A New, General Neighboring Optimal Guidance for Aerospace Vehicles

Mauro Pontani

Abstract This work describes and applies the recently introduced, general-purpose perturbative guidance termed variable-time-domain neighboring optimal guidance, which is capable of driving an aerospace vehicle along a specified nominal, optimal path. This goal is achieved by minimizing the second differential of the objective function (related to the flight time) along the perturbed trajectory. This minimization principle leads to deriving all the corrective maneuvers, in the context of an iterative closed-loop guidance scheme. Original analytical developments, based on optimal control theory and adoption of a variable time domain, constitute the theoretical foundation for several original features. The real-time feedback guidance at hand is exempt from the main disadvantages of similar algorithms proposed in the past, such as the occurrence of singularities for the gain matrices. The variable-time-domain neighboring optimal guidance algorithm is applied to two typical aerospace maneuvers: (1) minimum-time climbing path of a Boeing 727 aircraft and (2) interception of fixed and moving targets. Perturbations arising from nonnominal propulsive thrust or atmospheric density and from errors in the initial conditions are included in the dynamical simulations. Extensive Monte Carlo tests are performed, and unequivocally prove the effectiveness and accuracy of the variable-time-domain neighboring optimal guidance algorithm.

1 Introduction

The problem of driving an aerospace vehicle along a specified path leading to fulfilling the boundary conditions associated with the mission specifications requires defining the corrective actions aimed at compensating nonnominal flight conditions. This means that a feedback control law, or, equivalently, a closed-loop guidance algorithm, is to be defined, on the basis of the current state of the vehicle, evaluated at prescribed sampling times.

M. Pontani (✉)
Sapienza University of Rome, Rome, Italy
e-mail: mauro.pontani@uniroma1.it

Traditionally, two different approaches to guidance exist. Adaptive algorithms compute the flight trajectory at the beginning of each guidance interval, on the basis of feasibility or optimality criteria [3, 19]. Perturbative algorithms assume a specified nominal trajectory, and define the feedback control corrections aimed at maintaining the vehicle in the proximity of the nominal path [7, 9].

Neighboring Optimal Guidance (NOG) is a perturbative guidance concept that relies on the analytical second order optimality conditions, in order to find the corrective control actions in the neighborhood of the reference trajectory. This is an optimal trajectory that satisfies the first and second-order optimality conditions. In general, the neighboring optimal path originates from a perturbed state and is associated with the minimization of the second differential of the objective function. Several time-varying gain matrices, referring to the nominal trajectory, are defined, computed offline, and stored in the onboard computer. Only a limited number of works have been devoted to studying neighboring optimal guidance [1, 4–6, 18, 21]. In particular, a thorough treatment of NOG is due to Chuang [5], who proposed a simple formula for updating the time of flight, and used a very basic strategy to evaluate the gain matrices when the time of flight exceeds its nominal value. Hull [6, 7] supplied further relevant contributions to the topic, using a vector that contains the unknown parameters to optimize and proposing an analytical formulation for the update of the time of flight, albeit only at the initial time. A common difficulty encountered in implementing the NOG consists in the fact that the gain matrices become singular while approaching the final time. As a result, the real-time correction of the time of flight can lead to numerical difficulties so relevant to cause the failure of the guidance algorithm.

This work describes and applies the recently introduced [15, 16], general-purpose variable-time-domain neighboring optimal guidance algorithm (VTD-NOG), on the basis of the general theory of NOG described in [7]. Some fundamental, original features of VTD-NOG are aimed at overcoming the main difficulties related to the use of former NOG schemes, in particular the occurrence of singularities and the lack of an efficient law for the iterative real-time update of the time of flight. This is achieved by adopting a normalized time domain, which leads to defining a novel updating law for the time of flight, a new termination criterion, and a new analytical formulation for the sweep method. Two applications are considered, for the purpose of illustrating the new guidance algorithm: (1) minimum-time-to-climb path of a Boeing 727 aircraft and (2) interception of fixed and moving targets. Specifically, perturbations arising from the imperfect knowledge of the propulsive thrust and from errors in the initial conditions are included in the dynamical modeling. In addition, atmospheric density fluctuations are modeled for application (1). Extensive Monte Carlo (MC) tests are performed, with the intent of demonstrating the effectiveness and accuracy of the variable-time-domain neighboring optimal guidance algorithm.

2 Nominal Trajectory

The nominal trajectory of aerospace vehicles is computed in the absence of any perturbation. For the purpose of applying a neighboring optimal guidance, the nominal path is required to be an optimal trajectory that minimizes a specified objective function.

In general, the spacecraft trajectory is described through the time-varying, n-dimensional state vector $\mathbf{x}(t)$ and controlled through the time-varying, m-dimensional control vector $\mathbf{u}(t)$; the dynamical evolution over the time interval $[t_0, t_f]$ (with t_0 set to 0 and t_f unspecified) depends also on the time-independent, $\tilde{p}$-dimensional parameter vector $\tilde{\mathbf{a}}$. The governing state equations have the general form

$$\dot{\mathbf{x}} = \tilde{\mathbf{f}}(\mathbf{x}, \mathbf{u}, \tilde{\mathbf{a}}, t) \tag{1}$$

and are subject to q boundary conditions

$$\boldsymbol{\psi}(\mathbf{x}_0, \mathbf{x}_f, \tilde{\mathbf{a}}, t_f) = \mathbf{0} \tag{2}$$

where the subscripts "0" and "f" refer to t_0 and t_f . A feasible trajectory is a solution that obeys the state equations (1) and satisfies the boundary conditions (2).

The problem at hand can be reformulated by using the dimensionless normalized time τ defined as

$$\tau := \frac{t}{t_f} \quad \Rightarrow \quad \tau_0 \equiv 0 \leq \tau \leq 1 \equiv \tau_f \tag{3}$$

Let the dot denote the derivative with respect to τ hence forward. If $\mathbf{a} := \begin{bmatrix} \tilde{\mathbf{a}} & t_f \end{bmatrix}^T$ (and $p := \tilde{p} + 1$), the state equations (1) are rewritten as

$$\dot{\mathbf{x}} = t_f \tilde{\mathbf{f}}(\mathbf{x}, \mathbf{u}, \tilde{\mathbf{a}}, t_f \tau) =: \mathbf{f}(\mathbf{x}, \mathbf{u}, \mathbf{a}, \tau) \tag{4}$$

The objective functional to minimize has the following general form:

$$J = \phi(\mathbf{x}_0, \mathbf{x}_f, \mathbf{a}) + \int_0^1 L[\mathbf{x}(\tau), \mathbf{u}(\tau), \mathbf{a}, \tau] d\tau \tag{5}$$

The spacecraft trajectory optimization problem consists in identifying a feasible solution that minimizes the objective functional J, through selection of the optimal control law $\mathbf{u}^*(t)$ and the optimal parameter vector $\mathbf{a}^*$, i.e.

$$\{\mathbf{u}^*, \mathbf{a}^*\} = \arg\min_{\{\mathbf{u}, \mathbf{a}\}} J \tag{6}$$

2.1 First-Order Necessary Conditions for a Local Extremal

In order to state the necessary conditions for optimality, a Hamiltonian H and a function of the boundary conditions Φ are defined as [7]

$$H(\mathbf{x}, \mathbf{u}, \mathbf{a}, \boldsymbol{\lambda}, \tau) := L + \boldsymbol{\lambda}^T \mathbf{f} \qquad \Phi(\mathbf{x}_0, \mathbf{x}_f, \mathbf{a}, \boldsymbol{\upsilon}) := \phi + \boldsymbol{\upsilon}^T \boldsymbol{\psi} \tag{7}$$

where the time-varying, n-dimensional costate vector $\boldsymbol{\lambda}(\tau)$ and the time-independent, q-dimensional vector $\boldsymbol{\upsilon}$ are the adjoint variables conjugate to the state equations (4) and to the conditions (2), respectively.

In the presence of an optimal (locally minimizing) solution, the following conditions hold:

$$\mathbf{u}^* = \arg\min_{\mathbf{u}} H \tag{8}$$

$$\dot{\boldsymbol{\lambda}} = -\left[\frac{\partial H}{\partial \mathbf{x}}\right]^T \tag{9}$$

$$\boldsymbol{\lambda}_0 = -\left[\frac{\partial \Phi}{\partial \mathbf{x}_0}\right]^T \qquad \boldsymbol{\lambda}_f = \left[\frac{\partial \Phi}{\partial \mathbf{x}_f}\right]^T \tag{10}$$

$$\left[\frac{\partial \Phi}{\partial \mathbf{a}}\right]^T + \int_0^1 \left[\frac{\partial H}{\partial \mathbf{a}}\right]^T d\tau = \mathbf{0} \tag{11}$$

For the very general Hamiltonian (7) the Pontryagin minimum principle (8) yields the control variables as functions of the adjoint variables and the state variables; the relations (9) are the adjoint (or costate) equations, together with the related boundary conditions (10); (11) is equivalent to p algebraic scalar equations. If the control $\mathbf{u}$ is unconstrained, then (8) implies that H is stationary with respect to $\mathbf{u}$ along the optimal path, i.e.

$$H_{\mathbf{u}}^* = \mathbf{0}^T \tag{12}$$

Equations (8) through (11) are well established in optimal control theory (and are proven, for instance, in [7]), and allow translating the optimal control problem into a two-point boundary-value problem. Unknowns are the state $\mathbf{x}$, the parameter vector $\mathbf{a}$, and the adjoint variables $\boldsymbol{\lambda}$ and $\boldsymbol{\upsilon}$ (while the optimal control $\mathbf{u}^*$ is given by (8), as previously remarked). It is straightforward to demonstrate that the condition (11) is equivalent to

$$\boldsymbol{\mu}_f - \left[\frac{\partial \Phi}{\partial \mathbf{a}}\right]^T = \mathbf{0}, \quad \text{with} \quad \dot{\boldsymbol{\mu}} = -\left[\frac{\partial H}{\partial \mathbf{a}}\right]^T \quad \text{and} \quad \boldsymbol{\mu}_0 = \mathbf{0} \tag{13}$$

where $\boldsymbol{\mu}_0$ and $\boldsymbol{\mu}_f$ are, respectively, the initial and final value (at $\tau = 1$) of the time-varying $(p \times 1)$-vector $\boldsymbol{\mu}$.

2.2 Second-Order Sufficient Conditions for a Local Minimum

The derivation of the second-order optimality conditions involves the definition of an admissible comparison path, located in the neighborhood of the (local) nominal, optimal solution, associated with the state $\mathbf{x}^*$, costate $\boldsymbol{\lambda}^*$, and control $\mathbf{u}^*$. By definition, an admissible comparison path is a feasible trajectory that satisfies the equations of motion and the boundary conditions. A neighboring optimal path is an admissible comparison trajectory that satisfies also the optimality conditions. The nonexistence of alternative neighboring optimal paths is to be proven in order to guarantee optimality of the nominal solution [7, 8].

The first second-order condition is the Clebsch-Legendre sufficient condition for a minimum [7, 8], i.e. $H_{\mathbf{uu}}^* > 0$ (positive definiteness of $H_{\mathbf{uu}}^*$). In the necessary (weak) form the Hessian $H_{\mathbf{uu}}^*$ must be positive semidefinite.

