Adaptive control for autonomous ships with uncertain model and unknown propeller dynamics

Haseltalab, Ali; Negenborn, Rudy R.

DOI
10.1016/j.conengprac.2019.104116

Publication date
2019

Document Version
Accepted author manuscript

Published in
Control Engineering Practice

Citation (APA)
Haseltalab, A., & Negenborn, R. R. (2019). Adaptive control for autonomous ships with uncertain model and unknown propeller dynamics. *Control Engineering Practice*, *91*, Article 104116. https://doi.org/10.1016/j.conengprac.2019.104116

Adaptive Control for Autonomous Ships with Uncertain Model and Unknown Propeller Dynamics

Ali Haseltalab, Rudy R. Negenborn

Department of Maritime and Transport Technology, Delft University of Technology, Delft, the Netherlands. Emails: {a.haseltalab,r.r.negenborn}@tudelft.nl

Abstract

Motion control is one of the most critical aspects in the design of autonomous ships. During maneuvering, the dynamics of propellers as well as the craft hydrodynamical specifications experience sever uncertainties. In this paper, an adaptive control approach is proposed to control the motion and trajectory tracking of an autonomous vessel by adopting neural networks that is used for estimating the dynamics of the propellers and handling hydrodynamical uncertainties. Considering that the maneuvering model of a vessel resemble a nonlinear non-affine-in-control system, the proposed neural-based adaptive control algorithm is designed to estimate the nonlinear influence of the input function which in this case is the dynamics of propellers and thrusters. It is also shown that the proposed methodology is capable of handling state dependent uncertainties within the ship maneuvering model. A Lyapunov-based technique and Uniform Ultimate Boundedness are used to prove the correctness of the algorithm. To assess the method's performance, several experiments are considered including trajectory tracking simulations in the port of Rotterdam.

Keywords: Adaptive Control, Autonomous Vessels, Neural Networks, Propeller Dynamics.

1. Introduction

Autonomous Surface Vessels (ASVs) are types of ships that are capable of observing and sensing their surrounding environment to maneuver or carry

out dynamic positioning operations without intervention of human operators. Recently, the maritime industry has started to investigate the possibility of bringing ASVs into operation. Rolls Royce expects to be able to introduce a fully autonomous vessel by 2035 [1]. ASVs can be beneficial from several points of view such as crew cost and safety. Many dangerous operations can be carried out by ASVs where there is no operator on-board. ASVs are extensively investigated by the scientific community to address numerous existing challenges on the way of having an operational fully-autonomous ship. These researches fall into topics such as motion control of ASVs [2, 3], coordination between multiple ASVs [4, 5], and interaction of components inside ASVs [6].

One of the major challenges within the control of ASVs is the problem of uncertainties in the craft and its components model. Recently, several research works have been published to address this problem. In [7], a maneuvering model for a ship is extracted and an adaptive controller is implemented to control and estimate the ship parameters. In [8], the trajectory tracking problem is investigated using neural-adaptive control schemes where there exist several output constraints and parameter uncertainties in the craft model. A neural learning control strategy is adopted in [9] to guarantee trajectory tracking of an ASV with uncertainties in the model. In [10], a robust adaptive control strategy in combination with back-stepping and Lyapunov techniques is adopted to control the position of a ship in the presence of system uncertainties and unknown environmental disturbances. The use of fuzzy control approaches for adaptive track keeping is investigated in [11]. In [12], the performance of two different popular adaptive control algorithms for ASVs is compared where it is assumed that the vessel model is uncertain. Adoption of adaptive schemes for dynamic positioning is investigated in [13, 14] in the presence of uncertainty and unknown environmental disturbances. In [15], an adaptive scheme is proposed for pitch control of propellers to reduce fuel consumption. Adaptive control of ASVs with input constraints is investigated in [16]. Despite all these research projects, the use of adaptive control schemes in the maritime industry is still at its infancy.

The problem of uncertainty within propeller dynamics is not considered in any of the above works. It has been shown in the literature that the dynamics of propellers experience a relatively large amount of uncertainty during maneuvering of the vessel [17]. This makes the speed and position control of ASVs challenging. Considering the propellers shaft speed as the system input, the governing dynamical equation of the system is a non-affine in control system. As a result, the objective is to design a control algorithm that carries out the motion and position control of the ship by on-line approximation of propellers dynamics and handling hydrodynamical uncertainties within the vessel model.

