

Four Path Following Controllers for Rhombic Like Vehicles

Nuno Silva¹, Luca Baglivo², Alberto Vale¹, Mariolino De Cecco³

Abstract—This paper addresses the path following problem of a wheeled mobile robot with rhombic like kinematics (two drivable and steerable wheels) 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 the vehicle using Lyapunov functions. By fully exploiting the kinematic capabilities of this type of vehicles, 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. Simulated and experimental results are presented in the paper to illustrate the performance of the controllers. The main conclusions of these controllers are summarized, leading to a possible application in the vehicles that will operate in the remote handling missions of the International Thermonuclear Experimental Reactor (ITER).

I. INTRODUCTION

From the wide variety of nonholonomic wheeled mobile robots configurations available [1], rhombic like vehicles (RLVs) as described in [2] and depicted in Fig. 1, have been distinctly outside the spotlight of the scientific community. RLVs 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, RLVs are a set of nonholonomic vehicles which enable a decoupling between the vehicle's orientation and velocity vector, creating a controllable sideslip angle, β (Fig. 1).

Based on the enhanced maneuverability of a RLV, the work initiated in [2] was strengthened in [3], [4], [5] and [6], where the applicability of such a vehicle in the cluttered environments of ITER (International Thermonuclear Experimental Reactor) was studied. One of the many operating scenarios of ITER is the fusion reactor installed in the Tokamak Building (TB), being a 2D/3D view of level B1 of this building visible in Fig. 2. Remote Handling (RH) systems will play an important role in the ITER project and one of such systems is the Cask and Plug Remote Handling System (CPRHS) depicted in Fig. 1: a mobile vehicle responsible for RH operations of transportation of activated components and equipment inside the TB. The CPRHS can reach dimensions

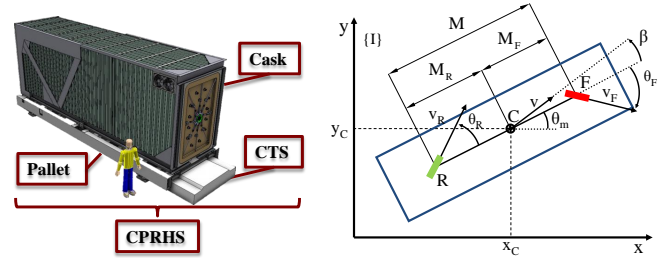


Fig. 1. CPRHS constitution (left) and RLV variables and parameters (right).

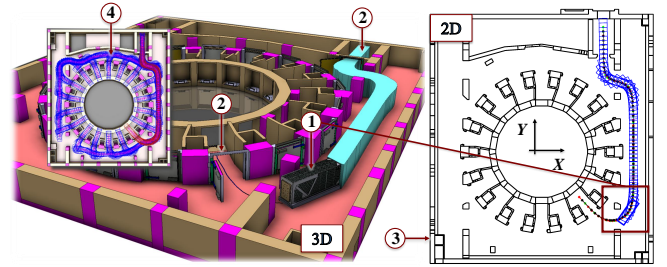


Fig. 2. Typical operating scenario for the RLV: 1- RLV (3D and 2D), 2- start and goal conditions, 3- 2D model of level B1 of TB, 4- optimized trajectories for each port in level B1 of TB (example trajectory in red).

of 8.50m x 2.62m x 3.70m (length, width, height) and a weight of 100T when fully loaded. In common, all the CPRHS are RLVs (Fig. 1) and have to operate within a narrow safety margin of 10 to 30 cm. 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) [6].

In [3], [4] and [5], a motion planning of the CPRHS in ITER was performed. The trajectories (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 sole purpose of the current paper to study different *path following* solutions for the CPRHS along the previously optimized trajectories. Moreover, this paper focuses on path following and not on trajectory tracking, since the objective is to steer the CPRHS along the optimized trajectories, while the vehicle's speed tracks the desired speed profile embedded in the trajectory. In contrast, trajectory tracking requires the CPRHS to track a time-parameterized reference [7], [8].

