Path Following Control of Rhombic Like Vehicles
Performance Assessment with Dynamic Vehicle Model

Nuno Silva[1], Alberto Vale[1] and Luca Baglivo[2]
[1]*Instituto de Plasmas e Fusão Nuclear, Instituto Superior Técnico, Universidade Técnica de Lisboa, Av. Rovisco Pais 1, 1049-001 Lisboa, Portugal*
[2]*Laboratory of Robotics and Systems in Engineering and Science, Instituto Superior Técnico, Universidade Técnica de Lisboa, Av. Rovisco Pais 1, 1049-001 Lisboa, Portugal*

Keywords: Rhombic Like Vehicles, Path Following, Dynamic Vehicle Modelling.

Abstract: This paper addresses the path following problem of a wheeled mobile robot with rhombic like kinematics (drivable and steerable wheels both at front and rear) operating in cluttered environments. Four path following controllers are developed to steer the kinematic model of a rhombic like vehicle (RLV) along a desired path: three are based on feedback laws derived at a kinematic level with geometrical inspiration; the fourth is a nonlinear controller built upon a kinematic model of a RLV using Lyapunov functions. All the developed controllers are capable of performing under two situations: when both wheels follow the same path, or when each wheel follows a different path. The dynamic modelling of a RLV is lastly introduced. Simulated results present a comparative performance assessment of the controllers while dealing, or not, with vehicle dynamics. The main conclusions of these controllers are summarized, leading to a possible application in the actual vehicles that will operate in the remote handling missions of the International Thermonuclear Experimental Reactor (ITER).

1 INTRODUCTION

Rhombic like vehicles (RLVs) as described in (Ribeiro and Lima, 1997) and depicted in Fig. 1, distinguish themselves from the typical unicycle and car like vehicles, by allowing to control the linear speed, v_i, and orientation, θ_i, of each wheel $i \in \{Rear, Front\}$. In fact, vehicles with rhombic like kinematics are a set of nonholonomic vehicles characterized for enabling a decoupling of the vehicle's orientation from the vehicle's velocity vector, creating a controllable sideslip angle, β (Fig. 1). In a RLV both wheels can follow the same path, referred as Line Guidance (LG), or each wheel can follow a different path, referred as Free Roaming (FR) (Vale et al., 2012). A RLV is particularly suitable for operations in cluttered scenarios where a high degree of vehicle mobility is essential. This issue is detailed in (Ribeiro and Lima, 1997) and strengthened in (Vale et al., 2012), where the applicability of a RLV in the cluttered environments of ITER (International Thermonuclear Experimental Reactor) was studied. Several RLVs will operate in Tokamak Building (TB) and Hot Cell Building (HCB) of ITER, where Fig. 2 illustrates one of the levels of TB. Remote Handling

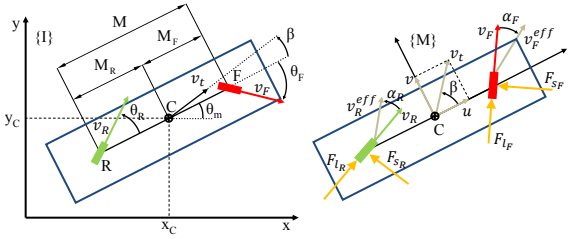

Figure 1: Rhombic like vehicle (RLV) kinematic (left) and dynamic (right) variables and parameters.

(RH) systems will play an important role in ITER and one of such systems is the Cask and Plug Remote Handling System (CPRHS) depicted in Fig. 2: a RLV responsible for RH operations of transportation of activated components and equipment inside the TB. The CPRHS can reach dimensions of 8.50m x 2.62m x 3.70m (length, width, height), a mass of 100T when fully loaded and has to operate within a narrow safety margin of 10 to 30 cm. It has to be noticed that the kinematic capabilities of RLVs can further be extended to load transport in a wide set of scenarios, such as storage warehouses, industrial sites and cargo harbours. In (Vale et al., 2012), a motion planning of the CPRHS in ITER is detailed. The tra-

Silva N., Vale A. and Baglivo L..
Path Following Control of Rhombic Like Vehicles - Performance Assessment with Dynamic Vehicle Model.
DOI: 10.5220/0004439500480057
In *Proceedings of the 10th International Conference on Informatics in Control, Automation and Robotics* (ICINCO-2013), pages 48-57
ISBN: 978-989-8565-71-6

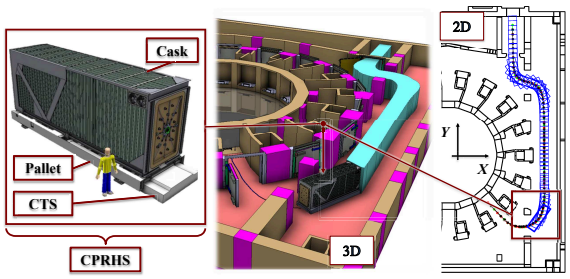

Figure 2: The CRPHS composition (left) and its typical operating scenario in a 3D and 2D view (middle and right, respectively).

jectories (path and speed profile) outputted from the aforementioned work are considered to be the optimal motion solution in the scenarios of ITER. It is the purpose of the current paper to study different *path following* solutions for a RLV with the characteristics of the CPRHS, along previously optimized trajectories. Moreover, this paper focuses on path following and not on trajectory tracking, since the objective is to steer a RLV along the optimized trajectories, while the vehicle's speed tracks the desired speed profile embedded in the trajectory. In contrast, trajectory tracking requires the vehicle to track a time-parameterized reference (Luca and Oriolo, 1998).

To solve the path following problem for unicycles, in (Micaelli and Samson, 1993; Lapierre et al., 2006) are employed nonlinear control techniques, either in the form of Lyapunov based methods or through feedback linearization. Noticeable performance results and proof of stability in relation to a feasible path are detailed. Concerning car like vehicles, (Luca and Oriolo, 1998) details a path following controller based on feedback linearization methods, which exploit the chained form representation of the kinematic model of a car like vehicle. Despite achieving good performances, in order to guarantee stability, some constraints must be made on the initial pose (position and orientation) of the vehicle and in the path itself. For All-Wheel-Steering vehicles, class to which RLVs belong to, (Hiraoka et al., 2009) explores the application of sliding mode controllers to the path following problem, improving robustness to uncertainties at a cost of compromising steering smoothness.