The adaptive control of systems with uncertain dynamical models has received extensive consideration by the academic community in the recent decades. Among the diverse methodologies to control a nonlinear uncertain system using adaptive strategies, adaptive control using Neural Networks (NN) has been recognized as a feasible scheme where the unmodeled dynamics are estimated by NN [18, 19]. In most of the published researches, the common assumption is that the system is affine-in-control, i.e., the control inputs appear linearly in the dynamical equations of the system. Notable works are [20, 21] where the proposed methodologies are designed based on feedback linearization. The difficulty with the control of non-affine uncertain systems where the control input appears as a nonlinear function $g(\cdot)$, is that the inverse of $g(\cdot)$, in general, does not have an explicit form, even if its existence can be shown with the *Implicit Function Theorem* [22]. Several strategies to control such non-affine systems have been proposed in the literature. In [22] and [23], the non-affine control problem is converted into an affine-in-control problem by defining a new control input that contains an integrator. On the other hand, in [24], NN is used to design an inverse controller while in [25], the inversion error is approximated using NN. Moreover, time-scale separation methods were applied in [26, 27].

Considering the vessel model and by building up on our previous research results in [28], an adaptive control methodology for a class of non-affine-in-control systems is proposed where the unknown nonlinear influence of the system input is estimated using NN, so that the ASV can follow the given trajectory with the

desired speed. In this paper, it is also shown that the proposed methodology is capable of handling state dependent uncertainties within the hydrodynamical model of the ship. To achieve these goals, the results in [21] for adaptive control of affine-in-control systems are extended to control of partially unknown non-affine systems. An algorithm is proposed to address the problem of controlling a class of non-affine systems where the dynamics of the input function $g(\cdot)$ is unknown or uncertain where $g(\cdot)$ is the generated thrust by the propeller. By the adoption of NN, particularly the results in [29] and the Weierstrass approximation theorem [30], the inverse of $g(\cdot)$ is calculated and by adopting a control law the stability of the system is guaranteed. For the stability analysis, the Lyapunov technique as well as *Uniform Ultimate Boundedness* are employed and it is then shown that the reference trajectory tracking error converges to a residual set. The algorithm transforms the system to an affine-in-control system and then, by approximating $g^{-1}(\cdot)$, estimates the feasible control input. It is also shown that this strategy is capable of handling state dependent uncertainties within the hydrodynamical model of the ship. In order to evaluate the performance of the algorithm, several experiments are carried out. Based on actual Automatic Identification System (AIS) data received from the Port of Rotterdam Authority, a maneuvering experiment is carried out. It is assumed that the ship model embeds a Direct Current (DC) power and propulsion system [31] in order to assess the interaction of the proposed algorithm with the on-board power and propulsion system. Moreover, a dynamic positioning experiment and a circular trajectory tracking experiment are performed. Compared to [28], this paper contains several novelties. Some of the significant novel aspects are:

1. The problem of uncertainty in the propeller model is discussed in-detail and reasons behind the adoption of an adaptive control scheme for trajectory tracking are discussed.

2. A more complex ship model is considered which comprises all maneuvering model elements in 3 degrees of freedom.

3. The problem formulation and the correctness proof of the algorithm are

carried out for Multi-Input Multi-Output (MIMO) systems, while in [28] the problem formulation was carried out for Single-Input Single-Output (SISO) systems.

4. In this paper, it is also shown that this strategy is capable of handling state dependent uncertainties.

5. For evaluating the performance of the algorithm, more sophisticated scenarios are considered where a trajectory of a real ship in the port of Rotterdam is used as a reference trajectory. Moreover, the interaction of the proposed methodology with on-board power and propulsion system of ships is evaluated.

The remainder of this paper is organized as follows. In Section 2, the problem of uncertainty in propeller dynamics is explained. In Section 3, the overall system is described and the problem is formulated. In Section 4, the algorithm is presented and its proof of correctness is given. Simulation experiments and results are discussed in Section 5. Concluding remarks and future research directions are given in Section 6.

2. Propellers Model and The Problem of Uncertainty

The propellers and thrusters are the main components for the generation of required forces to propel a ship aligned to its given referenced trajectory. Based on the propeller model, the required forces can be determined by introducing a proper shaft speed to the propellers and thrusters. As a result, the propeller shaft speed is treated as the system input. These actuators are also the main link between on-board power and propulsion system and surrounding environment of the ship. The relationship between the propeller shaft speed and the generated thrust and torque can be established based on the following relationships [32]:

$$T_{\mathrm{p}} = K_{\mathrm{T}} \rho D^4 |n_{\mathrm{p}}| n_{\mathrm{p}} \tag{1}$$