To solve the path following problem for unicycles, in [9], [10] 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. Simpler

¹N. Silva and A. Vale are with Instituto de Plasmas e Fusão Nuclear - Instituto Superior Técnico, Universidade Técnica de Lisboa, Av. Rovisco Pais 1, 1049-001 Lisboa, Portugal {nsilva,avale} at ipfn.ist.utl.pt

²L. Baglivo is with Laboratory of Robotics and Systems in Engineering and Science (LARSyS) - Instituto Superior Técnico, Universidade Técnica de Lisboa, Av. Rovisco Pais 1, 1049-001 Lisboa, Portugal {luca.baglivo} at isr.ist.utl.pt

³M. De Cecco is with DIMS - Mechanical and Structural Engineering Dept., University of Trento, Via Mesiano 77, 38123 Trento, Italy {mariolino.dececco} at ing.unitn.it

and less accurate controllers can be obtained employing geometric methods such as pure pursuit [11] or vector pursuit [12]. Though, given the strict safety margins of cluttered environments such as ITER, accuracy overcomes simplicity. Concerning car like vehicles, [8] 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, [13] explores the application of sliding mode controllers to the path following problem, improving robustness to uncertainties while compromising steering smoothness.

The only references to path following for RLVs are made in [9], [14] at a kinematic level only. In [9] 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 [14] 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 [14] via specific maneuvering embedded in the control design.

As novelty, 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 [9], the nonlinear controller proposed in this paper allows for a RLV to follow a path formed by poses. Moreover, in relation to [14] the 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 [14] where only $\dot{\theta}_R, \dot{\theta}_F$ and the longitudinal speed of vehicle, v , are used.

The paper is organized as follows: Section II details the kinematic model of a RLV and formulates the path following control problem; Section III presents the four path following controllers; Section IV lays out the results; and Section V outlines the conclusions and directions for future work.

II. PROBLEM FORMULATION

The subject of this section comprises a mathematical formulation of the kinematic model of the vehicle under study and precise path following problem statement.

A. 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 $\{I\}$, with (x_c, y_c) the coordinates of the center of the vehicle and θ_m the orientation of the vehicle. Also, consider v as the vehicle's total speed and β the controllable sideslip angle of the vehicle, both defined in $\{I\}$ (Fig. 1). A kinematic model for a RLV in $\{I\}$, that allows the simulation of the vehicle motion directly through the desired total speed v of the vehicle, instead of imposing an individual linear speed for each wheel, was introduced in [15] as follows:

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

where

$$\beta = \arctan \left(\frac{v_F \sin \theta_F + v_R \sin \theta_R}{2v_R \cos \theta_R} \right) \quad (2)$$

was obtained using [16], and

$$v = \frac{v_F \cos \theta_F + v_R \cos \theta_R}{2 \cos \beta} \quad (3)$$

is defined in the same way as in [15].

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:

$$v_F \cos \theta_F = v_R \cos \theta_R. \quad (4)$$

Moreover, the ability to control the sideslip angle β , results in the capability of the vehicle to move either with both wheels following the same path (LG), or with each wheel following a different path (FR) [6].

B. 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 $\{F\}$ frame that moves along the path, as depicted in Fig. 3. This type of approach can be found in [9] for wheeled mobile robots, where $\{F\}$ 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 [10] for wheeled mobile robots and in [17] for underwater vehicles, which starting from the grounds established in [9], lifts the initial condition

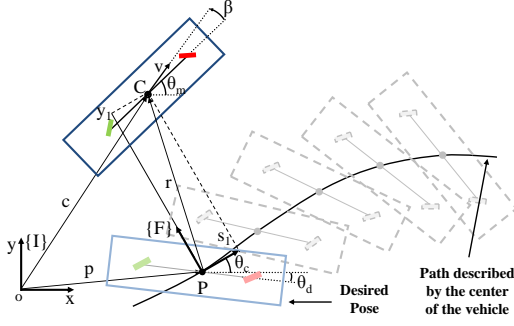


Fig. 3. RLV variables and $\{F\}$ frame definitions.

constraint entirely, by attaching $\{F\}$ to a virtual moving target along the path. This introduces an extra degree of freedom at the controller design stage.

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 $\{F\}$, 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 $\{I\}$ or as $(s_1, y_1, 0)$ in $\{F\}$. 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, one can obtain (\dot{s}_1, \dot{y}_1) as

$$\begin{cases} \dot{s}_1 = [\cos \theta_c & \sin \theta_c] \begin{bmatrix} \dot{x}_c \\ \dot{y}_c \end{bmatrix} - \dot{s}(1 - c_c(s)y_1) \\ \dot{y}_1 = [-\sin \theta_c & \cos \theta_c] \begin{bmatrix} \dot{x}_c \\ \dot{y}_c \end{bmatrix} - c_c(s)\dot{s}s_1 \end{cases} \quad (5)$$

By replacing (1) in (5), the kinematic model of a RLV in $\{F\}$ is obtained as follows

$$\begin{cases} \dot{s}_1 = -\dot{s}(1 - c_c(s)y_1) + v \cos \theta \\ \dot{y}_1 = -c_c(s)\dot{s}s_1 + v \sin \theta \\ \dot{\theta} = \omega_m + \dot{\beta} - c_c(s)\dot{s} \end{cases}, \quad (6)$$

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

This kinematic model, with s_1 not necessarily equal to zero, plays an important role in the development of the nonlinear controller presented in Section III-D.

