ITC 1/47
Journal of Information Technology and Control
Vol. 47 / No. 1 / 2018
pp. 118-130
DOI 10.5755/j01.itc.47.1.18021

Nonlinear Trajectory Tracking Control for Marine Vessels with Additive Uncertainties

Received 2017/07/05 | Accepted after revision 2018/01/08

http://dx.doi.org/10.5755/j01.itc.47.1.18021

Nonlinear Trajectory Tracking Control for Marine Vessels with Additive Uncertainties

Mario E. Serrano, Sebastian A. Godoy

Instituto de Ingeniera Qumica, Universidad Nacional de San Juan, Argentina, e-mails: serranoemanuel84@gmail.com; sgodoy@unsj.edu.ar

Daniel Gandolfo, Vicente A. Mut

Instituto de Automatica, Universidad Nacional de San Juan, Argentina, e-mails: dgandolfo@inaut.unsj.edu.ar; vmut@inaut.unsj.edu.ar

Gustavo J. E. Scaglia

Instituto de Ingeniera Qumica, Universidad Nacional de San Juan, Argentina, e-mail: gscaglia@unsj.edu.ar

Corresponding author: serranoemanuel84@gmail.com

The paper presents a nonlinear control law for a marine vessel to track a reference trajectory. In the wake of the results obtained in [19], an integrative approach is incorporated in the linear algebra methodology in order to reduce the effect of the uncertainty in the tracking error. This new approach does not increase the complexity of the design methodology. In addition, the zero convergence of tracking error under polynomial uncertainties is demonstrated. Simulation results under environmental disturbance and model mismatches are presented and discussed.

KEYWORDS: Trajectory tracking, nonlinear control, marine vessels, polynomial uncertainties.

Introduction

The past few decades have witnessed an increased research effort in the autonomous vehicles motion control area. A typical motion control problem is trajectory tracking [21, 16, 18], which is concerned with the design of control laws that force a vehicle to reach and follow a time parameterized reference (i.e., a geometric path with an associated timing law).

One challenge for trajectory tracking of a surface marine vessel stems from the fact that the system is often underactuated [3, 5, 24]. Conventional ships are usually

equipped with one or two main propellers for forward speed control, and rudders for ship course keeping. For ship maneuvering problems, such as path following and trajectory tracking, where we seek control for all three degrees of freedom (surge, sway, and yaw), the two controls cannot influence all three variables independently, thereby leading to underactuated control problems. In conventional way-point guidance systems, the output space is reduced such that the number of outputs equals the number of control inputs, and a fully actuated control problem can be formulated [8].

Another challenge in trajectory tracking problems of marine surface vessels is the inherent nonlinearity, from either the ship dynamics or path following kinematics. Many different nonlinear design methodologies have been attempted. Generally, backstepping techniques are the most chosen to solve tracking problems in marine vessels. In the work of Aguiar et al. [1], the authors proposed a solution for underactuated autonomous vehicles in the presence of possibly large modeling parametric uncertainty. They designed an adaptive supervisory control algorithm that combines logic-based switching with iterative Lyapunov-based techniques such as integrator backstepping. In [12], the authors address the trajectory tracking problem contemplating modeling errors in its implementation, an approach based in measurement techniques is proposed in that work. Two constructive backstepping design schemes were developed in [25] to solve stabilisation and tracking problems.

Alternative techniques such as adaptive dynamical sliding mode control and Quantitative Feedback Theory (QFT) have been reported in the literature to address the problem of trajectory tracking in the presence of uncertainties. In [22], the combination of backstepping and adaptive dynamical sliding mode control is proposed for dealing with the planar trajectory tracking control problem in marine vessels. Nicolau et al. [17] described the design steps of robust QFT autopilot for the course-keeping and course-changing control of a ship, in the presence of disturbances. In Lekkas and Fossen [11], two adaptive nonlinear observers are designed in order to estimate the ocean current components. The guidance algorithm uses this information and generates appropriate relative surge speed reference trajectories for minimizing the tracking error. A similar procedure is proposed in Yang et al. [23], where an observer is developed to provide an estimation of unknown disturbances. Then, this approach is applied to design a novel trajectory tracking robust controller through a vectorial backstepping technique.

In Serrano et al. [19], a linear algebra-based controller was proposed for tracking control in a ship. The originality of this control strategy is based on the linear algebra theory application to find the controller expression. The control law is obtained by solving a system of linear equations. It is easier to implement the developed algorithm in a real system because the use of discrete equations allows direct adaptation to any computer system or programmable device running sequential instructions to a programmable clock speed. However, this methodology does not consider additive uncertainties in the design stage. Thus, if the system is under modeling errors or disturbances in control actions, significant tracking errors can appear.

This work provides a positive answer to the challenging problem of designing controllers for trajectory tracking in multivariable nonlinear systems under additive uncertainties. The goal of this paper is to design a robust autopilot for ship trajectory tracking control, in the presence of disturbances. Thus, the main contribution of this work is to extend the techniques based on linear algebra presented in Serrano et al. [19], to systems under additive uncertainties. First, in order to deal with these uncertainties, some integrators have been added to the original controller. This achievement allows the tracking errors not to be affected by polynomial uncertainties. Next, the conditions for a system of linear equations have unique solution were analyzed. Finally, the control action were obtained by solving the linear system, even though the original system model is nonlinear.

Additionally, our controller shows to be robust under disturbances in the control actions. Due to its mathematical formulation, our approach can also be implemented as embedded (it does not compute higher order derivatives, exponentiation or complex trigonometric functions). Another contribution of this paper is the application of Monte Carlo (MC) based sampling experiment in the simulations. The controller parameters can be computed to minimize a cost index; here these are determined by using the Monte Carlo (MC) experiment. The theoretical results are validated and compared with the original methodology by simulations.

It is noteworthy that due to the above mentioned characteristics, the computing power required to perform the mathematical operations is low. Thence it is possible to implement the algorithm in any controller with low computing capacity. Furthermore, the developed algorithm is easier to be implemented in a real system because the use of discrete equations allows direct adaptation to any computer system or programmable device running sequential instructions at a programmable clock speed. Thus, one great advantage of this approach is the use of discrete-time equations, hence simplifying its implementation on a computer system. The proof of the zero-convergence of the tracking error under uncertainty is another main contribution of this work.

The paper is organized as follows. Section 2 describes the dynamic model for marine vessel and the linear algebra-based methodology proposed in Serano et al. [19]. The controller design methodology considering polynomial uncertainties is shown in Section 3. In Section 4, the controller parameters are tuned by the MC experiment and theoretical results are validated with simulation results of the control algorithm, followed by the conclusions in Section 5.

1. The Marine Vessel Model and Controller Design

1.1. The Description of the Surface Vessel Model

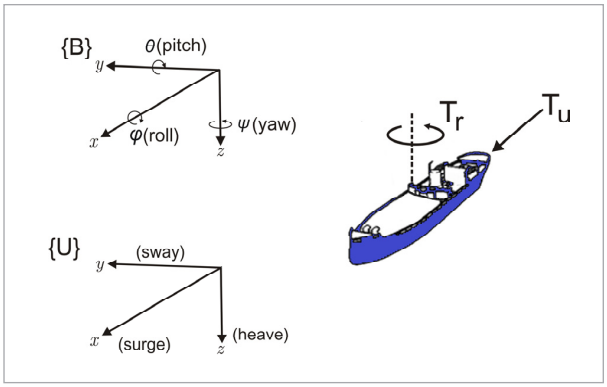
Consider an autonomous marine surface vessel whose kinematic and dynamic models are described as follows [19, 9]:

$$\begin{aligned}
\dot{x} &= u\cos(\psi) - v\sin(\psi)\\
\dot{y} &= u\sin(\psi) + v\cos(\psi)\\
\dot{\psi} &= r\\
\dot{u} &= \frac{m_{22}}{m_{11}}vr - \frac{d_{11}}{m_{11}}u + \frac{1}{m_{11}}T_u\\
\dot{v} &= -\frac{m_{11}}{m_{22}}ur - \frac{d_{22}}{m_{22}}v\\
\dot{r} &= \frac{m_{11}-m_{22}}{m_{33}}vu - \frac{d_{33}}{m_{33}}u + \frac{1}{m_{33}}T_r,
\end{aligned} \tag{1}$$

where $(x, y) \in R^2$ is the position of the ship given in an inertial frame and $\psi \in [0, 2\varpi)$ is the heading angle of the ship relative to the geographic North. The general kinematic equations of motion of the vehicle in the horizontal plane can be developed by using a global coordinate frame U and a body-fixed coordinate frame B, as depicted in Fig. 1. Here, u is the forward velocity (surge), v is the transverse velocity (sway) and r is the angular velocity in yaw. The parameters $m_{ii} > 0$ are given by the ship inertia and added mass effects. The parameters $d_{ii} > 0$ are given by the hydrodynamic damping. The available control inputs are the surge control force T_u and the yaw control moment T_r [19].

