V-bar and R-bar Glideslope Guidance Algorithms for Fixed-Time Rendezvous: A Linear Programming Approach

Yassine Ariba, Denis Arzelier, Laura Sofia Urbina, Christophe Louembet

V-bar and R-bar Glideslope Guidance Algorithms for Fixed-Time Rendezvous: A Linear Programming Approach

Yassine Ariba [*,**], **Denis Arzelier** [**], **Laura Sofia Urbina** [**], **Christophe Louembet** [***],

[*] *ICAM, 75 avenue de Grande Bretagne, 31300 Toulouse, France (email: yassine.ariba@icam.fr)*
[**] *CNRS, LAAS, 7 avenue du colonel Roche, F-31400 Toulouse, France and Univ de Toulouse, LAAS, F-31400 Toulouse, France (e-mails: yassine.ariba@laas.fr, arzelier@laas.fr, lsurbina@laas.fr).*
[***] *CNRS, LAAS, 7 avenue du colonel Roche, F-31400 Toulouse, France and Univ de Toulouse, UPS, F-31400 Toulouse, France (e-mail: christophe.louembet@laas.fr).*

Abstract: This paper presents minimum-fuel glideslope autonomous guidance algorithms for approaching a target evolving on a circular orbit. In the context of a chemical propulsion, the classical multipulse glideslope algorithm of Hablani is revisited and it is shown that when considering specific common directions for the glideslope such as V-bar and R-bar directions, a linear formulation of the circular minimum-fuel linearized rendezvous problem may be deduced. Unlike the classical glideslope algorithm for which there is no direct control on the fuel consumption, additional degrees of freedom and relevant decision variables may be identified by combining an analytical expression for the maximal guidance error and Hill-Clohessy-Wiltshire relative equations of motion. For a fixed-time rendezvous with a pre-assigned number of maneuvers, a fuel-optimal solution with a bounded guidance error is obtained by solving a linear programming problem. Numerical examples demonstrate the usefulness of the approach with respect to the classical ones when the approach corridor has to fulfill stringent geometrical restrictions such as line-of-sight constraints.

Keywords: Glideslope approach, impulsive control, circular rendezvous, Hill-Clohessy-Wiltshire equations, linear programming

1. INTRODUCTION

During the last fifteen years, there has been an increasing demand to perform autonomous rendezvous, proximity or On-Orbit Servicing operations between an active chaser spacecraft and a passive target spacecraft. Autonomy means that new challenges have to be met when designing appropriate guidance schemes. In particular, simplicity of onboard implementation while preserving optimality in terms of fuel consumption, is fundamental. For the proximal rendezvous phase, the assumption that the distance between the chaser and the target is small enough compared to the distance between the target and the center of attraction a linearized model for relative dynamics may be used to simplify the design of guidance schemes. In addition, if the maneuvers carried out on-board provide short high thrust pulses approximated as impulsive maneuvers, the design of efficient guidance scheme for the general rendezvous problem may be greatly simplified, Fehse (2003). The impulsive approximation for the thrust means that instantaneous velocity increments are applied to the chaser whereas its position remains unchanged.

This paper focuses on the fixed-time linearized minimum-fuel impulsive proximal rendezvous problem. Depending on various operational and safety constraints, different approaching trajectories may be envisioned and proposed to realize the proximal rendezvous. In the literature, common approach strategies and directions of approaching a target in the close range phase of the rendezvous mission are known as V-bar (curvilinear orbit direction as a straight line) and R-bar(direction of the center of attraction) approaches. This is mainly due to observability (line-of-sight constraints) and safety reasons imposing the requirement of a trajectory belonging to a cone-shaped approach corridor. One simple reference approaching trajectory complying to safety restrictions is known as the glideslope approach. A glideslope trajectory is a straight path connecting the current location of the chaser to its final destination which is a position of interest in space near the target. The glideslope approach has been first defined in the past for rendezvous and proximity operations involving the space shuttle, Pearson (1989). This preliminary study has been extended and generalized later for any direction in space and circular reference orbit in Hablani et al. (2002), Wang et al. (2007) and for elliptic reference orbit in Okasha and Newman (2011). Indeed, the results presented in Hablani et al. (2002) are well-known and define the so-called classical glideslope algorithm.

Our goal is to revisit the classical glideslope algorithm of Hablani in specific cases (V-bar and R-bar approaches) for which additional degrees of freedom helping to derive an optimization formulation may be defined. The inherent guidance error produced by the glideslope approach at each hump is bounded allowing to control it in a linear programming formulation of the guidance problem. Thanks to simple analytical derivations, two algorithms based on a linear programming formulation are designed and permit to obtain a minimum-fuel solution to the glideslope guidance problem while controlling the guidance error. The two proposed algorithms are compared to the classical glideslope algorithm and to analytical transfers from Fehse (2003) on two numerical examples

(one for the V-bar and one for the R-bar) illustrating the interest of this new approach. Note that the trajectories are considered in an open-loop setup in this paper whereas real trajectories undergo orbital disturbances and therefore, have to be closed-loop controlled.

Notations: $A \otimes B$ stands for the usual Kronecker product between matrices A and B. $\mathbb{O}_{p\times m}$, $\mathcal{I}_m$ and $\mathbf{1}_m$ denote respectively the null matrix of dimensions $p\times m$, the identity matrix of dimension m and the m-vector composed of ones. $\mathcal{M}_m$ and $\mathcal{L}_m$ are respectively defined as a $(m+1)\times m$ matrix and a m square matrix defined by:

$$\mathcal{M}_m = \begin{bmatrix} 1 & 0 & \cdots & 0 \\ -1 & 1 & \ddots & \vdots \\ 0 & -1 & \ddots & 0 \\ \vdots & \ddots & \ddots & 1 \\ 0 & \cdots & 0 & -1 \end{bmatrix}, \; \mathcal{L}_m = \begin{bmatrix} 0 & 0 & \cdots & \cdots & 0 \\ 1 & 0 & \ddots & & \vdots \\ 0 & \ddots & \ddots & \ddots & 0 \\ \vdots & \ddots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & 1 & 0 \end{bmatrix} \tag{1}$$

$\omega = \sqrt{\frac{\mu}{a^3}}$ is the mean motion of the circular reference orbit where a is its radius and μ is the standard gravitational parameter.