C. 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 (6). Given a LG or FR path to be followed parametrized in terms of its length and a desired speed profile $v_d(t) > v_{min} > 0$ for the vehicle speed v , 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 [3] and [5]. Moreover, the formulation of the control laws for β and ω_m requires a transformation $[\omega_m, \beta, v] \rightarrow [v_F, v_R, \theta_F, \theta_R]$, detailed in Section III, since, ultimately, one must determine the linear speeds and orientations of the wheels that drive the vehicle to the path.

III. 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.

A. Alonzo Kelly modified controller

Two feedback control laws are here introduced by the authors that solve the problem enunciated in Section II-C. The Alonzo Kelly Modified (AKM) path following controller draws its inspiration on the Inverse Kinematic Model (IKM) for RLVs proposed by A. Kelly in [18]. 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 [18] and presented here in (7), enables the transformation $[\omega_m, \beta, v] \rightarrow [v_F, v_R, \theta_F, \theta_R]$. To obtain the values of $v_F, v_R, \theta_F, \theta_R$, while respecting the rigid body constraint (4) 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 $i \in \{R, F\}$.

$$\begin{bmatrix} v_{Fx} \\ v_{Fy} \\ v_{Rx} \\ v_{Ry} \end{bmatrix} = \begin{bmatrix} \cos \beta & 0 \\ \sin \beta & M_F \\ \cos \beta & 0 \\ \sin \beta & -M_R \end{bmatrix} \begin{bmatrix} v \\ \omega_m \end{bmatrix} \quad (7)$$

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

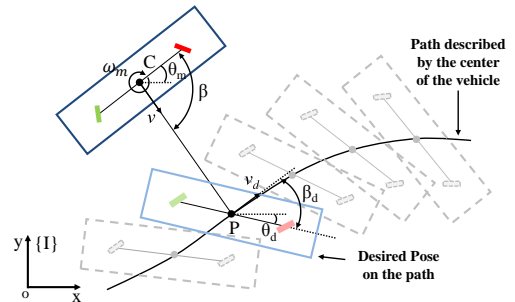


Fig. 4. Alonzo Kelly Modified controller representation and variables definition.

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 [14] is lifted entirely with the additional v_F, v_R control inputs introduced here. As mentioned previously, v is assumed to follow the desired speed profile embedded in the trajectory as described in [3] and [5].

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 detailed in (7) 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:

$$\beta = \arctan\left(\frac{P_y - C_y}{P_x - C_x}\right) - \theta_m. \quad (8)$$

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 (8) in (7), 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:

$$\omega_m = \frac{(\theta_d - \theta_m)k_{\omega_m}}{t_n}, \quad (9)$$

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 (8) and (9) 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} .

B. 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 [19]. 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 [19], the SMM employs two nonlinear feedback functions

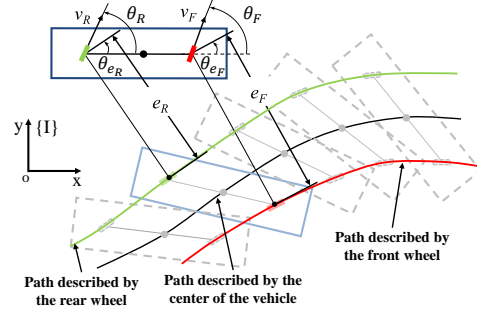


Fig. 5. Stanley Method Modified controller representation and variables definition.

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. 5. 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 [19].

The feedback control laws for θ_F and θ_R can be written, with reference to Fig. 5, as

$$\theta_F = \theta_{e_F} + \arctan\left(\frac{ke_F}{v_F}\right), \quad (10)$$

$$\theta_R = \theta_{e_R} + \arctan\left(\frac{ke_R}{v_R}\right), \quad (11)$$

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 [3] and [5]; 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. 5). The first term of (10) and (11) 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.