Figure 1

A Marine Vessel: a global coordinate frame U and a body-fixed coordinate frame B



1.2. A Linear Algebra Controller Design

The methodology proposed in Serrano et al. [19] aims to find the control actions T_u and T_r so that the marine vessel reaches and follows a reference trajectory (x_{ref}, y_{ref}). Next, the control design prcedure presented in [19] is described.

First, the model (1) is aproximated through the Euleriran approximation and rearranged in the matrix form:

$$\begin{bmatrix} 0 & 0\\ 0 & 0\\ 0 & 0\\ 0 & 0\\ \frac{1}{m_{11}} & 0\\ 0 & \frac{1}{m_{33}} \end{bmatrix} \begin{bmatrix} T_{u,n}\\ T_{r,n} \end{bmatrix} = \begin{bmatrix} \left(\frac{x_{n+1}-x_{n)}}{T_s}\right) - \left(u_n\cos(\psi_n) - v_n\sin(\psi_n)\right)\\ \left(\frac{y_{n+1}-y_n}{T_s}\right) - \left(u_n\sin(\psi_n) + v_n\cos(\psi_n)\right)\\ \left(\frac{\psi_{n+1}-\psi_n}{T_s}\right) - r_n\\ \left(\frac{v_{n+1}-v_n}{T_s}\right) - \left(-\frac{m_{11}}{m_{22}}u_n r_n - \frac{d_{22}}{m_{22}}v_n\right)\\ \left(\frac{u_{n+1}-u_n}{T_s}\right) - \left(\frac{m_{22}}{m_{11}}v_n r_n - \frac{d_{11}}{m_{11}}u_n\right)\\ \left(\frac{r_{n+1}-r_n}{T_s}\right) - \left(\frac{m_{11}-m_{22}}{m_{33}}v_n u_n - \frac{d_{33}}{m_{33}}r_n\right) \end{bmatrix}. \tag{2}$$

Second, consider the immediately reachable value of each state vector, as proposed here:

$$z_{n+1}=\begin{bmatrix}x_{n+1}\\ y_{n+1}\\ \psi_{n+1}\\ u_{n+1}\\ r_{n+1}\end{bmatrix}=\underbrace{\begin{bmatrix}x_{ref,n+1}\\ y_{ref,n+1}\\ \psi_{ref,n+1}\\ u_{ref,n+1}\\ r_{ref,n+1}\end{bmatrix}}_{z_{ref,n+1}}-k\underbrace{\begin{bmatrix}x_{ref,n}-x_n\\ y_{ref,n}-y_n\\ \psi_{ref,n}-\psi_n\\ u_{ref,n}-u_n\\ r_{ref,n}-r_n\end{bmatrix}}_{e_n=z_{ref,n}-z_n},\quad e_n=\begin{bmatrix}e_{x,n}\\ e_{y,n}\\ e_{\psi,n}\\ e_{u,n}\\ e_{r,n}\end{bmatrix}=\begin{bmatrix}x_{ref,n}-x_n\\ y_{ref,n}-y_n\\ \psi_{ref,n}-\psi_n\\ u_{ref,n}-u_n\\ r_{ref,n}-r_n\end{bmatrix}. \tag{3}$$

In (3), k represents a design parameter ($0 < k < 1$) and $\mathbf{e}_n$ denotes the tracking error. Note that:

- if k =0, $(z_{n+1}=z_{ref,n+1})$, the reference trajectory is reached in one step;
- if k =1, the error will remain constant, $(z_{n+1}-z_n=z_{ref,n+1}-z_{ref,n})$.

Considering Eq. (3), system (2) can be written in a more compact form as (4):

$$\begin{bmatrix}0 & 0\\ 0 & 0\\ 0 & 0\\ 0 & 0\\ \frac{1}{m_{11}} & 0\\ 0 & \frac{1}{m_{33}}\end{bmatrix}\begin{bmatrix}T_{u,n}\\ T_{r,n}\end{bmatrix}=\begin{bmatrix}\left(\frac{x_{n+1}-x_{n)}}{T_s}\right)-\left(u_n\cos(\psi_n)-v_n\sin(\psi_n)\right)\\ \left(\frac{y_{n+1}-y_n}{T_s}\right)-\left(u_n\sin(\psi_n)+v_n\cos(\psi_n)\right)\\ \left(\frac{\psi_{n+1}-\psi_n}{T_s}\right)-r_n\\ \left(\frac{v_{n+1}-v_n}{T_s}\right)-\left(-\frac{m_{11}}{m_{22}}u_nr_n-\frac{d_{22}}{m_{22}}v_n\right)\\ \left(\frac{u_{n+1}-u_n}{T_s}\right)-\left(\frac{m_{22}}{m_{11}}v_nr_n-\frac{d_{11}}{m_{11}}u_n\right)\\ \left(\frac{r_{n+1}-r_n}{T_s}\right)-\left(\frac{m_{11}-m_{22}}{m_{33}}v_nu_n-\frac{d_{33}}{m_{33}}r_n\right)\end{bmatrix}. \tag{4}$$

with,

$$\begin{aligned}\Delta_{x,ref}&=x_{ref,n+1}-k(x_{ref,n}-x_n)-x_n\\ \Delta_{y,ref}&=y_{ref,n+1}-k(y_{ref,n}-y_n)-y_n\\ \Delta_{\psi,ref}&=\psi_{ref,n+1}-k(\psi_{ez,n}-\psi_n)-\psi_n\\ \Delta_{v,ref}&=v_{ref,n+1}-k(v_{ez,n}-v_n)-v_n\\ \Delta_{u,ref}&=u_{ref,n+1}-k(u_{ref,n}-u_n)-u_n\\ \Delta_{r,ref}&=r_{ref,n+1}-k(r_{ref,n}-r_n)-r_n.\end{aligned}$$

At every sample time, the linear system of Eq. (4) is used to calculate the control action that ensures the tracking of the reference trajectories. To calculate $T_{u,n}$ and $T_{r,n}$, the system of equations (4) must have an exact solution. Thus, the values of the variables (u, ψ, r) must be determined in order that tracking errors tends to zero (see [19]).

The unknown variables (u_{ref}, ψ_{ref}, r_{ref}) of this system will be called sacrificed variables. The values adopted by such variables force the equation system (4) to have an exact solution. Then, in (4), the variables (u_{ref}, ψ_{ref}, r_{ref}) would be replaced by (u_{ez}, ψ_{ez}, r_{ez}), where (u_{ez}, ψ_{ez}, r_{ez}) represent the values so that (4) has an exact solution. Now, it is necessary to specify the conditions under which the system (4) has an exact solution.

System (4) is of the type **Au** = **b**. Thus, in order for (4) to have an exact solution, the column vector **b** must be a linear combination of the columns of **A**.