2. CLASSICAL GLIDESLOPE APPROACH FOR RENDEZVOUS

2.1 Relative motion dynamics

This work focuses on the close range phase of the spacecraft rendezvous mission, Fehse (2003). When relative navigation information is available to the chaser, the relative motion expressed in the Local-Vertical-Local-Horizontal (LVLH) frame centered at the target spacecraft position and depicted on Figure 1 is used to characterize the dynamics of the chaser.

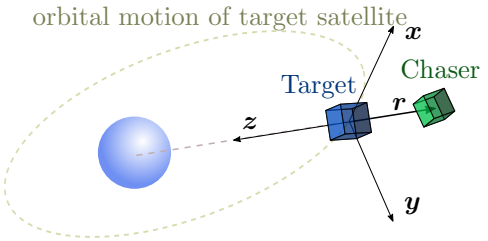

Fig. 1. LVLH frame for spacecraft rendezvous.

The vector $\boldsymbol{r}$ defines the relative position of the chaser with respect to the target. Without any further assumptions, the relative dynamics are described by a set of nonlinear differential equations, Alfriend (2010), which are hard to handle for control purposes. Considering that the two spacecraft are sufficiently close to each other (proximity assumption) and under Keplerian hypothesis, linear relative dynamics may be derived as shown in the seminal work of Lawden (1963). This set of equations are also known as the Tschauner-Hempel equations, Tschauner and Hempel (1964), that boil down to the well known Hill-Clohessy-Wiltshire equations, Clohessy and Wiltshire (1960), when the target spacecraft orbit is supposed to be circular. Defining the state vector of the relative dynamics by $X(\nu) = [\boldsymbol{r}(\nu),\ \boldsymbol{v}(\nu)]^T = [x,\ y,\ z,\ \frac{dx}{d\nu},\ \frac{dy}{d\nu},\ \frac{dz}{d\nu}]^T$ where time is replaced by the true anomaly ν as the independent variable, the autonomous relative motion of the chaser has the following Linear Time-Invariant state space representation, Alfriend (2010):

$$\frac{dX}{d\nu}(\nu) = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 2 \\ 0 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 3 & -2 & 0 & 0 \end{bmatrix} X(\nu). \tag{2}$$

The transition matrix $\Phi(\nu, \nu_0)$ of (2) is readily available such that $X(\nu) = \Phi(\nu, \nu_0)X(\nu_0)$ for $\nu \geq \nu_0$ where:

$$\Phi(\nu,\nu_0) = \left[\begin{array}{cccccc} 1 & 0 & 6(\delta\nu - s_\nu) & (4s_\nu - 3\delta\nu)/\omega & 0 & 2(1-c_\nu)/\omega \\ 0 & c_\nu & 0 & 0 & s_\nu/\omega & 0 \\ 0 & 0 & 4-3c_\nu & 2(c_\nu-1)/\omega & 0 & s_\nu/\omega \\ \hline 0 & 0 & 6\omega(1-c_\nu) & 4c_\nu - 3 & 0 & 2s_\nu \\ 0 & -\omega s_\nu & 0 & 0 & c_\nu & 0 \\ 0 & 0 & 3\omega s_\nu & -2s_\nu & 0 & c_\nu \end{array}\right] \tag{3}$$

where $\delta\nu = \nu - \nu_0$, $c_\nu = \cos(\nu - \nu_0)$ and $s_\nu = \sin(\nu - \nu_0)$. It is assumed that only the chaser is cooperative using 6 ungimbaled identical chemical thrusters. The use of chemical propulsion allows to idealize possible thrusts as impulsive maneuvers providing instantaneous velocity changes while the relative position remains unchanged during firing. Thus, a controlled trajectory composed of N impulses is described by the following equation:

$$X(\nu) = \Phi(\nu,\nu_0)X(\nu_0) + \sum_{i=1}^{N} \Phi(\nu,\nu_i)B\Delta v_i, \tag{4}$$

where $\nu_1 < \nu_2 < \cdots < \nu_N \leq \nu$ and Δv_i denotes the impulsive control applied at ν_i. $B = [\mathbb{O}_3\ \mathcal{I}_3]^T$ is the input matrix. It is well-known and it may be observed from equations (2) and (3) that the in-plane motion $(x - z)$ and the out-of-track motion are decoupled. This work will be focused on the in-plane motion for which fuel-optimal V-bar and R-bar glideslope guidance schemes will be derived. Hereafter, the following notation describing the in-plane free motion is adopted:

$$\begin{aligned} \begin{bmatrix} \boldsymbol{r}(\nu) \\ \boldsymbol{v}(\nu) \end{bmatrix} &= \left[\begin{array}{c|c} \Phi_{rr}(\nu,\nu_0) & \Phi_{rv}(\nu,\nu_0) \\ \hline \Phi_{vr}(\nu,\nu_0) & \Phi_{vv}(\nu,\nu_0) \end{array}\right] \begin{bmatrix} \boldsymbol{r}_0 \\ \boldsymbol{v}_0 \end{bmatrix} \\ &= \left[\begin{array}{cc|cc} 1 & 6(\delta\nu - s_\nu) & (4s_\nu - 3\delta\nu)/\omega & 2(1-c_\nu)/\omega \\ 0 & 4-3c_\nu & 2(c_\nu-1)/\omega & s_\nu/\omega \\ \hline 0 & 6\omega(1-c_\nu) & 4c_\nu - 3 & 2s_\nu \\ 0 & 3\omega s_\nu & -2s_\nu & c_\nu \end{array}\right] \end{aligned} \tag{5}$$

where $\boldsymbol{r} = [x\ z]^T$ and $\boldsymbol{v} = [v_x\ v_z]^T$.