Against to what was mentioned for the AKM controller, the formulation above for the SMM controller precludes the introduction of a lookahead distance, as 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.

C. 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

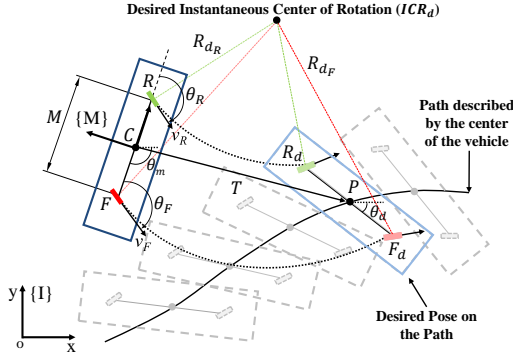


Fig. 6. Arc Path Following controller representation and variables definition.

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 II-C. The concept is captured in Fig. 6. A similar approach using circular path manifolds in a more general framework has been found in [20], whereas the use of clothoids is proposed in [21].

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

$$(ICR_d)_M = \begin{bmatrix} \frac{1}{2} \left(T_x + \frac{T_y}{\tan(\theta_e/2)} \right) \\ \frac{1}{2} \left(T_y + \frac{T_x}{\tan(\theta_e/2)} \right) \end{bmatrix}, \quad (12)$$

where $\theta_e = \theta_m - \theta_d$ and $T = [T_x, T_y]$ is the vector $P - C$ expressed in $\{M\}$ (see Fig. 6).

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

$$(Rd_F)_M = [Rd_{F_x} \quad Rd_{F_y}]^T = (F)_M - (ICR_d)_M, \quad (13)$$

$$(Rd_R)_M = [Rd_{R_x} \quad Rd_{R_y}]^T = (R)_M - (ICR_d)_M, \quad (14)$$

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

Thus the feedback control laws, one for each wheel, can be written, using (12)-(14), as

$$\theta_F = \arctan\left(\frac{Rd_{F_y}}{Rd_{F_x}}\right) - \frac{\pi}{2} \text{sign}(Rd_{F_y}), \quad (15)$$

$$\theta_R = \arctan\left(\frac{Rd_{R_y}}{Rd_{R_x}}\right) - \frac{\pi}{2} \text{sign}(Rd_{R_y}). \quad (16)$$

Expanding (15) and (16), the feedback control laws become:

$$\theta_F = \arctan\left(\frac{-T_y \tan(\theta_e/2) - T_x}{(M - T_x) \tan(\theta_e/2) + T_y}\right) + \frac{\pi}{2} \text{sign}\left(T_y + \frac{T_x}{\tan(\theta_e/2)}\right), \quad (17)$$

$$\theta_R = \arctan\left(\frac{-T_y \tan(\theta_e/2) - T_x}{-(M + T_x) \tan(\theta_e/2) + T_y}\right) + \frac{\pi}{2} \text{sign}\left(T_y + \frac{T_x}{\tan(\theta_e/2)}\right). \quad (18)$$

Equations (17) and (18) present singularities which can be easily solved. They come from three singular cases:

- 1) both numerator and denominator 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 (17)-(18) by their limit as $\theta_e \rightarrow 0$:

$$\theta_F = \theta_R = \arctan\left(\frac{T_y}{T_x}\right). \quad (19)$$

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 (12) and (17)-(18) 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 (Section III-A), the APF controller formulation enables the introduction of a look-ahead distance, i.e. it is not mandatory that P is the orthogonal projection of C on the path (Fig. 6). 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.

D. Nonlinear based controller

Two nonlinear feedback control laws have been developed by the authors to steer the kinematic model (6), along a desired path. The Nonlinear Control Based (NCB) path following controller, has its grounds on the work done in [9], [10] on path following control for kinematic models of wheeled mobile robots and in [17], [22] on the same topic but for autonomous underwater vehicles. In contrast to the work described in [9], 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 controllers developed in [17], [22] by deliberately controlling the rate of evolution of the vehicle's sideslip angle, β , 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 $\dot{\beta}$ can be embodied in the Lyapunov function candidate (see [10] and [17])

$$V_1 = \frac{1}{2} (s_1^2 + y_1^2) + \frac{1}{2\gamma} (\theta - \delta_\beta(y_1, v))^2, \quad (20)$$

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

$$\delta_\beta = -\theta_\beta \tanh(k_{\delta_\beta} y_1 v), \quad (21)$$

where $0 < \theta_\beta < \pi/2$ and k_{δ_β} is an arbitrary positive gain. Notice how equation (21) satisfies the first and second assumptions.