Next, the first two rows of system (4) are rewritten in the form:

$$\underbrace{\begin{bmatrix}\cos(\psi_n)\\ \sin(\psi_n)\end{bmatrix}}_{A}\underbrace{u_n}_{u}=\underbrace{\begin{bmatrix}\left(\frac{\Delta_{x,ref}}{T_s}\right)+v_n\sin(\psi_n)\\ \left(\frac{\Delta_{y,ref}}{T_s}\right)-v_n\cos(\psi_n)\end{bmatrix}}_{b}. \tag{5}$$

For (5) to have an exact solution, Eq. (6) must be fulfilled:

$$\tan(\psi_{ez,n})=\frac{\sin(\psi_{ez,n})}{\cos(\psi_{ez,n})}=\frac{\frac{\Delta_{y,ref}}{T_s}-v_n\cos(\psi_n)}{\frac{\Delta_{x,ref}}{T_s}+v_n\sin(\psi_n)} \tag{6}$$

and

$$\begin{aligned}u_{ez(n)}=&\left(\frac{\Delta_{y,ref}}{T_s}-v_n\cos(\psi_n)\right)\sin(\psi_{ez,n})+\\ &+\left(\frac{\Delta_{x,ref}}{T_s}+v_n\sin(\psi_n)\right)\cos(\psi_{ez,n}).\end{aligned} \tag{7}$$

Remark 1. Equation (6) establishes the conditions that ensure an exact solution for the model of Eq. (4), where $\psi_{ez,n+1}$ is the unknown variable. In Eq. (6), $\psi_{ez,n}$ is obtained at the current sampling time, while $\psi_{ez,n+1}$ can be calculated through Taylor approximations of zero, first, or second order, i.e.:

1 Zero-order

$$\psi_{ez,n+1} = \psi_{ez,n}.$$

2 First-order

$$\psi_{ez,n+1} = \psi_{ez,n} + \frac{d\psi_{ez}}{dt} T_s \approx \psi_{ez,n} + \frac{\left(\psi_{ez,n} - \psi_{ez,n-1}\right)}{T_s} T_s$$

$$\psi_{ez,n+1} = 2\psi_{ez,n} - \psi_{ez,n-1}.$$

3 Second-order

$$\psi_{ez,n+1} = \psi_{ez,n} + \left(\psi_{ez,n} - \psi_{ez,n-1}\right) + \frac{\left(\psi_{ez,n} - 2\psi_{ez,n-1} - \psi_{ez,n-2}\right)}{2}.$$

Next, the yaw velocity is analysed. Considering the third row of (4) and (6), we define:

$$r_{ez,n} = \frac{\psi_{ez,n+1} - k\left(\psi_{ez,n} - \psi_n\right) - \psi_n}{T_s}. \tag{8}$$

Finally, in Eq. (4) and considering Remark 1, the variables ($\psi_{ref,n+1}$, $u_{ref,n+1}$, $r_{ref,n+1}$) are replaced by ($\psi_{ez,n+1}$, $u_{ez,n+1}$, $r_{ez,n+1}$); and ($\psi_{ref,n}$, $u_{ref,n}$, $r_{ref,n}$) are replaced by ($\psi_{ez,n}$, $u_{ez,n}$, $r_{ez,n}$), thus leading to the following system of equations:

$$\begin{bmatrix} \frac{1}{m_{11}} & 0 \\ 0 & \frac{1}{m_{33}} \end{bmatrix} \begin{bmatrix} T_{u,n} \\ T_{r,n} \end{bmatrix} = \begin{bmatrix} \frac{\Delta_{u,ez}}{T_s} - \frac{m_{22}}{m_{11}} v_n r_n + \frac{d_{11}}{m_{11}} u_n \\ \frac{\Delta_{r,ez}}{T_s} - \frac{m_{11} - m_{22}}{m_{33}} v_n u_n + \frac{d_{33}}{m_{33}} r_n \end{bmatrix}, \tag{9}$$

where,

$$\Delta_{u,ez} = u_{ez,n+1} - k\left(u_{ez,n} - u_n\right) - u_n$$

$$\Delta_{r,ez} = r_{ez,n+1} - k\left(r_{ez,n} - r_n\right) - r_n.$$

Equation (9) allows to perform the calculation of the control action, T_u and T_r, which makes the tracking errors tend to zero in every sampling time. The control action is obtained by least squares (see [19]), i.e.:

$$\begin{bmatrix} T_{u,n} \\ T_{r,n} \end{bmatrix} = \left(A^T A\right)^{-1} A^T b$$

$$\begin{bmatrix} T_{u,n} \\ T_{r,n} \end{bmatrix} = \begin{bmatrix} m_{11}\left(\frac{\Delta_{u,ez}}{T_s} - \frac{m_{22}}{m_{11}} v_n r_n + \frac{d_{11}}{m_{11}} u_n\right) \\ m_{33}\left(\frac{\Delta_{r,ez}}{T_s} - \frac{m_{11} - m_{22}}{m_{33}} v_n u_n + \frac{d_{33}}{m_{33}} r_n\right) \end{bmatrix}. \tag{10}$$

Theorem 1. If the system behavior is ruled by (2) and the controller is designed considering Eqs. (6) to (9), the tracking error tends to zero ($e_n \to 0$) when $n \to \infty$.

Remark 2. Consider the geometric progression below:

$$a_{(1)} = ka_{(0)}$$
$$a_{(2)} = ka_{(1)} = k^2 a_{(0)}$$
$$\vdots \qquad .$$
$$a_{(n+1)} = ka_{(n)} = k^n a_{(0)}$$

If $0 < k < 1$ and $n \to \infty$ (with $n \in N$), then $a_{(n)} \to 0$.

The proof of Theorem 1 and the convergence to zero of the tracking errors is shown in detail in [19]. As is shown in the above study, if the control action (10) is replaced in the system model, then (11) is obtained (see Eq. (A.30) in [19]):

$$e_{n+1} = \begin{bmatrix} e_{x,n+1} \\ e_{y,n+1} \\ e_{\psi,n+1} \\ e_{u,n+1} \\ e_{r,n+1} \end{bmatrix} = k\, e_n + \underbrace{T_s \begin{bmatrix} h_{1,n} \\ h_{2,n} \\ h_{3,n} \\ 0 \\ 0 \end{bmatrix}}_{\text{Bounded that tends to zero}} \text{nonlinearity} \tag{11}$$

with,

$$h_{1,n} = -u_{ez,n} \sin\left(\psi_{ez,n} - \zeta e_{\psi,n}\right) e_{\psi,n} + \cos(\psi_n) e_{u,n}$$

$$h_{2,n} = -u_{ez,n} \cos\left(\psi_{ez,n} - \theta e_{\psi,n}\right) e_{\psi,n} + \sin\left(\psi_n\right) e_{u,n}$$

$$h_{3,n} = e_{r,n}$$

$$0 < \zeta < 1$$

$$0 < \theta < 1.$$

Equation (11) represents the sum of a linear system with a nonlinearity. Acording to [19], this nonlinearity tends to zero. Thence, if $0 < k < 1$ the tracking error tends to zero ($e_{x,n} \to 0$ and $e_{y,n} \to 0$ when $n \to \infty$(with $n \in N$)), because $e_{\psi,n} \to 0$, $e_{u,n} \to 0$ and $e_{u,n} \to 0$ (see Remark 2). Finally, it is thus demonstrated that $\mathbf{e}_{n+1} \to 0$ when $n \to \infty$ (with $n \in N$) [19].

2.3. A Summary for Controller Implementation

The next steps can be used to implement the control algorithm,

- Given ($x_{ref,n+i}$,$y_{ref,n+i}$) for i = 1, 2, and some initial conditions z_n,
- Compute ($\psi_{ez,n}$) using (6) and calculate $\psi_{ez,n+1}$ using the Taylor approximation (see Remark 1).

