Journal of
Marine Science and Engineering

MDPI

Article

Robust Model Predictive Control Based on Active Disturbance Rejection Control for a Robotic Autonomous Underwater Vehicle

Jaime Arcos-Legarda * and Álvaro Gutiérrez *

Escuela Técnica Superior de Ingenieros de Telecomunicación, Universidad Politécnica de Madrid, 28040 Madrid, Spain

* Correspondence: jaime.arcos@upm.es (J.A.-L.); a.gutierrez@upm.es (Á.G.)

Abstract: This work aims to develop a robust model predictive control (MPC) based on the active disturbance rejection control (ADRC) approach by using a discrete extended disturbance observer (ESO). The proposed technique uses the ADRC approach to lump disturbances and uncertainties into a total disturbance, which is estimated with a discrete ESO and rejected through feedback control. Thus, the effects of the disturbances are attenuated, and a model predictive control is designed based on a canonical model free of uncertainties and disturbances. The proposed control technique is tested through simulation into a robotic autonomous underwater vehicle (AUV). The AUV's dynamic model is used to compare the performance of a classical MPC and the combined MPC-ADRC. The evaluation results show evidence of the superiority of the MPC-ADRC over the classical MPC under tests of reference tracking, external disturbances rejection, and model uncertainties attenuation.

Keywords: model predictive control; active disturbance rejection control; extended state observer; autonomous underwater vehicles; underwater robotics

check for updates

Citation: Arcos-Legarda, J.; Gutiérrez, A. Robust Model Predictive Control Based on Active Disturbance Rejection Control for a Robotic Autonomous Underwater Vehicle. *J. Mar. Sci. Eng.* **2023**, *11*, 929. https://doi.org/10.3390/jmse11050929

Academic Editor: David Moreno-Salinas

Received: 30 March 2023
Revised: 24 April 2023
Accepted: 25 April 2023
Published: 26 April 2023

1. Introduction

Model predictive control (MPC) is based on the solution of an optimal control problem (OCP) for a prediction time horizon, subject to dynamic, states, and control constraints. The OCP attempts to find the best control sequence that produces the plant's desired behavior. It is solved at every sample time step, taking as initial conditions the system's current state. Then, the first element of the optimal control sequence is applied to the plant, and the entire optimization process is repeated at the subsequent sampled step with a moving horizon [1]. MPC was initially developed to meet the needs of industrial processes with relatively slow dynamics [2]; however, the development of computation capabilities of modern computers has impulsed the application of MPC to faster dynamic systems with complex dynamics. This is the case of MPC applied to autonomous underwater vehicles (AUV) [3,4], where the possibility to plan a control action according to the operation horizon's features allows for the development of an anticipated control action instead of reactive control strategies.

MPC has been successfully implemented in a wide variety of applications; however, its main drawback is that it relays on the prior knowledge of an accurate mathematical model of the system dynamics. Variations of the classical MPC have been proposed to attenuate the effect of model uncertainties with techniques of robust MPC for linear systems [5]. Likewise, the robust nonlinear MPC (NMPC) have been proposed with the use of extended Kalman filters that estimate the process noise [6]. Robustness against model mismatching or unknown disturbances in NMPC are generally considered as an inherent property [7,8].

Most robust NMPC schemes that consider model uncertainties in the design process are based on minimizing the worst-case scenario from a set of uncertain models; however, the application of these techniques results in a complex task due to the need to impose

stability constraints that could lead to an unfeasible solution during the search of an optimal control sequence [9]. In a complementary way, the adaptive model predictive control uses parametric estimation combined with robust MPC algorithms to gradually reduce the conservative control laws by adjusting the prediction model [10].

An alternative way to get a robust MPC is the estimation of disturbances through disturbance observers as high-gain observers [11], sliding mode disturbance observers [12], and nonlinear disturbance observers [13]. Although the disturbance observers in combination with predictive control have shown satisfactory results, their combination requires accurate prediction models and the evaluation of complex dynamic constraints. Alternatively, this paper proposes a model predictive control based on the active disturbance rejection control, which uses a discrete extended state observer to estimate the effects of model uncertainties and external disturbance. Additionally, the proposed control strategy uses a canonical dynamic model to predict the system behavior during the prediction horizon, which results in linear dynamic constraints.

Classical model-based control approaches rely on an accurate dynamic model of the system. However, in applications such as AUVs control, it is almost impossible to find a precise model, due to the hydrodynamic effects and the ocean flow currents, which produce model uncertainties and unknown external disturbances. Researchers have devoted a huge amount of effort to solving this problem. For instance, a sliding mode control (SMC) combined with an optimal thrust distribution on an over-actuated AUV was adopted to improve the robustness of dynamic positioning under the effects of the ocean current and model uncertainties [14]. The SMC has been also combined with disturbance observers to stabilize an AUV under time-varying external disturbance [15]. However, the SMC implementation needs to take special care of chattering in the control signals. Recently, adaptive control techniques based on backstepping have been developed for trajectory tracking task in AUVs [16,17]. The adaptive control approach has been also studied using fuzzy techniques with the design of fuzzy disturbance observer [18]. The use of disturbance observers has shown to be beneficial for the attenuation of disturbance effects. This is the case of the motion control strategy that integrates deep reinforcement learning and the extended state observer (ESO) to compensate for the model uncertainties into an optimal motion control policy [19]. Although the previously-mentioned controllers have effectively improve the closed-loop robustness of AUV motion control, there are opportunities to enhance the design process and results of control strategies for AUV. This is the case of combination of optimal control techniques like MPC with active disturbance rejection control strategies [20].

The remainder of this paper is organized as follows: The Section 2 describes the proposed control technique. Section 2.1 shows the design of the discrete extended state observer. Section 2.2 presents our proposed model predictive control based on the active disturbance rejection control approach. Section 2.3 finds a mathematical model for the autonomous underwater robot. Section 3 presents and discuses simulation results. Finally, Section 4 draws conclusions and future work.

2. Materials and Methods

The proposed control technique is a compound strategy that uses the features of a disturbance observer and an optimal control approach to finding a control sequence for a receding horizon that predicts the best control actions in every sampled time step. To develop the proposed technique, this section presents first a discrete extended state observer that estimates the system's states and a total disturbance that lumps the external disturbances and model uncertainties into a single signal. Then, a model predictive control (MPC) strategy based on the active disturbance rejection control (ADRC) approach is developed for a general system with a single input and a single output. Then, an autonomous underwater vehicle is modeled to describe the dynamics of an underactuated underwater robot, which is based on the four DoF Seabotix LBV150 ROV. This model is transformed to separate the controllable dynamics from the uncontrollable internal dynamics of the

system. This results in a simplified canonical model that is used to test the proposed MPC based on ADRC.

2.1. Discrete Extended State Observer

Let us consider a general nonlinear system, defined as

$$y_m^{(n)} = f_m\left(y_m, \dot{y}_m, \cdots, y_m^{(n-1)}\right) + g_m\left(y_m, \dot{y}_m, \cdots, y_m^{(n-1)}\right)u_m + \varphi\left(y_m, \dot{y}_m, \cdots, y_m^{(n-1)}, u_m\right) + \zeta, \tag{1}$$

where n is the order of the system, y_m is the control output, $y_m^{(n)}$ is the n-th time derivative of y_m, u_m is the control input, $f_m(\cdot)$ is the known dynamic model of the system, $\varphi(\cdot)$ represents the model uncertainties and unmodeled dynamics, and ζ represents the external disturbances.

Assuming bounded uncertainties, bounded disturbances, and bounded dynamic behavior, the signals $\varphi(\cdot)$, ζ, and $f_m(\cdot)$ can be lumped into a total disturbance ξ_m, such as that (1) is transformed into

$$y_m^{(n)} = g_m\left(y_m, \dot{y}_m, \cdots, y_m^{(n-1)}\right)u_m + \xi_m. \tag{2}$$

Then, a local approximation around a defined operation condition allows to express $g(\cdot)$ as a constant κ_m, which is valid in an open neighborhood around the operation point. Then, (2) is expressed as,

$$y_m^{(n)} = \kappa_m u_m + \xi_m. \tag{3}$$

This simplified model is used to design the extended state observer and feedback controller under the following assumptions:

Assumption 1. *The internal model of ξ_m can be approximated by a constant, at least during an infinitesimal period of time.*

Assumption 2. *ξ_m and its first time derivative are bounded.*

Assumption 3. *The first time derivative of the total disturbance is expressed as*

$$\frac{d}{dt}\xi_m = \epsilon, \tag{4}$$

where ϵ is a small constant, such as that $\epsilon \approx 0$.