In general, a neighboring optimal path located in the proximity of the optimal solution fulfills the feasibility equations (4) and (2) and the optimality conditions (8)–(11) to first order. This means that the state and costate displacements $\{\delta\mathbf{x}, \delta\boldsymbol{\lambda}\}$ (from the optimal solution) satisfy the linear equations deriving from (4) and (9),

$$\delta\dot{\mathbf{x}} = \mathbf{f_x}\delta\mathbf{x} + \mathbf{f_u}\delta\mathbf{u} + \mathbf{f_a}\delta\mathbf{a} \tag{14}$$

$$\delta\dot{\boldsymbol{\lambda}} = -H_{\mathbf{xx}}\delta\mathbf{x} - H_{\mathbf{xu}}\delta\mathbf{u} - H_{\mathbf{x\lambda}}\delta\boldsymbol{\lambda} - H_{\mathbf{xa}}\delta\mathbf{a} \tag{15}$$

in conjunction with the respective linear boundary conditions, derived from (2) and (10),

$$\boldsymbol{\psi}_{\mathbf{x}_f}\delta\mathbf{x}_f + \boldsymbol{\psi}_{\mathbf{x}_0}\delta\mathbf{x}_0 + \boldsymbol{\psi}_{\mathbf{a}}\delta\mathbf{a} = \mathbf{0} \tag{16}$$

$$\delta\boldsymbol{\lambda}_0 = -\Phi_{\mathbf{x}_0\mathbf{x}_0}\delta\mathbf{x}_0 - \Phi_{\mathbf{x}_0\mathbf{a}}\delta\mathbf{a} - \boldsymbol{\psi}_{\mathbf{x}_0}^T d\boldsymbol{v} \tag{17}$$

$$\delta\boldsymbol{\lambda}_f = \Phi_{\mathbf{x}_f\mathbf{x}_f}\delta\mathbf{x}_f + \Phi_{\mathbf{x}_f\mathbf{a}}\delta\mathbf{a} + \boldsymbol{\psi}_{\mathbf{x}_f}^T d\boldsymbol{v} \tag{18}$$

The fact that the Hamiltonian is stationary with respect to $\mathbf{u}$, i.e. $H_{\mathbf{u}}^* = \mathbf{0}^T$, yields

$$H_{\mathbf{ux}}\delta\mathbf{x} + H_{\mathbf{uu}}\delta\mathbf{u} + H_{\mathbf{ua}}\delta\mathbf{a} + H_{\mathbf{x\lambda}}\delta\boldsymbol{\lambda} = \mathbf{0} \tag{19}$$

Under the assumption that the Clebsch-Legendre condition is satisfied, (19) is solved for $\delta\mathbf{u}$

$$\delta\mathbf{u} = -H_{\mathbf{uu}}^{-1}\left(H_{\mathbf{ux}}\delta\mathbf{x} + H_{\mathbf{ua}}\delta\mathbf{a} + H_{\mathbf{x\lambda}}\delta\boldsymbol{\lambda}\right) \tag{20}$$

The parameter condition (11) is replaced by (13), leading to the following relations:

$$\delta\dot{\boldsymbol{\mu}} = -H_{\mathbf{ax}}\delta\mathbf{x} - H_{\mathbf{au}}\delta\mathbf{u} - H_{\mathbf{aa}}\delta\mathbf{a} - H_{\mathbf{a}\boldsymbol{\lambda}}\delta\boldsymbol{\lambda}, \quad \text{with} \tag{21}$$

$$\delta\boldsymbol{\mu}_0 = \mathbf{0}, \quad \delta\boldsymbol{\mu}_f = -\boldsymbol{\Phi}_{\mathbf{ax}_f}\delta\mathbf{x}_f - \boldsymbol{\Phi}_{\mathbf{aa}}\delta\mathbf{a} - \boldsymbol{\psi}_{\mathbf{a}}^T d\boldsymbol{\upsilon} \tag{22}$$

where (22) is written under the assumption that $\boldsymbol{\Phi}_{\mathbf{ax}_0} = \mathbf{0}$, condition that is met for the problems at hand. It is relatively straightforward to recognize that solving the equation system (14)–(19) and (21)–(22) is equivalent to solving the *accessory optimization problem* [7, 8], which consists in minimizing the second differential $d^2 J$. The solution process involves the definition of the sweep variables, through the following relations:

$$\delta\boldsymbol{\lambda} = \mathbf{S}\delta\mathbf{x} + \mathbf{R}d\boldsymbol{\upsilon} + \mathbf{m}d\mathbf{a}, \tag{23}$$

$$\mathbf{0} = \mathbf{R}\delta\mathbf{x} + \mathbf{Q}d\boldsymbol{\upsilon} + \mathbf{n}d\mathbf{a}, \tag{24}$$

$$\delta\boldsymbol{\mu} = \mathbf{m}\delta\mathbf{x} + \mathbf{n}^T d\boldsymbol{\upsilon} + \boldsymbol{\alpha}d\mathbf{a} \tag{25}$$

The matrices $\mathbf{S}$, $\mathbf{R}$, $\mathbf{m}$, $\mathbf{Q}$, $\mathbf{n}$, and $\boldsymbol{\lambda}$ must satisfy the sweep equations (not reported for the sake of conciseness), in conjunction with the respective boundary conditions (prescribed at the final time) [7, 8]. The variations $d\boldsymbol{\upsilon}$ and $d\mathbf{a}$ can be solved simultaneously at τ_0 (at which $\delta\boldsymbol{\mu}_0 = \mathbf{0}$, cf. (22)), to yield

$$\begin{bmatrix} d\boldsymbol{\upsilon} \\ d\mathbf{a} \end{bmatrix} = -\mathbf{V}_0^{-1}\mathbf{U}_0^T\delta\mathbf{x}_0 \quad \text{where} \quad \mathbf{U} := [\mathbf{R} \ \mathbf{m}] \quad \text{and} \quad \mathbf{V} := \begin{bmatrix} \mathbf{Q} & \mathbf{n} \\ \mathbf{n}^T & \boldsymbol{\alpha} \end{bmatrix} \tag{26}$$

If (26) is used at τ_0, then $\delta\boldsymbol{\lambda}_0 = \left(\mathbf{S}_0 - \mathbf{U}_0\mathbf{V}_0^{-1}\mathbf{U}_0^T\right)\delta\mathbf{x}_0$. Letting $\hat{\mathbf{S}} = \mathbf{S} - \mathbf{U}\mathbf{V}^{-1}\mathbf{U}^T$, the same sweep equation satisfied by $\mathbf{S}$ turns out to hold also for $\hat{\mathbf{S}}$, with boundary condition $\mathbf{S} \to \mathbf{0}$ as $\tau \to \tau_f \, (= 1)$. From the previous relation on $\delta\boldsymbol{\lambda}_0$ and $\delta\mathbf{x}_0$ one can conclude that $\delta\boldsymbol{\lambda}_0 \to \mathbf{0}$ as $\delta\mathbf{x}_0 \to \mathbf{0}$, unless $\hat{\mathbf{S}}$ tends to infinity at an internal time $\bar{\tau}\,(\tau_0 \leq \bar{\tau} < \tau_f)$, which is referred to as conjugate point. If $\delta\boldsymbol{\lambda}_0 \to \mathbf{0}$ and $\delta\mathbf{x}_0 \to \mathbf{0}$ then also $\delta\mathbf{u} \to \mathbf{0}$. In the end, if $\hat{\mathbf{S}} < \infty$, then no neighboring optimal path exists. This is the *Jacobi condition*. The use of $\mathbf{S}$ is not effective for the purpose of guaranteeing optimality. In fact, cases exist for which $\mathbf{S}$ becomes singular, while $\hat{\mathbf{S}}$ remains finite [7, 8], and this fully justifies the use of $\hat{\mathbf{S}}$.

It is worth remarking that, with the exception of the displacements $\{\delta\mathbf{x}, \delta\mathbf{u}, \delta\mathbf{a}, \ \delta\boldsymbol{\upsilon}, \delta\boldsymbol{\lambda}, \delta\boldsymbol{\mu}, \delta\mathbf{x}_0, \delta\mathbf{x}_f\}$, all the vectors and matrices reported in this section are evaluated along the nominal, optimal trajectory.

3 Variable-Time-Domain Neighboring Optimal Guidance

The iterative Variable-Time-Domain Neighboring Optimal Guidance (VTD-NOG) uses the optimal trajectory as the reference path, with the final intent of determining the control correction at each sampling time $\{t_k\}_{k=0,\dots,n_S}$. These are the times at which the displacement between the actual trajectory, associated with $\mathbf{x}$, and the nominal trajectory, corresponding to $\mathbf{x}^*$, is evaluated, to yield $d\mathbf{x}_k \equiv \delta\mathbf{x}_k = \mathbf{x}(t_k) - \mathbf{x}_k^*(t_k)$. The total number of sampling times, n_S, is unspecified, whereas the actual time interval between two successive sampling times is given and denoted with Δt_S, $\Delta t_S = t_{k+1} - t_k$. It is apparent that a fundamental ingredient needed to implement VTD-NOG is the formula for determining the overall time of flight $t_f^{(k)}$ at time t_k. This is equivalent to finding the time-to-go $\left(t_f^{(k)} - t_k\right)$ at t_k. The following subsection is focused on this issue.

3.1 Time-to-Go Updating Law and Termination Criterion

The fundamental principle that underlies the VTD-NOG scheme consists in finding the control correction $\delta\mathbf{u}(\tau)$ in the generic interval $[\tau_k, \tau_{k+1}]$ such that the second differential of J is minimized,

$$
\begin{aligned}
d^2 J = \int_{\tau_k}^1 & \begin{bmatrix} \delta\mathbf{x} \\ \delta\mathbf{u} \\ d\mathbf{a} \end{bmatrix}^T \begin{bmatrix} H_{\mathbf{xx}} & H_{\mathbf{xu}} & H_{\mathbf{xa}} \\ H_{\mathbf{ux}} & H_{\mathbf{uu}} & H_{\mathbf{ua}} \\ H_{\mathbf{ax}} & H_{\mathbf{au}} & H_{\mathbf{aa}} \end{bmatrix} \begin{bmatrix} \delta\mathbf{x} \\ \delta\mathbf{u} \\ d\mathbf{a} \end{bmatrix} d\tau \\
+ & \begin{bmatrix} d\mathbf{x}_k \\ d\mathbf{x}_f \\ d\mathbf{a} \end{bmatrix}^T \begin{bmatrix} \Phi_{\mathbf{x}_k\mathbf{x}_k} & \mathbf{0}_{n\times n} & \mathbf{0}_{n\times p} \\ \mathbf{0}_{n\times n} & \Phi_{\mathbf{x}_f\mathbf{x}_f} & \Phi_{\mathbf{x}_f\mathbf{a}} \\ \mathbf{0}_{p\times n} & \Phi_{\mathbf{ax}_f} & \Phi_{\mathbf{aa}} \end{bmatrix} \begin{bmatrix} d\mathbf{x}_k \\ d\mathbf{x}_f \\ d\mathbf{a} \end{bmatrix}
\end{aligned}
\tag{27}
$$

while holding the first-order expansions of the state equations, the related final conditions, and the parameter condition (i.e., the second of (22)). In contrast, the first of (22) cannot be used, because in general $\delta\boldsymbol{\mu}_k \neq \mathbf{0}$ at τ_k. Minimizing the objective (27) is equivalent to solving the accessory optimization problem, defined in the interval $[\tau_k, 1]$. This means that the relations reported in Sect. 2.2 need to be extended to the generic interval $[\tau_k, 1]$.