2.2 Hablani's classical glideslope approach for rendezvous

As mentioned in the introduction, the most cited reference on this topic is the paper Hablani et al. (2002) in which the so called classical inbound and outbound glideslope approaches for circular reference are presented in a general setup. This paper is only concerned by inbound decelerating glideslope trajectories and the main features of the classical glideslope algorithm are now recalled in this particular case. In the general case, this guidance trajectory is characterized by a straight line and its associated vector $\boldsymbol{\rho}(\nu) = \boldsymbol{r}_c(\nu) - \boldsymbol{r}_T$, defining the commanded path as illustrated in Figure 2. Defining $\boldsymbol{\rho}_0 = \boldsymbol{r}_0 - \boldsymbol{r}_T$, the unit vector $\boldsymbol{u}$ defines the direction of the straight path where:

$$\boldsymbol{u} = \left[\begin{array}{cc} \frac{x_T - x_0}{\|\boldsymbol{\rho}_0\|} & \frac{z_T - z_0}{\|\boldsymbol{\rho}_0\|} \end{array}\right]^T.$$

The chaser is commanded to reach $\boldsymbol{r}_T$ from $\boldsymbol{r}_0$ following a specific commanded profile for the distance to go $\|\boldsymbol{\rho}_0\| = \rho_0$:

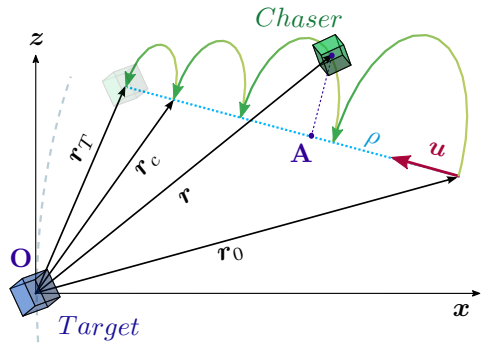

Fig. 2. Glideslope approach.

$$\rho(\nu) = \rho_0 e^{\frac{(\dot\rho_0 - \dot\rho_T)\nu}{\omega\rho_0}} + \frac{\dot\rho_T \rho_0}{\dot\rho_0 - \dot\rho_T}\left(e^{\frac{(\dot\rho_0 - \dot\rho_T)\nu}{\omega\rho_0}} - 1\right) \tag{6}$$

where $\dot\rho_0 < 0$, $\dot\rho_T < 0$ are respectively the initial and final commanded velocities and ρ_0 is the initial distance to go. These quantities are specified by the designer and inputs for the classical algorithm. Note that $\dot\rho_0 > \dot\rho_T$ and that $\rho_T = 0$ by definition of the trajectory. For a given set of these parameters, the basic principle of the classical algorithm is then to analytically compute a fixed number of impulses equally spaced in time over the transfer duration T. Each computed incremental velocity at $\boldsymbol{r}_k$ is obtained as $\Delta\boldsymbol{v}_k = \boldsymbol{v}_k^+ - \boldsymbol{v}_k^-$ where $\boldsymbol{v}_k^+$ is the departure velocity needed to go from $\boldsymbol{r}_k$ to $\boldsymbol{r}_{k+1}$ and $\boldsymbol{v}_k^-$ is the arrival velocity at $\boldsymbol{r}_k$. Both quantities are simply obtained by solving the autonomous Hill-Clohessy-Wiltshire (5) at each step k.

The classical glideslope algorithm is straightforward and rapid to implement but suffers from key shortcomings. In particular, it is important to notice that the actual trajectory of the chaser will not be strictly along the commanded straight line path but will exhibit humps between the N points where an impulsive maneuver is performed and located on the commanded path (cf. Figure 2). These humps coming from the natural relative motion of the chaser driven by the Hill-Clohessy-Wiltshire equations are nothing but lateral guidance position errors possibly occurring in and out-of-plane that cannot be directly controlled in the classical glideslope algorithm. In addition, if the initial and final commanded velocities of the glideslope profile are *a priori* given, there is no degree of freedom left to control the transfer time and the consumption when $X(0)$ and $X(\nu_T)$ are fixed. Indeed, the transfer time T is not fixed *a priori* but deduced from the initial and final commanded velocities $(\dot\rho_0, \dot\rho_T)$ and from the initial distance to go ρ_0:

$$T = \frac{\rho_0}{\dot\rho_0 - \dot\rho_T} \ln\left[\frac{\dot\rho_T}{\dot\rho_0}\right]. \tag{7}$$

The consumption itself is computed *a posteriori* from the velocity increments without any possibility to optimize it for given side conditions of the rendezvous. The objective of the paper is then to propose a new optimization algorithm for particular and common glideslope approaching directions as V-bar and R-bar allowing to take these important features into account.

- Minimize the fuel-consumption for a given set of initial and final rendezvous conditions and an *a priori* fixed time of transfer;
- Control the maximum guidance error by defining constraints on the humps profile.

2.3 Derivation of guidance error in orbital plane

Keeping in mind the requirements of the previous section, finding an analytical expression for the guidance error is a key technical step for the development of an efficient algorithm. Due to the decoupling property between in-plane and out-of-plane keplerian relative motions, this study will be limited to glideslope approaches defined in the orbital plane.

Let us define the point A in the $x-z$ plane (see Figure 2) as the orthogonal projection of the chaser position $\boldsymbol{r}$ on the glideslope straight line. We can thus write

$$\boldsymbol{OA} = (\boldsymbol{r} - \boldsymbol{r}_0)^T \boldsymbol{u}\, \boldsymbol{u} + \boldsymbol{r}_0,$$

and hence, guidance error is given by

$$\boldsymbol{\epsilon_T} = \boldsymbol{r} - \boldsymbol{OA}.$$

The guidance error norm is then deduced as:

$$\begin{aligned} \|\boldsymbol{\epsilon_T}\|^2 &= \left(\boldsymbol{\delta r} - [\boldsymbol{\delta r}^T \boldsymbol{u}]\, \boldsymbol{u}\right)^T \left(\boldsymbol{\delta r} - [\boldsymbol{\delta r}^T \boldsymbol{u}]\, \boldsymbol{u}\right) \\ &= \boldsymbol{\delta r}^T \boldsymbol{\delta r} - [\boldsymbol{\delta r}^T \boldsymbol{u}]^2 \\ &= \delta r_x^2(1-u_x^2) + \delta r_z^2(1-u_z^2) - 2u_x u_z \delta r_x \delta r_z \end{aligned} \tag{8}$$

with $\boldsymbol{\delta r} = \boldsymbol{r} - \boldsymbol{r}_0$, $\delta r_x = x - x_0$ and $\delta r_z = z - z_0$.