Then, based on such assumptions, the variable ξ_m can be used as an extended state of the dynamic system. This allows to define an extended state vector

$$\mathbf{x} := [y_m \quad \dot{y}_m \quad \ddot{y}_m \quad \cdots \quad y_m^{(n-1)}, \xi_m]^\intercal, \tag{5}$$

which produces an extended state space representation of the system as

$$\begin{array}{rcccl} \dot{y}_m & = & \dot{x}_1 & = & x_2, \\ \ddot{y}_m & = & \dot{x}_2 & = & x_3, \\ \vdots & = & \vdots & = & \vdots \\ y_m^{(n)} & = & \dot{x}_n & = & x_{n+1} + \kappa u, \\ \dot{\xi}_m & = & \dot{x}_{n+1} & = & \epsilon. \end{array} \tag{6}$$

In a compact form, the extended model is represented by a linear state space model given by:

$$\begin{array}{rcl} \dot{\mathbf{x}} & = & A\mathbf{x} + Bu + E, \\ y & = & C\mathbf{x}, \end{array} \tag{7}$$

where $A \in \mathbb{R}^{(n+1)\times(n+1)}$, $B \in \mathbb{R}^{n+1}$, $C \in \mathbb{R}^{n+1}$, $E \in \mathbb{R}^{n+1}$,

$$A = \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 1 \\ 0 & 0 & 0 & \cdots & 0 \end{bmatrix}, \quad B = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ \kappa \\ 0 \end{bmatrix}, \quad E = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ \epsilon \end{bmatrix}, \quad C^{\intercal} = \begin{bmatrix} 1 \\ 0 \\ \vdots \\ 0 \\ 0 \end{bmatrix}.$$

Now, let us define a sampling period T and use the Euler's method to find a discrete approximation of the extended linear model (7). This is,

$$\begin{array}{rcl} \mathbf{x}(k+1) & = & (I + TA)\mathbf{x}(k) + TBu(k) + TE, \\ y(k) & = & C\mathbf{x}(k), \end{array} \tag{8}$$

where k represents the discrete time variable.

Based on the discrete approximation of the simplified dynamic model (8), a discrete extended state observer (ESO) is proposed using the Luenberger's observer form. In this case, the observer estimates the system states, but also the total disturbance represented by the extended state x_{n+1}. Since the pair $(I + TA, C)$ is observable, the discrete ESO is proposed as follows:

$$\begin{array}{rcl} \hat{\mathbf{x}}(k+1) & = & (I + TA)\hat{\mathbf{x}}(k) + TBu(k) + L(y(k) - \hat{y}(k)), \\ \hat{y}(k) & = & C\hat{\mathbf{x}}(k), \end{array} \tag{9}$$

where $\hat{(\cdot)}$ denotes the estimated variable and $L := [l_1 \quad l_2 \quad \cdots \quad l_n]^{\intercal}$ is the vector of the observer's gains.

By subtracting (9) from (8), the estimation error dynamics is given by

$$\hat{\mathbf{e}}(k+1) = (I + TA - LC)\hat{\mathbf{e}} + TE, \tag{10}$$

where $\hat{\mathbf{e}} := \mathbf{x} - \hat{\mathbf{x}}$.

Under the assumption of a bounded and small ϵ, the stability of (10) depends of the eigenvalues of $(I + TA - LC)$. Then, the observer gains are selected in such a way that the eigenvalues of $(I + TA - LC)$ are inside a unitary circle of a complex z-plane, which results in eigenvalues with modulus less than one. This ensures the convergence of the estimated states to the real ones and an accurate estimation of the total disturbance. This estimation will be used in the feedback control law proposed during the next section.

2.2. Robust MPC and ADRC

A model predictive control based on active disturbance rejection control is performed by using the disturbance estimation provided by the discrete ESO. The states and disturbance estimation are used into an optimal control problem (OCP) that computes an optimal control sequence for a receding horizon. The model used to predict the system behavior during the optimization process is a canonical structure that dismisses the model uncertainties, external disturbances, and model nonlinearities. This solves the optimization problem with less computation effort than the typical nonlinear MPC approaches. Since the total disturbance is rejected through feedback control, the result is a robust model predictive control strategy.

A general description of the control loop is shown into the Figure 1, where the continuous mathematical model represents the system dynamics, whose output y is sampled with a sampling period T. The sampled output $y(k)$ and control signal $u(k)$ are used by the discrete ESO to estimate the system states and the total disturbance, which are a source input to solve the OCP that produces the robust MPC-ADRC control law.

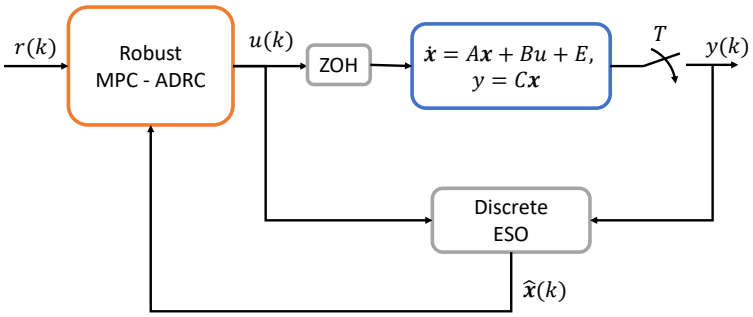


Figure 1. Control loop for a robust MPC based on ADRC and a discrete ESO.

The OCP is defined to find an optimal control sequence that minimizes an objective function, which is projected in a time horizon into the future and computed based on a simplified model of the system. Based on the assumption that the total disturbance estimation converges to the real disturbance, the simplified model takes a canonical form that is used as a prediction model, which is almost free of uncertainties and disturbances. This canonical model is given by

$$\begin{aligned} \mathbf{x}_s(k+1) &= A_s\mathbf{x}_s(k) + B_s u_s(k), \\ y_s(k) &= C_s\mathbf{x}_s(k), \end{aligned} \tag{11}$$

where $\mathbf{x}_s(k) := [y_m(k) \quad \dot{y}_m(k) \cdots y_m^{(n-1)}(k)]^\intercal$, $A_s \in \mathbb{R}^{n\times n}$, $B_s \in \mathbb{R}^n$, $C_s \in \mathbb{R}^n$,

$$A_s = \begin{bmatrix} 1 & T & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & T \\ 0 & 0 & \cdots & 1 \end{bmatrix}, \quad B_s = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ T\kappa \end{bmatrix}, \quad C_s^\intercal = \begin{bmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix}.$$

In order to formulate the OCP, let us define, first, the control sequence along the prediction horizon, which will be considered as a vector of search variables in the optimization process. This is

$$\mathbf{u}_s(k) := [u_s(0) \quad u_s(1) \quad \cdots \quad u_s(\rho) \quad \cdots \quad u_s(N-1)]^\intercal, \tag{12}$$

where N is the length, in samples, of the prediction horizon, $\rho \in \mathbb{Z}^+$ is the time discrete variable used inside the prediction horizon, which is set to zero in every sampling step and runs until $\rho = N-1$, this is $\rho = \{0, 1, 2, \cdots, N-1\}$. The prediction horizon and control sequence are depicted into the Figure 2.

In the formulation of the optimal control problem, the cost function is defined as

$$J_N(\mathbf{u}_s(k)) = \sum_{\rho=0}^{N-1} \Big(w_e e_s^2(\rho) + w_u u_s^2(\rho) \Big), \tag{13}$$

where $e_s(\rho) := r_s - y_s(\rho)$ is the tracking error, $r_s := r(k)$ is the reference at time k, w_e and w_u are positive constant weights for the error and control signals, respectively.

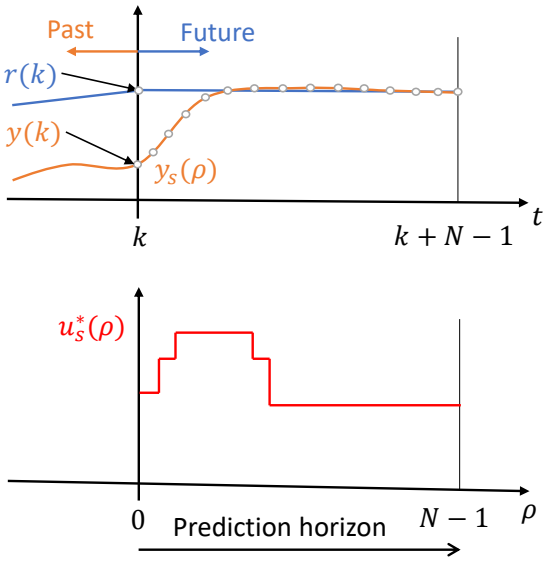


Figure 2. Prediction horizon and optimal control sequence for time k.

The optimal control problem is subject to a set of equality and inequality constraints. The equality constraints ensure the satisfaction of initial conditions and the dynamic behavior of the system during the projected horizon. At the same time, the inequality constraints ensure that the control actions are inside of the actuators' range of the action.