$$Q_{\mathrm{p}} = K_{\mathrm{Q}} \rho D^5 |n_{\mathrm{p}}| n_{\mathrm{p}}, \tag{2}$$

where T_{p} is the generated thrust, Q_{p} is the generated torque, n_{p} is the propeller shaft speed, D is the propeller diameter, and ρ is the water density. Parameters K_{T} and K_{Q} are thrust and torque coefficients, which are functions of propeller structure and advance ratio J [33], defined as:

$$K_{\mathrm{T}} = f_{K_T}(J, P/D, A_{\mathrm{e}}/A_{\mathrm{o}}, Z, R_{\mathrm{n}}, t_{\mathrm{c}})$$

$$K_{\mathrm{Q}} = f_{K_Q}(J, P/D, A_{\mathrm{e}}/A_{\mathrm{o}}, Z, R_{\mathrm{n}}, t_{\mathrm{c}}),$$

where P/D is the pitch ratio, $A_{\mathrm{e}}/A_{\mathrm{o}}$ is the blade area ratio, Z is the number of propeller blades, R_{n} is the Reynolds number of a characteristic ratio and t_{c} is the ratio of maximum propeller thickness to the length of the cord at a characteristic radius. Moreover, the advance ratio is defined as:

$$J = \frac{V_{\mathrm{a}}}{n_{\mathrm{p}} D},$$

where V_a is the advanced speed that is the speed of water passing through propellers found using the following equation:

$$V_a = (1 - w)U, \tag{3}$$

with U the forward speed of the vessel and w the wake friction, depending on the shape of the hull.

Functions $f_{K_{\mathrm{T}}}$ and $f_{K_{\mathrm{Q}}}$ were estimated in [33, 34] in terms of very long and complex polynomials. However, typically, these functions are approximated using J and open water diagram where the performance of propellers are assessed, i.e., K_T and K_Q are functions of J. Figure 1 shows an open water diagram of a fixed pitch propeller belonging to the Wageningen B systematic series.

The modeling of propellers has always been a challenge in the maritime industry where a thorough model has not been proposed so far (for more information on this please refer to [17] and references therein). During maneuvering of a vessel, the propellers behave differently compared to when sailing straight. When a ship turns, due to the presence of lateral velocity, the inflow to the propellers is slanting and not axial. As a result, the advance ratio J will decrease

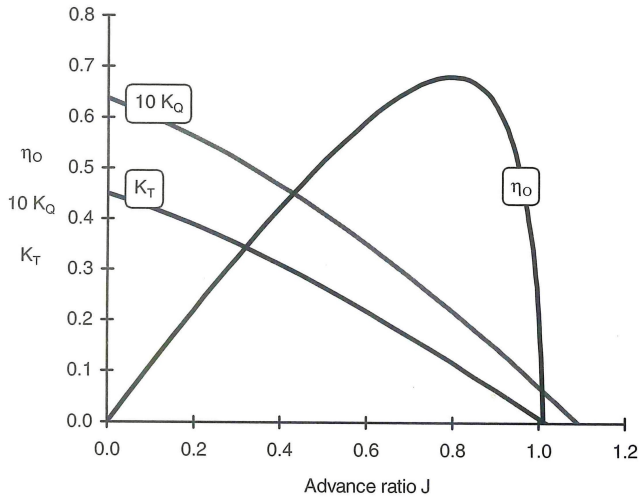


Figure 1: Open water diagram for Wageningen B 5 75 with pitch ratio 0.96 where η_o is the open water efficiency [35].

and more load is applied to propellers. Since, the open water diagram (and any other performance diagrams) is based on axial flow, they can not be used directly [36, 17]. Several analytical approaches have been proposed to solve this problem, however each of them contains a great amount of uncertainty.

Moreover, in a turn, the wake factor is also influenced. During straight courses, the wake is uniformly distributed but in a turn, the transversal velocity component is not dispersed uniformly and in the lower half of the propeller blade, the transversal velocity is way larger than the upper half [37, 17]. Figure 2 represents for a particular vessel the difference between the results of a propeller model and measured values [17], indicating the significant uncertainty in the model. In conventional ships, this problem might not be very critical since the control inputs are given by human operators. However in ASVs and during autopilot modes, this problem might result in inaccurate guidance. Since it has been shown in the literature that having an accurate and simple model for propellers is challenging, in this paper, the objective is to design an algorithm to control the ship maneuvering by on-line approximation of propellers dynamics.