The only references to path following for RLVs are made in (Micaelli and Samson, 1993; Morin and Samson, 2011) at a kinematic level only. In (Micaelli and Samson, 1993) nonlinear control methods are used to derive feedback control laws that enable one wheel of the RLV to converge to a reference path, whereas the other wheel determines the vehicle's orientation while following the path. In (Morin and Samson, 2011) the application of transverse functions to the control of RLVs is studied, placing a complementary constraint when the steering angles of both wheels are equal to $\pm\pi/2$, which is contoured using *practical* stabilization (Morin and Samson, 2011) via specific maneuvering embedded in the control design.

In this paper, and motivated by the abovementioned considerations, the authors propose four path following controllers for RLVs, which can be divided in two groups: three are categorized as geometrical controllers and are based on feedback laws derived at a kinematic level with geometrical inspiration; while the fourth is a nonlinear controller, built upon a kinematic model of the vehicle using Lyapunov functions. The geometrical controllers represent a good compromise between complexity *vs.* performance and enable a first line of awareness to the control problem. In relation to the nonlinear controller, two nonlinear feedback control laws are derived, acting independently on the vehicle's orientation and sideslip angle, and thereby extracting the full capabilities of a RLV. In striking contrast with the work done in (Micaelli and Samson, 1993), the nonlinear controller proposed in this paper allows for a RLV to follow a path formed by poses in a simple manner. Moreover, in relation to (Morin and Samson, 2011) the RLV kinematic constraint $\theta_R, \theta_F \neq \pm\pi/2$ is lifted by considering as control inputs v_R, v_F, θ_R and θ_F, as opposed to (Morin and Samson, 2011) where only $\dot{\theta}_R$, $\dot{\theta}_F$ and the longitudinal speed of the vehicle are used.

While operating in the scenarios of ITER, a RLV like the CPRHS is estimated to have a considerable speed for a vehicle with a mass up to 100T. Such kinetic energy in a vehicle with the dimensions and weight of the CPRHS, while operating under a 10 to 30 cm safety margin, represents a major challenge. Hence, a dynamic model of an RLV is detailed to enable a more realistic CPRHS simulation. The impact of simulated vehicle dynamics, mainly in the from of wheel slippage, in the developed four path following controllers, is compared against the strictly kinematic situation, where no slippage occurs.

The paper is organized as follows: Section 2 details the kinematic model of a RLV and formulates the path following control problem; Section 3 presents the four path following controllers; Section 4 presents the dynamic model of a RLV; Section 5 lays out the results; and Section 6 outlines the conclusions and directions for future work.

2 PROBLEM FORMULATION

This section comprises a mathematical formulation of the kinematic model of a RLV, essential for the de-

sign of the path following controllers of Section 3, followed by a precise problem statement.

2.1 Vehicle Kinematic Modelling

The formulation of a kinematic model for a RLV establishes the mathematical equations which relate the temporal variations of the vehicle pose, with the linear velocities on the wheels (i.e. steering plus the wheel linear speed). It consists on a pure geometrical study that is carried out without considering vehicle dynamic properties such as mass, inertia, slip or friction.

On this line, and referring to Fig. 1, consider the state vector $q = [x_c\, y_c\, \theta_m]$ as a representation of the vehicle pose in the frame $\{\mathrm{I}\}$, with (x_c, y_c) the coordinates of the center of the vehicle and θ_m the orientation of the vehicle. Also, consider v_t as the vehicle's total speed and β the controllable sideslip angle, both defined in $\{\mathrm{I}\}$ (Fig. 1). A kinematic model for a RLV in $\{\mathrm{I}\}$, that allows the simulation of the vehicle motion directly through the desired total speed v_t of the vehicle, instead of imposing an individual linear speed for each wheel, was introduced in (Wang, 2001) as follows:

$$\begin{bmatrix} \dot{x}_c \\ \dot{y}_c \\ \dot{\theta}_m \end{bmatrix} = \begin{bmatrix} \cos(\theta_m + \beta) \\ \sin(\theta_m + \beta) \\ \frac{\cos\beta[\tan\theta_F - \tan\theta_R]}{M} \end{bmatrix} v_t. \qquad (1)$$

This modelling entails that the wheels of the vehicle roll without slipping, a constraint inherent to the nonholonomy of RLVs, and also considers a rigid body constraint, common to this type of vehicles, as follows:

$$v_F \cos\theta_F = v_R \cos\theta_R. \qquad (2)$$

The possibility of directly control β, allows the vehicle to move either with both wheels following the same path (LG), or with each wheel following a different path (FR) (Vale et al., 2012).

2.2 Error Coordinates

A path following controller should reduce to zero i) the distance from the vehicle to the path and ii) the angle between the current orientation of the vehicle and the desired orientation at the path. This motivates the development of a kinematic model for a RLV in a Serret-Frenet $\{\mathrm{F}\}$ frame that moves along the path, as depicted in Fig. 3. This type of approach can be found in (Micaelli and Samson, 1993) for wheeled mobile robots, where $\{\mathrm{F}\}$ is attached to the path point closest to the vehicle. This implies that the initial position error of the vehicle in relation to the path, would have to be smaller than the smallest radius of curvature present in the path. The current work follows the approach taken in (Lapierre et al., 2006) for wheeled mobile robots and in (Lapierre and Jouvencel, 2008) for underwater vehicles, which starting from the grounds established in (Micaelli and Samson, 1993), lifts the initial condition constraint entirely, by attaching $\{\mathrm{F}\}$ to a virtual moving target along the path. This introduces an extra degree of freedom at the controller design stage.

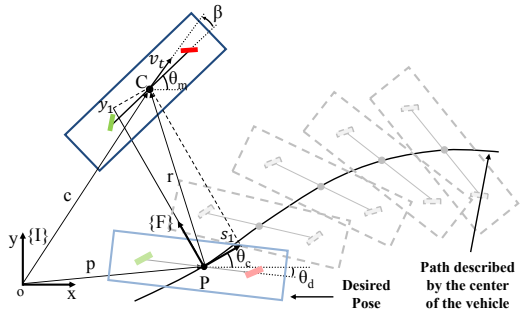

Figure 3: RLV variables and $\{\mathrm{F}\}$ frame definitions.