Then, the OCP is formulated as:

$$\min_{\mathbf{u}_s(k)} \quad J_N(\mathbf{u}_s(k)) \tag{14}$$

$$\text{s.t.:} \quad \mathbf{x}_s(\rho) = [\hat{x}_1 \quad \hat{x}_2 \quad \cdots \quad \hat{x}_n]^\intercal, \qquad \rho = 0, \tag{15}$$

$$\mathbf{x}_s(\rho+1) = A_s\mathbf{x}_s(\rho) + B_s u_s(\rho), \qquad \forall \rho \in [0, N-1], \tag{16}$$

$$-u_{max} + \tfrac{1}{\kappa}\hat{x}_{n+1} \leq u_s(\rho) \leq u_{max} + \tfrac{1}{\kappa}\hat{x}_{n+1}, \qquad \rho = 0, \tag{17}$$

$$-u_{max} \leq u_s(\rho) \leq u_{max}, \qquad \forall \rho \in [1, N-1], \tag{18}$$

where (14) is the cost function, (15) sets the initial conditions of the model to the state values computed by the discrete ESO at time k, (16) is a dynamic constraint that ensures the satisfaction of the system dynamics based on the canonical model (11), (17) imposes boundaries for the control signal at the first time step of the prediction horizon, these bounds are modified every sampled time step according to the total disturbance estimation, in such a way that the optimal control sequence allows to inject the rejection of the estimated disturbance. Likewise, (18) establishes the boundaries for the control actions over the prediction horizon between $\rho = 1$ and $\rho = N - 1$.

The implementation of the dynamic constraints implies the propagation of the initial conditions through all the prediction horizon, which results in a complex nonlinear structure that increases the computation time. In order to reduce the complexity of the dynamic constraints, a multiple shooting approach is used to transform the dynamic propagation into a set of equality constraints [21].

In order to implement the multiple shooting algorithm, let us define an extended set of search variables in a vector as

$$\mathbf{w} := \left[\mathbf{u}_s^\intercal(k) \quad \mathbf{x}_s^\intercal(0) \quad \mathbf{x}_s^\intercal(1) \quad \cdots \quad \mathbf{x}_s^\intercal(N)\right]^\intercal, \tag{19}$$

where $\mathbf{w}$ includes the control and state vectors sequences at $t = k$. This search vector increases the number of search variables but simplifies the nonlinearity propagation of

the system dynamics along the prediction horizon. Based on this approach, the system's dynamic constraints are included as a set of equality constraints as

$$\mathbf{h}_N(\mathbf{w}) := \begin{bmatrix} \mathbf{x}_s(0) - [\hat{x}_1 \quad \hat{x}_2 \quad \cdots \quad \hat{x}_n]^\intercal \\ \mathbf{x}_s(1) - (A_s\mathbf{x}_s(0) + B_s u_s(0)) \\ \vdots \\ \mathbf{x}_s(N) - (A_s\mathbf{x}_s(N-1) + B_s u_s(N-1)) \end{bmatrix} = \mathbf{0}, \tag{20}$$

and a set of inequality constraints defined as

$$\mathbf{g}_N(\mathbf{w}) := \begin{bmatrix} u_s(0) - u_{max} - \frac{1}{\kappa}\hat{x}_{n+1} \\ u_s(1) - u_{max} \\ u_s(2) - u_{max} \\ \vdots \\ u_s(N) - u_{max} \\ -u_s(0) - u_{max} + \frac{1}{\kappa}\hat{x}_{n+1} \\ -u_s(1) - u_{max} \\ -u_s(2) - u_{max} \\ \vdots \\ -u_s(N) - u_{max} \end{bmatrix} \leq \mathbf{0}. \tag{21}$$

Then, the OCP is transformed into a nonlinear programming problem (NLP) as

$$\begin{array}{lll} \min\limits_{\mathbf{w}} & J_N(\mathbf{w}) & \\ \text{s.t.:} & \mathbf{g}_N(\mathbf{w}) \leq \mathbf{0}, & \text{Inequality constraints,} \\ & \mathbf{h}_N(\mathbf{w}) = \mathbf{0}, & \text{Equality constraints.} \end{array}$$

The solution of the NLP results in an optimal sequence of control actions and a set of state vectors that minimize the cost function for the prediction horizon. Based on that solution, the control action for the current sampled time k is computed by subtracting the total disturbance effect, which was estimated by the discrete ESO, from the first element of the optimal control sequence found in the NLP. This allows to define a control law as

$$u(k) := u_s^*(0) - \frac{1}{\kappa}\hat{x}_{n+1}(k). \tag{22}$$

where $\mathbf{u}_s^*$ denotes the optimal control sequence and $u_s^*(0)$ is the first element of the sequence.

In order to compute the closed-loop behavior under the effect of the control signal proposed in (22), it is applied to the discrete model shown in (8), which results in a closed-loop dynamics describe by

$$\begin{array}{rcl} x_1(k+1) & = & x_1(k) + Tx_2(k), \\ x_2(k+1) & = & x_2(k) + Tx_3(k), \\ \vdots & = & \vdots \\ x_n(k+1) & = & x_n(k) + T\kappa u_s^*(0) + T(x_{n+1}(k) - \hat{x}_{n+1}(k)), \\ x_{n+1}(k+1) & = & x_{n+1}(k) + T\epsilon. \end{array} \tag{23}$$

The stability analysis of the closed-loop dynamics is divided into two parts. The first part considers the stability guaranteed by the selection of the ESO's gains. The appropriate selection of ESO's gains allows for defining an ultimate bound to the total disturbance estimation as

$$\lim_{k\to\infty} (x_{n+1}(k) - \hat{x}_{n+1}(k)) = \gamma, \tag{24}$$

where γ is a positive constant that can be arbitrarily reduced as much as the physical limitation in the hardware allows. The second part of the stability analysis considers the effect of the optimal control sequence $\mathbf{u}_s^*$. Since this control sequence is computed from a subset of control signals that guarantee the closed-loop stability through the convergence of the optimization problem system, then it is possible to affirm that the closed-loop is stable and the controlled variables converge to the vicinity of the target trajectory with an ultimate bound.

Finally, The a summary of the implementation procedure of the MPC based on ADRC is shown in the Algorithm 1. The following section shows the implementation of the algorithm on an autonomous underwater robot.

Algorithm 1 MPC-ADRC implementation

Step 0: Find a simulation model.
Step 1: Design ESO.
Step 1.1: Define the order of the design dynamic model n and find an approximation of the control input gain κ.
Step 1.2: Build the extended state model with the canonical form described in (7).
Step 1.3: Define a sampling period T according the Nyquist–Shannon sampling theorem and find a discrete model approximation of the system dynamics as (8).
Step 1.4: Assign the observer gains L such as the eigenvalues of $I + TA - LC$ are inside the unite circle in the complex z-plane and build the extended state observer ESO as shown in (9).
Step 2: MPC implementation.
Step 2.1: Define the weights w_e and w_u and build the cost function (13).
Step 2.2: Define the equality constraints (20) and inequalities constraints (21).
Step 2.3: Define the prediction horizon length N.
Step 2.4: Implement a nonlinear programming algorithm that solves the optimization problem at each sampling step.
Step 3: Evaluation.
Step 3.1: Build the control law as (22) and evaluate the closed-loop control using the simulation model.
Step 3.2: Evaluate the performance and robustness of the closed-loop, if it is necessary change observer gains L, weights w_e and w_u, or prediction horizon length N and repeat evaluation.

2.3. Autonomous Underwater Robot Model

In order to test the proposed robust MPC based on ADRC, an autonomous underwater robot is used as a test platform of the control strategy. This system has dynamic features that make it a suitable application of the robust MPC based on ADRC. First, the system dynamics is highly nonlinear with hydrostatic and hydrodynamic effects that difficult an accurate model identification. Second, the environment where the robot works has a high degree of uncertainty with unpredictable ocean currents, especially in the vicinity of reefs where the water flow is distorted by the rocks. These produce external disturbances on the robot dynamics, which must be rejected by the control system. A third reason to apply the proposed control strategy to the robotic AUV is the need to guarantee the satisfaction of safety constraints to avoid the collision of the robot with obstacles in the environment and to optimize the performance of the robot to increase its energy autonomy.

Let us consider the robotic autonomous underwater vehicle (AUV) shown in the Figure 3, which has four thrusters: two in the rear, one in the top, and one in the port (left side) [22]. The frames $\mathcal{I}$ and $\mathcal{B}$ shown in the figure represent the fixed inertial frame and the body frame, respectively.

