 34, quantity  $\Delta \mathbf{u}[k]$  or more precisely  $\Lambda \mathbf{u}[k]$ , is given by:

$$\Lambda \mathbf{u}[k] = -\mathbf{G}[k]\Lambda \mathbf{x}[k],$$

where:

$$\mathbf{G}[k] \stackrel{\text{def}}{=} (\mathbf{A}'[k]\mathbf{A}[k] + r_u^2 \mathbf{I})^{-1} \mathbf{A}'[k]\mathbf{B}[k], \quad (44)$$

where  $\mathbf{A}[k]$  and  $\mathbf{B}[k]$  are as in Equations 35. On substituting  $\Lambda \mathbf{u}[k]$  from Equation 44 into Relation 43, one will arrive at:

$$\begin{aligned} \Lambda \mathbf{x}[k+1] &= \mathbf{C}[k]\Lambda \mathbf{x}[k] + \mathbf{D}[k]\mathbf{n}[k], \\ \Lambda \mathbf{y}[k+1] &= \mathbf{K}[k]\Lambda \mathbf{x}[k] + \mathbf{L}[k]\mathbf{n}[k]. \end{aligned} \quad (45)$$

The coefficient matrices in Equations 45 are given by the following definitions:

$$\begin{aligned} \mathbf{C}[k] &\stackrel{\text{def}}{=} \boldsymbol{\Omega}_{\mathbf{x}}(\mathbf{Z}[k]) - \boldsymbol{\Omega}_{\mathbf{u}}(\mathbf{Z}[k])\mathbf{G}[k], \\ \mathbf{D}[k] &\stackrel{\text{def}}{=} \boldsymbol{\Omega}_{\mathbf{n}}(\mathbf{Z}[k]), \\ \mathbf{K}[k] &\stackrel{\text{def}}{=} \mathbf{F}_{\mathbf{x}}(\mathbf{Z}[k]) - \mathbf{F}_{\mathbf{u}}(\mathbf{Z}[k])\mathbf{G}[k], \\ \mathbf{L}[k] &\stackrel{\text{def}}{=} \mathbf{F}_{\mathbf{n}}(\mathbf{Z}[k]). \end{aligned} \quad (46)$$

By choosing  $k = 0$  as the moment of departure from the atmosphere, one will get  $\mathbf{n}[k] = 0 \forall k \geq 0$ . Thus, Equations 45 simplifies to  $\Delta\mathbf{x}[k+1] = \mathbf{C}[k]\Delta\mathbf{x}[k]$  and  $\Delta\mathbf{y}[k+1] = \mathbf{K}[k]\Delta\mathbf{x}[k]$  from which it now readily follows that:

$$\begin{aligned} \Delta\mathbf{y}[k+1] &= \mathbf{K}[k]\mathbf{C}[k-1] \cdots \mathbf{C}[0]\Delta\mathbf{x}[0] \\ &= \mathbf{M}[k]\Delta\mathbf{x}[0], \end{aligned}$$

where:

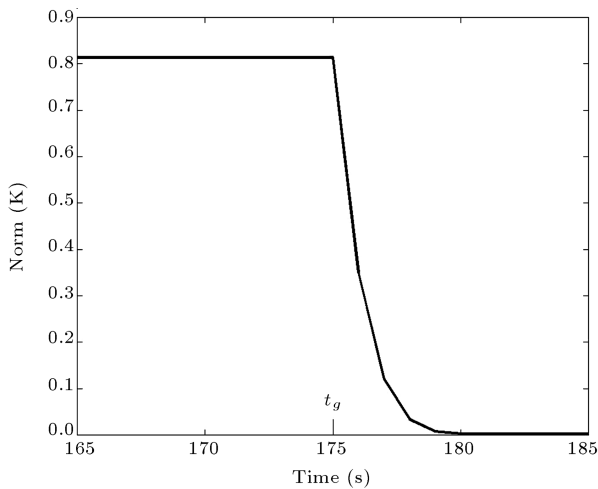
$$\mathbf{M}[k] \stackrel{\text{def}}{=} \mathbf{K}[k]\mathbf{C}[k-1] \cdots \mathbf{C}[0] \quad \forall k \geq 1. \quad (47)$$

On taking the norms of both sides of  $\Delta\mathbf{y}[k+1] = \mathbf{M}[k]\Delta\mathbf{x}[0]$  and using a basic property of norms, there follows the inequality:

$$\|\Delta\mathbf{y}[k+1]\| \leq \|\mathbf{M}[k]\| \times \|\Delta\mathbf{x}[0]\|. \quad (48)$$