3. MINIMUM-FUEL V-BAR GLIDESLOPE APPROACH VIA LINEAR PROGRAMMING

3.1 Constraint on guidance error

Considering a glideslope along a line parallel to the x-axis, the unitary direction vector u is $[1\ 0]^T$. Hence, the guidance error norm (8) is reduced to

$$\begin{aligned} \|\boldsymbol{\epsilon_T}\|^2 &= \delta r_z^2, \\ &= \left((4 - 3\cos\delta\nu)z_0 + \frac{2}{\omega}(\cos\delta\nu - 1)x_0' + \frac{1}{\omega}z_0' \sin\delta\nu - z_0\right)^2, \\ &= \left(3(1-\cos\delta\nu)z_0 + \frac{2}{\omega}(\cos\delta\nu - 1)x_0' + \frac{1}{\omega}z_0' \sin\delta\nu\right)^2. \end{aligned} \tag{9}$$

When the guidance error distance is maximal, the velocity vector $\boldsymbol{v}$ is parallel to the glideslope line. A maximum condition is then given by

$$z' = 0 \ \Leftrightarrow\ 3\omega z_0 \sin\delta\nu - 2x_0' \sin\delta\nu + z_0' \cos\delta\nu = 0. \tag{10}$$

Lemma 1. Let us define $\delta\nu_m$ the value of the true anomaly when guidance error is maximal and $\delta\nu_0$ the value of the true anomaly when guidance error is null, that is, when the satellite is back on the glideslope commanded path. Thus, $\delta\nu_0$ is the true anomaly interval for a hump. Considering a single hump, we have the relationship:

$$\delta\nu_0 = 2\ \delta\nu_m.$$

That is, the maximum occurs in the middle (in terms of true anomaly) of the hump.

From equation (9), guidance error distance can be readily upper-bounded with

$$\begin{aligned} |\epsilon_T| &= \left|(3 - 3\cos\delta\nu)z_0 + \frac{2}{\omega}(\cos\delta\nu - 1)x_0' + \frac{1}{\omega}z_0' \sin\delta\nu\right|, \\ &\leq \left|(3 - 3\cos\delta\nu_m)z_0 + \frac{2}{\omega}(\cos\delta\nu_m - 1)x_0' + \frac{1}{\omega}z_0' \sin\delta\nu_m\right|. \end{aligned}$$

3.2 Problem formulation

We aim now at combining motion equations, guidance error equations and identify relevant decision variables in order to formulate an optimization problem. The analysis of the previous paragraph provides an inequality that will

be used to specify maximum conditions on guidance error for each hump, $k = 0, \ldots, N-1$:

$$\left|(3 - 3\cos\delta\nu_m)z_0 + \frac{2}{\omega}(\cos\delta\nu_m - 1)x'_k + \frac{1}{\omega}z'_k \sin\delta\nu_m\right| \leq m_k$$

are equivalently

$$\begin{cases} (3 - 3\cos\delta\nu_m)z_0 + \frac{2}{\omega}(\cos\delta\nu_m - 1)x'_k + \frac{1}{\omega}z'_k \sin\delta\nu_m \leq m_k \\ -(3 - 3\cos\delta\nu_m)z_0 - \frac{2}{\omega}(\cos\delta\nu_m - 1)x'_k - \frac{1}{\omega}z'_k \sin\delta\nu_m \leq m_k \end{cases} \tag{11}$$

where m_k is given and specifies the maximal allowable guidance error during the $(k+1)^{\text{th}}$ maneuver. We recall that assuming the interval $\delta\nu_0$ is constant, the parameter $\delta\nu_m$ is constant too.

After each maneuver, the chaser must be back on the glideslope. Such a requirement can be enforced by N equations of the form

$$\boldsymbol{r}_{k+1} = \Phi_{rr}\boldsymbol{r}_k + \Phi_{rv}\boldsymbol{v}_k, \qquad k = 0, \ldots, N-1 \tag{12}$$

where

$$\boldsymbol{r}_k = \begin{bmatrix} x_k \\ z_0 \end{bmatrix}, \ \boldsymbol{v}_k = \begin{bmatrix} x'_k \\ z'_k \end{bmatrix}, \ \boldsymbol{r}_0 = \begin{bmatrix} x_0 \\ z_0 \end{bmatrix}, \ \boldsymbol{r}_N = \begin{bmatrix} x_T \\ z_0 \end{bmatrix}.$$

$\boldsymbol{r}_0$ and $\boldsymbol{r}_T$ are given and impose the initial and final positions. The z-component of $\boldsymbol{r}_k$ remains constant and equals z_0 since the glideslope is assumed to be parallel to the V-bar direction. The interval $\delta\nu_0$ between two impulses being constant, the transition matrix $\{\Phi_{rr}, \Phi_{rv}, \Phi_{vr}, \Phi_{vv}\}$ is constant. We consider $\boldsymbol{v}_k$ as the velocity vector resulting from the $(k+1)^{\text{th}}$ control impulse. Its components together with x_k form our decision variables. Hence, all positions on the glideslope, except the initial and final ones, are set free. Moreover, an additional equation may be specified so as to impose a final velocity condition $\boldsymbol{v}_T$:

$$\boldsymbol{v}_T = \boldsymbol{v}_N = \Phi_{vr}\boldsymbol{r}_{N-1} + \Phi_{vv}\boldsymbol{v}_{N-1} + \Delta\boldsymbol{v}_N. \tag{13}$$

$\boldsymbol{v}_N$ is thus the desired final velocity vector (and not a variable) while $\Delta\boldsymbol{v}_N$ is an extra variable representing the last impulse.