Consider Fig. 3, where P is an arbitrary path point to be followed and C is the CoG of the vehicle. Associated with P, consider the corresponding Serret-Frenet frame $\{\mathrm{F}\}$, with the signed curvilinear abscissa of P along the path denoted as s. One can either express C as $q = (X, Y, 0)$ in a selected inertial reference frame $\{\mathrm{I}\}$ or as $(s_1, y_1, 0)$ in $\{\mathrm{F}\}$. Stated equivalently, C can be given in (X, Y) or (s_1, y_1) coordinates. Furthermore, define θ_c as the angle of the tangent to the path at P, $\dot{\theta}_c = c_c(s)\dot{s}$ with $c_c(s)$ the path curvature as a function of s, and θ_d as the desired orientation of the vehicle at the path. Referring to the aforementioned literature and using (1), one can obtain $(\dot{s}_1, \dot{y}_1)$ as

$$\begin{cases} \dot{s}_1 = -\dot{s}(1 - c_c(s)y_1) + v_t\cos\theta \\ \dot{y}_1 = -c_c(s)\dot{s}s_1 + v_t\sin\theta \\ \dot{\theta} = \omega_m + \dot{\beta} - c_c(s)\dot{s} \end{cases}, \qquad (3)$$

where $\theta = \theta_m + \beta - \theta_c$ and $\omega_m = \dot{\theta}_m$.

This kinematic model of a RLV in $\{\mathrm{F}\}$, with s_1 not necessarily equal to zero, plays an important role in the development of the nonlinear controller presented in Section 3.4.

2.3 Path Following

With the background given in the current section, and with reference to Fig. 3, the path following problem under study can be formulated as:

Consider the kinematic model for RLVs given by (1) *and* (3)*. Given a LG or FR path to be followed parametrized in terms of its length and a desired speed profile* $v_{t_d}(t) > v_{t_{min}} > 0$ *for the vehicle*

total speed v_t, derive the feedback control laws for the linear speeds v_F, v_R and orientations θ_F, θ_R of the wheels, or alternatively β and ω_m, so that s_1, y_1 and θ tend to zero.

In the current paper, the desired speed profile mentioned above is embedded in the trajectory provided by the motion planning described in (Vale et al., 2012). Moreover, the formulation of the control laws for β and ω_m requires a transformation $[\omega_m, \beta, v_t] \rightarrow [v_F, v_R, \theta_F, \theta_R]$, detailed in Section 3, since, ultimately, one must determine the linear speeds and orientations of the wheels that drive the vehicle to the path.

3 PATH FOLLOWING CONTROL DESIGN

In what follows are unveiled the four path following controllers developed for RLVs, which represents the major novelty of this paper.

3.1 Alonzo Kelly Modified Controller

Two feedback control laws are here introduced by the authors that solve the problem enunciated in Section 2.3. The Alonzo Kelly Modified (AKM) path following controller draws its inspiration on the Inverse Kinematic Model (IKM) for RLVs proposed by A. Kelly in (Kelly, 2010). The formulation of the AKM controller herein presented is capable of both LG and FR, requiring only as input from the motion planning stage the path poses to be followed.

Referring to Fig. 1, the IKM described in (Kelly, 2010) and presented here in (4), enables the transformation $[\omega_m, \beta, v_t] \rightarrow [v_F, v_R, \theta_F, \theta_R]$. To obtain the values of $v_F, v_R, \theta_F, \theta_R$, while respecting the rigid body constraint (2) and, therefore, guaranteeing that the wheels roll without slipping, one can use the trigonometric relations $\theta_i = \arctan\left(\frac{v_{iy}}{v_{ix}}\right)$, for the wheels' orientation, and $v_i = \sqrt{v_{ix}^2 + v_{iy}^2}$, for the wheels' speed, with $i \in \{R, F\}$.

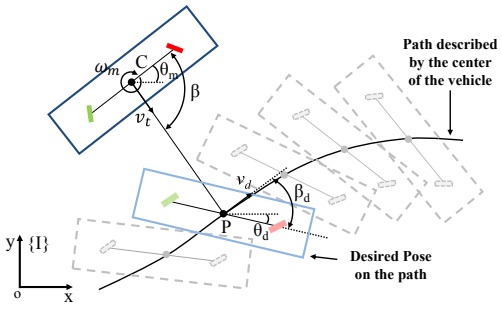

Figure 4: Alonzo Kelly Modified controller representation and variables definition.

$$\begin{bmatrix} v_{Fx} \\ v_{Fy} \\ v_{Rx} \\ v_{Ry} \end{bmatrix} = \begin{bmatrix} \cos\beta & 0 \\ \sin\beta & M_F \\ \cos\beta & 0 \\ \sin\beta & -M_R \end{bmatrix} \begin{bmatrix} v_t \\ \omega_m \end{bmatrix} \quad (4)$$

Hence, this IKM enables the formulation of a path following controller that only requires as input the desired pose of the vehicle at the path, $[P_x, P_y, \theta_d]$, and upon the control of β and ω_m, outputs the values of $v_F, v_R, \theta_F, \theta_R$ that enable the vehicle to converge and follow the desired path (Fig. 4). Moreover, the kinematic singularity $\theta_R, \theta_F \neq \pm\pi/2$ present in (Morin and Samson, 2011) is lifted entirely with the additional v_F, v_R control inputs introduced here. As mentioned previously, v_t is assumed to follow the desired speed profile embedded in the trajectory as described in (Vale et al., 2012).

A control law for β is used to direct the vehicle's velocity towards the desired path point, as depicted in Fig. 4. Referring to the IKM (4) and excluding, for the moment, the effect of ω_m in the velocity of each wheel, the β control law orientates the wheels in a manner that they point towards the desired path point. Such an objective can be embodied in the geometric law that follows:

$$\beta = \arctan\left(\frac{P_y - C_y}{P_x - C_x}\right) - \theta_m. \quad (5)$$

where (P_x, P_y) are the coordinates of the center of the path pose being followed and (C_x, C_y) the coordinates of the center of the vehicle (Fig. 4). By replacing (5) in (4), the wheels' velocity and, therefore, the vehicle's velocity is, at all times, directed towards the desired path pose.