As shown in Figure 2, the graph of  $\|\mathbf{M}[k]\|$  becomes rapidly decreasing with increasing  $k$ , as soon as the vehicle leaves the atmosphere. The graph also suggests  $\|\mathbf{M}[k]\|$  converges to zero, as  $k \rightarrow \infty$ . Since matrices  $\mathbf{C}[k]$ ,  $\mathbf{K}[k]$  and  $\mathbf{M}[k]$  are predetermined and not dependent upon or affected by the actual trajectory or errors therein, it may be concluded that:

$$\lim_{k \rightarrow \infty} \Delta\mathbf{y}[k] = 0 \quad \forall \Delta\mathbf{x}(0) \in \mathbb{R}^7. \quad (49)$$



**Figure 2.**  $\|\mathbf{M}[k]\|$  monotonically decreases as  $k$  increases and it converges to zero as  $k \rightarrow \infty$ .

The limiting property (Equation 49) asserts that in a linearized discrete-time model, the closed-loop control law obtained from  $\Delta\mathbf{u}[k] = -\mathbf{G}[k]\Delta\mathbf{x}[k]$  does indeed bring the actual output variables very close to the desired, i.e. nominal values for all errors in the initial state vector. Therefore, it is possible to conclude that, if the sampling period is sufficiently small, the same assertion must be true in the cases of the corresponding nonlinear discrete-time and the original nonlinear continuous-time models, for all initial state vector errors belonging to a sufficiently small neighborhood of the origin of the state space,  $\mathbb{R}^7$ . It is worth noting that in spite of Equation 49,  $\lim_{k \rightarrow \infty} \Delta\mathbf{x}[k] \neq 0$ , i.e. some state variables do not approach their nominal values as  $k \rightarrow \infty$ .

Equations 36 and 49 have important implications. However, there is another equally important practical consideration as follows. It is not enough for the output error to converge to zero in the limit as  $k \rightarrow \infty$ ; the output error must become negligible before the rocket runs out of fuel. In other words, the fuel depletion time must not be less than the orbit insertion time. The length of time it takes for the output error to become negligible depends on the size of the initial state error. Therefore, it may be helpful to have an estimate, however crude, of how large the initial state error can become before the closed-loop guidance algorithm fails to correct the trajectory by the fuel depletion time. Denoting the acceptable tolerance in output error by  $\varepsilon$  and the fuel depletion time by  $k_F$  what has just been said may be restated in terms of the inequality,  $\|\Delta\mathbf{y}[k_F]\| < \varepsilon$ . A sufficient condition for the last inequality is now obtained with the aid of Inequality 48 as follows:

$$\|\Delta\mathbf{x}[0]\| < \frac{\varepsilon}{\|\mathbf{M}[k_F - 1]\|}. \quad (50)$$

Thus, the guidance algorithm will be able to bring the actual output vector within the  $\varepsilon$ -neighborhood of the nominal output by the insertion time, if the initial state error is less than  $\varepsilon/\|\mathbf{M}[k_F - 1]\|$ .

## RESULTS AND DISCUSSION

Performance of the guidance method discussed in previous sections will be examined here through graphs and tables.

Off-line computation of the nominal path is based on the assumption that there are no disturbances at any time during the entire flight. As is well-known, however, disturbances are present at all times during the entire flight, stemming from two main sources: wind, which can be significant particularly during the so-called endoatmospheric phase, and uncertainty in the thrust magnitude of the engine, which exists always.



The wind is modeled in terms of its components in the NED coordinate system. Thus, for example, the wind velocity along the west-east direction is taken to be a scalar random process:

$$\nu_E[t] = \nu_{E0}(1 + 0.2n_E[t]), \quad (51)$$

where  $\nu_{E0}$  is a constant and  $n_E[t]$  is another scalar random process generated through the dynamical equation:

$$n_E[t] = \zeta_E n_E[t-1] + \sqrt{1 - \zeta_E^2} z_E[t], \quad (52)$$

corresponding to a digital filter with a pole at  $z = \zeta_E$  ( $-1 < \zeta_E < 1$ ), a zero at  $z = 0$ , a gain of  $\sqrt{1 - \zeta_E^2}$ , and driven by the zero-mean unit-variance white Gaussian process  $z_E[t]$ . The initial condition  $n_E[0]$  in Equation 52 is a  $N(0, 1)$  random variable statistically independent with the  $z_E[t]$  process. Thus  $n_E[t]$  and, hence,  $\nu_E[t]$  are Gauss-Markov random processes with  $E\{n_E[t]\} = 0$ ,  $\text{var}\{n_E[t]\} = 1$ ,  $E\{\nu_E[t]\} = \nu_{E0}$ , and  $\text{var}\{\nu_E[t]\} = 0.04\nu_{E0}^2$ . What has been said about the wind in a west-east direction holds true also for wind velocities  $\nu_N[t]$  and  $\nu_U[t]$  in the south-north and down-up directions, respectively; only the subscript  $E$  needs to be replaced by  $N$  for the south-north or  $U$  for the down-up wind, as the case may be. The input white processes,  $z_E[t]$ ,  $z_N[t]$  and  $z_U[t]$ , are independent. The  $N(0, 1)$  initial conditions,  $n_E[0]$ ,  $n_N[0]$  and  $n_U[0]$ , are also independent. Moreover, independence holds also between the set of input processes and the set of initial conditions. Neither the means,  $\nu_{E0}$ ,  $\nu_{N0}$  and  $\nu_{U0}$ , nor the filter poles,  $\zeta_E$ ,  $\zeta_N$  and  $\zeta_U$ , need be the same in

general though, of course, it is possible for the two or three of them to be taken equal in simulations. Effects of wind in different directions for both Open-Loop (OL) and Closed-Loop (CL) strategies are presented in Tables 2 and 3, where “open-loop” means, of course, use of the nominal control during the entire powered flight, and “closed-loop” implies use of the proposed guidance scheme.

To evaluate performance, values of  $V$ , as well as errors in  $h$ ,  $\gamma$  and  $i$  at the moment of insertion have been shown in Tables 2 and 3 for simulations of both OL and CL strategies. “Error” means difference between desired and actual values. The target being a sun-synchronous orbit, the desired values for  $h(T)$ ,  $\gamma(T)$  and  $i(T)$  are respectively, 200 km, 0 deg and 97.5 deg. In Tables 2 and 3, the wind speeds are, respectively, 6.5 m/s (known as a “moderate breeze” in the Beaufort scale) and 22 m/s (strong gale).

As the results indicate, if OL guidance is chosen, then, west-east winds can create great errors in parameters values at the moment of orbit insertion. This is primarily due to the fact that the launch path is close to a polar orbit. Errors, particularly in  $i(T)$  and  $\gamma(T)$ , are so large, the mission may be considered as failed for the OL case. The results also demonstrate the significant improvement brought about by the CL algorithm at the cost, however, of a slightly lower speed at orbit insertion.

In reality, winds along down-up directions are seldom as strong as those in west-east or south-north directions. However, to see the effect of wind direction alone on guidance performance and for comparison

**Table 2.** Errors at the orbit insertion moment,  $T$ , for wind speed of 6.5 m/s.

Wind Direction		Altitude Error $\Delta h(T)$ (m)	Flight Path Angle Error $\Delta \gamma(T)$ (deg)	Inclination Error $\Delta i(T)$ (deg)	Velocity $V(T)$ (m/s)
North	OL	-730	-0.136	-0.028	7798.2
	CL	0.87	$-2.46 \times 10^{-5}$	$-6.68 \times 10^{-6}$	7795.6
South	OL	730.0	0.136	0.028	7793.0
	CL	-0.929	$1.47 \times 10^{-5}$	$-5.60 \times 10^{-6}$	7795.6
East	OL	86.5	0.016	-0.130	7795.3
	CL	-0.44	$4.16 \times 10^{-6}$	$-5.89 \times 10^{-6}$	7795.6
West	OL	-88.0	-0.016	0.130	7795.9
	CL	-0.099	$-3.40 \times 10^{-6}$	$-6.1 \times 10^{-6}$	7795.6
Up	OL	126.1	0.022	$-6.15 \times 10^{-6}$	7795.2
	CL	7.36	$-1.67 \times 10^{-4}$	$-1.06 \times 10^{-6}$	7795.6
Down	OL	-126.2	-0.022	$-6.15 \times 10^{-6}$	7795.9
	CL	-7.40	$-1.56 \times 10^{-4}$	$-1.73 \times 10^{-6}$	7795.5

purposes, down-up winds in simulations were taken to be as strong as those in other directions. A first conclusion is that, even for OL guidance, the errors caused by down-up winds are not as large as those caused by west-east or south-north directions. Another conclusion is that CL guidance does succeed in significantly reducing errors.

The thrust magnitude of each stage is assumed

to be within a few percent of its nominal value. Table 4 shows the effects of thrust magnitude (static) deviations on the accuracy of orbit insertion. For OL guidance, the error in  $i(T)$  is not so great, however, the error in  $\gamma(T)$  is large enough for the OL guidance to be considered as failed. In this case, too, the CL guidance algorithm greatly reduces the errors caused by thrust magnitude deviations.