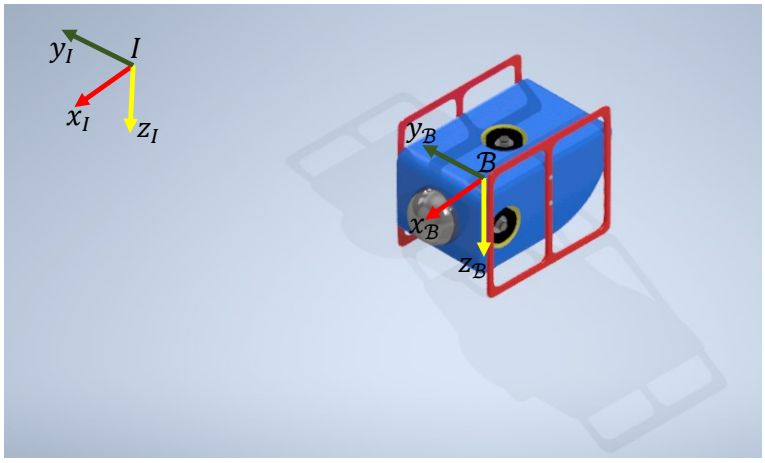


Figure 3. Robotic autonomous underwater vehicle.

In order to find a mathematical model that describes the dynamic behavior of the robotic AUV, the motion variables are defined according with the SNAME notation for a submerged body through a fluid [23]. In this sense, the position of the robot is defined as $\boldsymbol{\eta} := [\boldsymbol{\eta}_1^\intercal \quad \boldsymbol{\eta}_2^\intercal]^\intercal$, where $\boldsymbol{\eta}_1 \in \mathbb{R}^3$ represents the position with respect to the inertial frame $\mathcal{I}$ and $\boldsymbol{\eta}_2 \in \mathcal{S}^3$ represents the robot's attitude, as well known as pose. To describe the position of the robot, the three Cartesian coordinates are used such as that $\boldsymbol{\eta}_1 := [x \quad y \quad z]^\intercal$, where x, y, and z represent the motion in the directions of *surge*, *sway*, and *heave*, respectively. Likewise, the attitude is represented with the Euler angles as $\boldsymbol{\eta}_2 := [\phi \quad \theta \quad \psi]^\intercal$, where ϕ, θ, and ψ represent *roll*, *pitch*, and *yaw*, respectively.

Then, the Newton-Euler's method is used to analyze the motion of a submerged rigid-body with six degrees of freedom (DoF) [24]. The mathematical model that describes the AUV motion is given by

$$\boldsymbol{M}\dot{\boldsymbol{\nu}} + \boldsymbol{C}(\boldsymbol{\nu})\boldsymbol{\nu} + \boldsymbol{D}(\boldsymbol{\nu})\boldsymbol{\nu} + \boldsymbol{g}(\boldsymbol{\eta}) = \boldsymbol{\Gamma}, \tag{25}$$

$$\dot{\boldsymbol{\eta}} = \boldsymbol{J}(\boldsymbol{\eta})\boldsymbol{\nu}, \tag{26}$$

where $\boldsymbol{\nu} := [\boldsymbol{\nu}_1^\intercal \quad \boldsymbol{\nu}_2^\intercal]^\intercal \in \mathbb{R}^6$ denotes the linear and angular velocities with $\boldsymbol{\nu}_1 := [u \quad v \quad w]^\intercal$ and $\boldsymbol{\nu}_2 := [p \quad q \quad r]^\intercal$, respectively. $\boldsymbol{\Gamma}$ designates a vector of generalized forces and moments defined as: $\boldsymbol{\Gamma} := [\boldsymbol{f}^\intercal \quad \boldsymbol{\tau}^\intercal]^\intercal$, where the body-fixed forces in x, y, and z are represented as $\boldsymbol{f} := [X \quad Y \quad Z]^\intercal$ and the body-fixed moments in ϕ, θ, and ψ are defined as $\boldsymbol{\tau} := [K \quad M \quad N]^\intercal$.

Since the submerged body is exposed to hydrostatic and hydrodynamic effects, the mass matrix includes the rigid body mass and inertial in M_{RB}, and an added mass term in M_A, which represents the mass of fluid surrounding the robot that it must move to deflect the resistance of the fluid. This is modeled as a virtual mass added to the rigid body as:

$$\boldsymbol{M} := \boldsymbol{M}_{RB} + \boldsymbol{M}_A, \tag{27}$$

where $\boldsymbol{M}_{RB} \in \mathbb{R}^{6\times6}$ and $\boldsymbol{M}_A \in \mathbb{R}^{6\times6}$. Likewise, the Coriolis and centrifugal effects are represented as

$$\boldsymbol{C}(\boldsymbol{\nu}) := \boldsymbol{C}_{RB} + \boldsymbol{C}_A, \tag{28}$$

where $\boldsymbol{C}_{RB} \in \mathbb{R}^{6\times6}$ designates the effects of the rigid-body and $\boldsymbol{C}_A \in \mathbb{R}^{6\times6}$ designates the effects of the added mass. An additional factor in the description of the AUV is the viscous damping. This is a hydrodynamic effect that produces dissipative moments and forces. The main sources of damping are the skin friction and the interaction with waves and internal currents. The total effects of damping are lumped into a drag matrix $\boldsymbol{D}(\boldsymbol{\nu})$, which is approximated as

$$\boldsymbol{D}(\boldsymbol{\nu}) := \boldsymbol{D}_q(\boldsymbol{\nu}) + \boldsymbol{D}_l, \tag{29}$$

where $\boldsymbol{D}_q(\boldsymbol{\nu}) \in \mathbb{R}^{6\times6}$ and $\boldsymbol{D}_l \in \mathbb{R}^{6\times6}$ are the quadratic and linear drag matrices, respectively.

The term $\boldsymbol{g}(\boldsymbol{\eta})$ describes the restoring forces acting in the robot. It includes the gravitational effects represented by a force W passing through the center of gravity (CG) and a buoyancy force B passing through the center of buoyancy (CB). The restoring forces and moments are expressed in the body-fixed frame as

$$\boldsymbol{g}(\boldsymbol{\eta}) = \begin{bmatrix} (W-B)\sin\theta \\ -(W-B)\cos\theta\sin\phi \\ -(W-B)\cos\theta\cos\phi \\ -(y_g W - y_b B)\cos\theta\cos\phi + (z_g W - z_b B)\cos\theta\sin\phi \\ (z_g W - z_b B)\sin\phi + (x_g W - x_b B)\cos\theta\cos\phi \\ -(x_g W - x_b B)\cos\theta\sin\phi - (y_g W - y_b B)\sin\theta \end{bmatrix}, \tag{30}$$

where $W = mg$, $B = \rho_w g \nabla$, m is the sum of the robot mass and the mass of water in the ballast tanks, g is the acceleration of the gravity, ρ_w is the water density, ∇ is the volume of liquid displaced by the robot, x_g, y_g, z_g, and x_b, y_b, z_b, are the coordinates of the CG and CB with respect to the frame $\mathcal{B}$. In an effort to simplify the dynamic interaction between these two forces, the CG and CB are considered coincident in the origin of $\mathcal{B}$; however, in general, it is desirable to design AUV with the CB ahead of the CG to guarantee controllability and stability in pitch.

The Jacobian matrix $\boldsymbol{J}(\boldsymbol{\eta})$ transforms the relative velocities expressed in $\mathcal{B}$ into the inertial reference frame $\mathcal{I}$. This is

$$\dot{\boldsymbol{\eta}} = \boldsymbol{J}(\boldsymbol{\eta})\boldsymbol{\nu}, \tag{31}$$

$$\Downarrow \tag{32}$$

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \\ \dot{\theta} \\ \dot{\phi} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} \boldsymbol{J}_1(\boldsymbol{\eta}_2) & \mathbf{0}_{3\times3} \\ \mathbf{0}_{3\times3} & \boldsymbol{J}_2(\boldsymbol{\eta}_2) \end{bmatrix} \begin{bmatrix} u \\ v \\ w \\ p \\ q \\ r \end{bmatrix}, \tag{33}$$

where $\boldsymbol{J}_1(\boldsymbol{\eta}_2) := \mathbf{R}(\boldsymbol{\eta}_2) \in \mathcal{SO}(3)$ and $\boldsymbol{J}_2(\boldsymbol{\eta}_2) \in \mathbb{R}^{3\times3}$ are a rotational and a transformation matrices, respectively.

Since the underwater thrusters installed on the robot are not strictly aligned with the body-fixed frame axis, the propulsion forces do not produce a pure vector of generalized forces and moments in $\boldsymbol{\Gamma}$. Then, it is necessary to find a mathematical expression that relates the propulsion force vector $\mathbf{f}_{th}$ with the $\boldsymbol{\Gamma}$. This is

$$\boldsymbol{\Gamma} := T_{th}\mathbf{f}_{th}, \tag{34}$$

where T_{th} is known as an allocation matrix, which depends on the robot's architecture and the location of the thrusters. Additional details about the model can be found in [22,25].