A control law for ω_m is responsible for orientating the vehicle with the desired orientation at the path. A function that embodies this objective, can be formulated as follows:

$$\omega_m = \frac{(\theta_d - \theta_m) k_{\omega_m}}{t_n}, \quad (6)$$

with t_n the simulation step and k_{ω_m} a positive gain that tunes the vehicle's response rate to orientation variations.

Another degree of freedom inherent to the AKM controller is the choice of the path point to follow. In the formulation of (5) and (6) is chosen the closest path point P, this being the orthogonal projection of the C on the path (Fig. 4). Though, by introducing an offset on the path point to be followed, referred in the literature as lookahead distance, the performance

of the controller can be significantly improved. In fact, due to the impossibility of the vehicle's wheels to instantaneously change orientation or speed, the introduction of a lookahead distance might guarantee that when the vehicle actually reaches the path point "looked at", the pose becomes the desired one. Moreover, the lookahead distance can be fixed or speed dependant, depending on whether there is a varying speed profile, as it happens in the present case.

In brief, as tunable parameters the AKM controller has the choice of the lookahead point P and the gain k_{ω_m}.

3.2 Arc Path Following Controller

A geometric control method has been developed by the authors, named Arc Path Following (APF) controller, capable of both LG and FR. The key idea of APF is to exploit the high degree of mobility of RLVs by computing a simple but effective feedback control law based on geometric and kinematic reasoning. Since the vehicle is free to define its Instantaneous Center of Rotation (ICR) in $\mathbb{R}^2$, it is in any case possible (unless specific physical constraints exist) to compute the generalized circular path that steers the vehicle from the current pose to the desired one. Thereby this strategy solves the problem enunciated in Section 2.3. The concept is captured in Fig. 5. A similar approach using circular path manifolds in a more general framework has been found in (Kim and Minor, 2007), whereas the use of clothoids is proposed in (De Cecco et al., 2007).

The general solution to the desired *ICR* (ICR_d) expressed in the mobile reference frame $\{\mathrm{M}\}$, arises as

$$(ICR_d)_M = \begin{bmatrix} \frac{1}{2}\left(T_x + \frac{T_y}{\tan(\theta_e/2)}\right) \\ \frac{1}{2}\left(T_y + \frac{T_x}{\tan(\theta_e/2)}\right) \end{bmatrix}, \quad (7)$$

where $\theta_e = \theta_m - \theta_d$ and $T = [T_x, T_y]$ is the vector $P - C$ expressed in $\{\mathrm{M}\}$ (see Fig. 5).

The desired radius of curvature for the front (Rd_F) and rear (Rd_R) wheels, expressed in $\{\mathrm{M}\}$, are

$$(Rd_F)_M = \begin{bmatrix} Rd_{F_x} & Rd_{F_y} \end{bmatrix}^T = (F)_M - (ICR_d)_M, \quad (8)$$

$$(Rd_R)_M = \begin{bmatrix} Rd_{R_x} & Rd_{R_y} \end{bmatrix}^T = (R)_M - (ICR_d)_M, \quad (9)$$

with $(F)_M = \left[\frac{M}{2}\ 0\right]$ and $(R)_M = \left[-\frac{M}{2}\ 0\right]$.

Thus the feedback control laws, one for each wheel, can be written, using (7)-(9), as

$$\theta_F = \arctan\left(\frac{Rd_{F_y}}{Rd_{F_x}}\right) - \frac{\pi}{2}\mathrm{sign}\left(Rd_{F_y}\right), \quad (10)$$

$$\theta_R = \arctan\left(\frac{Rd_{R_y}}{Rd_{R_x}}\right) - \frac{\pi}{2}\mathrm{sign}\left(Rd_{R_y}\right). \quad (11)$$

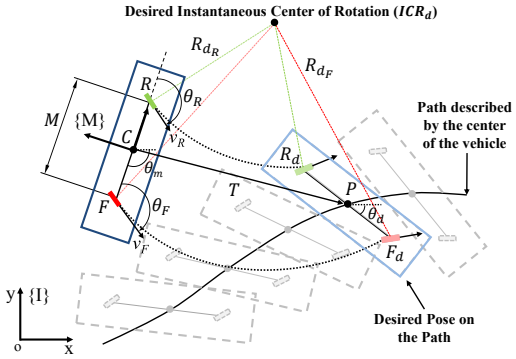

Figure 5: Arc Path Following controller representation and variables definition.

Expanding (10) and (11), the feedback control laws become:

$$\theta_F = \arctan\left(\frac{-T_y\tan(\theta_e/2) - T_x}{(M - T_x)\tan(\theta_e/2) + T_y}\right) + \\ + \frac{\pi}{2}\mathrm{sign}\left(T_y + \frac{T_x}{\tan(\theta_e/2)}\right), \quad (12)$$

$$\theta_R = \arctan\left(\frac{-T_y tan(\theta_e/2) - T_x}{-(M + T_x)\tan(\theta_e/2) + T_y}\right) + \\ + \frac{\pi}{2}\mathrm{sign}\left(T_y + \frac{T_x}{\tan(\theta_e/2)}\right). \quad (13)$$

Equations (12) and (13) present singularities which can be easily solved. They come from three singular cases:

1. both numerator and denomitator in the arctan functions are zero;
2. $\theta_e = 0$ in the *sign* function, that is when the current vehicle attitude is equal to the desired one.
3. $T_x = T_y = 0$ in the *sign* function, that is when the current reference point is equal to the desired one.

Case 1) occurs when either the current position of the forward or the rear wheel are coincident with the desired ones. To solve this case it is sufficient to set, respectively, $ICR_d = (F)_M$ or $ICR_d = (R)_M$.

Case 2) is solved by substituting (12)-(13) by their limit as $\theta_e \to 0$:

$$\theta_F = \theta_R = \arctan\left(\frac{Ty}{Tx}\right). \quad (14)$$

Case 3) occurs when the reference point of the current and desired vehicle pose are coincident ($C = P$). This is solved by setting $ICR_d = (C)_M$.

It has to be noticed that all the singularities described above do not consist in discontinuities of the

control laws. They are degeneracy cases of the solutions expressed by (7) and (12)-(13) due to special geometric configurations in the problem definition. By the above considerations, the control law is smooth for every possible vehicle pose.

Similarly to the AKM controller, the APF controller formulation enables the introduction of a lookahead distance, i.e. it is not mandatory that P is the orthogonal projection of C on the path (Fig. 5). This represents an extra degree of freedom in the controller that can be exploited to achieve smoother and better path following performances.