**Table 3.** Errors at the orbit insertion moment,  $T$ , for wind speed of 22 m/s.

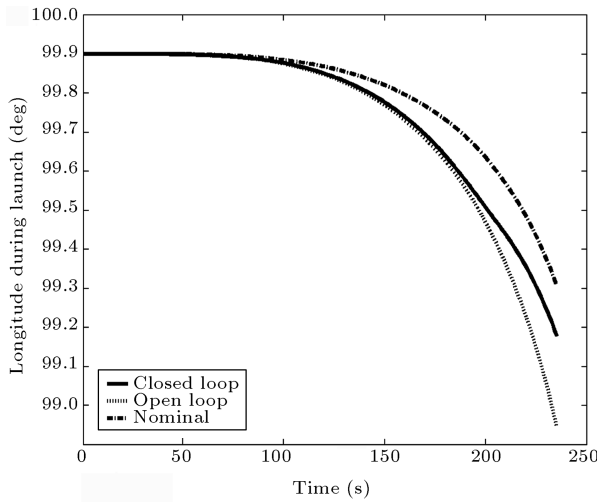
Wind Direction		Altitude Error $\Delta h(T)$ (m)	Flight path angle Error $\Delta \gamma(T)$ (deg)	Inclination Error $\Delta i(T)$ (deg)	Velocity $V(T)$ (m/s)
North	OL	$-8.91 \times 10^3$	-1.64	-0.33	7827.1
	CL	9.25	$-2.08 \times 10^{-4}$	$-1.17 \times 10^{-5}$	7795.5
South	OL	$8.97 \times 10^3$	1.70	0.36	7763.6
	CL	-12.85	$2.75 \times 10^{-4}$	$-1.52 \times 10^{-6}$	7795.4
East	OL	956.9	0.179	-1.60	7792.1
	CL	-42.1	$-9.16 \times 10^{-4}$	$1.90 \times 10^{-5}$	7793.1
West	OL	$-1.18 \times 10^3$	-0.22	1.58	7799.8
	CL	-37.7	$8.20 \times 10^{-4}$	$1.63 \times 10^{-6}$	7793.2
Up	OL	$1.54 \times 10^3$	0.273	$-6.13 \times 10^{-6}$	7791.4
	CL	90.11	-0.002	$-6.0 \times 10^{-5}$	7794.1
Down	OL	$-1.55 \times 10^3$	-0.275	$-6.16 \times 10^{-6}$	7799.8
	CL	-90.79	0.002	$4.81 \times 10^{-5}$	7795.1

**Table 4.** Errors at the orbit insertion moment,  $T$ , caused by thrust magnitude error.

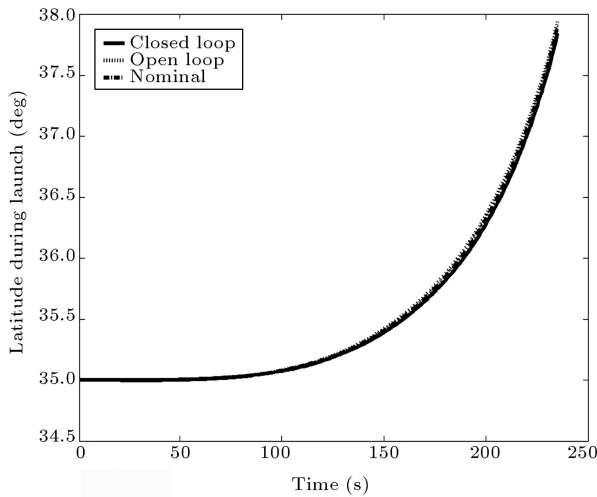
Thrust Error (%)		Altitude Error $\Delta h(T)$ (m)	Flight Path Angle Error $\Delta \gamma(T)$ (deg)	Heading Angle Error $\Delta i(T)$ (deg)	Velocity $V(T)$ (m/s)
+1	OL	$1.19 \times 10^4$	1.86	0	7742.7
	CL	$2.73 \times 10^3$	-0.067	-0.0016	7856.3
-1	OL	$-1.23 \times 10^4$	-1.94	0	7750.6
	CL	$-2.74 \times 10^3$	0.068	0.0016	7734.6
+5	OL	$5.62 \times 10^4$	8.55	0	8049.5
	CL	$1.36 \times 10^4$	-0.336	-0.0083	8096.8
-5	OL	$-6.62 \times 10^4$	-10.5	0	7594.8
	CL	$-1.38 \times 10^4$	0.34	0.0081	7488.2
+8	OL	$8.61 \times 10^4$	12.86	0	8220.3
	CL	$2.16 \times 10^4$	-0.535	-0.0132	8274.6
-8	OL	$-1.16 \times 10^5$	-17.84	0	7501.8
	CL	$-2.27 \times 10^4$	0.56	0.013	7301.4

To consider a more realistic scenario in the following round of simulations, the thrust magnitude of each stage is 1% lower than its nominal value and the wind has components in west-east and south-north directions given, of course, by Equations 36 and 37, with  $\nu_{E0} = \nu_{N0} = \frac{15}{\sqrt{2}} = 10.61$  m/s (a moderate gale) and  $\zeta_E = \zeta_N = 0.9$ . The results are exhibited in Figures 3 to 12. Solid, dashed and solid-dashed graphs in each figure correspond to CL, OL and purely nominal simulations, respectively.