In the Lyapunov function V_1 adopted, the first term $\frac{1}{2} (s_1^2 + y_1^2)$ 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 aims to shape the approach angle $\beta = \theta - (\theta_m - \theta_c)$ of the velocity of the vehicle to the path, as a function of the 'lateral' distance y_1 and speed v , by forcing it to follow a desired orientation profile embedded in the function δ_β . The parameter γ accounts only for normalization purposes. Moreover, the first assumption provides an adequate reference sign definition in order 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. Assumption A.2 specifies that the desired approach angle vanishes as y_1 goes to zero, thus imposing the condition that the vehicle's velocity must be tangent to the path when the lateral distance y_1 is 0. Finally, the last assumption states that the vehicle does not tend to a state of rest, case where the controllability cannot be guaranteed.

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

$$\dot{s} = v \cos \theta + k_s s_1, \quad (22)$$

$$\dot{\beta} = \dot{\delta}_\beta + c_c(s) \dot{s} - \dot{\theta}_m - \gamma y_1 v \frac{\sin \theta - \sin \delta_\beta}{\theta - \delta_\beta} - k_\beta (\theta - \delta_\beta), \quad (23)$$

with k_β and k_s positive gains. This yields

$$\dot{V}_1 = -k_s s_1^2 + y_1 v \sin \delta_\beta - k_\beta \frac{(\theta - \delta_\beta)^2}{\gamma} \leq 0, \quad (24)$$

where the presence of the term $y_1 v \sin \delta_\beta$ in the previous equation justifies A.1. Moreover, (24) depicts the importance of the virtual target control law (22) on guaranteeing global convergence to the path.

Therefore, the kinematic control laws (22)-(23) 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 [9], [10] and [17].

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

$$V_2 = \frac{1}{2\gamma} (\theta_m - \theta_d - \delta_{\omega_m}(y_1, v))^2, \quad (25)$$

under the assumption (A.4) $\delta_{\omega_m}(0, v) = 0$ and with δ_{ω_m} defined in a similar manner as in (21).

The Lyapunov function V_2 adopted, captures the orientation error between the current orientation of the vehicle and the desired orientation at the path, which must be reduced to zero. As in the case of the control law for β , δ_{ω_m} aims to shape the approach angle $\theta_m - \theta_d$ of the vehicle orientation in relation to the desired orientation at the path, as a function of y_1 and v . The parameter γ accounts only for normalization purposes. Moreover, assumption A.4 guarantees that the vehicle's orientation must be equal to the desired orientation at the path when the lateral distance y_1 is 0, i.e. as $y_1 \rightarrow 0 \Rightarrow \theta_m \rightarrow \theta_d$.

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

$$\omega_m = \dot{\delta}_{\omega_m} + \dot{\theta}_d - k_{\omega_m} (\theta_m - \theta_d - \delta_{\omega_m}), \quad (26)$$

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

$$\dot{V}_2 = -k_{\omega_m} \frac{(\theta_m - \theta_d - \delta_{\omega_m})^2}{\gamma} \leq 0. \quad (27)$$

Hence, control law (26) 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 [9], [10] and [17].

To determine the values of v_F, v_R, θ_F and θ_R , the IKM (7) is used. The required value of ω_m is obtained directly from the control law (26), while β is obtained by integrating the output of the control law (23). Similarly to the AKM controller, v is provided by the trajectory.

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

IV. RESULTS

This section illustrates the simulated and experimental performance of the four path following controllers described in Section III. 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 [3], [4] and [5]. The vehicle simulated was the CPRHS, while following a trajectory to port 16 in level B1 of TB of ITER (Fig. 7-bottom left). The path includes both LG and FR: from the lift exit (A in Fig. 7), along the corridor (B in Fig. 7) and until the maneuver (C in Fig. 7) it is a LG path, while the entrance to port 16 is realized under FR (D in Fig. 7). A maneuver exists when the vehicle stops and changes its motion direction, in order to achieve a specified orientation, as illustrated in Fig. 7-bottom left. To deal with maneuvers at the path following stage, the path is divided $n + 1$ sub-paths, where n is the number of maneuvers. This enables each controller of Section III to deal solely with one sub-path at the time without discontinuities.