- Compute $(u_{ez,n}, r_{ez,n})$ using (7) and (8), respectively; calculate $(u_{ez,n+1}, r_{ez,n+1})$ using the Taylor approximation (see Remark 1).
- Compute the control actions $T_{u,n}$ and $T_{u,n}$ according to (10).

3. A Controller Design Underuncertainty

Now, an additive uncertainty is incorporated into the model of the system, and an approach to eliminate its influence on the tracking error is proposed. Considering (2), the following marine vessel model is assumed:

$$\begin{bmatrix} x_{n+1} \\ y_{n+1} \\ \psi_{n+1} \\ v_{n+1} \\ u_{n+1} \\ r_{n+1} \end{bmatrix} = \begin{bmatrix} x_n \\ y_n \\ \psi_n \\ v_n \\ u_n \\ r_n \end{bmatrix} + T_s \begin{bmatrix} u_n \cos(\psi_n) - v_n \sin(\psi_n) \\ u_n \sin(\psi_n) + v_n \cos(\psi_n) \\ r_n \\ -\frac{m_{11}}{m_{22}} u_n r_n - \frac{d_{22}}{m_{22}} v_n \\ \frac{m_{22}}{m_{11}} v_n r_n - \frac{d_{11}}{m_{11}} u_n \\ \frac{m_{11} - m_{22}}{m_{33}} v_n u_n - \frac{d_{33}}{m_{33}} r_n \end{bmatrix} + \tag{12}$$

$$+T_s \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ \frac{1}{m_{11}} & 0 \\ 0 & \frac{1}{m_{33}} \end{bmatrix} \begin{bmatrix} T_{u,n} \\ T_{r,n} \end{bmatrix} + E_n; \quad E_n = \begin{bmatrix} E_{x,n} \\ E_{y,n} \\ E_{\psi,n} \\ E_{v,n} \\ E_{u,n} \\ E_{r,n} \end{bmatrix},$$

where $\mathbf{E}_n$ is the additive uncertainty. Notice that the additive uncertainty can be used to model perturbed systems as well as a wide class of model mismatches.

Taking into account that the mismatch might depend on the state and on the input of the system, consider a real plant $z_{n+1} = f(z_n, u_n)$. The additive uncertainty can be expressed by $E_n = f(z_n, u_n) - \hat{f}(z_n, u_n)$, where $\hat{f}(z_n, u_n)$ is the discrete-time nonlinear model of the system. Note that if, as it will be assumed, $\mathbf{z}$ and $\mathbf{u}$ are bounded and f is a Lipschitz continuous function, then E_n can be modeled as a bounded uncertainty [15, 13].

Now, the procedure for the controller design developed in the above section is applied to the model under uncertainty in order to analyze the effect of the uncertainty in the system. Replacing $T_{u,n}$ and $T_{r,n}$ from (10) in (12), after some simple operations, it yields

$$e_{n+1} = \begin{bmatrix} e_{x,n+1} \\ e_{y,n+1} \\ e_{\psi,n+1} \\ e_{u,n+1} \\ e_{r,n+1} \end{bmatrix} = k\, e_n + T_s \begin{bmatrix} h_{1,n} \\ h_{2,n} \\ h_{3,n} \\ 0 \\ 0 \end{bmatrix} - \begin{bmatrix} E_{x,n} \\ E_{y,n} \\ E_{\psi,n} \\ E_{u,n} \\ E_{r,n} \end{bmatrix}. \tag{13}$$

Now, looking at (13), a direct effect of the additive uncertainty on the tracking error can be seen.

3.1. The Integral Action

In order to reduce the effect of $\mathbf{E}_n$, some integrators of the tracking errors in the system state variables will be introduced, depending on the time variation hypothesis of $\mathbf{E}_n$. It is assumed that $\mathbf{E}_n$ is unknown and each component is an m-order polynomial.

Remark 3. The first order difference of E_n is defined as $\delta E_n = E_{n+1} - E_n$, the second order difference as $\delta^2 E_n = \delta(\delta E_n) = \delta(E_{n+1} - E_n) = E_{n+2} - 2E_{n+1} + E_n$, and as a rule, the q-th order difference is defined as $\delta^q E_n = \delta(\delta^{q-1} E_n)$.

Remark 4. The q-th difference of a $q-1$ order polynomial is zero.

Let us consider a constant uncertainty $\mathbf{E}_n = const$. That means $\boldsymbol{\delta}\mathbf{E}_n = \mathbf{E}_{n+1} - \mathbf{E}_n = \mathbf{0}$. In this case, an integrator for each state variable will force the error to converge to zero. Denoting by $\mathbf{e}(t)$ the continuous time error in the state vector, define

$$U_{1,n+1} = U_{1,n} + \int_{nT_s}^{(n+1)T_s} e(t)dt \cong U_{1,n} + T_s e_n \tag{14}$$

as the integral of the error. The control action (10) will be computed assuming a new term in (3), such as $\mathbf{z}$

$$z_{n+1} = z_{ref,n+1} - k\underbrace{\left(z_{ref,n} - z_n\right)}_{e_n} + K_1 U_{1,n+1}, \tag{15}$$

where k and K_1 are, respectively, the proportional and integral constants of the control actions.

After incorporating the integral term in the design methodology, the controller design algorithm given in Section 2.3 is applied. Then, the same procedure is carried out to obtain (6), (7) and (8) is carried out to compute the new heading angle, forward velocity and angular velocity.

The heading angle will be computed as

$$\tan(\psi_{ez,n}) = \frac{\frac{\Delta_{y,ref} + K_{1,y}U_{1,y,n+1}}{T_s} - v_n \cos(\psi_n)}{\frac{\Delta_{x,ref} + K_{1,x}U_{1,x,n+1}}{T_s} + v_n \sin(\psi_n)}. \quad (16)$$

The reference forward velocity is

$$u_{ez(n)} = \left(\frac{\Delta_{y,ref} + K_{1,y}U_{1,y,n+1}}{T_s} - v_n \cos(\psi_n)\right)\sin(\psi_{ez,n}) + \left(\frac{\Delta_{x,ref} + K_{1,x}U_{1,x,n+1}}{T_s} + v_n \sin(\psi_n)\right)\cos(\psi_{ez,n}). \quad (17)$$

The angular velocity that makes the tracking errors tend to zero must be

$$r_{ez,n} = \frac{\Delta_{\psi,ez} + K_{1,\psi}U_{1,\psi,n+1}}{T_s}. \quad (18)$$

Finally, T_u and T_r are obtained using least squares.

$$\begin{bmatrix} \frac{1}{m_{11}} & 0 \\ 0 & \frac{1}{m_{33}} \end{bmatrix} \begin{bmatrix} T_{u,n} \\ T_{r,n} \end{bmatrix} = \begin{bmatrix} \frac{\Delta_{u,ez} + K_{1,u}U_{1,u,n+1}}{T_s} - \frac{m_{22}}{m_{11}} v_n r_n + \frac{d_{11}}{m_{11}} u_n \\ \frac{\Delta_{r,ez} + K_{1,r}U_{1,r,n+1}}{T_s} - \frac{m_{11} - m_{22}}{m_{33}} v_n u_n + \frac{d_{33}}{m_{33}} r_n \end{bmatrix}. \quad (19)$$

Replacing the control actions $(T_{u,n}, T_{r,n})$ of (19) in (12), and after some simple operations, it yields

$$e_{n+2} + \left(-k + K_1 T_s - 1\right)e_{n+1} + k e_n = \underbrace{T_s \begin{bmatrix} h_{1,n+1} - h_{1,n} \\ h_{2,n+1} - h_{2,n} \\ h_{3,n+1} - h_{3,n} \\ 0 \\ 0 \end{bmatrix}}_{\text{Bounded nonlinearity that tends to zero}} - \underbrace{(E_{n+1} - E_n)}_{\delta E_n = 0}. \quad (20)$$

Therefore, k and K_1 are chosen in order to ensure the stability of the linear system represented in the left-hand side of (20); that is, the zeros of this polynomial (r_i) q should be inside the unit circle. Then $\sqrt{e_{x,n}^2 + e_{y,n}^2} \to 0$, as $n \to \infty$.That is, the tracking error tends to zero despite of uncertainties, if they are constant.