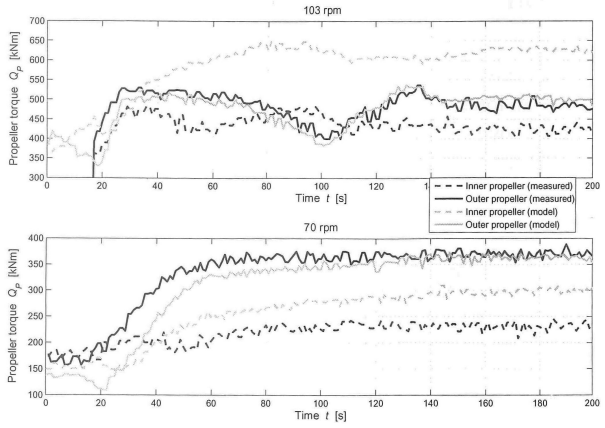


Figure 2: The difference between measured propeller torque and the outcome of the model during a turn [17].

3. System Description and Problem Formulation

In this section, the ship motion dynamics in 3 Degrees of Freedom (DoF) is presented where actuators (propellers and thrusters) shaft speeds are regarded as control input variables and ship position and speed are output variables. In this section, it is assumed that the relationship between propellers shaft speeds and generated torque and thrust is unknown. Then, the problem is formulated where the aim is to control the ship motion by estimating the propellers dynamics.

3.1. ASV Dynamics

In the context of this paper, the 3DoF motion of the ship is considered. The ship model can then be represented as:

$$\begin{aligned} &\dot{\eta}_s(t) = R\big(\eta_s(t)\big)V(t) \\ &M_s\dot{V}(t) + C_s\big(V(t)\big)V(t) + D_s\big(V(t)\big)V(t) = \tau_s, \end{aligned} \tag{4}$$

where $\eta_s(t) = [x_{\rm s}(t), y_{\rm s}(t), \delta_{\rm s}(t)]$ is a vector with the position and orientation of the ship at time t, $V(t) = [u(t), v(t), r(t)]^T$ is the 3DoF ship speed and τ_s is the vector of forces applied to the ship center of gravity. M_s is the Inertial Mass

matrix which consists of Rigid Body matrix M_{RB} and Added Mass matrix M_{A},

$$M_s = M_{\mathrm{RB}} + M_{\mathrm{A}} \tag{5}$$

where

$$M_{\mathrm{RB}} = \begin{bmatrix} m_{\mathrm{s}} & 0 & 0 \\ 0 & m_{\mathrm{s}} & m_{\mathrm{s}} x_{\mathrm{g}} \\ 0 & m_{\mathrm{s}} x_{\mathrm{g}} & I_{\mathrm{z}} \end{bmatrix}, M_{\mathrm{A}} = \begin{bmatrix} -X_{\dot{\mathrm{u}}} & 0 & 0 \\ 0 & -Y_{\dot{\mathrm{v}}} & -Y_{\dot{\mathrm{r}}} \\ 0 & -N_{\dot{\mathrm{v}}} & -N_{\dot{\mathrm{r}}} \end{bmatrix}. \tag{6}$$

m_{s} is the mass of the vessel, x_{g} is the distance between the center of gravity of the vessel to the center of the body-fixed coordinate frame.

$C_{\mathrm{s}}(\cdot)$ represents Coriolis and Centrifugal matrices which consists of rigid-body and added Coriolis and centripetal parts as:

$$C_{\mathrm{s}}(V) = C_{\mathrm{RB}}(V) + C_{\mathrm{A}}(V), \tag{7}$$

where

$$\begin{aligned} C_{\mathrm{RB}}(V) &= \begin{bmatrix} 0 & 0 & -m_{\mathrm{s}}(x_{\mathrm{g}} r + v) \\ 0 & 0 & m_{\mathrm{s}} u \\ m_{\mathrm{s}}(x_{\mathrm{g}} r + v) & -m_{\mathrm{s}} u & 0 \end{bmatrix} \\ C_{\mathrm{A}}(V) &= \begin{bmatrix} 0 & 0 & c_{13}(V) \\ 0 & 0 & c_{23}(V) \\ -c_{13}(V) & -c_{23}(V) & 0 \end{bmatrix}, \end{aligned} \tag{8}$$

with $c_{13}(V) = Y_{\dot{\mathrm{v}}} v + \frac{1}{2}(N_{\dot{\mathrm{v}}} + Y_{\dot{\mathrm{r}}})$ and $c_{23}(V) = -X_{\dot{\mathrm{u}}} u$.