The path used in the simulations is a discrete path formed by a set of consecutive poses, as depicted in Fig. 7-bottom

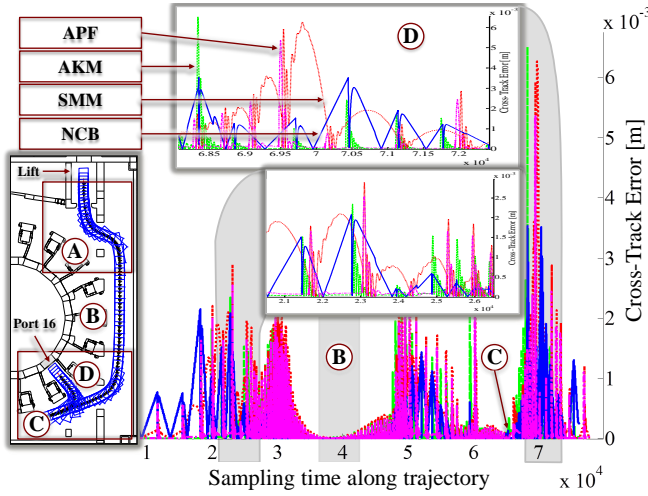


Fig. 7. Cross-track error of the four path following controllers in a trajectory to port 16 in level B1 of TB of ITER, focusing: A - lift exit in LG; and D - port 16 entrance in FR. B denotes the straight corridor and C the maneuver point.

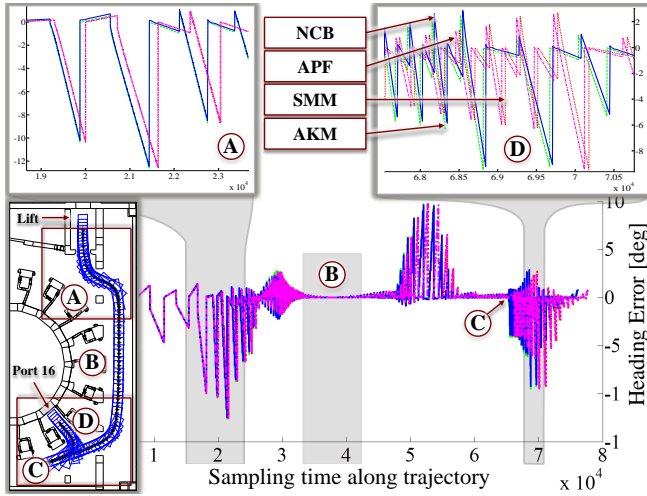


Fig. 8. Heading error of the four path following controllers in a trajectory to port 16 in level B1 of TB of ITER, focusing: A - lift exit in LG; and D - port 16 entrance in FR. B denotes the straight corridor and C the maneuver point.

left. This path was calculated using the motion planning realized in [3], [4] and [5] on path optimization for RLVs.

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 behaviour of wheel actuators.

In Fig. 7 is shown the cross-track error of the CPRHS while following the chosen trajectory to port 16, with the control cycle established at 50 Hz. Firstly, it must be noted that all the path following controllers perform within a 6 mm cross track error, which corresponds to 6% of the minimum safety margin of 10 cm established for this scenario. Also, the most critical areas, these being lift exit and port entrance, are the ones where more error occurs. In these areas, the peaks result from the discontinuities due to the linear path

interpolation performed. The cones and parabolas seen for the NCB and SMM controllers, respectively, denote the non-utilization of lookahead distance in a simulation which includes actuators dynamics. In fact, and attending the small scale of errors presented, it becomes impossible for these controllers to further minimize the cross-track error without including actuators dynamics in the formulation of the controllers. The performances of all four controllers are very satisfying, though the AKM and APF controllers achieve the smallest Root-Mean-Square Deviation, mainly due to the utilization of a lookahead distance. The same analysis can be extended to the heading error results depicted in Fig. 8, where, however, the differences between each controller are more subtle, presenting more akin performances.

In Fig. 9 is depicted the experimental setup used. In order to provide a common ground with the simulation results, the walls of half of the level B1 of TB of ITER were replicated at a 1:25 scale, with special focus to the surroundings of ports 16 and 17. The vehicle used was a 1:25 scale model of the CPRHS depicted in Fig. 1, built using LEGO® MINDSTORMS®. To determine, in real time, the pose of the vehicle in the scenario created, was used a webcam, tracking software and visual markers placed on top of the prototype and in the scenario. The software used was a branch of TES with communication protocols to acquire data and command the vehicle. Similarly to the simulation, the control cycle was set to 50 Hz. Notice that, the experimental results are more reliable due to including uncertainties not considered in the simulations.