The optimization problem consists in finding the values of

- velocity vectors $\boldsymbol{v}_k$ for $k = 0, \ldots, N-1$,
- positions x_k between two manoeuvres for $k = 1, \ldots, N-1$,
- the final impulse $\Delta\boldsymbol{v}_N$,

minimizing the overall consumption. This latter is based on impulses:

$$\min \quad \sum_{k=0}^{N} \|\Delta\boldsymbol{v}_k\|_1,$$

with $\Delta\boldsymbol{v}_k = \boldsymbol{v}_k - \boldsymbol{v}_{k_-}$. This criterion can be expressed with respect to the decision variables:

$$\min \|\boldsymbol{v}_0 - \boldsymbol{v}_{0_-}\|_1 + \sum_{k=1}^{N-1} \|\boldsymbol{v}_k - \Phi_{vr}\boldsymbol{r}_{k-1} - \Phi_{vv}\boldsymbol{v}_{k-1}\|_1 + \|\Delta\boldsymbol{v}_N\|_1, \tag{14}$$

where $\boldsymbol{v}_{0_-}$ is the initial velocity vector. This cost function involving absolute values can be transformed into a linear function with the introduction of new variables and inequality constraints, Bertsimas and Tsitsiklis (1997).

3.3 Minimum-fuel LP algorithm

In this paragraph, the optimization to be addressed is stated and an algorithm is proposed. This algorithm computes appropriate impulses to transfer the satellite from an initial position $\boldsymbol{r}_0$ to a final position $\boldsymbol{r}_T$ while minimizing the consumption. Considering the decision variables previously defined, the cost criterion (14) and gathering constraints on guidance errors (11), on positions (12) and on the final velocity (13), a linear programming problem can be built.

$$\begin{aligned} \min \ & c^T X \\ \text{s.t. } & AX \leq b, \ A_{eq}X = b_{eq} \end{aligned} \tag{15}$$

The vector X puts together all the decision variables,

$$X = \left[\, x_1 \ \ldots \ x_{N-1} \middle| \boldsymbol{v}_0^T \ \ldots \ \boldsymbol{v}_{N-1}^T \middle| \Delta\boldsymbol{v}_N^T \middle| d_0^T \ \ldots \ d_N^T \,\right]^T, \tag{16}$$

where $d_i \in \mathbb{R}^2$ are extra variables used to deal with the absolute values in the cost function. This cost criterion is then reduced to the sum of d_i, $i = 0, \ldots, N$, and the vector c is defined as

$$c^T = \left[\, \mathbb{O}_{1\times 3N+1} \ 1 \ \ldots \ 1 \,\right].$$

The equality constraint is built on equations (12) and (13), while the inequality constraint is built on equation (11) and additional inequalities introduced to cope with the absolute values in the cost function, Bertsimas and Tsitsiklis (1997). The different matrices involved in the linear programming problem are defined as:

$$A_{eq} = \left[\begin{array}{c|c|c|c} \mathcal{M}_{N-1} \otimes C_1 & -\mathcal{I}_N \otimes \Phi_{rv} & \mathbb{O}_{2N\times 2} & \mathbb{O}_{2N\times(2N+2)} \\ \hline \mathbb{O}_{2\times(N-1)} & \mathbb{O}_{2\times(2N-2)} \ \Phi_{vv} & \mathcal{I}_{2\times 2} & \mathbb{O}_{2\times(2N+2)} \end{array}\right] \tag{17}$$

and

$$b_{eq} = \left[\begin{array}{c} \Phi_{rr}\boldsymbol{r}_0 - C_2 z_0 \\ \mathbf{1}_{N-2} \otimes (\Phi_{rr} - \mathcal{I}_2)C_2 z_0 \\ \Phi_{rr}C_2 z_0 - \boldsymbol{r}_T \\ \hline \boldsymbol{v}_T - \Phi_{vr}C_2 z_0 \end{array}\right], \tag{18}$$

where $C_1 = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$ and $C_2 = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$.

$$A = \left[\begin{array}{c|c|c|c} \mathbb{O}_{4(N+1)\times N-1} & \overline{A} & \begin{matrix} \mathbb{O}_{4N\times 2} \\ \mathcal{I}_2 \\ -\mathcal{I}_2 \end{matrix} & \mathcal{I}_{N+1} \otimes \begin{bmatrix} -\mathcal{I}_2 \\ -\mathcal{I}_2 \end{bmatrix} \\ \hline \mathbb{O}_{2N\times N-1} & \mathcal{I}_N \otimes M & \mathbb{O}_{2N\times 2} & \mathbb{O}_{2N\times 2(N+1)} \end{array}\right] \tag{19}$$

where $\overline{A} = \begin{bmatrix} \mathcal{L}_N \otimes \begin{bmatrix} -\Phi_{vv} \\ \Phi_{vv} \end{bmatrix} + \mathcal{I}_N \otimes \begin{bmatrix} \mathcal{I}_2 \\ -\mathcal{I}_2 \end{bmatrix} \\ \mathbb{O}_{2\times 2N} \\ \mathbb{O}_{2\times 2N} \end{bmatrix}$ and

$$b = \left[\begin{array}{c} \boldsymbol{v}_{0_-} \\ -\boldsymbol{v}_{0_-} \\ \Phi_{vr}\boldsymbol{r}_0 \\ -\Phi_{vr}\boldsymbol{r}_0 \\ \mathbf{1}_{N-2} \otimes \begin{bmatrix} \Phi_{vr}C_2 z_0 \\ -\Phi_{vr}C_2 z_0 \end{bmatrix} \\ \mathbb{O}_{2\times 1} \\ \mathbb{O}_{2\times 1} \\ \hline M_0 \\ \vdots \\ M_{N-1} \end{array}\right] \tag{20}$$

with

$$\begin{aligned} M &= \begin{bmatrix} \frac{2}{\omega}\big(\cos\delta\nu_m - 1\big) & \frac{1}{\omega}\sin\delta\nu_m \\ -\frac{2}{\omega}\big(\cos\delta\nu_m - 1\big) & -\frac{1}{\omega}\sin\delta\nu_m \end{bmatrix}, \\ M_k &= \begin{bmatrix} m_k - \big(3 - 3\cos\delta\nu_m\big)z_0 \\ m_k + \big(3 - 3\cos\delta\nu_m\big)z_0 \end{bmatrix}. \end{aligned} \tag{21}$$

Remark 1. Note that expressions of above matrices have been slightly simplified because we have $\Phi_{rr}C_1 = C_1$ and $\Phi_{vr}C_1 = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$.