Other than the linear expansion of the state and costate equations, the related boundary conditions, and the second relation of (22), also Eqs. (23)–(25), (19), and (20) remain unchanged. However, now (26) is to be evaluated at τ_k and becomes

$$
\begin{bmatrix} d\boldsymbol{v} \\ d\mathbf{a} \end{bmatrix} = -\mathbf{V}_k^{-1}\mathbf{U}_k^T \delta\mathbf{x}_k - \mathbf{V}_k^{-1}\boldsymbol{\Theta}\,\delta\boldsymbol{\mu}_k, \quad \text{with} \quad \boldsymbol{\Theta} = \begin{bmatrix} \mathbf{0}_{q\times p} \\ \mathbf{I}_{p\times p} \end{bmatrix}
\tag{28}
$$

because $\delta\boldsymbol{\mu}_k \neq \mathbf{0}$ (unlike $\delta\boldsymbol{\mu}_0 = \mathbf{0}$). The latter relation supplies the corrections $d\boldsymbol{v}$ and $d\mathbf{a}$ at τ_k as functions of the gain matrices $\mathbf{U}$ and $\mathbf{V}$ (defined in (26)), evaluated at τ_k, and $\delta\boldsymbol{\mu}_k$ (coming from the numerical integration of (21) in the preceding interval $[\tau_{k-1}, \tau_k]$). Equation (28) contains the updating law of the total flight time t_f, which is included as a component of $\mathbf{a}$. Hence, if $dt_f^{(k)}$ denotes the correction on t_f^* evaluated at τ_k, then $t_f^{(k)} = t_f^* + dt_f^{(k)}$. As the sampling interval Δt_S is specified, the general formula for τ_k is

$$\tau_k = \sum_{j=0}^{k-1} \frac{\Delta t_S}{t_f^{(j)}} \tag{29}$$

The overall number of intervals n_S is found at the first occurrence of the following condition, associated with the termination of VTD-NOG:

$$\sum_{j=0}^{n_S-1} \frac{\Delta t_S}{t_f^{(j)}} \geq 1 \qquad \Rightarrow \qquad \tau_{n_S} = 1 \tag{30}$$

It is worth stressing that the updating formula (28) derives directly from the natural extension of the accessory optimization problem to the time interval $[\tau_k, 1]$. In addition, the introduction of the normalized time τ now reveals its great utility. In fact, all the gain matrices are defined in the normalized interval $[0,1]$ and cannot become singular. Moreover, the limiting values $\{\tau_k\}_{k=0,\dots,n_S-1}$ are dynamically calculated at each sampling time using (29), while the sampling instants in the actual time domain are specified and equally spaced (cf. Fig. 1). Also the termination criterion (30) has a logical, consistent definition, and corresponds to the upper bound of the interval $[0,1]$, to which τ is constrained.
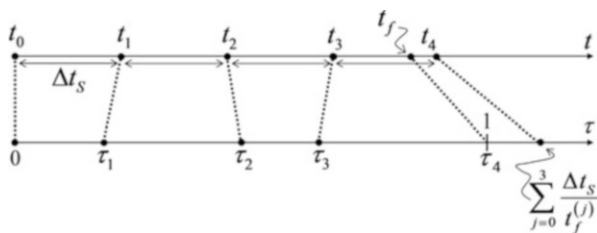

Fig. 1 Illustrative sketch of the relations between the two time domains

3.2 Modified Sweep Method

The definition of a neighboring optimal path requires the numerical backward integration of the sweep equations [7]. However, as previously remarked, the matrix $\hat{\mathbf{S}}$ has practical utility, because $\mathbf{S}$ may become singular while $\hat{\mathbf{S}}$ remains finite in the interval [0,1[. Therefore, a suitable integration technique is based on using the classical sweep equations in the interval $[\tau_{sw}, 1]$ (where τ_{sw} is sufficiently close to $\tau_f = 1$) and then switching to $\hat{\mathbf{S}}$. However, due to (28), new relations are to be derived for $\hat{\mathbf{S}}$ and the related matrices.

With this intent, the first step consists in combining (28) with (23)–(24), and leads to obtaining

$$\delta\boldsymbol{\lambda} = \left(\hat{\mathbf{S}} - \mathbf{W}\mathbf{m}^T\right)\delta\mathbf{x} - \mathbf{W}\mathbf{n}^T d\boldsymbol{\upsilon} - \mathbf{W}\boldsymbol{\alpha} d\mathbf{a}, \quad \text{with} \quad \mathbf{W} := \mathbf{U}\mathbf{V}^{-1}\boldsymbol{\Theta} \qquad (31)$$

This relation replaces (23).

Equation (31) is to be employed repeatedly in the derivation of new sweep equations. The related analytical developments are described in full detail in [15], and lead to attaining the following modified sweep equations:

$$\dot{\hat{\mathbf{S}}} = -\hat{\mathbf{S}}\mathbf{A} + \hat{\mathbf{S}}\mathbf{B}\hat{\mathbf{S}} + \left[\hat{\mathbf{S}}\mathbf{D}\boldsymbol{\alpha}^{-1} + \mathbf{W}\mathbf{F}\boldsymbol{\alpha}^{-1} + \mathbf{E}\boldsymbol{\alpha}^{-1}\right]\mathbf{m}^T - \mathbf{W}\mathbf{E}^T - \mathbf{W}\mathbf{D}^T\hat{\mathbf{S}} - \mathbf{C} - \mathbf{A}^T\hat{\mathbf{S}} \qquad (32)$$

$$\dot{\mathbf{R}}^T = \mathbf{R}^T\mathbf{B}\hat{\mathbf{S}} - \mathbf{R}^T\mathbf{A} - \mathbf{R}^T\mathbf{B}\mathbf{W}\mathbf{m}^T \qquad (33)$$

$$\dot{\mathbf{m}}^T = -\mathbf{m}^T\mathbf{A} + \mathbf{m}^T\mathbf{B}\hat{\mathbf{S}} - \mathbf{m}^T\mathbf{B}\mathbf{W}\mathbf{m}^T - \mathbf{E}^T - \mathbf{D}^T\hat{\mathbf{S}} + \mathbf{D}^T\mathbf{W}\mathbf{m}^T \qquad (34)$$

$$\dot{\mathbf{Q}} = -\mathbf{R}^T\mathbf{B}\mathbf{W}\mathbf{n}^T \qquad (35)$$

$$\dot{\mathbf{n}} = -\mathbf{R}^T\left(\mathbf{D} + \mathbf{B}\mathbf{W}\boldsymbol{\alpha}\right) \qquad (36)$$

$$\dot{\boldsymbol{\alpha}} = \mathbf{D}^T\mathbf{W}\boldsymbol{\alpha} - \mathbf{F} - \mathbf{m}^T\mathbf{B}\mathbf{W}\boldsymbol{\alpha} - \mathbf{m}^T\mathbf{D} \qquad (37)$$

The gain matrices involved in the sweep method, i.e. $\mathbf{S}$, $\hat{\mathbf{S}}$, $\mathbf{R}$, $\mathbf{Q}$, $\mathbf{n}$, $\mathbf{m}$, and $\boldsymbol{\alpha}$, can be integrated backward in two steps:

1. in $[\tau_{sw}, 1]$ the equations of the classical sweep method [7, 15], with the respective boundary conditions, are used.
2. in $[0, \tau_{sw}]$ (32) through (37) are used; $\mathbf{R}$, $\mathbf{Q}$, $\mathbf{n}$, $\mathbf{m}$, and $\boldsymbol{\alpha}$ are continuous across the time τ_{sw}, whereas $\hat{\mathbf{S}}$ is given by $\hat{\mathbf{S}} := \mathbf{S} - \mathbf{U}\mathbf{V}^{-1}\mathbf{U}^T$; in this work, τ_{sw} is set to 0.99.

3.3   Preliminary Offline Computations and Algorithm Structure

The implementation of VTD-NOG requires several preliminary computations that can be completed offline and stored in the onboard computer. First of all, the optimal trajectory is to be determined, together with the related state, costate, and control variables, which are assumed as the nominal ones. In the time domain τ these can be either available analytically or represented as sequences of equally spaced values, e.g. $\mathbf{u}_i^* = \mathbf{u}^*(\tau_i)$ $(i = 0, \ldots, n_D; \ \tau_0 = 0$ and $\tau_{n_D} = 1)$. However, in the presence of perturbations, VTD-NOG determines the control corrections $\delta\mathbf{u}(\tau)$ in each interval $[\tau_k, \tau_{k+1}]$, where the values $\{\tau_k\}$ never coincide with the equally spaced values $\{\tau_i\}$ used for $\mathbf{u}_i^*$. Hence, regardless of the number of points used to represent the control correction $\delta\mathbf{u}(\tau)$ in $[\tau_k, \tau_{k+1}]$, it is apparent that a suitable interpolation is to be adopted for the control variable $\mathbf{u}^*$ (provided that no analytical expression is available). In this way, the value of $\mathbf{u}^*$ can be evaluated at any arbitrary time in the interval $0 \le \tau \le 1$. For the same reason also the nominal state $\mathbf{x}^*$ and costate $\boldsymbol{\lambda}^*$ need to be interpolated. If a sufficiently large number of points is selected (e.g., $n_D = 1001$), then piecewise linear interpolation is a suitable option. The successive step is the analytical derivation of the matrices

$$\left\{ \begin{array}{l} \mathbf{f_x}, \mathbf{f_u}, \mathbf{f_a}, H_{\mathbf{xx}}, H_{\mathbf{xu}}, H_{\mathbf{x\lambda}}, H_{\mathbf{xa}}, H_{\mathbf{ux}}, H_{\mathbf{uu}}, H_{\mathbf{ua}}, H_{\mathbf{u\lambda}}, H_{\mathbf{ax}}, H_{\mathbf{au}}, H_{\mathbf{aa}}, H_{\mathbf{a\lambda}}, \\ \boldsymbol{\psi}_{\mathbf{x}_f}, \boldsymbol{\psi}_{\mathbf{x}_0}, \boldsymbol{\psi}_{\mathbf{a}}, \Phi_{\mathbf{x}_0\mathbf{x}_0}, \Phi_{\mathbf{x}_0\mathbf{a}}, \Phi_{\mathbf{x}_f\mathbf{x}_f}, \Phi_{\mathbf{x}_f\mathbf{a}}, \Phi_{\mathbf{ax}_f}, \Phi_{\mathbf{aa}} \end{array} \right\} \quad (38)$$

Then, they are evaluated along the nominal trajectory and linearly interpolated, as well as $\mathbf{A}$, $\mathbf{B}$, $\mathbf{C}$, $\mathbf{D}$, $\mathbf{E}$, and $\mathbf{F}$, whose expressions are reported in [17]. Subsequently, the two-step backward integration of the sweep equations described in Sect. 3.2 is performed, and yields the gain matrices $\hat{\mathbf{S}}$, $\mathbf{R}$, $\mathbf{m}$, $\mathbf{Q}$, $\mathbf{n}$, and $\boldsymbol{\alpha}$, using also the analytic expressions of $\mathbf{W}$, $\mathbf{U}$, and $\mathbf{V}$. The linear interpolation of all the matrices not yet interpolated concludes the preliminary computations.