The experimental results obtained are shown in Fig. 10, for the four path following controllers, in a trajectory to port 16 (Fig. 10-top left). Similarly to the simulation, both the cross-track error and the heading error register the highest values at port 16 entrance (C in Fig. 10), after the maneuver (B in Fig. 10). As expected, the errors are bigger than the simulated ones for all the four controllers, due to the errors inherent to the localization system used, communication lags between the software and the vehicle and actuators dynamics. Though, all the controllers perform within 25% of the 10 cm

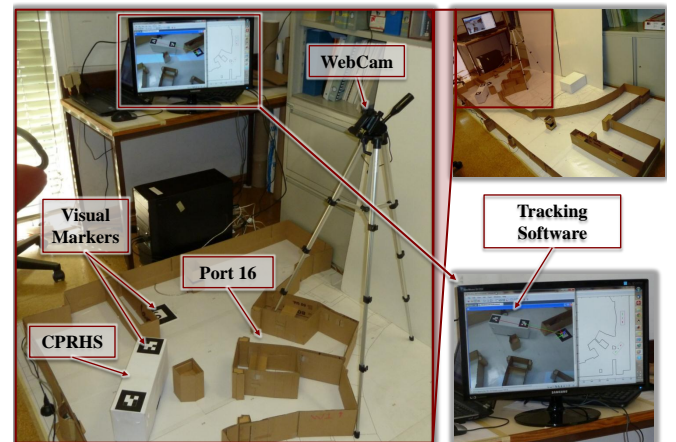


Fig. 9. Experimental setup used for port 16 in level B1 of TB of ITER.

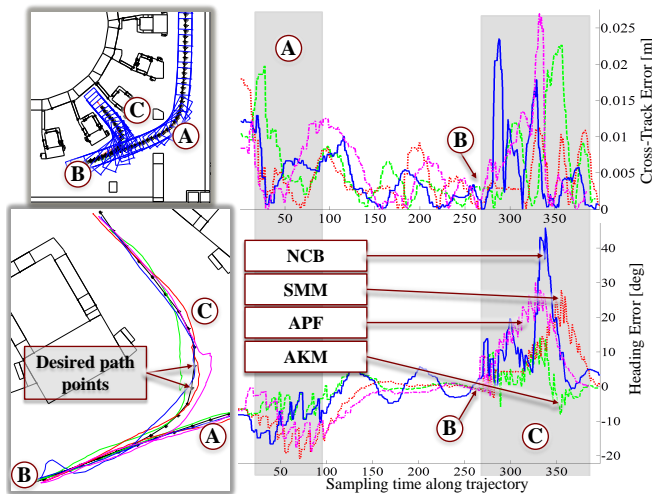


Fig. 10. Experimental cross-track error (top right) and heading error (bottom left) of the four path following controllers while approaching and entering port 16 in level B1 of TB of ITER. In the bottom left is shown the path described by each controller in comparison to the desired path.

safety margin established for the scenario.

A supporting video for the current paper can be found at [23], depicting the simulated and experimental performance of the APF controller.

V. CONCLUSIONS

Four path following controllers for RLVs were developed, implemented and tested, both in simulation and experimentally. All the controllers perform within the strict safety margin constraints established for the cluttered scenarios of ITER and, overall, attain good results. The study of four distinct controllers enables the conclusion that both the AKM and APF controllers slightly stand out from the other two, providing a good and smooth path following performance, due to the inclusion of a lookahead distance which contours the 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 amongst the four controllers possesses the biggest room for improvement, by enabling an extension to include vehicle and actuators dynamics through backstepping techniques.

Moreover, a possible future route would pass by merging the four path following controllers, with a supervisory controller deciding, on each moment, which one to employ given the situation at hand. Also, the inclusion of dynamic considerations in the controllers formulations must be addressed and experimental tests should be performed at a larger scale.