The procedure to run the glideslope method proposed in this paper is summarized in Algorithm 1. Given an

initial $\{\boldsymbol{r}_0, \boldsymbol{v}_{0\_}\}$ and a final $\{\boldsymbol{r}_T, \boldsymbol{v}_T\}$ configuration, a transfer time T and a number N of maneuvers, the algorithm computes the optimal impulse sequence $\Delta\boldsymbol{v}_k$, $k = 0, \ldots, N$, minimizing the consumption while ensuring a maximal guidance error m_k for each manoeuvre.

Algorithm 1: Minimum-fuel LP glideslope

Data: $\boldsymbol{r}_0$, $\boldsymbol{r}_T$, $\boldsymbol{v}_0^-$, $\boldsymbol{v}_T$, N, T, ω, $m = [m_0, \ldots, m_{N-1}]$
Initialization: $\delta t = T/N$; $\delta\nu_0 = \omega\ \delta t$; $\delta\nu_{\max} = \delta\nu_0/2$;
$\left[\Phi_{rr}, \Phi_{rv}, \Phi_{vr}, \Phi_{vv}\right] \leftarrow$ compute transition matrix for δt and ω;
for $k = 0$ **to** $N-1$ **do**
  $M_k \leftarrow$ compute matrix with m_k, z_0 and $\delta\nu_{\max}$;
$A \leftarrow$ compute matrix with Φ_{vv}, $\delta\nu_{\max}$, ω and N;
$b \leftarrow$ compute vector with Φ_{vr}, $\boldsymbol{v}_{0\_}$, M_k, $\boldsymbol{r}_0$ and N;
$A_{eq} \leftarrow$ compute matrix with Φ_{rv}, Φ_{vv} and N;
$b_{eq} \leftarrow$ compute vector with Φ_{rr}, Φ_{vr}, $\boldsymbol{r}_0$, $\boldsymbol{r}_T$, $\boldsymbol{v}_T$ and N;
$c \leftarrow$ compute vector with N;
Solve: the linear programming problem, and store the optimal solution in X
if *a solution is found* **then**
  $\boldsymbol{v}_k \leftarrow$ extracted from X, $k = 0, \ldots, N-1$;
  $\Delta\boldsymbol{v}_N \leftarrow$ extracted from X;
  for $k = 0$ **to** $N-1$ **do**
    $\boldsymbol{r}_{k+1} \leftarrow \Phi_{rr}\boldsymbol{r}_k + \Phi_{rv}\boldsymbol{v}_k$;
    $\boldsymbol{v}_{k+1\_} \leftarrow \Phi_{vr}\boldsymbol{r}_k + \Phi_{vv}\boldsymbol{v}_k$;
    $\Delta\boldsymbol{v}_k \leftarrow \boldsymbol{v}_k - \boldsymbol{v}_{k\_}$;
else
  increase some m_k, update M_k and b, and re-run the solver;

4. EXTENSION TO R-BAR APPROACH

4.1 Constraint on excursion

In that case, the unitary direction vector u is $[0\ 1]^T$. Hence, the guidance error norm (8) reduces to:

$$\begin{aligned}\|\boldsymbol{\epsilon_T}\|^2 &= \delta r_x^2, \\ &= \left(6z_0(\delta\nu - \sin\delta\nu) + \frac{1}{\omega}(4\sin\delta\nu - 3\delta\nu)x_0' \right. \\ &\quad \left. + \frac{2}{\omega}(1-\cos\delta\nu)z_0'\right)^2.\end{aligned} \tag{22}$$

The following lemma will be used in the next paragraph to provide an inequality constraint on the excursion for each hump. Unlike the V-bar case, a simple relationship between the true anomaly $\delta\nu_m$ when the guidance error is maximal and the maneuver interval $\delta\nu_0$ cannot be established. So, a conservative bound is introduced.

Lemma 2. Let us define $\delta\nu_0$ the value of the true anomaly when guidance error is 0, that is, when the satellite is back on the glideslope commanded path. Considering a single hump such that $\delta\nu_0 \in [0\ \arccos\frac{3}{4}]$, a conservative upper-bound of the guidance error is given by:

$$|\boldsymbol{\epsilon_T}| \leq \alpha_1|x_0'| + \alpha_2|z_0'| + \alpha_3|z_0|,$$

with $\alpha_1 = |\frac{1}{\omega}(4\sin\delta\nu_0 - 3\delta\nu_0)|$, $\alpha_2 = |\frac{2}{\omega}(1-\cos\delta\nu_0)|$ and $\alpha_3 = |6(\delta\nu_0 - \sin\delta\nu_0)|$.

Although in the above lemma the range of $\delta\nu_0$ is restricted to $[0\ \arccos\frac{3}{4}]$, this result can be easily extended to a full revolution.