Figures 3 to 7 show graphs versus time  $t$  of the state-variables  $\lambda(t)$  (longitude),  $\tau(t)$  (latitude),  $h(t)$  (altitude),  $\gamma(t)$  (flight-path angle) and  $\psi(t)$  (heading angle) during the launch, i.e. for  $t_0 \leq t \leq T$ . Figures 8 and 9 show the graphs versus  $t$  of control variables  $\alpha(t)$  (angle of attack) and  $\mu(t)$  (velocity-roll angle), again for  $t_0 \leq t \leq T$ . Clearly,  $\alpha(t)$  and  $\mu(t)$  are identically zero in OL simulations, and near zero during the endoatmospheric phase of CL



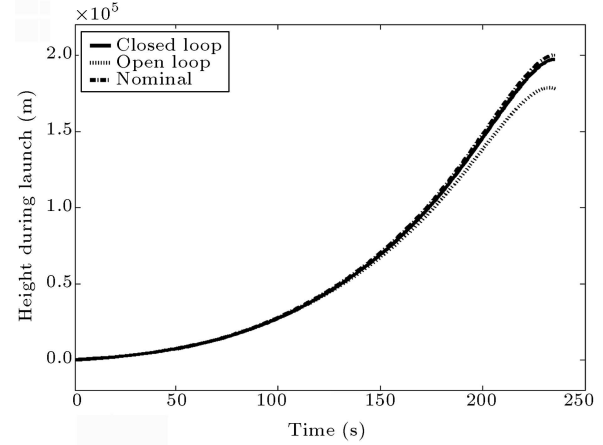
**Figure 3.** Longitude during launch.



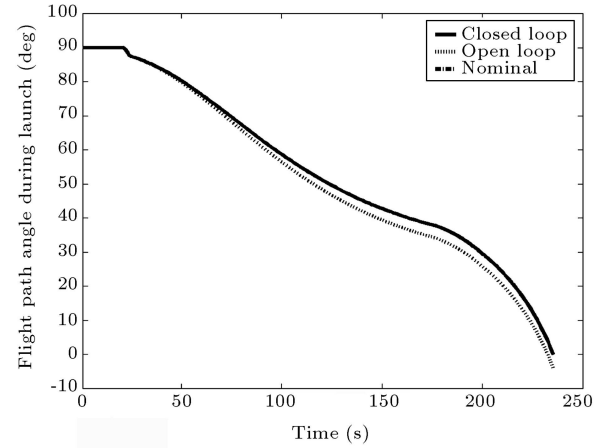
**Figure 4.** Latitude during launch.

simulations. However,  $\alpha(t)$  and  $\mu(t)$  do take on large nonzero values during the exoatmospheric phase of CL simulations in order to effect guidance toward the target orbit.

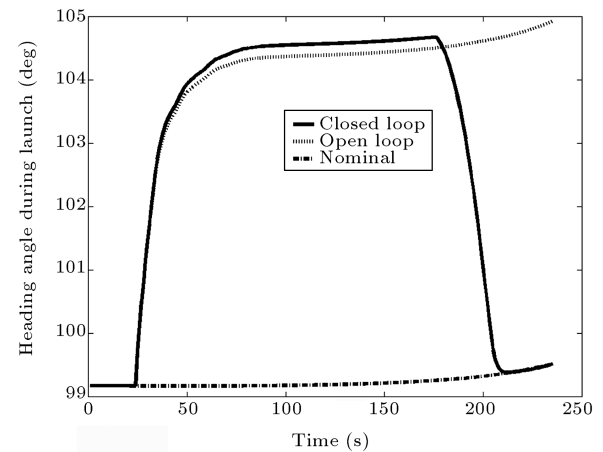
The amplitude of variations in  $\alpha(t)$  and  $\mu(t)$  are about 40 deg and 30 deg, respectively, implying that