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

- [1] P. Muir and C. Neuman, "Kinematic modeling of wheeled mobile robots," *Journal of robotic systems*, vol. 4, no. 2, pp. 281–340, 1987.
- [2] M. Ribeiro and P. Lima, "Conceptual study on flexible guidance and navigation for ITER remote handling transport casks," in *Proceedings of the 17TH IEEE/NPSS Symposium on Fusion Engineering*, vol. 2, San Diego, Oct. 1997, pp. 969–972.
- [3] D. Fonte, F. Valente, A. Vale, and I. Ribeiro, "A motion planning methodology for rhombic-like vehicles for ITER remote handling operations," *Intelligent Autonomous Vehicles*, vol. 7, no. 1, pp. 443–448, 2010.
- [4] —, "Path optimization for rhombic-like vehicles: An approach based on rigid body dynamics," in *Proceedings of the 15th International Conference on Advanced Robotics (ICAR)*, Tallinn, June 2011, pp. 106–111.
- [5] F. Valente, A. Vale, D. Fonte, and I. Ribeiro, "Optimized trajectories of the transfer cask system in ITER," *Fusion Engineering and Design*, vol. 86, no. 9–11, pp. 1967–1970, Oct. 2011.
- [6] A. Vale, F. Valente, I. Ribeiro, J. Ferreira, and R. Ventura, "Motion planning and localization approaches for mobile robot navigation in ITER," in *Proceedings of Robótica 2012*, Guimarães, Apr. 2012, pp. 75–80.
- [7] A. Aguiar and J. Hespanha, "Performance limitations in reference tracking and path following for nonlinear systems," *Automatica*, vol. 44, no. 3, pp. 598–610, Mar. 2008.
- [8] A. D. Luca and G. Oriolo, "Feedback control of a nonholonomic car-like robot," in *Robot motion planning and control*, J.-P. Laumond, Ed. Springer, 1998, ch. 4th.
- [9] A. Micaelli and C. Samson, "Trajectory tracking for unicycle-type and two-steering-wheels mobile robots," INRIA, Tech. Rep., 1993.
- [10] L. Lapierre, D. Soetanto, and A. Pascoal, "Non-Singular Path-Following Control of a Unicycle in the Presence of Parametric Modeling Uncertainties," *International Journal of Robust and Nonlinear Control*, vol. 16, no. 10, pp. 485–503, 2006.
- [11] J. Snider, "Automatic steering methods for autonomous automobile path tracking," Robotics Institute, Tech. Rep. February, Feb. 2009.
- [12] J. Wit, C. D. Crane, and D. Armstrong, "Autonomous ground vehicle path tracking," *Journal of Robotic Systems*, vol. 21, no. 8, pp. 439–449, Aug. 2004.
- [13] T. Hiraoka, O. Nishihara, and H. Kumamoto, "Automatic path-tracking controller of a four-wheel steering vehicle," *Vehicle System Dynamics*, vol. 47, no. 10, pp. 1205–1227, Oct. 2009.
- [14] P. Morin and C. Samson, "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)*, no. 3, Orlando, Dec. 2011, pp. 7349–7355.
- [15] D. Wang, "Trajectory Planning for a Four-Wheel-Steering Vehicle," in *Proceedings of the IEEE International Conference On Robotics And Automation*, vol. I-IV, Seoul, May 2001, pp. 3320–3325.
- [16] D. Fonte, "Motion Planning for a Rhombic-Like Vehicle Operating in ITER Scenarios," Ph.D. dissertation, Instituto Superior Técnico, 2011.
- [17] L. Lapierre, D. Soetanto, and A. Pascoal, "Nonlinear path-following control of an AUV," *Ocean Engineering*, vol. 34, no. 11–12, pp. 1734–1744, Aug. 2007.
- [18] A. Kelly, "A Vector Algebra Formulation of Kinematics of Wheeled Mobile Robots," Robotics Institute, Tech. Rep., Aug. 2010.
- [19] S. Thrun, M. Montemerlo, and H. Dahlkamp, "Stanley: The robot that won the DARPA Grand Challenge," *Journal of Field Robotics*, vol. 23, no. 9, pp. 661–692, Sept. 2006.
- [20] Y. Kim and M. a. Minor, "Path Manifold-based Kinematic Control of Wheeled Mobile Robots Considering Physical Constraints," *The International Journal of Robotics Research*, vol. 26, no. 9, pp. 955–975, Sept. 2007.
- [21] M. De Cecco, E. Bertolazzi, G. Miori, R. Oboe, and L. Baglivo, "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)*, Angers, May 2007, pp. 11–18.
- [22] L. Lapierre and B. Jouvencel, "Robust nonlinear path-following control of an AUV," *IEEE Journal of Oceanic Engineering*, vol. 33, no. 2, pp. 89–102, Apr. 2008.
- [23] http://youtu.be/_Hpk8cMqX-s.