Since the robot considered in this paper has four thrusters and can move in six DoF, then the system is underactuated. This implies that the robot is partially controllable and part of its dynamics is not directly affected by the propulsion of the actuators. According to the geometrical distribution of the thrusters, the robot is underactuated in the pitch and roll angles. Therefore, the following assumptions are adopted:

Assumption 4. *It is assumed that the dynamics of the underactuated DoF are stabilized with complementary elements such as fairwater planes.*

Assumption 5. *High speed and sharp turns are avoided to evade instability effects in the vertical plane.*

In order to adapt the proposed control strategy to the robotic AUV, a transformation of the model is executed such as that the nonlinearities, model uncertainties, and external disturbances are shown as input equivalent signals and the controllable dynamics is isolated from the uncontrollable one. First, let us write the model as general nonlinear system

$$\dot{\mathbf{x}}_r = \mathbf{f}_r(\mathbf{x}_r) + \mathbf{g}_r\mathbf{u}_r + \boldsymbol{\zeta}_r, \tag{35}$$

where $\boldsymbol{\zeta}_r$ represents the unmodeled dynamics and external disturbances, $\mathbf{x}_r = [\boldsymbol{\eta}^\intercal \quad \dot{\boldsymbol{\eta}}^\intercal]^\intercal$, $\mathbf{u}_r = \mathbf{f}_{th}$,

$$\mathbf{f}_r(\mathbf{x}_r) = \begin{bmatrix} \dot{\boldsymbol{\eta}} \\ \nabla \boldsymbol{J}(\boldsymbol{\eta})\dot{\boldsymbol{\eta}} + \boldsymbol{M}^{-1}(-\boldsymbol{C}(\boldsymbol{\nu})\boldsymbol{\nu} - \boldsymbol{D}(\boldsymbol{\nu})\boldsymbol{\nu} - \boldsymbol{g}(\boldsymbol{\eta})) \end{bmatrix}, \quad \mathbf{g}_r(\mathbf{x}_r) = \begin{bmatrix} 0 \\ \boldsymbol{M}^{-1}T_{th} \end{bmatrix},$$

and

$$\boldsymbol{\nu} = \boldsymbol{J}^{-1}(\boldsymbol{\eta})\dot{\boldsymbol{\eta}}.$$

Then, the control output is defined as:

$$\mathbf{h}(\mathbf{x}_r) := \begin{bmatrix} x^* - x \\ y^* - y \\ z^* - z \\ \psi^* - \psi \end{bmatrix}, \tag{36}$$

where $(\cdot)^*$ denotes the control reference.

By applying successive time derivatives to the control output until the control input is explicit, the system dynamics takes the form

$$\frac{d}{dt}\mathbf{h}(\mathbf{x}_r) = \nabla\mathbf{h}(\mathbf{x}_r)\dot{\mathbf{x}}_r, \tag{37}$$

$$\frac{d}{dt}\mathbf{h}(\mathbf{x}_r) = \nabla\mathbf{h}(\mathbf{x}_r)\mathbf{f}_r(\mathbf{x}_r), \tag{38}$$

$$\frac{d}{dt}\mathbf{h}(\mathbf{x}_r) = \mathbf{L}_{\mathbf{f}_r}\mathbf{h}(\mathbf{x}_r), \tag{39}$$

$$\frac{d^2}{dt^2}\mathbf{h}(\mathbf{x}_r) = \mathbf{L}^2_{\mathbf{f}_r}\mathbf{h}(\mathbf{x}_r) + \mathbf{L}_{\mathbf{g}_r}\mathbf{L}_{\mathbf{f}_r}\mathbf{h}(\mathbf{x}_r)\mathbf{u}_r + \mathbf{L}_{\zeta_r}\mathbf{L}_{\mathbf{f}_r}\mathbf{h}(\mathbf{x}_r), \tag{40}$$

where $\nabla\mathbf{h}(\mathbf{x}_r)$ is the gradient of $\mathbf{h}(\mathbf{x}_r)$, $\mathbf{L}_{\mathbf{f}_r}\mathbf{h}(\mathbf{x}_r)$ is the Lie derivative of $\mathbf{h}(\mathbf{x}_r)$ along $\mathbf{f}_r(\mathbf{x}_r)$, $\mathbf{L}^2_{\mathbf{f}_r}\mathbf{h}(\mathbf{x}_r)$ is its second Lie derivative, and $\mathbf{L}_{\mathbf{g}_r}\mathbf{L}_{\mathbf{f}_r}\mathbf{h}(\mathbf{x}_r)$ is a decoupling matrix, which is locally invertible [26].

Based on the normal form described in (39) and (40), a simplified model is proposed to design the control system. In this case, a canonical model is assumed as the base model for the control design. This model is dynamically decoupled, affine to the control input, and it includes a vector of lumped disturbances as described in its state space form:

$$\dot{\boldsymbol{\varphi}} = \begin{bmatrix} \mathbf{0}_{4\times4} & \mathbf{I}_{4\times4} \\ \mathbf{0}_{4\times4} & \mathbf{0}_{4\times4} \end{bmatrix}\boldsymbol{\varphi} + \boldsymbol{\kappa}\mathbf{u}_r + \boldsymbol{\xi}, \tag{41}$$

where the state vector for the actuated DoF is

$$\boldsymbol{\varphi} := \begin{bmatrix} \mathbf{h}(\mathbf{x}_r) \\ \mathbf{L}_{\mathbf{f}_r}\mathbf{h}(\mathbf{x}_r) \end{bmatrix}, \tag{42}$$

the nonlinearities, model uncertainties, and external disturbances are lumped in a vector of total disturbances as

$$\boldsymbol{\xi} := \begin{bmatrix} \mathbf{0} \\ \mathbf{L}^2_{\mathbf{f}_r}\mathbf{h}(\mathbf{x}_r) + \mathbf{L}_{\zeta}\mathbf{L}_{\mathbf{f}_r}\mathbf{h}(\mathbf{x}_r) \end{bmatrix}, \tag{43}$$

and the decoupling matrix $\mathbf{L}_{\mathbf{g}_r}\mathbf{L}_{\mathbf{f}_r}\mathbf{h}(\mathbf{x}_r)$ is approximated by a diagonal matrix with control input gains as,

$$\mathbf{L}_{\mathbf{g}_r}\mathbf{L}_{\mathbf{f}_r}\mathbf{h}(\mathbf{x}_r) \approx \boldsymbol{\kappa} := \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ \kappa_1 & 0 & 0 & 0 \\ 0 & \kappa_2 & 0 & 0 \\ 0 & 0 & \kappa_3 & 0 \\ 0 & 0 & 0 & \kappa_4 \end{bmatrix}. \tag{44}$$

This model represents a multiple-input multiple-output system; however, since the dynamics of each DoF are considered decoupled, the control can be designed following the procedures described in the Section 2.2 for each element of the following set of single-input single-output systems

$$\begin{aligned} \begin{bmatrix} \dot{\varphi}_i \\ \dot{\varphi}_{i+4} \end{bmatrix} &= \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}\begin{bmatrix} \varphi_i \\ \varphi_{i+4} \end{bmatrix} + \begin{bmatrix} 0 \\ \kappa_i \end{bmatrix} u_{r_i} + \begin{bmatrix} 0 \\ \xi_i \end{bmatrix}, \\ h_i &= \begin{bmatrix} 1 & 0 \end{bmatrix}\begin{bmatrix} \varphi_i \\ \varphi_{i+4} \end{bmatrix}, \end{aligned} \qquad \forall\, i = \{1,2,3,4\}, \tag{45}$$

which are canonical models to be implemented as dynamic constraints during the prediction horizon.

3. Results

The proposed robust MPC based on ADRC is evaluated with three simulation scenarios that test the capabilities of the controller to track references, reject external disturbances, and attenuate the effects of model uncertainties. In order to perform such evaluations, a simulation model is built with the parameters in Table 1, which are adapted from the unsupervised online identification performed in [22]. An input control gain approximation, around the equilibrium point of the rest position, is performed by measuring the open loop gain of the controlled variables. The results of such approximation allow finding the parameters needed for control design shown in Table 2. Then, observer gains are selected such as that ultimately bounded behavior is achieved by the estimation error in (10). In this way, the gains in L are computed by defining the eigenvalues $\lambda(I + TA - LC)$ with a modulus inferior to one, this is $|\lambda(I + TA - LC)| < 1$. Then, a try-and-error procedure has been undertaken to determine a prediction horizon of $N = 10$ and a set of cost function's weights as presented in Table 3. These tuning parameters are not necessarily optimal; however, their performance is good enough for the control purpose.

In the event of physical implementation, it is necessary to consider the elapsed time to solve the optimal control problem. Since the solution must be completed every sampling step before the next sampling is taken, the stopping criteria set in the optimization software must be configured so that the computation time is the shortest possible.