In brief, the APF controller has the choice of the lookahead point P as its only tunable parameter.

3.3 Stanley Modified Controller

The Stanley Method Modified (SMM) path following controller, is an extension to RLVs of the path following controller used by Stanford University's autonomous car entry in the DARPA Grand Challenge, named Stanley (Thrun et al., 2006). This controller is capable of both LG and FR, though, it requires as inputs the path to be followed by both wheels, in contrast with the other controllers here presented that only require the path poses.

Considering the path following controller described in (Thrun et al., 2006), the SMM employs two nonlinear feedback functions of the cross-track errors e_F and e_R, respectively, which are a measure of the distance from the center of each wheel to the nearest point on the respective wheel path, as shown in Fig. 6. The SMM approach results in two control laws, one for each wheel, responsible for converging e_F and e_R to zero and for which exponential convergence can be shown (Thrun et al., 2006).

The feedback control laws for θ_F and θ_R can be written, with reference to Fig. 6, as

$$\theta_F = \theta_{e_F} + \arctan\left(\frac{ke_F}{v_F}\right), \tag{15}$$

$$\theta_R = \theta_{e_R} + \arctan\left(\frac{ke_R}{v_R}\right), \tag{16}$$

where: θ_F, θ_R are the required outputs; v_F, v_R are the desired front and rear wheel speeds, respectively, embedded in the trajectory as described in (Vale et al., 2012); k is a gain parameter, chosen equally for both wheels to maintain vehicle symmetry; e_F, e_R represent the cross-track errors for the front and rear wheel, respectively; and $\theta_{e_F}, \theta_{e_R}$ describe the orientation of the nearest path segment, measured relative to the vehicle's own orientation (see Fig. 6). The first term of (15) and (16) simply keeps the wheel parallel to the respective wheel path. When the cross-track error is non-zero, the second term adjust the steering of each wheel in (nonlinear) proportion to the cross-track error: the larger this error, the stronger the steering response towards the respective wheel path.

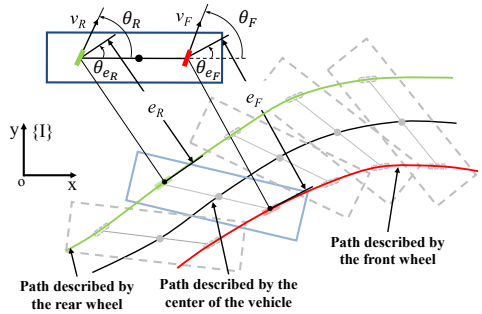

Figure 6: Stanley Method Modified controller representation and variables definition.

Against to what was mentioned for the AKM controller, the formulation above for the SMM controller precludes the introduction of a lookahead distance. The reference path point to be followed by each wheel must inevitably be the orthogonal projections of the contact points of the wheels in the respective wheel path.

In brief, the SMM controller has the gain k as its only tunable parameter.

3.4 Nonlinear Based Controller

Two nonlinear feedback control laws have been developed by the authors to steer the kinematic model (3), along a desired path and at constant speed ($\dot{v}_t = 0$). The Nonlinear Control Based (NCB) path following controller, has its grounds on the work done in (Micaelli and Samson, 1993; Lapierre et al., 2006) on path following control for kinematic models of wheeled mobile robots and in (Lapierre and Jouvencel, 2008) on the same topic but for autonomous underwater vehicles. In contrast to the work described in (Micaelli and Samson, 1993), the usage of the virtual target principle enables the formulation of the path following problem in a non-singular manner, thus guaranteeing global convergence to the path. Moreover, the NCB controller sets apart from the kinematic controller developed in (Lapierre and Jouvencel, 2008) by deliberately controlling the rate of evolution of the vehicle's sideslip angle, $\dot{\beta}$, separately from the vehicle's angular speed, ω_m. This enables the vehicle to converge and follow a desired LG or FR path, exploiting the full maneuverability of a RLV.

Referring to Fig 3, the objective of deriving a control law for β can be embodied in the Lyapunov function candidate (see (Lapierre et al., 2006))

$$V_1 = \frac{1}{2}\left(s_1^2 + y_1^2\right) + \frac{1}{2\gamma}\left(\theta - \delta_\beta(y_1, v_t)\right)^2, \quad (17)$$

under the assumptions (A.1) $y_1 v_t \sin\delta_\beta(y_1, v_t) \leq 0, \forall y_1 \forall v_t$, (A.2) $\delta_\beta(0, v_t) = 0$ and (A.3) $\lim_{t\to\infty} v_t(t) \neq 0$. Also, let the desired approach angle function δ_β be defined by

$$\delta_\beta = -\theta_\beta \tanh\left(k_{\delta_\beta} y_1 v_t\right), \quad (18)$$

where $0 < \theta_\beta < \pi/2$ and k_{δ_β} is an arbitrary positive gain. Notice how equation (18) satisfies the first and second assumptions.

In (17) the first term captures the distance between the vehicle and the path, which must be reduced to 0. In turn, this term also precludes the introduction of a lookahead distance in the NCB controller. The second term shapes the approach angle of β to the path as a function of the 'lateral' distance y_1 and speed v_t, by forcing it to follow a desired orientation profile embedded in (18). The parameter γ accounts only for normalization purposes. The first assumption provides an adequate reference sign definition to drive the vehicle to the path, i.e. turn the vehicle left when on the right side of the path, and turn right in the other situation. A.2 imposes the condition that β must be such that the vehicle's velocity is tangent to the path when $y_1 = 0$, regardless of the vehicle orientation. Finally, the last assumption states that the vehicle does not tend to a state of rest, case when the controllability cannot be guaranteed.

Consider the following kinematic control laws for s, the virtual moving target, and β as:

$$\dot{s} = v_t \cos\theta + k_s s_1, \quad (19)$$

$$\dot{\beta} = \dot{\delta}_\beta + c_c(s)\dot{s} - \dot{\theta}_m - \gamma y_1 v_t \frac{\sin\theta - \sin\delta_\beta}{\theta - \delta_\beta} - k_\beta(\theta - \delta_\beta), \quad (20)$$

with k_β and k_s positive gains. This yields

$$\dot{V}_1 = -k_s s_1^2 + y_1 v_t \sin\delta_\beta - k_\beta \frac{\left(\theta - \delta_\beta\right)^2}{\gamma} \leq 0, \quad (21)$$

where the presence of the term $y_1 v_t \sin\delta_\beta$ in the previous equation justifies A.1. Moreover, (21) depicts the importance of the virtual target control law (19) on guaranteeing global convergence to the path.