3.2. Two Integral Actions

Let us now consider that the uncertainty can be modeled by a function where the second order difference is zero, such that $\delta^2\mathbf{E}_n = \delta(\delta\mathbf{E}_n) = \delta(\mathbf{E}_{n+1} - \mathbf{E}_n) = \mathbf{E}_{n+2} - 2\mathbf{E}_{n+1} + \mathbf{E}_n = \mathbf{0}$. Then, a double integrator should be introduced in a similar way to (14), defining the integrating variables $\mathbf{U}_1$, $\mathbf{U}_2$

$$U_{2,n+1} = U_{2,n} + \int_{nT_s}^{(n+1)T_s} U_1(t)dt \cong U_{2,n} + T_s U_{1,n+1}. \quad (21)$$

In this case, the control action (10) will be computed assuming an additional term in (3), such as $\mathbf{z}$

$$z_{n+1} = z_{ref,n+1} - k\underbrace{\left(z_{ref,n} - z_n\right)}_{e_n} + K_1 U_{1,n+1} + K_2 U_{2,n+1}, \quad (22)$$

where k, K_1 and K_2 are constants and represent the proportional, integral and double integral control parameters. Operating as before, and taking into account that $\delta^2\mathbf{E}_n = \mathbf{0}$, the error dynamics can be expressed by

$$e_{n+3} + \left(-k + T_s\left(K_1 + T_s K_2\right) - 2\right)e_{n+2} + \left(2k - T_s K_1 + 1\right)e_{n+1} - k e_n = \underbrace{T_s \begin{bmatrix} h_{1,n+2} - 2h_{1,n+1} + h_{1,n} \\ h_{2,n+2} - 2h_{2,n+1} + h_{2,n} \\ h_{3,n+2} - 2h_{3,n+1} + h_{3,n} \\ 0 \\ 0 \end{bmatrix}}_{\text{Bounded nonlinearity that tends to zero}} - \underbrace{\delta^2 E_n}_{=0}. \quad (23)$$

Now, as can be seen in (23), under the assumption of constant or linear varying uncertainty, $\delta^2\mathbf{E}_n = \mathbf{0}$, and the uncertainty has no influence on the error dynamics. The controller parameters k, K_1 and K_2 are constant chosen in order to ensure the stability of the linear system represented in the left-hand side of (23),

as shown in the previous case. Following a similar reasoning, if the uncertainties can be approximated with a $q-1$ order polynomial, the influence of $\mathbf{E}_n$ on $\mathbf{e}_n$ will be eliminated by introducing q integrators.

Remark 5. The controller parameters can be chosen differently for each state variable, as pointed out in (3) for k, but not relevant benefit is obtained in the examples developed later on. The same could be done with all the controller parameters k, K_1 and K_2.

3.3. Multiple Integral Actions

The previous sections added new integrator terms in the design methodology presented in [19]. This approach can be extended to a number p of integrators. Assume that the uncertainty can be modeled by a function where the difference of order p is zero, such that $\delta^p E_n = \delta\left(\delta^{p-1} E_n\right) = 0$. Then, a *p–integrator* should be introduced in a similar way to (21), defining the integrating variables $\mathbf{U}_1$, $\mathbf{U}_2$,..., $\mathbf{U}_p$:

$$U_{p,n+1} = U_{p,n} + \int_{nT_s}^{(n+1)T_s} U_{p-1}(t)dt \cong U_{p,n} + T_s U_{p-1,n+1}.$$

In this case, the control action (10) will be computed assuming p additional terms, such as

$$z_{d,n+1} = z_{ref,n+1} - k\underbrace{\left(z_{ref,n} - z_n\right)}_{e_n} + \sum_{i=1}^{p} K_i U_{i,n+1}, \qquad (24)$$

where k, K_1, K_2,..., K_p are, respectively, the proportional and integrals control actions. Operating as before, and taking into account that $\delta^p \mathbf{E}_n = \mathbf{0}$, the error dynamics can be expressed by $p-1$

$$e_{n+p+1} + \sum_{i=1}^{p-1} f_i\left(k, K_1, ..., K_p, T_s\right) e_{n+2} - k e_n = \\ = T_s \underbrace{\begin{bmatrix} \sum_{j=0}^{p}\sum_{i=0}^{j} \binom{j}{i} (-1)^i h_{1,n-j-i-1} \\ \sum_{j=0}^{p}\sum_{i=0}^{j} \binom{j}{i} (-1)^i h_{2,n-j-i-1} \\ \sum_{j=0}^{p}\sum_{i=0}^{j} \binom{j}{i} (-1)^i h_{3,n-j-i-1} \\ 0 \\ 0 \end{bmatrix}}_{\text{Bounded nonlinearity that tends to zero}} - \underbrace{\delta^p E_n}_{=0}. \qquad (25)$$

Thence, under the assumption of constant or linear varying uncertainty, $\delta^p \mathbf{E}_n = \mathbf{0}$, the uncertainty has no influence on the error dynamics. The controller parameters k, K_1, K_2,..., K_p are chosen in order to ensure the stability of the linear system represented in the left-hand side of (25), as shown in the above cases.

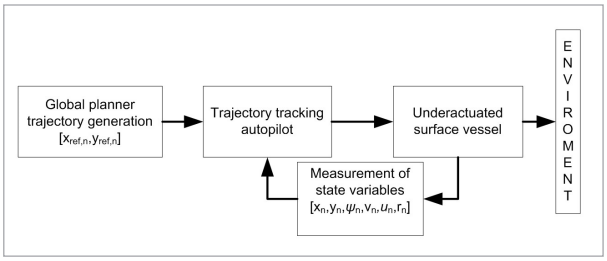
4. Simulations Results

The simulation results for the performance evaluation of the trajectory tracking controller proposed in the previous section are presented in this section. Only two integrators ($p = 2$) will be deemed in the simulation section for practical reasons.

As was already discussed, the behavior of the controlled system depends on the parameters k, K_1, K_2. Thus, in this work, and in order to determine values for the parameters of the controller for $p = 0, 1, 2$, the Monte Carlo Randomized Algorithm used in [6] is applied.

The control approach is applied on the original time-continuous system, as shown in Fig. 2. The marine vessel configuration is obtained from recent papers [9, 19, 13]. It has a length of 1.19 m, a mass of 17.6 kg and is represented for the following parameters: $m_{11} = 19\ kg$, $m_{22} = 35.2\ kg$, $m_{33} = 4.2\ kg$, $d_{11} = 4\ kg/s$, $d_{22} = 1\ kg/s$, $d_{33} = 10\ kg/s$. In order to perform realistic simulations, saturation levels in the control signals are imposed [10, 19]: $T_{u,max} = 1N$, $T_{u,min} = -1N$, $T_{r,max} = 1Nm$, $T_{r,min} = -1Nm$.

Figure 2
The architecture of the trajectory tracking controller



4.1. The Monte Carlo Randomized Algorithm

In the field of systems and control, probabilistic methods have been found to be useful, especially for problems related to robustness of uncertain systems [20]. One of these methods, the Monte Carlo Randomized Algorithm, is widely used in many fields such as

the radioactive decay, systems of interacting atoms, the traffic on roads, etc. [2]. In the control area, Monte Carlo methods allow to estimate an expectation value and they provide effective tools for the analysis of probabilistically robust control schemes.