On the basis of the optimal reference path, at each time τ_k the VTD-NOG algorithm determines the time of flight and the control correction. More specifically, after setting the actual sampling time interval Δt_S, at each τ_k $(k = 0, \ldots, n_S - 1; \ \tau_0 = 0)$ the following steps implement the feedback guidance scheme:

1. Evaluate $\delta\mathbf{x}_k$.
2. Assume the value of $\delta\boldsymbol{\mu}$ calculated at the end of the previous interval $[\tau_{k-1}, \tau_k]$ as $\delta\boldsymbol{\mu}_k$ $(\delta\boldsymbol{\mu}_0 = \mathbf{0})$.
3. Calculate the correction $dt_f^{(k)}$ and the updated time of flight $t_f^{(k)}$.
4. Calculate the limiting value τ_{k+1}.
5. Evaluate $\delta\boldsymbol{\lambda}_k$.
6. Integrate numerically the linear differential system composed of (14), (15), and (21).
7. Determine the control correction $\delta\mathbf{u}(\tau)$ in $[\tau_k, \tau_{k+1}]$ through (20).
8. Points 1 through 7 are repeated after increasing k by 1, until (30) is satisfied.
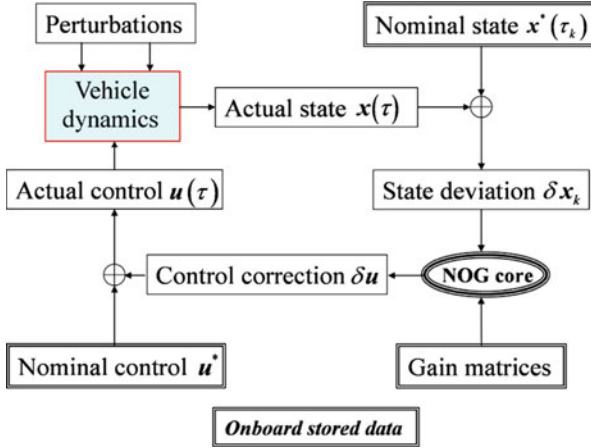
Fig. 2 Block diagram of VTD-NOG

Figure 2 portrays a block diagram that illustrates the feedback structure of VTD-NOG. The control and flight time corrections depend on the state displacement $\delta\mathbf{x}$ (evaluated at specified times) through the time-varying gain matrices, computed offline and stored onboard.

4   Minimum-Time-to-Climb Path of a Boeing 727 Aircraft

As a first example, the variable-time-domain neighboring optimal guidance is applied to the minimum-time ascent path of a Boeing 727 aircraft, which is a mid-size commercial jet aircraft. Its propulsive and aerodynamics characteristics are interpolated on the basis of real data, and come from [2].

4.1   Problem Definition

The aircraft motion is assumed to occur in the vertical plane. In addition, due to the low flight altitude, the flat-Earth approximation is adopted, and the gravitational force is considered constant ($g = 9.80665 \, \text{m/s}^2$). In light of these assumptions, the equations of motion are [20]

$$z\prime = v \sin \gamma \tag{39}$$
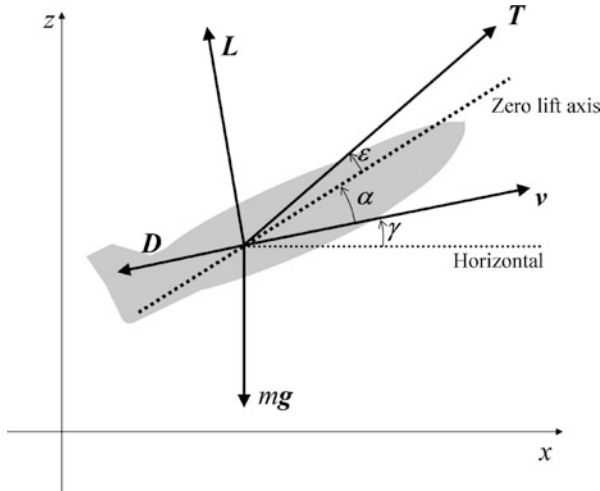
$$x\prime = v \cos \gamma \tag{40}$$

Fig. 3 Thrust, lift, and drag forces, with the related angles

$$\gamma\prime = -\frac{g}{v}\cos\gamma + \frac{L}{mv} + \frac{T}{mv}\sin(\alpha + \epsilon) \tag{41}$$

$$v\prime = -g\sin\gamma - \frac{D}{m} + \frac{T}{m}\cos(\alpha + \epsilon) \tag{42}$$

where z, x, γ, and v denote, respectively, the altitude, range, flight path angle, and velocity of the aircraft at hand, and $\prime$ is the derivative with respect to the actual time t; D and L represent the magnitudes of the aerodynamic drag and lift (whose direction is illustrated in Fig. 3), whereas T denotes the thrust magnitude, and m is the mass, which is assumed constant and equal to 81,647 kg. The angle ϵ is portrayed in Fig. 3 as well, and identifies the thrust direction with respect to the zero lift axis; α is the angle of attack. The two aerodynamic forces are functions of (1) the (dimensionless) lift and drag coefficients c_L and c_D, (2) the atmospheric density ρ, (3) the instantaneous velocity v, and (4) the reference surface area S, equal to 145 m^2 [2], according to the following relations:

$$L = \frac{1}{2}c_L\rho Sv^2 \quad \text{and} \quad D = \frac{1}{2}c_D\rho Sv^2 \tag{43}$$

Due to low altitude, the atmospheric density at sea level is used for the entire time of flight. The two coefficients c_L and c_D depend on the angle of attack and are interpolated in the following fashion [2]:

$$c_L = c_{L0} + c_{L1}\alpha + c_{L2}(\alpha - \alpha_1)^2 \tag{44}$$

$$c_D = c_{D0} + c_{D1}\alpha + c_{D2}\alpha^2 \tag{45}$$

where the values of the constant quantities $\{c_{L0}, c_{L1}, c_{L2}, \alpha_1, c_{D0}, c_{D1}, c_{D2}\}$ are reported in [2]. Lastly, the thrust magnitude T depends on the instantaneous velocity, and is interpolated as well,

$$T = c_{T0} + c_{T1}v + c_{T2}v^2 \tag{46}$$

where the constant values of $\{c_{T0}, c_{T1}, c_{T2}\}$ are again reported in [2].

The minimum-time-to-climb problem consists in finding the optimal time history of the control angle α that minimizes the time t_f needed to reach a given altitude in horizontal flight, with a prescribed velocity. This means that the objective function is simply

$$J = t_f \tag{47}$$

(the initial time is set to 0). The final conditions are partially specified,

$$z_f = \bar{z}_f \,(= 609.6\,\text{m}), \quad \gamma_f = 0\,\text{deg}, \quad v_f = \bar{v}_f \,(= 128\,\text{m/s}) \tag{48}$$

whereas the initial conditions are completely known,

$$z_0 = 0\,\text{m}, \quad x_0 = 0\,\text{m}, \quad \gamma_0 = 0\,\text{deg}, \quad v_0 = 128\,\text{m/s} \tag{49}$$

As the range x does not appear in the right-hand sides of the equations of motion nor in the final conditions, x is ignorable; as a result, the state vector $\mathbf{x}$ is given by $\mathbf{x} = [z \quad \gamma \quad v]^T$, while the control vector $\mathbf{u}$ includes only α ($\mathbf{u} \equiv \alpha$). The state equations can be rewritten in terms of τ-derivatives,

$$\dot{z} = t_f v \sin \gamma \tag{50}$$

$$\dot{\gamma} = t_f \left[-\frac{g}{v} \cos \gamma + \frac{L}{mv} + \frac{T}{mv} \sin(\alpha + \epsilon) \right] \tag{51}$$

$$\dot{v} = t_f \left[-g \sin \gamma - \frac{D}{m} + \frac{T}{m} \cos(\alpha + \epsilon) \right] \tag{52}$$

The right-hand sides of (50)–(52) form the vector $\mathbf{f}$.

4.2 Optimal Trajectory

The VTD-NOG algorithm requires the preliminary determination of the optimal trajectory, which is assumed as the nominal path, together with the related optimal control, state, and costate vectors (cf. Sect. 2).

For the dynamical system at hand the Hamiltonian H and the function Φ are

$$H = \lambda_1 t_f v \sin\gamma + \lambda_2 t_f \left[-\frac{g}{v}\cos\gamma + \frac{L}{mv} + \frac{T}{mv}\sin(\alpha + \epsilon) \right]$$

$$+ \lambda_3 t_f \left[-g\sin\gamma - \frac{D}{m} + \frac{T}{m}\cos(\alpha + \epsilon) \right] \tag{53}$$

$$\Phi = \upsilon_1(z_f - \bar{z}_f) + \upsilon_2\gamma_f + \upsilon_3(v_f - \bar{v}_f) \tag{54}$$

The adjoint equations assume the form

$$\dot{\lambda}_1 = 0 \quad \Rightarrow \quad \lambda_1 = \lambda_{1,0} \tag{55}$$

$$\dot{\lambda}_2 = t_f \left[-\lambda_1 v\cos\gamma - \frac{\lambda_2 g}{v}\sin\gamma + \lambda_3 g\cos\gamma \right] \tag{56}$$

$$\dot{\lambda}_3 = t_f \left\{ -\lambda_1\sin\gamma - \frac{\lambda_2}{mv}\left[\frac{\partial T}{\partial v}\sin(\alpha + \epsilon) + \frac{\partial L}{\partial v} - \frac{T}{v}\sin(\alpha + \epsilon) - \frac{L}{v} \right] \right\}$$

$$+ t_f \left\{ -\lambda_2\frac{g\cos\gamma}{v^2} - \lambda_3\left[-\frac{\partial D}{\partial v} + \frac{\partial T}{\partial v}\cos(\alpha + \epsilon) \right] \right\} \tag{57}$$

The respective boundary conditions (10) do not add any further information (since the state components are completely specified at the final time), and therefore they are not reported. It is worth remarking that the derivatives of L, D, and T with respect to α and v can be easily expressed using (43)–(46) and are continuous. Moreover, the fact that H is stationary with respect to α at the optimal solution yields

$$\lambda_2\frac{T}{mv}\cos(\alpha + \epsilon) + \frac{\lambda_2}{mv}\frac{\partial L}{\partial\alpha} - \frac{\lambda_3}{m}\frac{\partial D}{\partial\alpha} - \lambda_3\frac{T}{m}\sin(\alpha + \epsilon) = 0 \tag{58}$$

Finally, the parameter condition (11) leads to

$$\int_0^1 \boldsymbol{\lambda}^T\frac{\partial \mathbf{f}}{\partial t_f}d\tau + 1 = 0 \tag{59}$$

However, the parameter condition can be proven to be ignorable. As a first step, the components of $\boldsymbol{\lambda}$ are homogeneous in the adjoint equations (55)–(57). This circumstance implies that if an optimization algorithm is capable of finding some initial value of $\boldsymbol{\lambda}$ such that $\boldsymbol{\lambda}_0 = k_\lambda\boldsymbol{\lambda}_0^*$ ($k_\lambda > 0$) (where * denotes optimality), then the same proportionality holds between $\boldsymbol{\lambda}$ and $\boldsymbol{\lambda}^*$ at any τ. Moreover, the control $\mathbf{u}$ can be found through (58), which yields the same solution if $\boldsymbol{\lambda}$ replaces $\boldsymbol{\lambda}^*$. This circumstance implies that if $\boldsymbol{\lambda}$ is proportional to $\boldsymbol{\lambda}^*$ then the final conditions are fulfilled at the minimum final time t_f^*. In contrast, the parameter condition is violated, because the integral of (59) turns out to be

$$\int_0^1 \boldsymbol{\lambda}^T \frac{\partial \mathbf{f}}{\partial t_f} d\tau = k_\lambda \int_0^1 \boldsymbol{\lambda}^{*T} \frac{\partial \mathbf{f}}{\partial t_f} d\tau = -k_\lambda \neq -1 \tag{60}$$

Therefore, provided that the proportionality condition holds, the optimal control $\mathbf{u}^*$ can be determined without considering the parameter condition, which is ignorable as an equality constraint and can be replaced by the inequality constraint

$$\int_0^1 \boldsymbol{\lambda}^T \frac{\partial \mathbf{f}}{\partial t_f} d\tau < 0 \tag{61}$$

because k_λ is an arbitrary positive constant. Once the (nonoptimal) values of the costate variables (fulfilling the proportionality condition) have been determined, the correct (optimal) values can be recovered after calculating k_λ by means of (60).