Therefore, the kinematic control laws (19)-(20) drive s_1, y_1 and θ asymptotically to zero, i.e. the vehicle converges asymptotically to the path. The proof builds similarly to the ones found in (Micaelli and Samson, 1993; Lapierre et al., 2006) and (Lapierre and Jouvencel, 2008).

The second objective of deriving a control law for ω_m can be embodied in the Lyapunov function candidate

$$V_2 = \frac{1}{2\gamma}\left(\theta_m - \theta_d - \delta_{\omega_m}(y_1, v_t)\right)^2, \quad (22)$$

under the assumption (A.4) $\delta_{\omega_m}(0, v_t) = 0$ and with δ_{ω_m} defined in a similar manner as in (18).

The Lyapunov function (22) captures the orientation error between the current orientation of the vehicle and the desired orientation at the path, which must be reduced to zero. A.4 guarantees that the vehicle's orientation must be equal to the desired orientation at the path when $y_1 = 0$.

By making the kinematic control law for $\omega_m = \dot{\theta}_m$, required for Lyapunov stability, as

$$\omega_m = \dot{\delta}_{\omega_m} + \dot{\theta}_d - k_{\omega_m}(\theta_m - \theta_d - \delta_{\omega_m}), \quad (23)$$

with k_{ω_m} a positive gain, $\dot{V}_2$ becomes

$$\dot{V}_2 = -k_{\omega_m} \frac{(\theta_m - \theta_d - \delta_{\omega_m})^2}{\gamma} \leq 0. \quad (24)$$

Hence, control law (23) drives $\theta_m - \theta_d$ asymptotically to zero, i.e. the vehicle's orientation converges asymptotically to the desired orientation at the path. Similarly, the proof is alike to the ones found in (Micaelli and Samson, 1993; Lapierre et al., 2006) and (Lapierre and Jouvencel, 2008).

To determine the values of v_F, v_R, θ_F and θ_R, the IKM (4) is used. The required value of ω_m is obtained directly from the control law (23), while β is obtained by integrating the output of the control law (20). Similarly to the AKM controller, v_t is provided by the trajectory.

In brief, the NCB controller has the following tunable parameters: the normalization parameter γ, the gains k_{δ_β}, $k_{\delta_{\omega_m}}$, k_β, k_s and k_{ω_m}, and the approach angles θ_β and θ_{ω_m}.

4 DYNAMIC MODELLING

The RLV kinematic models (1) and (3), are able to capture the essentials of the motion of RLVs. However, the non-slipping assumptions in which the derivation of the kinematic model was based, will hardly be satisfied in reality if the vehicle's kinetic energy is significant. Hence, the dynamic formulation that follows aims to provide a more accurate simulation scenario for the test of the controllers developed in Section 3. In the present case, the RLV is regarded as a rigid vehicle with a strictly planar motion. Hence, it has three degrees of freedom: x and y translation, and a rotation about the z-axis. The Newton-Euler equations of motion for a rigid vehicle in the vehicle

frame {M}, attached to the vehicle at its mass center C, are (Jazar, 2008):

$$\begin{cases} F_x = m\dot{u} - m\omega_m v \\ F_y = m\dot{v} + m\omega_m u \\ M_z = \ddot{\theta}_m I_{zz} \end{cases} . \quad (25)$$

The symbols m and I_{zz} represent, respectively, the mass and moment of inertia, around the z axis, of the vehicle. In the frame {M}, u and v denote the longitudinal and transverse speeds of the vehicle, respectively.

To determine the forces F_x, F_y and moment M_z, must be noticed that the vehicle motion is a consequence of forces acting at the vehicle's wheels. In fact, the forces acting on the vehicle can be categorized as: transverse forces, F_{s_i}, and longitudinal forces, F_{l_i}, acting on the wheels, with $i \in \{R,F\}$. This formulation entails, with reference to Fig. 1:

$$\begin{aligned} m\dot{u} - m\omega_m v =& F_{l_F}\cos\theta_F - F_{s_F}\sin\theta_F \\ &+ F_{l_R}\cos\theta_R - F_{s_R}\sin\theta_R, \end{aligned} \quad (26)$$

$$\begin{aligned} m\dot{v} + m\omega_m u =& F_{l_F}\sin\theta_F + F_{s_F}\cos\theta_F \\ &+ F_{l_R}\sin\theta_R + F_{s_R}\cos\theta_R, \end{aligned} \quad (27)$$

$$\begin{aligned} \ddot{\theta}_m I_{zz} =& F_{l_F}\sin\theta_F M_F + F_{s_F}\cos\theta_F M_F \\ &- F_{l_R}\sin\theta_R M_R - F_{s_R}\cos\theta_R M_R. \end{aligned} \quad (28)$$

To calculate F_{s_i} and F_{l_i}, is followed the approach taken in (Fonte, 2011), where a simplified linear tire model is used, as follows:

$$F_{s_i} = C_s\alpha_i,\ \alpha_i = \left(\theta_i - \arctan\left(\frac{v \pm \dot{\theta}_m M_i}{u}\right)\right), \quad (29)$$

$$F_{l_i} = C_l\sigma_i,\ \sigma_i = \left(\frac{v_i - v_i^{eff}\cos\alpha_i}{\max\left(|v_i|, \left|v_i^{eff}\cos\alpha_i\right|\right)}\right). \quad (30)$$

The following definitions apply (Fonte, 2011): $C_l = f(F_{z_i})$ and $C_s = f(F_{z_i})$ are the longitudinal and cornering stiffness, respectively, which depend on the normal force applied at the wheel, F_{z_i}, and considered equal for both wheels ($F_{z_F} = F_{z_R}$); α_i is the wheel slip angle; σ_i is the wheel longitudinal slip ratio; v_i^{eff} is the effective rolling speed of wheel i given by $v_i^{eff} = \sqrt{u^2 + \left(v \pm \dot{\theta}_m M_i\right)^2}$; v_i is the inputted wheel speed (at the actuator level); and $\pm$ is positive for $i = F$ and negative for $i = R$.