Table 1. Autonomous underwater robot's parameters.

Parameter	Symbol	Value	Units
Mass	m	8.0	kg
Inertia	I_{xx}, I_{yy}, I_{zz}	0.1589, 0.1589, 0.1589	kg·m^2
Buoyancy	B	79.5881	N
Linear drag	X_u, Y_v, Z_w,	−13.6040, −18.1106, −17.1828	–
Linear drag	K_p, M_q, N_r,	−16.4146, −16.4146, −16.4146	–
Quadratic drag	$X_{\|u\|u}$, $Y_{\|v\|v}$, $Z_{\|w\|w}$,	−17.8534, −1.0594, −3.6482	–
Quadratic drag	$K_{\|p\|p}$, $M_{\|q\|q}$, $N_{\|r\|r}$,	−10.3483, −10.3483, −10.3483	–
Added mass	$X_{\dot{u}}$, $Y_{\dot{v}}$, $Z_{\dot{w}}$	1.7532, 0.6636, 2.898	kg

Table 2. Input control gain parameters.

Parameter	κ_1	κ_2	κ_3	κ_4
Value	0.0967	0.0550	0.0575	0.5000

Table 3. Control tuning parameters.

Controller	**Observer Gains (L)**	**Function Cost Weights (w_e, w_u)**
Control surge	[0.8400 23.3600 215.0400]	(10000, 0.01)
Control sway	[0.8400 23.3600 215.0400]	(50000, 0.01)
Control heave	[0.5700 10.6275 64.7425]	(10000, 0.01)
Control yaw	[0.5700 10.6275 64.7425]	(8000, 0.1)

The first evaluation test compares the performance and robustness of the proposed control strategy with an MPC without the assistance of the discrete ESO. Figure 4 shows the controlled outputs of both controllers tracking predefined trajectories. This simulation also compares the capabilities of the controllers to reject external disturbances. At the time $t = 15$ s, the controlled linear positions are perturbed with an external force of 1 N and the yaw angle with a moment of 1 Nm.

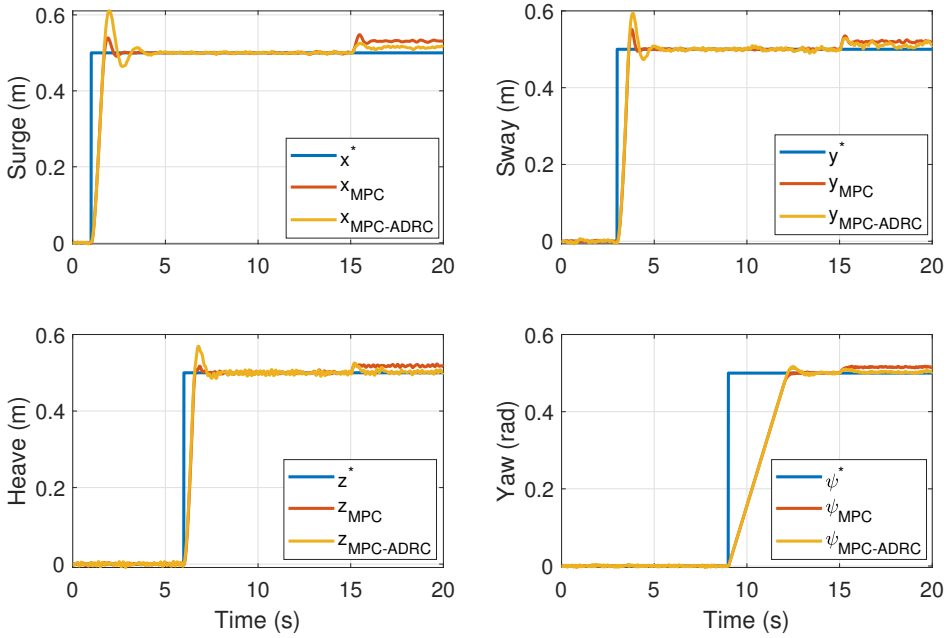


Figure 4. Tracking trajectory and external disturbance rejection.

Both controllers, the MPC and the MPC-ADRC, have bounded control actions with $-|u_{max}| \leq u \leq |u_{max}|$, which allows comparing the robustness of each controller with a time integral of the square error defined as

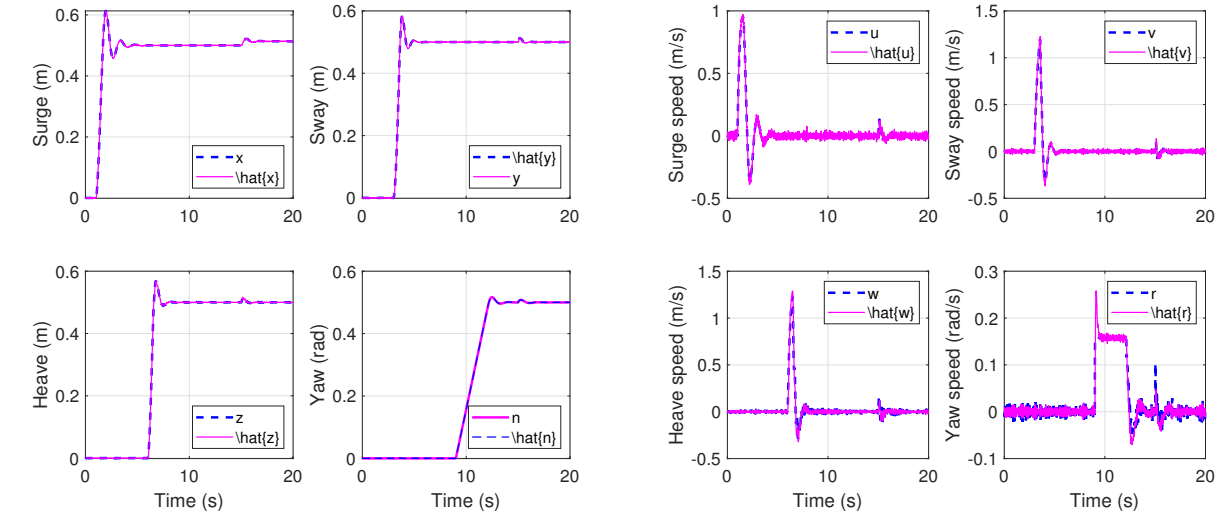
$$ISE = \int_{t=15}^{t=20} \mathbf{h}^{\mathsf{T}}(\mathbf{x}_r)\mathbf{h}(\mathbf{x}_r)dt. \tag{46}$$

The ISE indexes of the MPC and MPC-ADRC are shown in Table 4. These results show evidence of better disturbance rejection of the MPC based on ADRC, with an ISE index equivalent to 24.75% of the MPC index.

Table 4. ISE index during external disturbance rejection

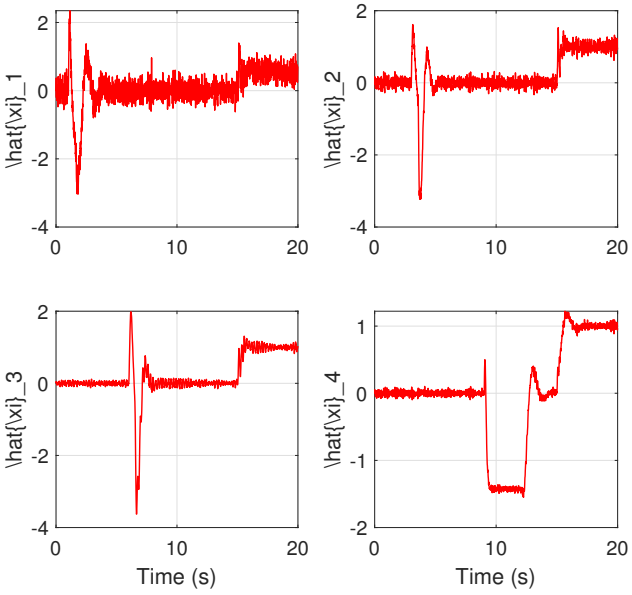
MPC	**MPC-ADR**
0.0097	0.0024

The ESO's estimations and control signals of the MPC based on ADRC during the first evaluation are shown in Figure 5. The current position coordinates, disturbed by white noise, and their estimations are shown in Figure 5a. These position estimations result in filtered measurements, which are suitable to implement the control techniques. Figure 5b shows the current speeds and their estimations, in this figure it is shown the convergence of the estimated signals to the real speed values. Figure 5c shows the total disturbances estimation with the evolution from the estimation of model uncertainties during the first 15 seconds and the reaction to estimate the external disturbances for $t \geq 15$ s. The control signals are shown in Figure 5d, where the boundaries on the control signals are satisfied, which is evidence of satisfaction of the inequality constraints in the optimal control problem. These control signals show the controller's reaction to reject the external disturbances in $t \geq 15$ s.