In the end, the problem of determining the minimum-time-to-climb path can be reformulated as a two-point boundary-value problem, in which the unknowns are the initial values of three adjoint variables, as well as the time of flight, i.e. $\{\lambda_{1,0}, \lambda_{2,0}, \lambda_{3,0}, t_f\}$. The boundary conditions are represented by (48), accompanied by the inequality constraint (61). Once the optimal parameter set has been determined, the state and costate equations can be integrated, using (58) to express the control angle α as a function of the adjoint variables.

The optimal parameter set is determined by means of a simple implementation of swarming algorithm (PSO). This is a heuristic optimization technique, based on the use of a population of individuals (or particles). Selection of the globally optimal parameters is the result of a number of iterations, in which the individuals share their information. This optimization approach is extremely intuitive and easy-to-implement. Nevertheless, in the scientific literature several papers [10–14] prove that the use of this method is effective for solving trajectory optimization problems. A set of canonical units is employed for the problem at hand: the distance unit (DU) and time unit (TU) are

$$DU = \frac{2m}{\rho S} \quad \text{and} \quad TU = \sqrt{\frac{2m}{\rho g S}} \tag{62}$$

The search space is defined by the inequalities $-1 \leq \lambda_{k,0} \leq 1$ and $1 \, \text{TU} \leq t_f \leq 6 \, \text{TU}$. It is worth remarking that ignorability of the parameter condition allows defining arbitrarily the range in which the initial values of the adjoint variables are sought. Their correct values (fulfilling also the parameter condition (59)) can be recovered a posteriori, as discussed previously. PSO is used with the intent of minimizing the objective
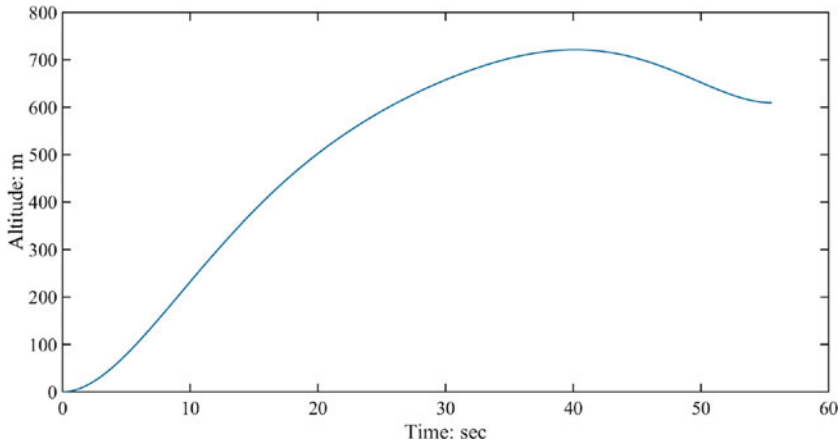
$$\tilde{J} = \sum_{k=1}^3 |d_k| \tag{63}$$

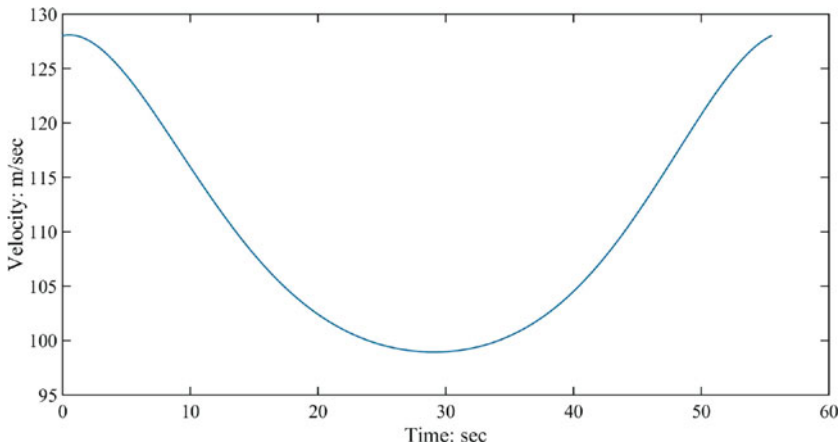**Fig. 4** 727 aircraft: nominal altitude

Fig. 5 727 aircraft: nominal velocity

where each term d_k represents a final constraint violation. While minimization of (63) ensures feasibility, enforcement of the necessary conditions for optimality guarantees that the solution found by PSO is (at least locally) optimal.

The PSO algorithm yields a solution associated with $\tilde{J} = 8.458 \times 10^{-7}$. The corresponding optimal time histories of the state and control components are portrayed in Figs. 4, 5, 6, and 7. From their inspection it is apparent that the minimum climbing path is composed of two phases: an initial ascent phase, up to an altitude greater than the final one, followed by a diving phase. The minimum time to climb equals 55.5 s.
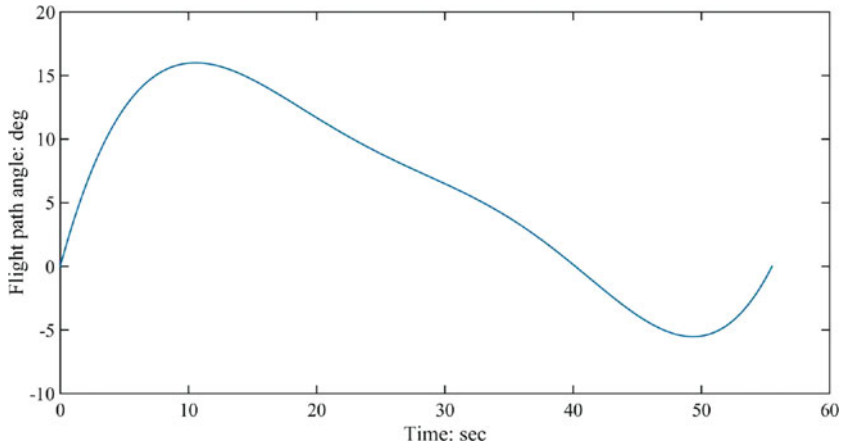
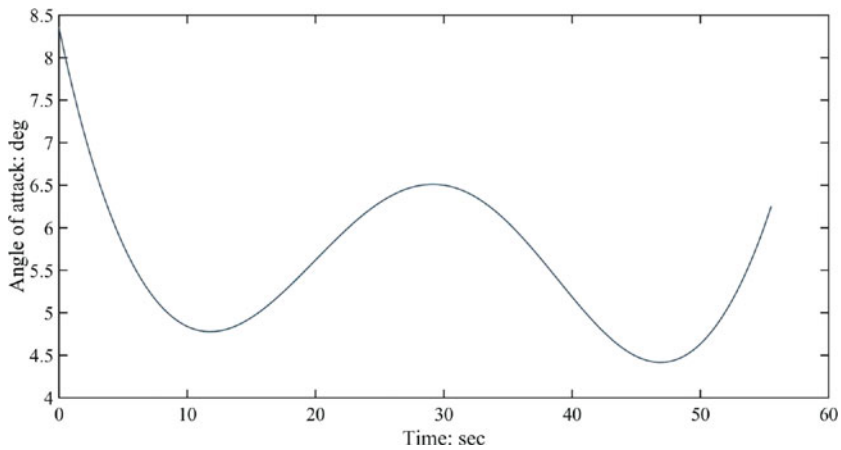**Fig. 6** 727 aircraft: nominal flight path angle

Fig. 7 727 aircraft: optimal control time history

4.3 Application of VTD-NOG

The neighboring optimal guidance algorithm proposed in this work is applied to the minimum-time-to-climb path of the Boeing 727 aircraft. Perturbations from the nominal situation are considered, in order to simulate a realistic scenario. In particular, perturbations on the initial state, thrust magnitude, and atmospheric density are taken into account. Several Monte Carlo campaigns are run, with the intent of obtaining some useful statistical information on the accuracy of the algorithm at hand, in the presence of the previously mentioned deviations, which are simulated stochastically. Monte Carlo campaigns test the guidance algorithm by running a significant number of numerical simulations. Each perturbed quantity in

the initial state is associated with a Gaussian distribution, with mean value equal to the respective nominal one and with a specific σ-value, which is related to the statistical dispersion about the mean value. The σ-values are

$$\Delta z^{(\sigma)} = 5\,\text{m}, \quad \Delta v^{(\sigma)} = 5\,\text{m/s}, \quad \Delta \gamma^{(\sigma)} = 5\,\text{deg} \tag{64}$$

In the numerical simulations the deviations from the nominal values are constrained to the intervals $\left[0, 2\Delta\chi^{(\sigma)}\right]$ ($\chi = z$ or γ) and $\left[-2\Delta v^{(\sigma)}, 2\Delta v^{(\sigma)}\right]$. A different approach was chosen for the perturbation of the thrust magnitude. In fact, usually the thrust magnitude exhibits small fluctuations. This time-varying behavior is modeled through a trigonometric series with random coefficients,

$$T_{pert} = T \left\{ 1 + \sum_{k=1}^{5} a_k \sin(2k\pi\tau) + \sum_{k=1}^{5} a_{k+5} \cos(2k\pi\tau) \right\} \tag{65}$$

where T_{pert} denotes the perturbed thrust, whereas a_k represents a random number with Gaussian distribution, zero mean, and standard deviation equal to 0.01. The atmospheric density fluctuations are modeled through a similar trigonometric series,

$$\rho_{pert} = \rho \left\{ 1 + \sum_{k=1}^{5} b_k \sin(2k\pi\tau) + \sum_{k=1}^{5} b_{k+5} \cos(2k\pi\tau) \right\} \tag{66}$$

where ρ_{pert} denotes the perturbed density, whereas b_k represents a random number with the same statistic properties as a_k.