**Figure 5.** Altitude during launch.



**Figure 6.** Flight path angle during launch.



**Figure 7.** Heading angle during launch.

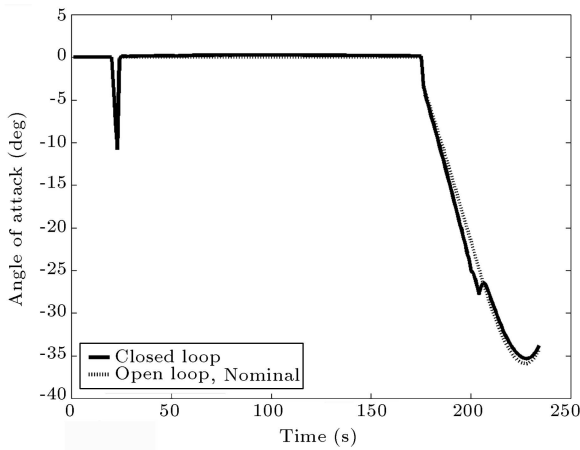


Figure 8. Angle of attack during launch.

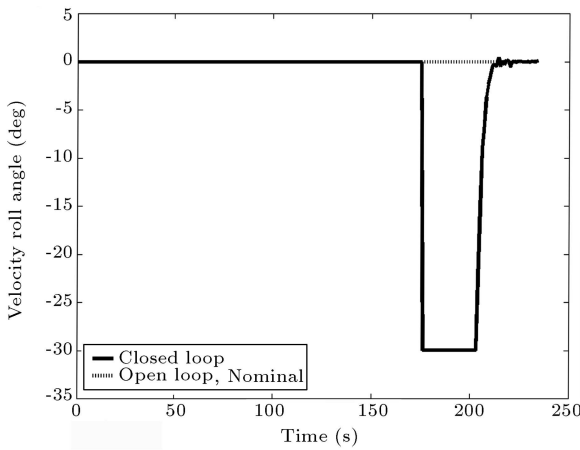


Figure 9. Velocity roll angle during launch.

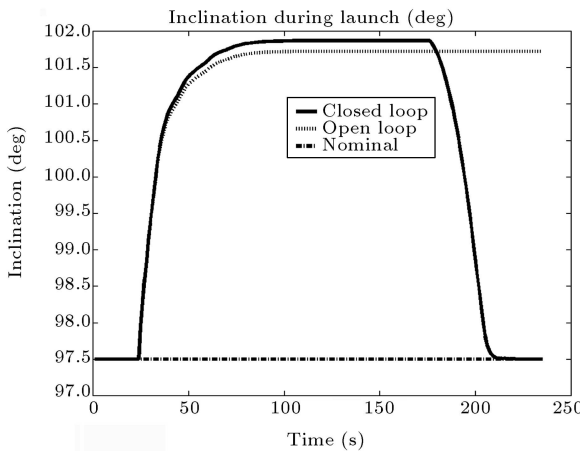


Figure 10. Inclination during launch.

the vehicle may undergo intensive maneuvering. This, however, occurs outside the atmosphere and will not endanger the vehicle. Figure 10 depicts the variations of inclination  $i(t)$  during launch. Figures 11 and 12 show the latitude and velocity for a longer period of time, namely from lift-off to 7500 seconds after orbit

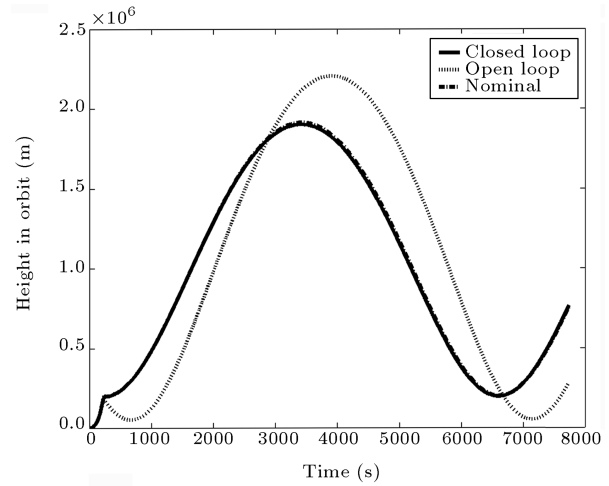


Figure 11. Altitude of satellite in orbit.