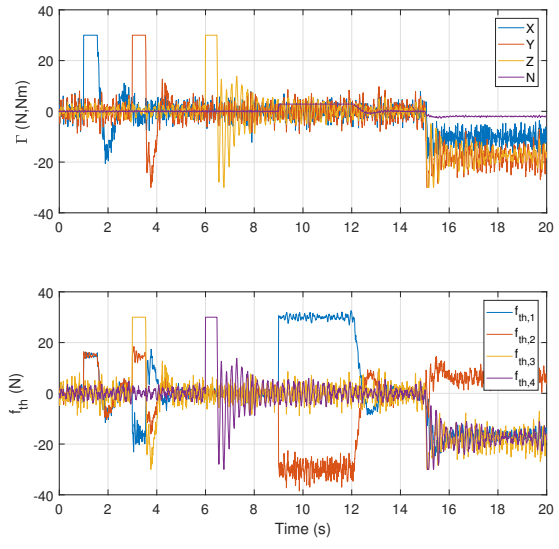


(**a**) Position estimation

(**b**) Speed estimation



(**c**) Disturbances estimation



(**d**) Generalized and thrusters forces

Figure 5. Extended state observer estimations and control signals.

The second evaluation tests the robustness of the proposed controller against model uncertainties with simulations of parameter variations. In this evaluation, the trajectory tracking task performed during the first evaluation is repeated 21 times with values of the robot's mass homogeneously distributed between 50% and 150% of its nominal value.

Figure 6 shows the behavior of the system's controlled output during all the simulations. The responses of the parameter variation form a cyan envelope around the nominal response in magenta, which is evidence of a bounded behavior around the nominal response and also shows robustness against model uncertainties.

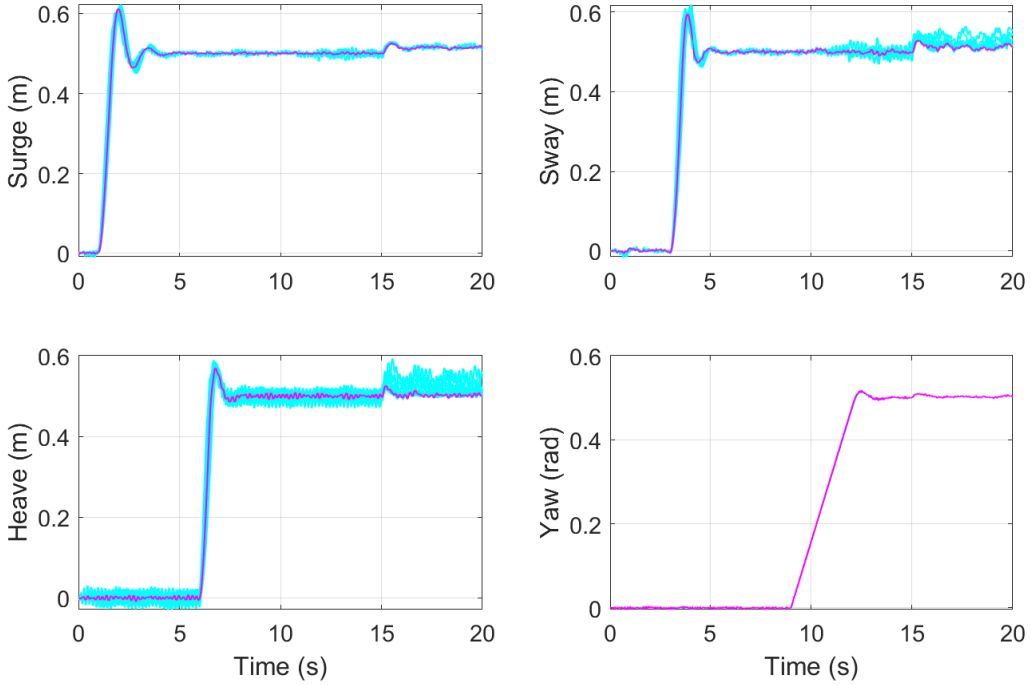
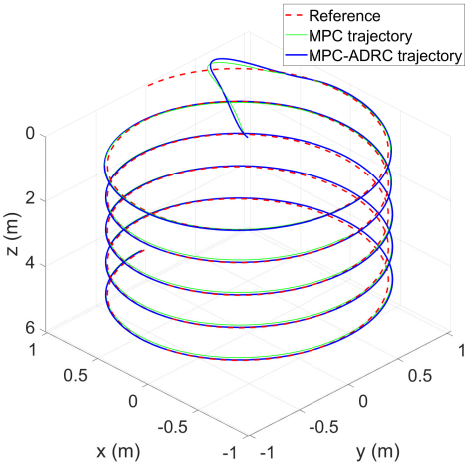
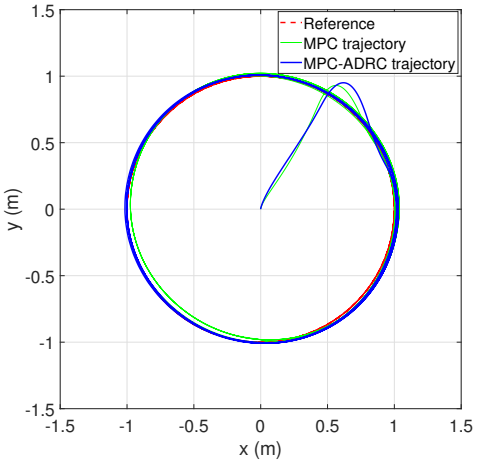


Figure 6. Robustness against model uncertainties. Mass variation between 50% and 150% of the nominal value.

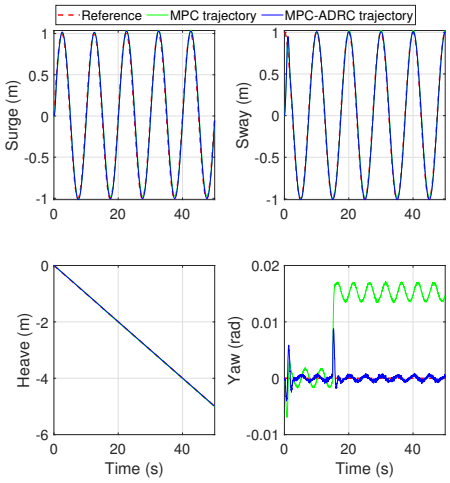
The final simulation executes a trajectory tracking task that imposes a spiral pattern starting the robot at $(0, 0, 0)$. The spiral reference is shown in Figure 7a, where the MPC without assistance and the MPC based on ADRC perform a transient response and then the trajectory tracking is executed. The vertical position changes at a constant speed of 0.1 m/s and goes from 0 m to 5 m. Horizontally, the robot moves in a circle centered at $(0, 0)$ with a radius of 1 m. In this plane, a complete circumference is tracked every 10 s, and the robot's position evolves according to the top view shown in Figure 7b. To test the robustness against external disturbances, an external force of 1 N is injected into the three-axis at $t = 15$ s The behavior of the robot shows that both the MPC without assistance and the MPC based on ADRC accomplish tracking the trajectory; however, our control strategy achieves a better disturbance rejection, especially in the robot's heading as shown in the yaw angle of the Figure 7c. In order to verify the satisfaction of the robot's physical limitation during the test of the MPC based on ADRC, the robot's velocity is shown in Figure 7d, where the maximum speed is bounded under a maximum of 1.5 m/s, which is an acceptable value for the robotic AUV. Likewise, the control signals are shown in Figure 7e, where the satisfaction of the inequality constraints of the optimal control problem is evidenced.



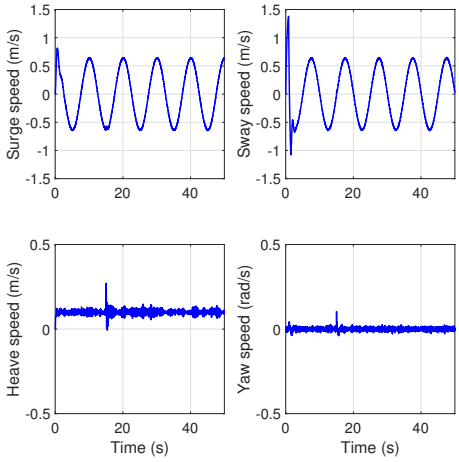
(**a**) Trajectory tracking of a spiral pattern



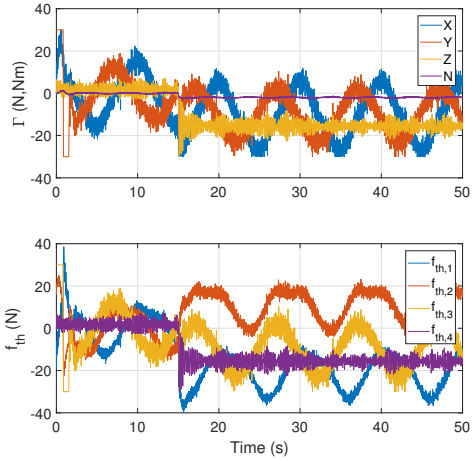
(**b**) Top view of the robot's trajectory



(**c**) Position tracking



(**d**) Speed tracking



(**e**) Generalized and thrusters forces

Figure 7. Trajectory tracking with external disturbance rejection.