At the end of the algorithmic process described in Sect. 3.3, two statistical quantities are evaluated, i.e. the mean value and the standard deviation for all of the outputs of interest. The symbols $\bar{\chi}$ and $\chi^{(\sigma)}$ will denote the mean value and standard deviation of χ ($\chi = z$ or v or γ or t_f) henceforth. Five campaigns are performed, each including 100 runs. The first four campaigns (MC1 through MC4) use a sampling time $\Delta t_S = 2\,\text{s}$, whereas MC5 adopts a sampling time $\Delta t_S = 1\,\text{s}$. MC1 assumes only perturbations of the initial state. The second campaign (MC2) considers the thrust fluctuations, whereas MC3 takes into account only the atmospheric density perturbations. MC4 and MC5 include all the deviations from the nominal flight conditions (with different sampling times). Table 1 summarizes the results for the five Monte Carlo campaigns and reports the related statistics. Application of VTD-NOG to the problem of interest leads to excellent results, with modest errors on the desired final conditions. More specifically, inspection of Table 1 points out that errors on the initial conditions are corrected very effectively; however, also the remaining results exhibit modest deviations at the final time. The latter is extremely close to the optimal value, and this is an intrinsic characteristic of VTD-NOG, which employs first order expansions of the state and costate equations in the proximity of the optimal solution. As a final remark, in the presence of all of

Table 1 Statistics on the time of flight and the errors on altitude, velocity, and flight path angle

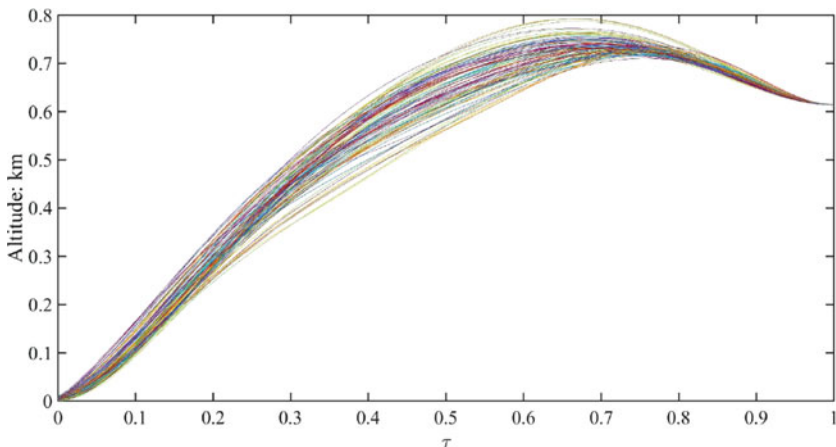| Statistics	MC1	MC2	MC3	MC4	MC5
$\Delta \bar{z}_f$ (cm)	1.2	−11.5	8.8	−5.3	1.1
$\Delta z_f^{(\sigma)}$ (cm)	6.3	39.7	14.9	57.3	22.4
$\Delta \bar{v}_f$ (cm/s)	−6.1	−5.9	−6.7	−14.7	−9.7
$\Delta v_f^{(\sigma)}$ (cm/s)	1.7	6.9	6.6	41.6	13.2
$\Delta \bar{\gamma}_f$ (deg)	−0.225	−0.232	−0.268	−0.487	−0.304
$\Delta \gamma_f^{(\sigma)}$ (deg)	0.099	0.166	0.123	1.205	0.356
$\bar{t}_f$ (s)	53.79	53.40	55.34	53.77	53.76
$t_f^{(\sigma)}$ (s)	4.97	0.13	0.33	4.95	4.92

Fig. 8 727 aircraft: altitude time histories obtained in MC5

the perturbations (MC4 and MC5), decreasing the sampling time implies a reduction of the final errors. Figures 8, 9, 10, and 11 depict the perturbed state components and control angle, obtained in MC5.

5 Interception

As a second application of VTD-NOG, this section considers the interception of a target by a maneuvering vehicle in exoatmospheric flight or in the presence of negligible aerodynamic forces, e.g. an intercepting rocket operating at high altitudes. Both the pursuing vehicle and the target are modeled as point masses, in the context of a three-degree-of-freedom problem.
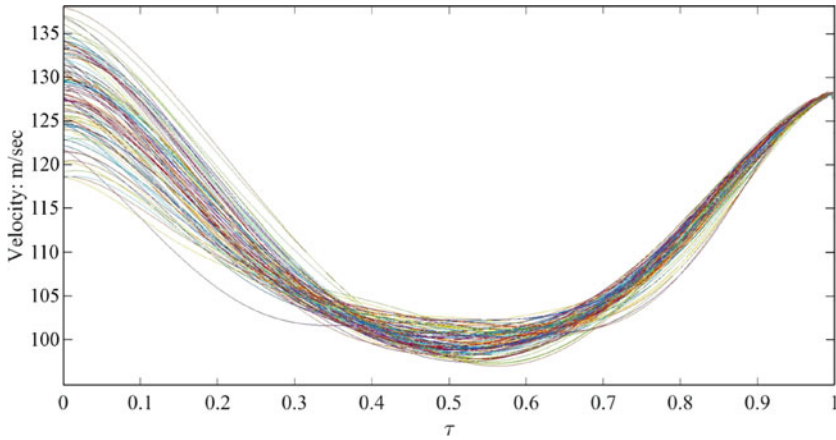
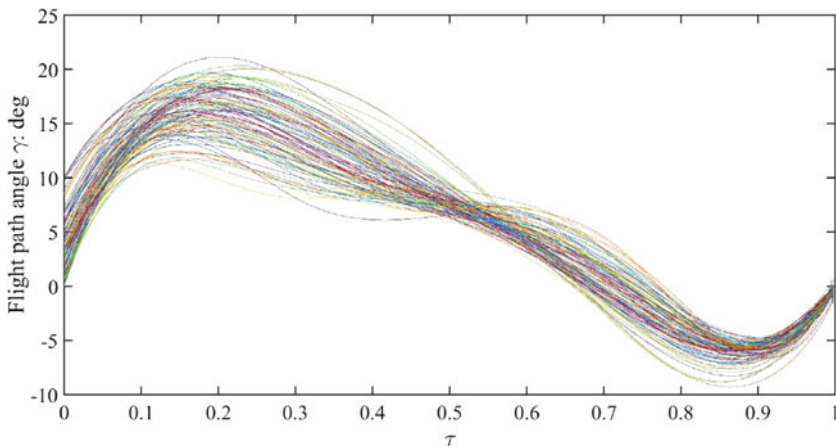**Fig. 9** 727 aircraft: velocity time histories obtained in MC5

Fig. 10 727 aircraft: time histories of the flight path angle obtained in MC5

5.1 Problem Definition

Under the assumption that interception occurs in a sufficiently short time interval, the flat Earth approximation can be adopted again. This means that the Cartesian reference frame can be defined as follows: the x_1-axis is aligned with the local upward direction, the x_2-axis is directed eastward, and the x_3-axis is aligned with the local North direction. As a result, the Cartesian equations of motion for the intercepting rocket are
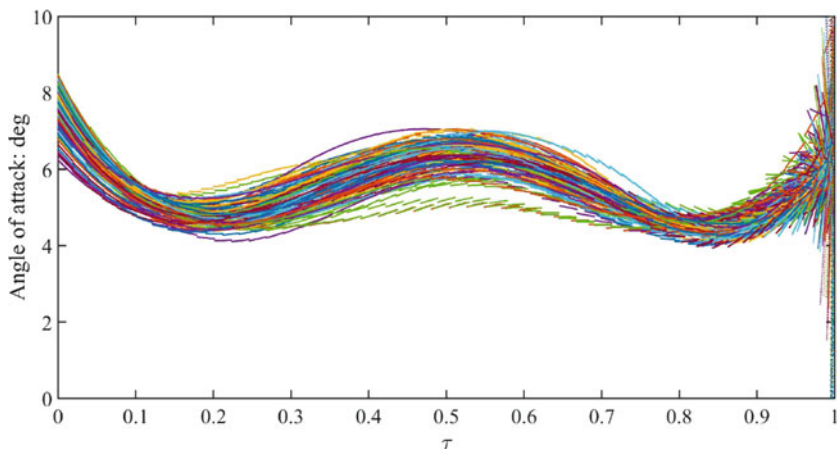
$$\dot{x}_1 = t_f x_4 \tag{67}$$

Fig. 11 727 aircraft: time histories of the angle of attack obtained in MC5

$$\dot{x}_2 = t_f x_5 \tag{68}$$

$$\dot{x}_3 = t_f x_6 \tag{69}$$

$$\dot{x}_4 = t_f(-g + a_T \cos u_2 \sin u_1) \tag{70}$$

$$\dot{x}_5 = t_f a_T \cos u_2 \cos u_1 \tag{71}$$

$$\dot{x}_6 = t_f a_T \sin u_2 \tag{72}$$

where the derivatives are written with respect to τ, $\{x_1, x_2, x_3\}$ are the three position coordinates, $\{x_4, x_5, x_6\}$ are the corresponding velocity components, and t_f represents the time of flight up to interception. The symbol g denotes the magnitude of the (constant) gravitational force at the reference altitude, whereas the thrust acceleration has magnitude a_T and direction identified through the two angles u_1 and u_2. The target position is assumed as known, and therefore it is expressed by three specified functions of the dimensionless time τ,

$$x_1^{(T)} = f_1(\tau), \qquad x_2^{(T)} = f_2(\tau), \qquad x_3^{(T)} = f_3(\tau) \tag{73}$$

While the state vector contains the position and velocity components $\{x_i\}_{i=1,\ldots,6}$, the control vector is $\mathbf{u} = [u_1 \quad u_2]^T$, and the parameter vector $\mathbf{a}$ includes only t_f (as in the previous application). As the time is to be minimized, the objective function is

$$J = t_f \tag{74}$$

5.2 Optimal Trajectory

The first-order conditions for optimality are obtained after introducing the Hamiltonian H and the function of the boundary conditions Φ, according to (7)

$$H = \lambda_1 t_f x_4 + \lambda_2 t_f x_5 + \lambda_3 t_f x_6 + \lambda_4 t_f (-g + a_T \cos u_2 \sin u_1)$$

$$+ \lambda_5 t_f a_T \cos u_2 \cos u_1 + \lambda_6 t_f a_T \sin u_2 \tag{75}$$

$$\Phi = \upsilon_1 \left[x_{1f} - f_1(1) \right] + \upsilon_2 \left[x_{2f} - f_2(1) \right] + \upsilon_3 \left[x_{3f} - f_3(1) \right] \tag{76}$$

where the subscript f refers to the value of the respective variable at the final time. The adjoint equations (9) in conjunction with the respective boundary conditions (10) for λ_4 through λ_6 lead to

$$\dot{\lambda}_1 = 0 \quad \Rightarrow \quad \lambda_1 = \lambda_{1,0} \tag{77}$$

$$\dot{\lambda}_2 = 0 \quad \Rightarrow \quad \lambda_2 = \lambda_{2,0} \tag{78}$$

$$\dot{\lambda}_3 = 0 \quad \Rightarrow \quad \lambda_3 = \lambda_{3,0} \tag{79}$$

$$\dot{\lambda}_4 = -\lambda_{1,0} t_f \quad \Rightarrow \quad \lambda_4 = \lambda_{1,0} t_f (1 - \tau) \tag{80}$$

$$\dot{\lambda}_5 = -\lambda_{2,0} t_f \quad \Rightarrow \quad \lambda_5 = \lambda_{2,0} t_f (1 - \tau) \tag{81}$$

$$\dot{\lambda}_6 = -\lambda_{3,0} t_f \quad \Rightarrow \quad \lambda_6 = \lambda_{3,0} t_f (1 - \tau) \tag{82}$$

where $\lambda_{i,0}$ denotes the (unknown) initial value of the adjoint variable λ_i. Then, the Pontryagin minimum principle yields

$$u_2 = -\arcsin \frac{\lambda_{3,0}}{\sqrt{\lambda_{1,0}^2 + \lambda_{2,0}^2 + \lambda_{3,0}^2}} \tag{83}$$

$$\sin u_1 = -\frac{\lambda_{1,0}}{\sqrt{\lambda_{1,0}^2 + \lambda_{2,0}^2}} \quad \text{and} \quad \cos u_1 = -\frac{\lambda_{2,0}}{\sqrt{\lambda_{1,0}^2 + \lambda_{2,0}^2}} \tag{84}$$