Because of its nature, these types of algorithms can give an erroneous result with a non zero probability. Thus, a natural question arises about how many simulations should be performed to ensure the correct answer. Under a sufficiently large sample size N, a probabilistic statement can be made as shown below:

Theorem 2. (Tempo and Ishii [20]) Let $\varepsilon, \delta \in (0, 1)$, where ε is an a priori specified accuracy, and $1-\delta$ is the confidence interval. If

$$N \geq \left\lceil \frac{\log \frac{1}{\delta}}{\log \frac{1}{1-\varepsilon}} \right\rceil, \tag{26}$$

then the empirical maximum satisfies the following inequality with probability greater than $1 - \delta$

$$\text{Prob}_{\Delta}\left\{J(\Delta) \leq \hat{J}_{\max}\right\} \geq 1-\varepsilon. \tag{27}$$

That is,

$$\text{Prob}_{\Delta(1,..,N)}\left\{\text{Prob}_{\Delta}\left\{J(\Delta) \leq \hat{J}_{\max}\right\} \geq 1-\varepsilon\right\} > 1-\delta, \tag{28}$$

where J is the performance function and $\hat{J}_{\max}$ is the *empirical maximun*. For further details, see [20].

The theorem says that the empirical maximum is an estimate of the true value within an a priori specified accuracy ε with confidence δ if the sample size N satisfies (26). The algorithm may not produce an approximately correct answer, but the probability of this event is no greater than δ. It is worthy to emphasize that, in Theorem 2, the sample size N is finite and, moreover, is not dependent on the size of the uncertain set $\mathbf{B}$, the structured set of uncertainty matrices, and the probability density function $f_{\Delta}(\Delta)$, but only on ε and δ. In the next section, (26) is used to estimate the number of simulations.

4.2. A Monte Carlo Experiment

In this subsection, the Monte Carlo method is applied to select an appropiate set of controller parameters. Even though the optimum is not guaranteed, the Monte Carlo Experiment (MCE) provides an approximate solution based on a large number of trials (M). In this paper, a confidence value (δ) of 0.01, and an accuracy of 0.007 (ε) is adopted. Then, from (26), it is necessary to make 1 000 simulations. Hence, 1 000 values of each parameter ranging from 0 to 1 were simulated. This parameter range ensures convergence to zero tracking error (see the proof of Theorem 2).

The aim of MCE is to find the parameter values (k, K_1 and K_2) optimizing a defined cost function. An idea widely used in the literature is to consider the cost incurred by the tracking error [6, 4]. Let Φ be a desired trajectory, where $\#\Phi$ is the number of points of such trajectory. Let $C_x^{\Phi} = \sum_{i=0}^{\#\Phi}\left(\frac{1}{\#\Phi}\right)\left(x_{(i)} - x_{ref(i)}\right)^2$ be the squared error for the *x-coordinate*; $C_y^{\Phi} = \sum_{i=0}^{\#\Phi}\left(\frac{1}{\#\Phi}\right)\left(y_{(i)} - y_{ref(i)}\right)^2$ the squared error for the *y-coordinate*. Thus, the cost function can be represented by the combination of them with the aim to reduce the tracking error:

$$C^{\Phi} = \sum_{i=0}^{\#\Phi}\left(\frac{1}{\#\Phi}\right)\left(\left(x_{(i)} - x_{ref(i)}\right)^2 + \left(y_{(i)} - y_{ref(i)}\right)^2\right). \tag{29}$$

Thus, the objective is to find k, K_1, K_2,..., and K_p, in such way that C^{Φ} is minimized. To this end, in this work the MCE is carried considering $p = 0, 1, 2$ in (24). The MC experiment allows finding empirically the parameter values minimizing the cost function.

Below, the considerations made in the MCE are provided in more detail:

- The model mismatch between (1) and the Ship behavior is represented by the uncertainty $\mathbf{E}_n$, with high (unknown) order difference.
- The simulations are performed using MatLab software platform. The simulations are performed with $p = 1, 2$ and will be called C2 and C3, respectively. Note that the controller implementation with $p = 0$ corresponds to that presented in [19] and it will be called C1.
- For each controller, 1 000 simulations are carried out. All simulations are implemented with the same desired trajectory Φ. In this section, an eight-shaped trajectory is considered. The sampling time used is $T_s = 0.1s$.
- For each simulation, the controller parameters are constant chosen in a random way, such that the roots (r_i) of the linear systems defined in the right-

hand side of (11) and the left-hand side of (20) and (23) are $r_i \in (a,b)$, where b should be < 1 to ensure system stability (error convergence) and $a > 0$ and not too low for proper marine vessel response. In our case, $a = 0.5$ and $b = 0.99$ were empirically chosen considering a trade off between the speed of convergence to zero of tracking errors and a soft ship response. That is, all the roots are $r_i = rand(0.5,0.99)$.

- For $p = 0$ (controller proposed in [19]), 1 000 simulations are performed and the controller parameter is chosen as $k = r_1$, where r_1 is the root of the linear system in (11). The k value remains fixed during each simulation.
- For $p = 1$, 1 000 simulations are performed again. In each simulation, the controller parameters are constant chosen according to (20):

$$k = r_1 r_2$$
$$K_1 = \frac{-r_1 - r_2 + r_1 r_2 + 1}{T_s}. \quad (30)$$

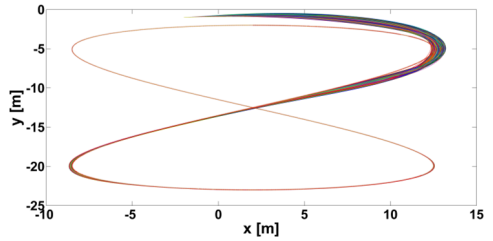
- For $p = 2$, 1 000 simulations are performed. In each simulation, the controller parameters are chosen constant according to (23):

$$k = r_1 r_2 r_3$$
$$K_1 = \frac{2r_1 r_2 r_3 + 1 - r_1 r_2 - r_1 r_3 - r_2 r_3}{T_s} \quad (31)$$
$$K_2 = \frac{-r_1 - r_2 - r_3 - r_1 r_2 r_3 + 1 + r_1 r_2 + r_1 r_3 + r_2 r_3}{T_s^2},$$

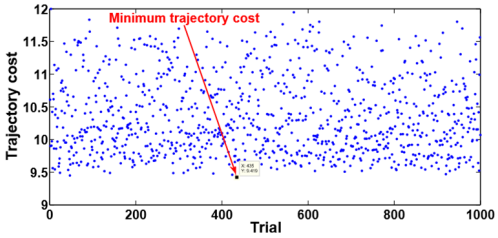
where r_1, r_2 and r_3 are the roots of the linear system in (23).

Figure 3(b) shows the results of the 1 000 simulations when $p = 1$. The results show the values taken by the cost function for each simulation; scattered values are obtained due to the randomness with which the parameters were chosen in each simulation. The minimum cost obtained for C2 is $C^{\Phi} = 9.419$. Figures 3(a) and 3(c) shows the trajectory tracking for each controller (C2 and C3) in the 1 000 simulations carried out. Figure 3(d) shows that the lowest cost obtained by C3, with $p = 2$, corresponds to $C^{\Phi} = 8.161$. By inspection it can be seen that, in general, all the cost values obtained by C3 are under the minimum value obtained when $p = 1$ (C2).

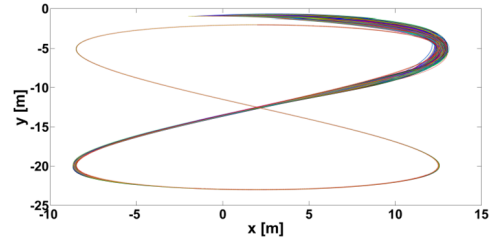
Figure 3
The results of Monte Carlo Experiment



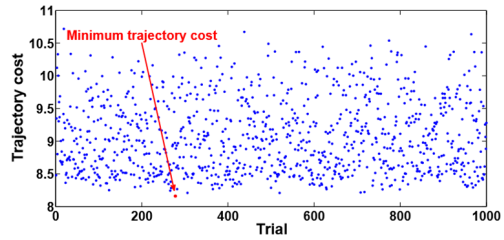
(a) The trajectory of C2 for the 1 000 trials



(b) The trajectory cost of C2 for the 1 000 trials



(c) The trajectory of C3 for the 1 000 trials



(d) The trajectory cost of C3 for the 1 000 trials

The analysis of the results shows that the performance of the controller improves as p increases. Thus, the results obtained by the MCE to choose the controller parameters verify the theoretical results obtained in the previous section. Table 1 shows the summary of the results obtained with each controller and the controller parameters used in the minimum trajectory cost simulation.