The combined slip effects, which occur when both $F_{s_i}, F_{l_i} \neq 0$, is taken into account by clipping (29) and (30) as stated in (Jazar, 2008):

$$F_{s_i}^2 + F_{l_i}^2 = \left(\mu F_{z_i}\right)^2, \quad (31)$$

where μ is the friction coefficient between the tire and the contact surface.

Furthermore, a discrete first order model in the linear speed of the wheels and a discrete second order model in the orientation of the wheels, is used to simulate the typical behavior of wheel actuators.

5 RESULTS

This section illustrates the simulated performance of the four path following controllers described in Section 3. The simulations were performed using the Trajectory Evaluator and Simulator (TES) software developed under the scope of previous grants of Remote Handling with the domestic agency Fusion for Energy (F4E), as described in (Vale et al., 2012). The vehicle simulated was a RLV with the characteristics of the CPRHS, while following a trajectory to port 14 in level B1 of TB of ITER (Fig. 7). The trajectory used in the simulations is a discrete FR path, at constant speed, formed by a set of consecutive poses, as depicted in Fig. 7. This trajectory was calculated using the motion planning realized in (Vale et al., 2012).

The performance of the four controllers is assessed by measuring the cross-track error of the center of the vehicle, e_C, along the path, as depicted in Figs. 8 and 9. To determine the influence of vehicle dynamics in the performance of the controllers, two series of tests were realized: on the first, the vehicle total speed, v_t, is set constant to 0.15 m/s and the mass is 50T (Fig. 8); on the second test, v_t is set to 0.4 m/s and the mass is 100T (Fig. 9). On both series of tests, each of the controllers is tested individually under two situations, these being using the kinematic model (1),

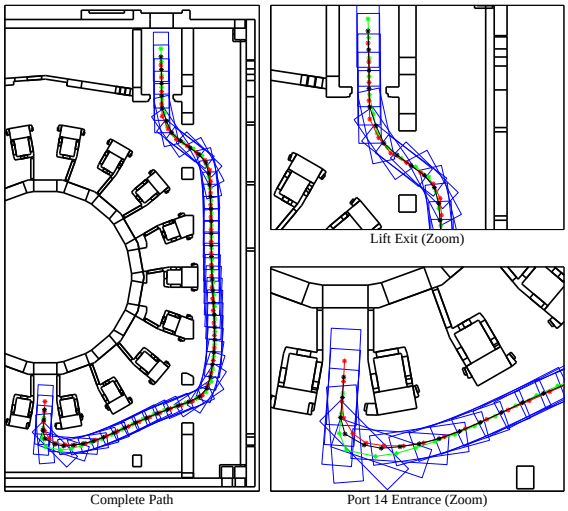

Figure 7: Path used in the simulation: FR path from lift to port 14 in level B1 of TB of ITER.

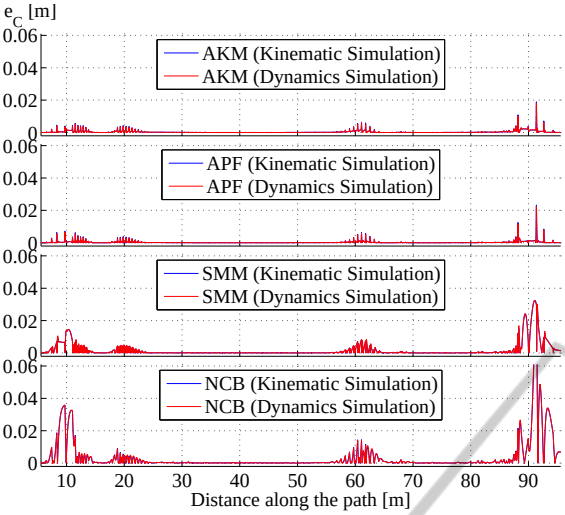

Figure 8: Cross-track error, e_C, comparison for each of the four controllers, while simulating, or not, vehicle dynamics, in a trajectory to port 14 in level B1 of TB of ITER. The vehicle speed is 0.15 m/s and the mass is 50T.

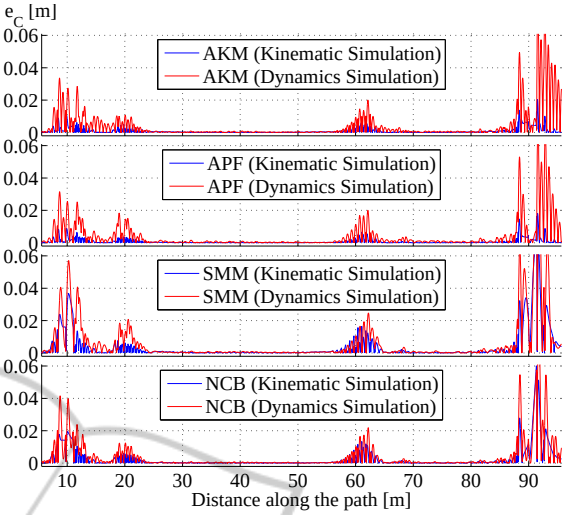

Figure 9: Cross-track error, e_C, comparison for each of the four controllers, while simulating, or not, vehicle dynamics, in a trajectory to port 14 in level B1 of TB of ITER. The vehicle speed is 0.4 m/s and the mass is 100T.

or using the dynamic model (25), for simulating the motion of a RLV. Though, it is important to mention that actuators dynamics is always present across the tests and the controllers gains are kept the same.

As it can be seen in Figs. 8 and 9, the controllers performance deteriorates as v_t increases and while negotiating turns (zoomed areas in Fig. 7). More importantly, Fig. 8 demonstrates how wheel slippage at 0.15 m/s for a 50T RLV is negligible, since the performance of the controllers is the same regardless of the utilization of a kinematic or dynamic model for vehicle simulation. In contrast, Fig. 9 depicts how the performance of the controllers worsens at 0.4 m/s for a 100T RLV if a dynamic vehicle model, instead of a kinematic one, is used for simulation. This result enhances the importance of including vehicle dynamic considerations in the design of the controllers.