These relations imply that the optimal thrust direction is time-independent, regardless of the (known) target position. It is relatively straightforward to prove that for the present application the remaining necessary conditions coming from (10) are

useless for the purpose of identifying the optimal solution, in the sense that they do not lead to establishing any new relation among the unknowns of the problem, i.e., $\{\lambda_{1,0}, \lambda_{2,0}, \lambda_{3,0}, t_f\}$. Also the parameter condition (11) can be proven to be ignorable, in a way similar to that used in Sect. 4.2. Moreover, as the two angles u_1 and u_2 are constant, they can be considered as the unknown quantities in place of $\{\lambda_{1,0}, \lambda_{2,0}, \lambda_{3,0}\}$. Under the assumption that a_T is constant, integration of (67)–(72) leads to obtaining the following explicit solution for x_1, x_2, and x_3:

$$x_1 = x_{1,0} + x_{4,0}t_f\tau + \frac{1}{2}a_T(t_f\tau)^2 \cos u_2 \sin u_1 - \frac{1}{2}g(t_f\tau)^2 \tag{85}$$

$$x_2 = x_{2,0} + x_{5,0}t_f\tau + \frac{1}{2}a_T(t_f\tau)^2 \cos u_2 \cos u_1 \tag{86}$$

$$x_3 = x_{3,0} + x_{6,0}t_f\tau + \frac{1}{2}a_T(t_f\tau)^2 \sin u_2 \tag{87}$$

These expressions are evaluated at $\tau = 1$, then they are set equal to the respective position coordinates of the target at $\tau = 1$. From (85)–(87) one obtains the following equations:

$$(a_T^2 - g^2)t_f^4 + 4gx_{4,0}t_f^3 - 4t_f^2(x_{4,0}^2 + x_{5,0}^2 + x_{6,0}^2) - 4gt_f^2 [f_1(1) - x_{1,0}]$$

$$+8x_{4,0}t_f [f_1(1) - x_{1,0}] + 8x_{5,0}t_f [f_2(1) - x_{2,0}] + 8x_{6,0}t_f [f_3(1) - x_{3,0}] \tag{88}$$

$$-4 [f_1(1) - x_{1,0}]^2 - 4 [f_2(1) - x_{2,0}]^2 - 4 [f_3(1) - x_{3,0}]^2 = 0$$

$$u_2 = \arcsin \frac{2 [f_3(1) - x_{3,0}] - 2x_{6,0}t_f}{a_T t_f^2} \tag{89}$$

$$\cos u_1 = \frac{2 [f_2(1) - x_{2,0}] - 2x_{5,0}t_f}{a_T t_f^2 \cos u_2} \quad \text{and} \quad \sin u_1 = \frac{2 [f_1(1) - x_{1,0}] - 2x_{4,0}t_f + gt_f^2}{a_T t_f^2 \cos u_2} \tag{90}$$

Depending on the analytical form of f_1, f_2, and f_3, (88) can either represent a transcendental equation or simplify to a polynomial equation of fourth degree. Once (88) has been solved, calculation of the optimal thrust angles is straightforward, by means of (89) and (90).

5.3 Application of VTD-NOG

The guidance algorithm described in this work is applied to the interception problem in the presence of nonnominal flight conditions. In particular, perturbations on the initial state and oscillations of the thrust acceleration magnitude over time are

modeled. Several Monte Carlo campaigns are run, with the intent of obtaining some useful statistical information on the accuracy of the algorithm at hand, in the presence of the previously mentioned deviations, which are simulated stochastically. The nominal initial position is perturbed by a random vector $\boldsymbol{\rho}$: its magnitude ρ is associated with a Gaussian distribution, with standard deviation equal to 5 m and maximal value never exceeding 10 m, whereas the corresponding unit vector $\hat{\rho}$ has direction uniformly distributed over a unit sphere. Similarly, the nominal initial velocity is perturbed by a random vector $\mathbf{w}$: its magnitude w is associated with a Gaussian distribution, with standard deviation equal to 5 m/s and maximal value never exceeding 10 m/s, whereas the corresponding unit vector $\hat{w}$ has direction uniformly distributed over a unit sphere. A different approach is adopted for the perturbation of the thrust acceleration. As the thrust magnitude (and the related acceleration a_T) exhibits fluctuations, the perturbed thrust acceleration is modeled through a trigonometric series,

$$a_{T,pert} = a_T \left\{ 1 + \sum_{k=1}^{5} c_k \sin(2k\pi\tau) + \sum_{k=1}^{5} c_{k+5} \cos(2k\pi\tau) \right\} \tag{91}$$

where a_T is the nominal thrust acceleration, whereas the corresponding (time-varying) perturbed value $a_{T,pert}$ is actually used in the MC simulations. The coefficients $\{c_k\}_{k=1,\dots,10}$ have a random Gaussian distribution with zero mean and a standard deviation equal to 0.01. At the end of the algorithmic process described in Sect. 3.3, two statistical quantities are evaluated, i.e. the mean value and the standard deviation for all of the outputs of interest (with a notation similar to that adopted for the preceding application).

In this section, three different targets are taken into account. For each of them, four Monte Carlo campaigns have been performed, each including 100 runs. The first campaign (MC1) assumes only perturbations of the initial state. The second campaign (MC2) considers only oscillations of the thrust acceleration magnitude, while the third and fourth campaigns (MC3 and MC4) include both types of perturbations, with different sampling time intervals ($\Delta t_S = 1$ s and $\Delta t_S = 0.5$ s, respectively).

Fixed Target As a first special case, a fixed target is considered. This means that the three functions f_1, f_2, and f_3 equal three prescribed values $x_1^{(T)}, x_2^{(T)}$, and $x_3^{(T)}$. As a result, (88) assumes the form of a fourth degree equation; its smallest real root represents the minimum time to interception.

In the numerical example that follows, the reference altitude (needed for defining the value of g) is set to the initial altitude of the intercepting rocket, whereas $a_T = 2g$. The initial state of the pursuing vehicle is

$$x_{1,0} = 30 \text{ km} \quad x_{2,0} = 0 \text{ km} \quad x_{3,0} = 3 \text{ km} \quad x_{4,0} = x_{5,0} = x_{6,0} = 0.1 \text{ km/s} \tag{92}$$
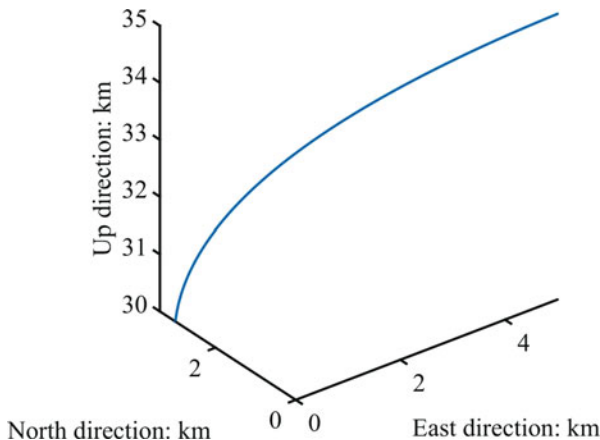
Fig. 12 Optimal interception trajectory of a fixed target

Table 2 Fixed target: statistics on the time of flight and the miss distance

	Δt_S (s)	$\bar{d}_f$ (m)	$d_f^{(\sigma)}$ (m)	$\bar{t}_f$ (s)	$t_f^{(\sigma)}$ (s)
MC1	1	1.71	1.34	30.58	0.22
MC2	1	1.24	0.36	30.57	0.10
MC3	1	1.74	1.04	30.60	0.25
MC4	0.5	1.05	1.50	30.60	0.27

whereas the target position is

$$x_1^{(T)} = 35 \,\text{km} \quad x_2^{(T)} = 5 \,\text{km} \quad x_3^{(T)} = 0 \,\text{km} \tag{93}$$

The minimum time of flight up to interception turns out to equal 30.58 s, whereas the two optimal thrust angles are $u_1^* = 73.3$ deg and $u_2^* = -41.8$ deg. Figure 12 portrays the optimal intercepting trajectory.

Application of VTD-NOG yields results associated with the statistics summarized in Table 2. They regard the miss distance d_f (at the end of nonnominal paths) and the time of flight. Inspection of Table 2 reveals that VTD-NOG generates accurate results, with modest values of the miss distance, which decreases by 40 % from MC3 to MC4. Figure 13 portrays the time evolution of the corrected control angles, obtained in MC3.

Falling Target The second special case assumes a target in free fall (e.g., a ballistic missile at high altitudes), with given initial conditions, denoted with $\left\{x_{i,0}^{(T)}\right\}_{i=1,\dots,6}$. This means that the three functions f_1, f_2, and f_3 are

$$f_1 = x_{1,0}^{(T)} + x_{4,0}^{(T)} t_f \tau - \frac{1}{2} g (t_f \tau)^2 \quad f_2 = x_{2,0}^{(T)} + x_{5,0}^{(T)} t_f \tau \quad f_3 = x_{3,0}^{(T)} + x_{6,0}^{(T)} t_f \tau \tag{94}$$

As a result, (88) assumes again the form of a fourth degree equation; its smallest real root represents the minimum time to interception.
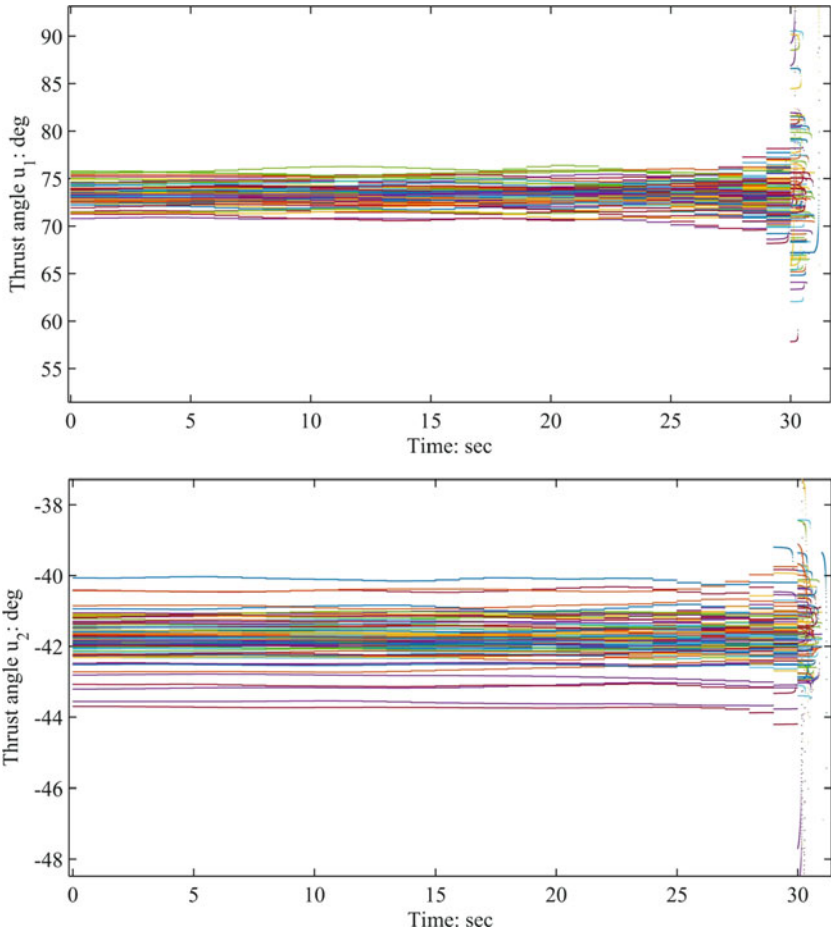
Fig. 13 Fixed target interception: time histories of the control angles obtained in MC3