Table 1
The summary of the Monte Carlo Experiment simulations

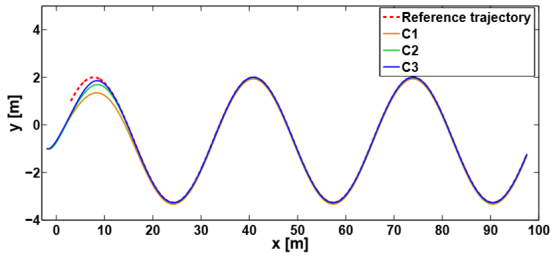
Controller \ Results	Minimum Cost	Controllers Parameters
C2	$C^{\Phi} = 9.419$	$k = 0.883$ $K_1 = 0.027$
C3	$C^{\Phi} = 8.161$	$k = 0.8827$ $K_1 = 0.0269$ $K_2 = 0.0011$

4.3. Simulation Results Considering Model Mismatches

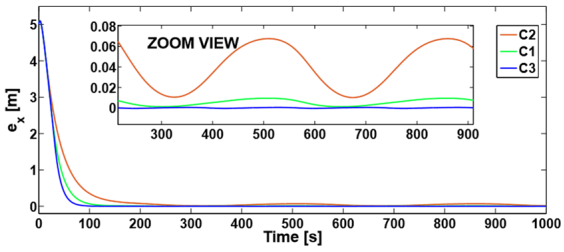
In this work, we are especially interested in situations for which there is parametric uncertainty in the model of the vehicle. Typical parameters for which this uncertainty is high, include mass and added mass for underwater vehicles which may be subject to large variations according to the payload configuration, and friction coefficients that are usually strongly dependent on the environmental conditions [17]. Thus, the system is simulated considering model mismatches. The model parameters were altered 15% of its nominal value: $m_{11} = 21.5\ kg$, $m_{22} = 40\ kg$, $m_{33} = 4.8\ kg$, $d_{11} = 4.6\ kg/s$, $d_{11} = 4.6\ kg/s$, $d_{22} = 1.15\ kg/s^{-1}$, $d_{33} = 11.5\ kg/s^{-1}$.

A sinusoidal reference trajectory is generated with a forward velocity of $u = 0.1\ m/s$. The reference trajectory starts at $(x_{ref}(0), y_{ref}(0)) = (3\ m, 1\ m)$, the sampling time T_s used for the simulation is 0.1 sec. and the initial position of the ship is $(x(0), y(0)) = (-2m, -1m)$. Figure 4(a) shows how the ship reaches and follows the reference trajectory even in the presence of undesirable disturbances under the command of C1, C2 and C3. As can be seen, the controllers reach and follow the desired trajectory without unexpected oscillations. Figures 4(b) and 4(c) show that C3 in the sequel has the lowest cost error when compared with the rest of the controller. By inspection, when the controller has two integrators (C3), the tracking error is the lowest and present a better performance against unwanted disturbances compared with C1 and C2.

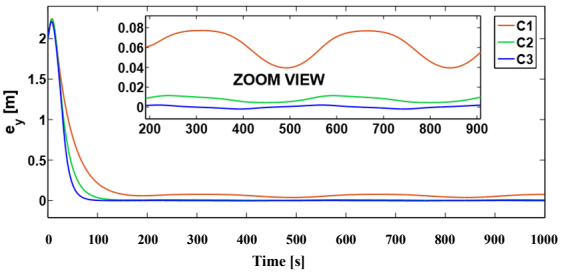
Figure 4
The results of simulations considering model mismatches



(a) The trajectory trackign of the ship



(b) The tracking error in *x-coordinate*



(c) The tracking error in *y-coordinate*

4.4. The Simulation Results Under Environmental Disturbances

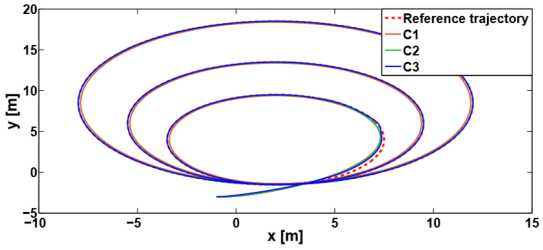
Finally, a curvature test is performed. Three circle trajectories with different radius were used in this work, as shown in Fig. 5(a). The inner trajectory has a radius of $r = 5.5\ m$, the medium one $r = 7.5\ m$ and the last one $r = 10\ m$. The initial position of the ship is at $(x_{ref}(0), y_{ref}(0)) = (-1\ m, -3\ m)$ and the trajectory begins in the position $(x(0), y(0)) = (2\ m, -1.5\ m)$.

To verify and illustrate the theoretical results, the proposed control law is tested in front of small environmental disturbances induced by a wave, wind and an ocean current [19, 7]. We simulate the controllers C1, C2 and C3 with the same parameters of the controller selected above. The environmental disturbances acting on the surge, sway and yaw dynamics are given by $T_{wu} = 0.005 m_{11} rand(\cdot)$, $T_{wv} = 0.0025 m_{22} rand(\cdot)$, $T_{wr} = 0.02 m_{33} rand(\cdot)$, where $rand(\cdot)$ is the random noise with a magnitude of 1 and zero lower bound [19, 7]. This choice results in non-zero-mean disturbances. The above disturbances are represented as follows:

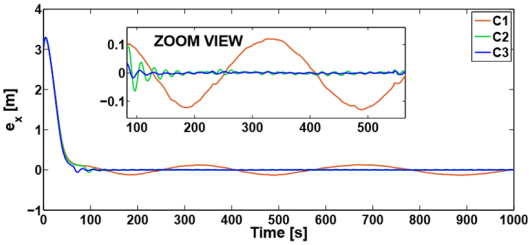
$$\dot{u} = \frac{m_{22}}{m_{11}} vr - \frac{d_{11}}{m_{11}} u + \frac{1}{m_{11}} T_u + \frac{1}{m_{11}} T_{wu}$$
$$\dot{v} = -\frac{m_{11}}{m_{22}} ur - \frac{d_{22}}{m_{22}} v + \frac{1}{m_{22}} T_{wv} \quad (32)$$
$$\dot{r} = \frac{m_{11} - m_{22}}{m_{33}} vu - \frac{d_{33}}{m_{33}} u + \frac{1}{m_{33}} T_r + \frac{1}{m_{33}} T_{wr}.$$

The effects of the disturbances introduced on the dynamic response are first illustrated in Fig. 5. The reference trajectory and the results of the controllers are shown in Fig. 5(a). As can be seen, all controllers reach and follow the desired trajectory. However, the performance of controllers with integral action is superior to C1 (controller proposed in [19]). Figures 5(b) and 5(c) show the plots of the tracking error in the *x–coordinate* and *y–coordinate* according to each controller used in the test for the three curvatures shown in Fig. 5(a).