Overall, the AKM and APF controllers attain similar performances and surpass the NCB and SMM controllers. This demonstrates how the inclusion of a lookahead distance contours the non consideration of vehicle and actuators dynamics in the path following controllers formulation. To achieve the results shown for the AKM and APF controllers, a 0.09 m lookahead distance was used. This allowed a superior performance while using a kinematic model for vehicle simulation. Though, it is also the reason of the strong oscillations while using a dynamic model for simulation (Fig. 9). In fact, a high lookahead distance (e.g. 1 m) increases e_C, but enables the controller to better stabilize the vehicle under high wheel slippage. Table 1 summarizes the controllers performance at port 14 entrance (Fig. 7), this being the most critical part of the path.

Table 1: RMSE of e_C for each of the controllers at port 14 entrance, while using a kinematic, or a dynamic, model for vehicle simulation. Units are in [mm].

Controllers	Kinematic Model		Dynamic Model	
	0.15 m/s	0.40 m/s	0.15 m/s 50T	0.40 m/s 100T
AKM	1.8	4.3	1.7	25.2
APF	2.1	3.6	1.9	22.5
SMM	12.1	23.2	12.2	36.1
NCB	25.7	16.1	25.8	26.4

In relation to the NCB controller, a comparison between Figs. 8 and 9 reveals that, while using a kinematic model for simulation, the NCB performance is better at 0.4 m/s than at 0.15 m/s. Although it seems contradictory, this is justified by (18), which depicts how, for the same gains, a stronger path deviation correction is achieved if the vehicle total speed, v_t, is higher. This issue can be mitigated at a kinematic level by increasing the NCB gains. Though, while dealing with vehicle dynamics, high gains would increase vehicle instability and decrease path following performance.

It should be noticed that all the four path following controllers perform well within the 30 cm safety margin established for the scenario, even while dealing with the dynamics of a 100T vehicle at 0.4 m/s. The maximum e_C registered was 11.95 cm for the SMM controller at 0.4 m/s with dynamics.

6 CONCLUSIONS

Four path following controllers for RLVs were developed and implemented. Alongside, the dynamic modelling of a RLV was realized. The tests aimed to assess the controllers performance while using either the kinematic or the dynamic vehicle model for simulation. All the controllers perform within the strict safety margin constraints established for the cluttered scenarios of ITER and attain good results, even in the presence of some wheel slippage. The study of four distinct controllers enables the conclusion that both the AKM and APF controllers slightly stand out from the other two, due to the inclusion of a lookahead distance which contours the vehicle and actuators dynamics issue. The SMM controller still enables a good performance, while being very simple, and is a good choice if the path of the wheels is available. The NCB controller also achieves a good performance and possesses the biggest room for improvement, by enabling the inclusion of vehicle and actuators dynamics through backstepping techniques.

The results were obtained only in simulation and a future route would pass by performing experimental tests at a small scale, to begin with, and, ultimately, at a larger scale. The results show that when the vehicle's mass and speed is significant, the controllers performance deteriorates. Hence, an improved version of the controllers is being developed, that takes into account dynamic considerations at the controllers design stage.

ACKNOWLEDGEMENTS

The work was supported by FCT in the frame of the Contract of Associate Laboratories of Instituto de Plasmas e Fusão Nuclear – Laboratório Associado/IST (Pest-OE/SADG/LA0010/2011).

REFERENCES

De Cecco, M., Bertolazzi, E., Miori, G., Oboe, R., and Baglivo, L. (2007). PC-sliding for vehicles path planning and control - Design and evaluation of robustness to parameters change and measurement uncertainty. In *Proceedings of the 4th International Conference on Informatics in Control, Automation and Robotics (ICINCO)*, pages 11–18, Angers.

Fonte, D. (2011). *Motion Planning for a Rhombic-Like Vehicle Operating in ITER Scenarios*. PhD thesis, Instituto Superior Técnico.

Hiraoka, T., Nishihara, O., and Kumamoto, H. (2009). Automatic path-tracking controller of a four-wheel steering vehicle. *Vehicle System Dynamics*, 47(10):1205–1227.

Jazar, R. N. (2008). *Vehicle Dynamics: Theory and Application*. Springer.

Kelly, A. (2010). A Vector Algebra Formulation of Kinematics of Wheeled Mobile Robots. Technical report, Robotics Institute.

Kim, Y. and Minor, M. a. (2007). Path Manifold-based Kinematic Control of Wheeled Mobile Robots Considering Physical Constraints. *The International Journal of Robotics Research*, 26(9):955–975.

Lapierre, L. and Jouvencel, B. (2008). Robust nonlinear path-following control of an AUV. *IEEE Journal of Oceanic Engineering*, 33(2):89–102.

Lapierre, L., Soetanto, D., and Pascoal, A. (2006). Non-Singular Path-Following Control of a Unicycle in the Presence of Parametric Modeling Uncertainties. *International Journal of Robust and Nonlinear Control*, 16(10):485–503.

Luca, A. D. and Oriolo, G. (1998). Feedback control of a nonholonomic car-like robot. In Laumond, J.-P., editor, *Robot motion planning and control*, chapter 4th. Springer.

Micaelli, A. and Samson, C. (1993). Trajectory tracking for unicycle-type and two-steering-wheels mobile robots. Technical report, INRIA.

Morin, P. and Samson, C. (2011). Control of two-steering-wheels vehicles with the Transverse Function approach. In *Proceedings of the 50th IEEE Conference on Decision and Control and European Control Conference (CDC-ECC)*, number 3, pages 7349–7355, Orlando.

Ribeiro, M. and Lima, P. (1997). Conceptual study on flexible guidance and navigation for ITER remote handling transport casks. In *Proceedings of the 17TH IEEE/NPSS Symposium on Fusion Engineering*, volume 2, pages 969–972, San Diego.

Thrun, S., Montemerlo, M., and Dahlkamp, H. (2006). Stanley: The robot that won the DARPA Grand Challenge. *Journal of Field Robotics*, 23(9):661–692.

Vale, A., Valente, F., Ribeiro, I., Ferreira, J., and Ventura, R. (2012). Motion planning and localization approaches for mobile robot navigation in ITER. In *Proceedings of Robótica 2012*, pages 75–80, Guimarães.

Wang, D. (2001). Trajectory Planning for a Four-Wheel-Steering Vehicle. In *Proceedings of the IEEE International Conference On Robotics And Automation*, volume I-IV, pages 3320–3325, Seoul.