In the numerical example that follows, the reference altitude is set to the initial altitude of the rocket, whereas $a_T = 3g$. The initial state of the pursuing vehicle is

$$x_{1,0} = 15\,\text{km} \quad x_{2,0} = x_{3,0} = 0\,\text{km} \quad x_{4,0} = 0\,\text{km/s} \quad x_{5,0} = x_{6,0} = 0.2\,\text{km/s} \quad (95)$$

whereas the initial state of the target is given by

$$
\begin{aligned}
x_{1,0}^{(T)} &= 30\,\text{km} \quad x_{2,0}^{(T)} = x_{3,0}^{(T)} = 1\,\text{km} \\
x_{4,0}^{(T)} &= -1\,\text{km/s} \quad x_{5,0}^{(T)} = x_{6,0}^{(T)} = 0.1\,\text{km/s}
\end{aligned}
\tag{96}
$$

The minimum time of flight up to interception turns out to equal 12.68 s, whereas the two optimal thrust angles are $u_1^* = 96.6$ deg and $u_1^* = -6.5$ deg.

Table 3 Falling target: statistics on the time of flight and the miss distance

	Δt_S (s)	$\bar{d}_f$ (m)	$d_f^{(\sigma)}$ (m)	$\bar{t}_f$ (s)	$t_f^{(\sigma)}$ (s)
MC1	1	0.46	0.27	12.67	0.03
MC2	1	1.81	1.40	12.68	0.01
MC3	1	1.83	1.41	12.67	0.03
MC4	0.5	1.47	1.12	12.68	0.03

The guidance algorithm is applied again in the presence of nonnominal flight conditions. The same Monte Carlo campaigns performed for the previous case are repeated for the application at hand. Table 3 summarizes the results for the four Monte Carlo campaigns and reports the related statistics, with regard to the miss distance (at the end of nonnominal paths) and the time of flight. It is worth remarking that decreasing the sampling time (cf. Table 3) leads again to reducing the mean miss distance. As in the previous application, the actual times of flight are extremely close to the minimal value (12.68 s).

Moving Target The third special case assumes a target that describes a circular path at constant altitude (e.g., an unmanned aerial vehicle). This means that

$$f_1 = x_1^{(T)} \qquad f_2 = x_{2C}^{(T)} + R_T \cos(\omega t_f \tau + \varphi) \qquad f_3 = x_{3C}^{(T)} + R_T \sin(\omega t_f \tau + \varphi) \quad (97)$$

where $x_1^{(T)}$ denotes the target constant altitude, $\left(x_{2C}^{(T)}, x_{3C}^{(T)}\right)$ identify the center of the circular path, whereas ω and φ define, respectively, the angular rate of rotation and the initial angular position. In this case, (88) assumes the form of a transcendental equation, to be solved numerically (for instance, through the Matlab native function *fsolve*). However, numerical solvers need a suitable approximate guess solution to converge to a refined result. For the application at hand, this guess can be easily supplied. In fact, if the radius R_T is sufficiently small, one can assume that the target is located at $\left(x_1^{(T)}, x_{2C}^{(T)}, x_{3C}^{(T)}\right)$. In the presence of a fixed target, an analytical solution exists, and derives from solving a fourth degree equation, as already explained in Sect. 5.2. This solution is used as a guess for the moving target.

In the numerical example that follows, the reference altitude is set to the initial altitude of the rocket, whereas $a_T = 3g$. The initial state of the pursuing vehicle is

$$\begin{aligned} x_{1,0} &= 9\,\text{km} \quad x_{2,0} = x_{3,0} = 0\,\text{km} \\ x_{4,0} &= 0.01\,\text{km/s} \quad x_{5,0} = 0.1\,\text{km/s} \quad x_{6,0} = 0\,\text{km/s} \end{aligned} \quad (98)$$

whereas the fundamental parameters of the target are

$$x_1^{(T)} = 10\,\text{km} \quad x_{2C}^{(T)} = x_{3C}^{(T)} = 2\,\text{km} \quad 2\pi\omega^{-1} = 60\,\text{s} \quad R_T = 0.5\,\text{km} \quad \varphi = 0 \quad (99)$$

The minimum time of flight up to interception turns out to equal 14.76 s, whereas the two optimal thrust angles are $u_1^* = 74.4\,\text{deg}$ and $u_2^* = 51.5\,\text{deg}$.

Table 4 Moving target: statistics on the time of flight and the miss distance

	Δt_S (s)	$\bar{d}_f$ (m)	$d_f^{(\sigma)}$ (m)	$\bar{t}_f$ (s)	$t_f^{(\sigma)}$ (s)
MC1	1	0.49	0.34	14.73	0.22
MC2	1	2.03	1.31	14.76	0.10
MC3	1	2.15	1.47	14.74	0.25
MC4	0.5	1.90	1.09	14.76	0.11

The guidance algorithm is applied again in the presence of nonnominal flight conditions. The same Monte Carlo campaigns performed for the previous case are repeated for the application at hand. Table 4 summarizes the results for the four Monte Carlo campaigns and reports the related statistics, with regard to the miss distance (at the end of nonnominal paths) and the time of flight. It is worth remarking that decreasing the sampling time (cf. Table 4) leads again to reducing the mean miss distance.

6 Concluding Remarks

This work describes and applies the recently introduced, general-purpose variable-time-domain neighboring optimal guidance algorithm. Usually, all the neighboring optimal guidance schemes require the preliminary determination of an optimal trajectory. Moreover, complex analytical developments accompany the implementation of this kind of perturbative guidance. However, the main difficulties encountered in former formulations of neighboring optimal guidance are the occurrence of singularities for the gain matrices and the challenging implementation of the updating law for the time-to-go. A fundamental original feature of the variable-time-domain neighboring optimal guidance is the use of a normalized time scale as the domain in which the nominal trajectory and the related vectors and matrices are defined. As a favorable consequence, the gain matrices remains finite for the entire time of flight and no extension of their domain is needed. Moreover, the updating formula for the time-to-go derives analytically from the natural extension of the accessory optimization problem associated with the original optimal control problem. This extension leads also to obtaining new equations for the sweep method, which provides all the time-varying gain matrices, computed offline and stored in the onboard computer. In this mathematical framework, the guidance termination criterion finds a logical, consistent definition, and corresponds to the upper bound of the interval to which the normalized time is constrained. Two applications are considered in the paper: (a) minimum-time-to-climb path of a Boeing 727 aircraft and (b) minimum-time exoatmospheric interception of fixed or moving targets. In both cases (especially for (a)), as well as in alternative applications already reported in the scientific literature [16, 17], the variable-time-domain neighboring optimal guidance yields very satisfactory results, with runtime (per simulation) never exceeding the time of flight. This means that VTD-NOG

actually represents an effective, general algorithm for the real-time determination of the corrective actions aimed at maintaining an aerospace vehicle in the proximity of its optimal path.

References

1. Afshari, H.H., Novinzadeh, A.B., Roshanian, J.: Determination of nonlinear optimal feedback law for satellite injection problem using neighboring optimal control. Am. J. Appl. Sci. **6**(3), 430–438 (2009)
2. Bryson, A.E.: Dynamic Optimization. Addison Wesley Longman, Boston (1999)
3. Calise, A.J., Melamed, N., Lee, S.: Design and evaluation of a three-dimensional optimal ascent guidance algorithm. J. Guid. Control. Dyn. **21**(6), 867–875 (1998)
4. Charalambous, C.B., Naidu, S.N., Hibey, J.L.: Neighboring optimal trajectories for aeroassisted orbital transfer under uncertainties. J. Guid. Control. Dyn. **18**(3), 478–485 (1995)
5. Chuang, C.-H.: Theory and computation of optimal low- and medium-thrust orbit transfers. NASA-CR-202202, NASA Marshall Space Flight Center, Huntsville (1996)
6. Hull, D.G.: Robust neighboring optimal guidance for the advanced launch system. NASA-CR-192087, Austin (1993)
7. Hull, D.G.: Optimal Control Theory for Applications. Springer, New York (2003)
8. Jo, J.-W., Prussing, J.E.: Procedure for applying second-order conditions in optimal control problems. J. Guid. Control. Dyn. **23**(2), 241–250 (2000)
9. Kugelmann, B., Pesch, H.J.: New general guidance method in constrained optimal control, part 1: numerical method. J. Optim. Theory Appl. **67**(3), 421–435 (1990)
10. Pontani, M., Conway, B.A.: Particle swarm optimization applied to space trajectories. J. Guid. Control. Dyn. **33**(5), 1429–1441 (2010)
11. Pontani, M., Conway, B.A.: Particle swarm optimization applied to impulsive orbital transfers. Acta Astronaut. **74**, 141–155 (2012)
12. Pontani, M., Conway, B.A.: Optimal finite-thrust rendezvous trajectories found via particle Swarm algorithm. J. Spacecr. Rocket. **50**(6), 1222–1234 (2013)
13. Pontani, M., Conway, B.A.: Optimal low-thrust orbital maneuvers via indirect swarming method. J. Optim. Theory Appl. **162**(1), 272–292 (2014)
14. Pontani, M., Ghosh, P., Conway, B.A.: Particle swarm optimization of multiple-burn rendezvous trajectories. J. Guid. Control. Dyn. **35**(4), 1192–1207 (2012)
15. Pontani, M., Cecchetti, G., Teofilatto, P.: Variable-time-domain neighboring optimal guidance, part 1: algorithm structure. J. Optim. Theory Appl. **166**(1), 76–92 (2015)
16. Pontani, M., Cecchetti, G., Teofilatto, P.: Variable-time-domain neighboring optimal guidance, part 2 application to lunar descent and soft landing. J. Optim. Theory Appl. **166**(1), 93–114 (2015)
17. Pontani, M., Cecchetti, G., Teofilatto, P.: Variable-time-domain neighboring optimal guidance applied to space trajectories. Acta Astronaut. **115**, 102–120 (2015)
18. Seywald, H., Cliff, E.M.: Neighboring optimal control based feedback law for the advanced launch system. J. Guid. Control. Dyn. **17**(3), 1154–1162 (1994)
19. Teofilatto, P., De Pasquale, E.: A non-linear adaptive guidance algorithm for last-stage launcher control. J. Aerosp. Eng. **213**, 45–55 (1999)
20. Vinh, N.X.: Optimal Trajectories in Atmospheric Flight. Elsevier, New York (1981)
21. Yan, H., Fahroo, F., Ross, I.: Real-time computation of neighboring optimal control laws. AIAA Guidance, Navigation and Control Conference, Monterey. AIAA Paper 2002–465 (2002)