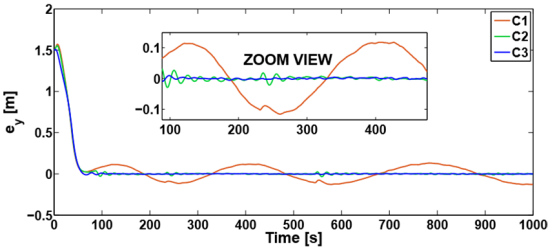
Figure 5
The results of simulation considering environmental disturbances



(a) The trajectory tracking of the ship



(b) The tracking error in *x – coordinate*



(c) The tracking error in *y – coordinate*

5. Conclusion

A new control law for trajectory tracking in marine vessels under uncertainties was presented. To deal with the uncertainties, a new term has been incorporated into the methodology presented in Serrano et al. [19]. This new approach allows reducing the effect of uncertainties in the tracking error. To tune the controller, the Monte Carlo experiment was used, and a cost function that depends on tracking errors was minimized. The proposed controllers are easy to implement, making them suitable for implementation in low-profile processors.

To demonstrate the effectiveness of the proposed methodology, several simulation tests were carried out incorporating different sources of uncertainty. The performance of C2 and C3 is noteworthy, while the complexity of the algorithms is not excessive. Finally, the proof of convergence to zero of the tracking errors has been included, ensuring that the task being performed will be accomplished accordingly.

Acknowledgements

This work was funded by the CONICET (National Council for Scientific Research), Argentina. The authors thank to the Institute of Chemical Engineering of the National University of San Juan, Argentina.

References

1. Aguiar, A. P., Hespanha, J. P. Trajectory Tracking and Path-following of Underactuated Autonomous Vehicles with Parametric Modeling Uncertainty. IEEE Transactions on Automatic Control, 2007, 52(8), 1362–1379. https://doi.org/10.1109/TAC.2007.902731

2. Barat, A., Ruskin, H. J., Crane, M. Probabilistic Models for Drug Dissolution. Part 1. Review of Monte Carlo and Stochastic Cellular Automata Approaches. Simulation Modelling Practice and Theory, 2006, 14(7), 843 – 856. https://doi.org/10.1016/j.simpat.2006.01.004

3. Belleter, D., Paliotta, C., Maggiore, M., Pettersen, K. Path Following for Underactuated Marine Vessels. IFAC-PapersOnLine, 2016, 49(18), 588–593. https://doi.org/10.1016/j.ifacol.2016.10.229
4. Blažič, S. A Novel Trajectory Tracking Control Law for Wheeled Mobile Robots. Robotics and Autonomous Systems, 2011, 59(11), 1001–1007. https://doi.org/10.1016/j.robot.2011.06.005
5. Candeloro, M., Lekkas, A. M., Sørensen, A. J. A Voronoi Diagram Based Dynamic Path Planning System for Underactuated Marine Vessels. Control Engineering Practice, 2017, 61, 41–54. https://doi.org/10.1016/j.conengprac.2017.01.007
6. Cheein, F. A., Scaglia, G. Trajectory Tracking Controller Design for Unmanned Vehicles: A New Methodology. Journal of Field Robotics, 2014, 31(6), 861–887. https://doi.org/10.1002/rob.21492
7. Do, K., Jiang, Z., Pan, J. Robust Global Stabilization of Underactuated Ships on a Linear Course: State and Output Feedback. International Journal of Control, 2003, 76(1), 1–17. https://doi.org/10.1080/0020717021000048233
8. Fossen, T. I., Breivik, M., Skjetne, R. Line of Sight Path Following of Underactuated Marine Craft. Proceedings of the 6th IFAC MCMC, Girona, Spain, 2003, 244–249. https://doi.org/10.1016/S1474-6670(17)37809-6
9. Ghommam, J., Mnif, F., Derbel, N. Global Stabilisation and Tracking Control of Underactuated Surface Vessels. IET, Control Theory & Applications, 2010, 4(1), 71–88. https://doi.org/10.1049/iet-cta.2008.0131
10. Lefeber, E., Pettersen, K. Y., Nijmeijer, H. Tracking Control of an Underactuated Ship. IEEE Transactions on Control Systems Technology, 2003, 11(1), 52–61. https://doi.org/10.1109/TCST.2002.806465
11. Lekkas, A. M., Fossen, T. I. Trajectory Tracking and Ocean Current Estimation for Marine Underactuated Vehicles. 2014 IEEE Conference on Control Applications (CCA), Juan Les Antibes, France, 8-10 October, 2014, 905–910. https://doi.org/10.1109/CCA.2014.6981451
12. Li, Z., Sun, J., Oh, S. Design, Analysis and Experimental Validation of a Robust Nonlinear Path Following Controller for Marine Surface Vessels. Automatica, 2009, 45(7), 1649–1658. https://doi.org/10.1016/j.automatica.2009.03.010
13. Ma, B. L. Global κ-Exponential Asymptotic Stabilization of Underactuated Surface Vessels. Systems & Control Letters, 2009, 58(3), 194–201. https://doi.org/10.1016/j.sysconle.2008.10.011
14. Mayne, D. Q., Rawlings, J. B., Rao, C. V., Scokaert, P. O. Constrained Model Predictive Control: Stability and Optimality. Automatica, 2000, 36(6), 789–814. https://doi.org/10.1016/S0005-1098(99)00214-9
15. Michalek, M. M., Kozlowski, K. R. Feedback Control Framework for Car-like Robots Using the Unicycle Controllers. Robotica, 2004, 30(4), 517–535. https://doi.org/10.1017/S0263574711000750
16. Moe, S., Caharija, W., Pettersen, K. Y., Schjolberg, I. Path Following of Underactuated Marine Surface Vessels in the Presence of Unknown Ocean Currents. 2014 American Control Conference, Portland, USA, 4-6 June, 2014, 3856-3861. https://doi.org/10.1109/ACC.2014.6858984
17. Nicolau, V., Miholca, C., Aiordachioaie, D., Ceanga, E. QFT Autopilot Design for Robust Control of Ship Course Keeping and Course Changing Problems. Journal of Control Engineering and Applied Informatics, 2006 7(1), 44–56.
18. Paliotta, C., Lefeber, E., Pettersen, K. Y. Trajectory Tracking of Under Actuated Marine Vehicles. 2016 IEEE 55th Conference on Decision and Control (CDC), Las Vegas, USA, 12-14 December, 2016, 5660-5667.
19. Serrano, M. E., Scaglia, G. J., Mut, V., Ortiz, O. A., Jordan, M. Tracking Trajectory of Underactuated Surface Vessels: A Numerical Method Approach. Control Engineering and Applied Informatics, 2004, 15(4), 15–25.
20. Tempo, R., Ishii, H. Monte Carlo and Las Vegas Randomized Algorithms for Systems and Control: An Introduction. European Journal of Control, 2007, 13(2-3), 189–203. https://doi.org/10.3166/ejc.13.189-203
21. Wang, L., Sun, T., Yu, X. Trajectory Linearization Based Robust Model Predictive Control for Constrained Unmanned Surface Vessels. Information Technology and Control, 2016, 45(4), 412–418.
22. Xu, J., Wang, M., Qiao, L. Dynamical Sliding Mode Control for the Trajectory Tracking of Underactuated Unmanned Underwater Vehicles. Ocean Engineering, 2015, 105, 54–63. https://doi.org/10.1016/j.oceaneng.2015.06.022
23. Yang, Y., Du, J., Liu, H., Guo, C., Abraham, A. A Trajectory Tracking Robust Controller of Surface Vessels with Disturbance Uncertainties. IEEE Transactions on Control Systems Technology, 2014, 22(4), 1511–1518. https://doi.org/10.1109/TCST.2013.2281936
24. Zhang, G., Zhang, X. Practical Robust Neural Path Following Control for Underactuated Marine Vessels with Actuators Uncertainties. Asian Journal of Control, 2017, 19(1), 173–187. https://doi.org/10.1002/asjc.1345
25. Zhang, Z., Wu, Y. Further Results on Global Stabilisation and Tracking Control for Underactuated Surface Vessels with Non-Diagonal Inertia and Damping Matrices. International Journal of Control, 2015, 88(9), 1679–1692. https://doi.org/10.1080/00207179.2015.1013061