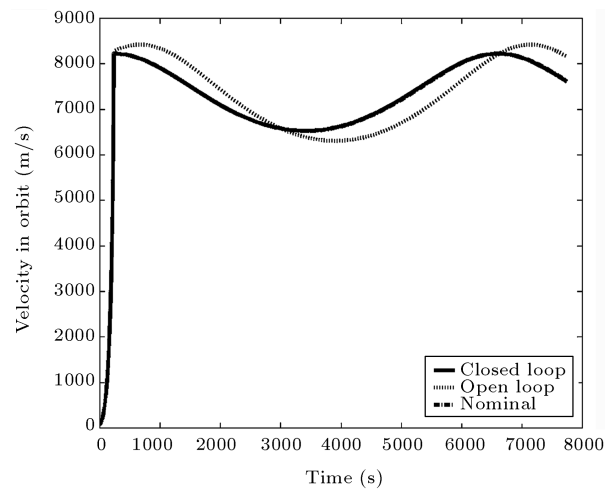


Figure 12. Velocity of satellite in orbit.

insertion. These figures make it possible to observe what happens to the satellite during one complete cycle around the Earth after insertion into the target orbit. The performance of the proposed guidance algorithm may be further verified by comparing the CL and OL graphs as follows. For the OL graphs, it can be seen from Figures 10 and 11 that  $i \approx 101.7$  deg instead of 97.5 deg, and  $|h|_{\min} \approx 50$  km, which means that the satellite will return to the atmosphere and eventually fall. For the CL graphs, on the other hand, one obtains  $i(T) \approx 97.5$  deg,  $|h|_{\min} \approx 200$  km from the same figures. Furthermore, in the OL graphs,  $\gamma(T) \approx -4$  deg, implying a failed launch, while in the CL case,  $\gamma(T) \approx 0.01$  deg which is an acceptable error.

## CONCLUSION

A guidance algorithm for sending a small satellite to a sun synchronous LEO was devised with the aims of simplicity of onboard computations and accuracy of orbit insertion. The onboard computations involve

no more than multiplying at each sampling time a  $2 \times 7$  gain matrix by a  $7 \times 1$  data vector. The gain matrices are computed off-line and stored in the onboard computer. Data are obtained through onboard measurements. The proposed algorithm performs accurate orbit insertion against wind disturbances of speeds of at least 22 m/s and thrust magnitude errors of at least  $\pm 5\%$ . Extending the method to launches to non-LEO environments or to other orbits can be subjects for future research.

## NOMENCLATURE

$a$	speed of sound
$a$	LTL parameter in Equation 28
$b$	LTL parameter in Equation 28
$c_i$	mass burning rate of the $i$ th stage
$\mathbf{f}$	a function representing dynamics
$g$	gravitational acceleration
$h$	altitude from sea level
$h_1$	air density constant
$h_2$	air pressure constant
$m$	mass
$n$	random process
$\mathbf{n}$	disturbance vector
$p$	air pressure
$p_s$	air pressure at sea level
$q$	dynamic pressure
$r$	distance from Earth center
$t$	time
$t_0$	launch start time
$t_v$	end of vertical launch time
$\mathbf{u}$	control vector
$w$	wind force
$\mathbf{x}$	state vector
$\mathbf{y}$	output vector
$z$	white random process
$\mathbf{A}$	see Equation 46
$\mathbf{B}$	see Equation 47
$A_b$	aerodynamic reference area
$A_e$	engine exit area of launch vehicle
$A_L$	lateral cross section area of launch vehicle
$C_D$	drag coefficient
$C_{D0}$	drag coefficient constant
$C_{D1}$	drag coefficient constant
$C_{D2}$	drag coefficient constant
$C_{D3}$	drag coefficient constant
$C_L$	lift coefficient

$C_{L1}$	lift coefficient constant
$C_{L2}$	lift coefficient constant
$D$	drag
$\mathbf{F}$	a function representing the final output
$L$	lift
$M$	Mach number
$\mathbf{M}$	matrix relating output vector to state vector
$\mathbf{Q}$	scaling matrix
$T$	thrust
$T$	orbit insertion moment
$T_{\text{vac}}$	thrust in vacuum
$V$	velocity
$Z$	denoting nominal values of state and control vectors
$\alpha$	angle of attack
$\beta$	ratio of specific heat of air
$\gamma$	flight path angle
$\zeta$	see Equation 49
$\theta$	angle
$\lambda$	longitude
$\mu$	velocity roll angle
$\nu$	wind speed
$\nu_0$	wind speed mean
$\rho$	air density
$\rho_s$	air density at sea level
$\tau$	latitude
$\psi$	heading angle
$\omega$	Earth rotating rate
$\omega_p$	see Equation 26
$\Delta$	difference between actual and nominal value

## Subscripts

nom	nominal value
$E$	east component
$N$	north component
$U$	up component

## REFERENCES

1. Dubourg, V., Kainov, V., Thoby, M., Silkin, O. and Solovey, V. "The DEMETER micro satellite launch campaign: A cheap access to space", *Advances in Space Research*, **37**, pp. 754-760 (2006).
2. Freeborn, P., Kinnersley, M. and Zorina, A. "The ROCKOT launch vehicle-the competitive launch solution for small Earth observation satellites into low Earth orbits", *Acta Astronautica*, **56**, pp. 315-325 (2005).