4.2 Formulation and algorithm

The problem formulation for the R-bar approach follows the same process as in Subsection 3.3. The set of N motion equations (12) with

$$\boldsymbol{r}_k = \begin{bmatrix} x_0 \\ z_k \end{bmatrix},\ \boldsymbol{v}_k = \begin{bmatrix} x_k' \\ z_k' \end{bmatrix},\ \boldsymbol{r}_0 = \begin{bmatrix} x_0 \\ z_0 \end{bmatrix},\ \boldsymbol{r}_N = \begin{bmatrix} x_0 \\ z_T \end{bmatrix}.$$

allows the satellite to go along the glideslope line. For the R-bar approach, the z-component is changing while the x-component remains constant. Thus, equations (12), (13) and (14) are considered again and combined with the new appropriate vector of decision variables:

$$X = \left[\ z_1\ \ldots\ z_{N-1} \middle| v_0^T\ \ldots\ v_{N-1}^T \middle| \Delta v_N^T \middle| d_0^T\ \ldots\ d_N^T\ \right]^T.$$

Again, $d_i \in \mathbb{R}^2$ are extra variables used to deal with the absolute values in the cost function. Eventually, a minimum-fuel R-bar glideslope approach is formulated as a linear programming problem (15) with matrices defined as follows.

$$A_{eq} = \left[\begin{array}{c|c|c|c} \tilde{A} & -\mathcal{I}_N \otimes \Phi_{rv} & \mathbb{O}_{2N\times 2} & \mathbb{O}_{2N\times 2N+2} \\ \hline \mathbb{O}_{2\times N-2}\ \ \Phi_{vr}C_2 & \mathbb{O}_{2\times 2N-2}\ \ \Phi_{vv} & \mathcal{I}_{2\times 2} & \mathbb{O}_{2\times 2N+2} \end{array}\right], \tag{23}$$

with $\tilde{A} = \begin{bmatrix} \mathcal{I}_{N-1}\otimes C_2 \\ \mathbb{O}_{2\times N-1} \end{bmatrix} + \begin{bmatrix} \mathbb{O}_{2\times N-1} \\ \mathcal{I}_{N-1}\otimes -\Phi_{rr}C_2 \end{bmatrix}$ and

$$b_{eq} = \left[\begin{array}{c} \Phi_{rr}\boldsymbol{r}_0 - C_1 x_0 \\ \mathbb{O}_{2(N-2)\times 1} \\ C_1 x_0 - \boldsymbol{r}_T \\ \hline \boldsymbol{v}_T \end{array}\right], \tag{24}$$

$$A = \left[\begin{array}{c|c|c|c} \hat{A} & \overline{A} & \begin{array}{c}\mathbb{O}_{4N\times 2} \\ \mathcal{I}_2 \\ -\mathcal{I}_2\end{array} & \mathcal{I}_{N+1}\otimes\begin{bmatrix}-\mathcal{I}_2 \\ -\mathcal{I}_2\end{bmatrix} \\ \hline \begin{array}{c}\mathbb{O}_{8\times N-1} \\ \mathcal{I}_{N-1}\otimes M_1\end{array} & \mathcal{I}_N\otimes M_2 & \mathbb{O}_{8N\times 2} & \mathbb{O}_{8N\times 2(N+1)} \end{array}\right] \tag{25}$$

where

$$\overline{A} = \begin{bmatrix} \mathcal{L}_N \otimes \begin{bmatrix}-\Phi_{vv} \\ \Phi_{vv}\end{bmatrix} + \mathcal{I}_N \otimes \begin{bmatrix}\mathcal{I}_2 \\ -\mathcal{I}_2\end{bmatrix} \\ \mathbb{O}_{2\times 2N} \\ \mathbb{O}_{2\times 2N} \end{bmatrix},$$

$$\hat{A} = \begin{bmatrix} \mathbb{O}_{8\times N-2} & \mathbb{O}_{8\times 1} \\ \mathcal{I}_{N-2}\otimes\begin{bmatrix}-\Phi_{vr}C_2 \\ \Phi_{vr}C_2\end{bmatrix} & \mathbb{O}_{4(N-2)\times 1} \\ \mathbb{O}_{4\times N-2} & \mathbb{O}_{4\times 1} \end{bmatrix},$$

$$M_1 = \begin{bmatrix}\mathbf{1}_4 \\ -\mathbf{1}_4\end{bmatrix}\alpha_3,\ M_2 = \mathbf{1}_2 \otimes \begin{bmatrix}\alpha_1 & \alpha_2 \\ \alpha_1 & -\alpha_2 \\ -\alpha_1 & \alpha_2 \\ -\alpha_1 & -\alpha_2\end{bmatrix},$$

and

$$b = \left[\begin{array}{c} \boldsymbol{v}_{0\_} \\ -\boldsymbol{v}_{0\_} \\ \Phi_{vr}\boldsymbol{r}_0 \\ -\Phi_{vr}\boldsymbol{r}_0 \\ \mathbb{O}_{4(N-1)\times 1} \\ \hline \mathbf{1}_8 m_0 - M_1 z_0 \\ \mathbf{1}_8 m_1 \\ \vdots \\ \mathbf{1}_8 m_{N-1} \end{array}\right]. \tag{26}$$

5. NUMERICAL EXAMPLES

This scenario, extracted from Hablani et al. (2002), is a tangential impulse transfer along V-bar direction and considers a target flying an orbit at 400km altitude, which corresponds to an orbital rate of $\omega \approx 0.001$ rad/s. The standard glideslope algorithm is used to transfer the chaser from $x = -500$ m to $x = -100$ m in 10 maneuvers (N) along 9 min ($T = 540$ s). This example's parameters are set as follows:

$$\boldsymbol{r}_0 = \begin{bmatrix}-500 \\ 0 \\ -20\end{bmatrix},\ \boldsymbol{r}_T = \begin{bmatrix}-100 \\ 0 \\ -20\end{bmatrix},\ \boldsymbol{v}_{0\_} = \begin{bmatrix}0 \\ 0 \\ 0\end{bmatrix}.$$

These parameters have been injected in Algorithm 1 to compute the 10 pulses, and a final velocity has been specified as well ($\boldsymbol{v}_T = \mathbf{0}$). Figure 3 shows the chaser trajectories for three different methods in the $x-z$ plane: the standard glideslope defined in Hablani et al. (2002), the two impulses approach from Fehse (2003) and our

	N	2	3	4	10	20
cons. m/s	Std	3.61	4.76	5.63	8.09	9.32
	Alg 1	2.26	2.29	2.30	2.31	2.31
exc. m	Std	26.2	16.2	11.1	2.67	0.78
	Alg 1	13.8	6.2	3.5	0.56	0.15

Table 1.