4. Conclusions

A robust model predictive control was developed using the active disturbance rejection control approach to estimate a total disturbance that lumps the nonlinearities, model uncertainties, and external disturbances into a unified signal. This signal is estimated using a discrete extended disturbance observer that also finds an estimation of the state vector. Then, the proposed control strategy uses the estimated disturbance and states to find an optimal control sequence for a receding horizon by using a simplified canonical model, which simplifies the dynamic constraints during the solution of the optimal control problem. The first element of the optimal control sequence and the disturbance estimation are used to define a control law, which is imposed through feedback control action that guarantees reference tracking, external disturbance rejection, and attenuation of model uncertainties.

To evaluate the proposed control strategy, a robotic autonomous underwater vehicle was used as a test platform. A protocol of three evaluation scenarios was executed in simulation to test the robustness of the proposed control. The performance of the MPC based on ADRC is compared with an MPC without the assistance of the discrete ESO. The evaluation tests show evidence of effective trajectory tracking and robustness against external disturbances and parameter uncertainties. Future work will explore the physical implementation of the proposed control strategy into an evaluation prototype of a robotic autonomous underwater vehicle.

Author Contributions: Conceptualization, J.A.-L.; Formal analysis, J.A.-L.; Funding acquisition, Á.G.; Investigation, J.A.-L. and Á.G.; Methodology, J.A.-L. and Á.G.; Project administration, Á.G.; Software, J.A.-L.; Supervision, Á.G.; Validation, J.A.-L.; Writing—original draft, J.A.-L.; Writing—review & editing, Á.G. All authors have read and agreed to the published version of the manuscript.

Funding: This work has been supported by project Nautilus by Grant PID2020-112502RB-C41 funded by MCIN/AEI/10.13039/501100011033. First author was supported by European Union with the plan Next Generation EU, the Spain *Ministerio de Universidades* with RD 289/2021.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Acknowledgments: European Union for the plan Next Generation EU and the Spain *Ministerio de Universidades*.

Conflicts of Interest: The authors declare no conflict of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript; or in the decision to publish the results.

References

1. Camacho, E.F.; Alba, C.B. *Model Predictive Control*; Springer Science & Business Media: Berlin/Heidelberg, Germany, 2013.
2. Qin, S.J.; Badgwell, T.A. A survey of industrial model predictive control technology. *Control Eng. Pract.* **2003**, *11*, 733–764.
3. Shen, C.; Shi, Y.; Buckham, B. Trajectory tracking control of an autonomous underwater vehicle using Lyapunov-based model predictive control. *IEEE Trans. Ind. Electron.* **2017**, *65*, 5796–5805.
4. Yan, Z.; Gong, P.; Zhang, W.; Wu, W. Model predictive control of autonomous underwater vehicles for trajectory tracking with external disturbances. *Ocean Eng.* **2020**, *217*, 107884.
5. Campo, P.J.; Morari, M. Robust model predictive control. In Proceedings of the 1987 American Control Conference, Minneapolis, MN, USA, 10–12 June 1987; pp. 1021–1026.
6. Nagy, Z.K.; Braatz, R.D. Robust nonlinear model predictive control of batch processes. *AIChE J.* **2003**, *49*, 1776–1786.
7. Allgower, F.; Findeisen, R.; Nagy, Z.K. Nonlinear model predictive control: From theory to application. *J. Chin. Inst. Chem. Eng.* **2004**, *35*, 299–316.
8. Heshmati-Alamdari, S.; Karras, G.C.; Marantos, P.; Kyriakopoulos, K.J. A robust predictive control approach for underwater robotic vehicles. *IEEE Trans. Control Syst. Technol.* **2019**, *28*, 2352–2363. [CrossRef]
9. Chen, H. A Feasible Moving Horizon $\mathcal{H}_\infty$ Control Scheme for Constrained Uncertain Linear Systems. *IEEE Trans. Autom. Control* **2007**, *52*, 343–348. [CrossRef]
10. Adetola, V.; DeHaan, D.; Guay, M. Adaptive model predictive control for constrained nonlinear systems. *Syst. Control Lett.* **2009**, *58*, 320–326. [CrossRef]

11. Zhang, G.; Yan, W.; Gao, J.; Liu, C. High-gain observer-based model predictive control for cross tracking of underactuated autonomous underwater vehicles. In Proceedings of the 2016 IEEE International Conference on Underwater System Technology: Theory and Applications (USYS), Penang, Malaysia, 13–14 December 2016; pp. 115–120.
12. Gao, J.; Zhang, G.; Wu, P.; Zhao, X.; Wang, T.; Yan, W. Model predictive visual servoing of fully-actuated underwater vehicles with a sliding mode disturbance observer. *IEEE Access* **2019**, *7*, 25516–25526. [CrossRef]
13. Yang, H.; Deng, F.; He, Y.; Jiao, D.; Han, Z. Robust nonlinear model predictive control for reference tracking of dynamic positioning ships based on nonlinear disturbance observer. *Ocean Eng.* **2020**, *215*, 107885. [CrossRef]
14. Vu, M.T.; Le, T.H.; Thanh, H.L.N.N.; Huynh, T.T.; Van, M.; Hoang, Q.D.; Do, T.D. Robust position control of an over-actuated underwater vehicle under model uncertainties and ocean current effects using dynamic sliding mode surface and optimal allocation control. *Sensors* **2021**, *21*, 747. [CrossRef] [PubMed]
15. Zheng, J.; Song, L.; Liu, L.; Yu, W.; Wang, Y.; Chen, C. Fixed-time sliding mode tracking control for autonomous underwater vehicles. *Appl. Ocean Res.* **2021**, *117*, 102928. [CrossRef]
16. An, S.; Wang, L.; He, Y.; Yuan, J. Adaptive Backstepping Sliding Mode Tracking Control For Autonomous Underwater Vehicles With Input Quantization. *Adv. Theory Simul.* **2022**, *5*, 2100445.
17. Zhang, Y.; Liu, J.; Yu, J. Adaptive asymptotic tracking control for autonomous underwater vehicles with non-vanishing uncertainties and input saturation. *Ocean Eng.* **2023**, *276*, 114280. [CrossRef]
18. Wang, H.D.; Zhai, Y.X.; Shah, U.H.; Karkoub, M.; Li, M. Adaptive fuzzy control of underwater vehicle manipulator system with dead-zone band input nonlinearities via fuzzy performance and disturbance observers. *Ocean Eng.* **2023**, *277*, 114194. [CrossRef]
19. Huang, F.; Xu, J.; Wu, D.; Cui, Y.; Yan, Z.; Xing, W.; Zhang, X. A general motion controller based on deep reinforcement learning for an autonomous underwater vehicle with unknown disturbances. *Eng. Appl. Artif. Intell.* **2023**, *117*, 105589. [CrossRef]
20. Lamraoui, H.C.; Qidan, Z. Path following control of fully-actuated autonomous underwater vehicle in presence of fast-varying disturbances. *Appl. Ocean Res.* **2019**, *86*, 40–46. [CrossRef]
21. Bock, H.G.; Plitt, K.J. A multiple shooting algorithm for direct solution of optimal control problems. *IFAC Proc. Vol.* **1984**, *17*, 1603–1608. [CrossRef]
22. Karras, G.C.; Marantos, P.; Bechlioulis, C.P.; Kyriakopoulos, K.J. Unsupervised online system identification for underwater robotic vehicles. *IEEE J. Ocean. Eng.* **2018**, *44*, 642–663. [CrossRef]
23. Snam, E. *Nomenclature for Treating the Motion of a Submerged Body through a Fluid*; Technical and Research Bulletin 1–5; The Society of Naval Architects and Marine Engineers: Jersey City, NJ, USA, 1950.
24. Fossen, T.I. *Handbook of Marine Craft Hydrodynamics and Motion Control*; John Wiley & Sons: Hoboken, NJ, USA, 2011.
25. Fossen, T.I. Guidance and Control of Ocean Vehicles. Ph.D. Thesis, University of Trondheim, Trondheim, Norway, 1999.
26. Arcos-Legarda, J.; Cortes-Romero, J.; Beltran-Pulido, A.; Tovar, A. Hybrid disturbance rejection control of dynamic bipedal robots. *Multibody Syst. Dyn.* **2019**, *46*, 281–306. [CrossRef]