3. Marrdonny, M. and Mobed, M. "A guidance algorithm for launch to equatorial orbit", *Aircraft Engineering and Aerospace Technology*, **81**(2), pp. 137-148 (2009).
4. Sinha, S.K. and Shrivastava, S.K. "Optimal explicit guidance of multistage launch vehicle along three-dimensional trajectory", *Journal of Guidance, Control and Dynamics*, **13**(3), pp. 394-403 (1990).
5. Battin, R.H., *An Introduction to the Mathematics and Methods of Astrodynamics*, AIAA Education series, New York (1987).
6. Madeira, F. and Rios-Neto, A. "Guidance and control of a launch vehicle using a stochastic gradient projection method", *Automatica*, **36**, pp. 427-438 (2000).
7. Pamadi, B.N., Taylor, L.W. and Price, D.B. "Adaptive guidance for an aero-assisted boost vehicle", *Journal of Guidance, Control and Dynamics*, **13**(4), pp. 586-595 (1990).
8. Lu, P. "Nonlinear trajectory tracking guidance with application to a launch vehicle", *Journal of Guidance, Control and Dynamics*, **19**(1), pp. 99-106 (1996).
9. Lu, P., Sun, H. and Tsai, B. "Closed-loop endoatmospheric ascent guidance", *Journal of Guidance, Control and Dynamics*, **26**(2), pp. 283-294 (2003).
10. Wiesel, W.E., *Spaceflight Dynamics*, McGraw-Hill, Singapore (1997).
11. Pamadi, B.N. "Simple guidance method for single stage to low earth orbit", *Journal of Guidance, Control and Dynamics*, **18**(6), pp. 1420-1426 (1995).
12. Gräßlin, M.H., Telaar, J. and Schöttle, U.M. "Ascent and reentry guidance concept based on NLP-methods", *Acta Astronautica*, **55**, pp. 461-471 (2004).
13. Feeley, T.S. and Speyer, J.L. "Techniques for developing approximate optimal advanced launch system guidance", *Journal of Guidance, Control and Dynamics*, **17**(5), pp. 889-896 (1994).
14. Seywald, H. and Cliff, E.M. "Neighboring optimal control based feedback law for the advanced launch system", *Journal of Guidance, Control and Dynamics*, **17**(6), pp. 1154-1162 (1994).
15. Calise, A.J., Melamed, N. and Lee, S. "Design and evaluation of a three-dimensional optimal ascent guidance algorithm", *Journal of Guidance, Control and Dynamics*, **21**(6), pp. 865-875 (1998).
16. Leung, M.S.K. and Calise, A.J. "Hybrid approach to near-optimal launch vehicle guidance", *Journal of Guidance, Control and Dynamics*, **17**(5), pp. 881-888 (1994).
17. Gath, P.F. and Calise, A.J. "Optimization of launch vehicle ascent trajectories with path constraints and coast arcs", *Journal of Guidance, Control and Dynamics*, **24**(2), pp. 296-304 (2001).
18. Park, S. "Launch vehicle trajectories with a dynamic pressure constraint", *Journal of Spacecraft and Rockets*, **35**(6), pp. 765-773 (1998).
19. Shaver, D.A. and Hull, D.G. "Advanced launch system trajectory optimization using suboptimal control", *Proceedings of the AIAA Guidance, Navigation, and Control Conference, AIAA, Washington, DC* (AIAA Paper 90-3413), pp. 892-901 (1990).

## BIOGRAPHIES

**Mohammad Marrdonny** was born in 1975 in Iran. He received his Ph.D. from the Department of Electrical Engineering at Sharif University of Technology (S.U.T), Tehran in 2009, where he also received his B.S. and M.S. degrees in Electronics and Control Engineering in 1998 and 2000, respectively. His field of interest is: Navigation and Guidance of Aerospace Vehicles.

**Mohammad Mobed** was born in 1954 in Iran. He received his B.S. degree from Sharif University of Technology, and his M.S. and Ph.D. degrees from The University of Michigan, Ann Arbor, all in Electrical Engineering in 1977, 1979 and 1988, respectively. He was a researcher and an instructor at the Iran Telecommunications Research Center, Amirkabir University of Technology, and Tafresh University, during the period from 1994 to 1996. He has been with the Electrical Engineering Department, Sharif University of Technology, as an Assistant Professor since 1997. His interests are in Systems aspects of Communications and Control.