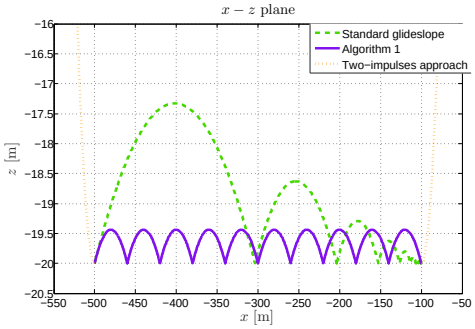

Fig. 3. *Three glideslope transfers along V-bar.*

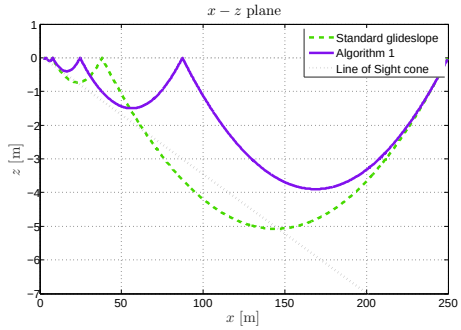

Fig. 4. *Hablani's glideslope (green) and new algorithm transfer (violet) along V-bar with a LOS constraint*

Algorithm 1. In this simulation, the m_k has been chosen identically equal to 1m. It is interesting to note in that configuration that the algorithm adjusts pulses to be regularly spaced. We remind that x_k, $0 < k < N$, are not constrained (and not ordered) and could be anywhere on the glideslope line. Regarding the consumption, Algorithm 1 outperforms the standard glideslope, reducing the cost from 8.09m/s to 2.31m/s. Note that we have included in the consumption of the standard approach an extra impulse to end up the manoeuvre with the desired final velocity vector $\boldsymbol{v}_T$. The consumption of the two tangential impulses approach is very low, 0.12m/s, but it results in a very large excursion ($\approx$ 125m). Table 1 compares numerical simulations of the standard glideslope and Algorithm 1 for various number of impulses. For every test in Table 1, we set m_k identically equal to 20m.

A second scenario, inspired from Di Cairano et al. (2012), is considered. In this case, from an initial position $(x_0, z_0) = (250, 0)$, the chaser must be transferred to a target platform while remaining within a Line-of-Sight (LOS) cone. The docking port is located at the position coordinates $(2.5, 0)$. The half angle of the LOS cone is 2deg and its vertex is 0.5m behind the docking port. The simulations of the standard glideslope and Algorithm 1, for $N = 5$ and $T = 480s$, are shown in Figure 4. Specifying the following upper-bounds on guidance error allows to maintain the spacecraft within the LOS cone.

$$m = [\,5 \;\; 1.5 \;\; 0.4 \;\; 0.1 \;\; 0.03\,]$$

Moreover, the resulting consumption is 3.87m/s, lower than the standard glideslope 4.90m/s.

6. CONCLUSION

In this paper, a revisited solution to the classical Hablani glideslope algorithm, Hablani et al. (2002), for the impulsive close range rendezvous in a circular orbit framework is proposed for the the V-bar and R-bar approaches. It is aimed at tackling the two main drawbacks of this classical algorithm which are the lack of control on the bounds over the inherent guidance errors and the impossibility of minimizing the fuel consumption.

In response to these issues, a new problem formulation is developed, where the relative dynamics, the guidance error expressions and the identification of relevant decision variables are combined in order to derive a linear programming formulation of the guidance problem. This new proposed glideslope algorithm includes important design features like the specification of maximum conditions on the guidance error for each hump.

Two simulations are performed and compared to the classical glideslope algorithm given by Hablani et al. (2002) and to the two-impulses approach by (Fehse, 2003). The key features of the exposed results are a significant improved fuel consumption with respect to Hablani's algorithm and a user-defined bound profile on the maximum guidance error, which turns out to be very useful when dealing when visibility constraints.

7. ACKNOWLEDGEMENTS

The Authors would like to thank Jean-Claude Berges from CNES and Damiana Losa from Thales Alenia Space for the grants that partly supports this activity.

REFERENCES

Alfriend, K. (ed.) (2010). *Spacecraft Formation Flying: Dynamics, Control and Navigation.* Astrodynamics Series. Elsevier, Burlington, USA.

Bertsimas, D. and Tsitsiklis, J. (1997). *Introduction to Linear Optimization.* Athena Scientific, 1st edition.

Clohessy, W. and Wiltshire, R. (1960). Terminal guidance system for satellite rendezvous. *Journal of the Astronautical Sciences*, 27(9), 653–658.

Di Cairano, S., Park, H., and Kolmanovsky, I. (2012). Model predictive control approach for guidance of spacecraft rendezvous and proximity maneuvering. *International Journal of Robust and Nonlinear Control*, 22, 1398–1427.

Fehse, W. (ed.) (2003). *Automated rendezvous and docking of spacecraft.* Cambridge Aerospace Series. Cambridge University Press, Cambridge, UK.

Hablani, H., Tapper, M., and David J. Dana-Bashian, D. (2002). Guidance and relative navigation for autonomous rendezvous in a circular orbit. *Journal of Guidance, Control and Dynamics*, Vol. 25(No. 3).

Lawden, D. (1963). *Optimal trajectories for space navigation.* Butterworth, London, England.

Okasha, M. and Newman, B. (2011). Guidance, navigation and control for satellite proximity operations using tschauner-hempel equations. In *AIAA Guidance, Navigation and Control Conference*. Portland, Oregon, USA.

Pearson, D. (1989). The glideslope approach. *Advances in Astronautical Sciences*, 69, 109–123.

Tschauner, J. and Hempel, P. (1964). Optimale beschleunigungs-programme fur des rendezvous manover. *Astronautica Acta*, 5-6, 296–307.

Wang, F., Cao, X., and Chen, X. (2007). Guidance algorithms for the near-distance rendezvous of on-orbit-servicing spacecraft. *Transactions of Japanese Society for Aeronautical and Space Sciences*, 50(167), 9–17.