The Damping matrix D_s is constructed by addition of two linear and non-linear matrices, i.e.,

$$D_{\mathrm{s}}(V) = D_{\mathrm{L}} + D_{\mathrm{NL}}(V) \tag{9}$$

where

$$D_{\mathrm{L}} = \begin{bmatrix} -X_{\mathrm{u}} & 0 & 0 \\ 0 & -Y_{\mathrm{v}} & -Y_{\mathrm{r}} \\ 0 & -N_{\mathrm{v}} & -N_{\mathrm{r}} \end{bmatrix}$$
$$D_{\mathrm{NL}}(V) = \begin{bmatrix} -d_{11}(V) & 0 & 0 \\ 0 & -d_{22}(V) & -d_{23}(V) \\ 0 & -d_{32}(V) & -d_{33}(V) \end{bmatrix}. \tag{10}$$

with $d_{11}(V) = X_{|u|u}|u| + X_{uuu}u^2$, $d_{22}(V) = Y_{|v|v}|v| + Y_{|r|v}|r|$, $d_{23}(V) = Y_{|v|r}|v| + Y_{|r|r}|r|$, $d_{32}(V) = N_{|v|v}|v| + N_{|r|v}|r|$ and $d_{33}(V) = N_{|v|r}|v| + N_{|r|r}|r|$. For more information on the model and the parameters, the reader is referred to [3, 7].

Matrix $R(\eta)$ is a Jacobian matrix that transforms ship velocity from body-fixed to inertial velocities, defined as:

$$R(\eta_s) = \begin{bmatrix} \cos(\delta) & -\sin(\delta) & 0 \\ \sin(\delta) & \cos(\delta) & 0 \\ 0 & 0 & 1 \end{bmatrix}, \tag{11}$$

in which δ is the ship heading angle, τ_s is the vector of forces applied to the ship center of gravity, i.e.,

$$\tau_s = \begin{bmatrix} \tau_x \\ \tau_y \\ \tau_\delta \end{bmatrix}, \tag{12}$$

where τ_{x} and τ_{y} are surge and sway forces and τ_δ is yaw moment, all applied to the gravity center of the ship.

For the sake of simplicity, it is assumed that the propellers are not rotatable. As a result, the relationship between the produced thrust by actuators (propellers and thrusters) and the vector of forces is [3]:

$$\tau_s = T_{3\times m} \begin{bmatrix} g_1(n_1) \\ \vdots \\ g_m(n_m) \end{bmatrix}, \tag{13}$$

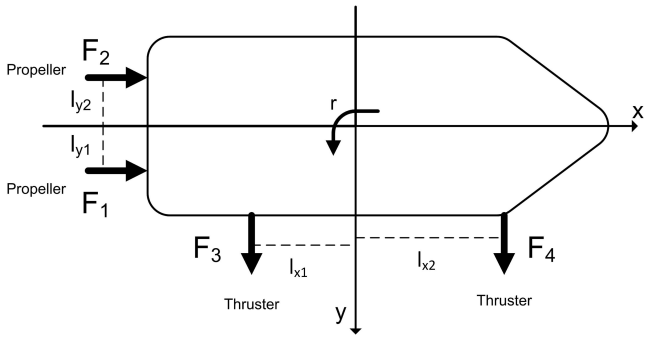


Figure 3: An ASV with two propellers (F_1 and F_2), one side thruster (F_3) and a bow thruster (F_4).

where $g_1, \ldots, g_m$ are actuators dynamics, $n_1, \ldots, n_m$ are actuators shaft speeds, m is the number of actuators, and T is the thrust configuration matrix defined as:

$$T = \begin{bmatrix} t_1 & \ldots & t_m \end{bmatrix}, \tag{14}$$

with t_1, t_2, ..., t_m column vectors for standard actuators. If the actuator is a propeller, then:

$$t_i = \begin{bmatrix} 1 \\ 0 \\ -l_{\mathrm{y}} \end{bmatrix}; \tag{15}$$

if the actuator is a stern or bow thruster, then:

$$t_i = \begin{bmatrix} 0 \\ 1 \\ l_{\mathrm{x}} \end{bmatrix}, \tag{16}$$

where l_y and l_x are actuator positions in the ASV reference frame (Figure 3). Since, generally, T is not a square matrix the solution to the problem of unconstrained thrust allocation to non-rotatable actuators can be found using the pseudo-inverse of T:

$$\tau_{\mathrm{ac}} = T^{\mathrm{T}}(TT^{-1})^{-1}\tau_{\mathrm{s}}. \tag{17}$$

where τ_{ac} is the vector of generated thrust by propelling actuators.

In this paper, it is assumed that $g_1, \ldots, g_m$ are unknown functions. For the algorithm design, the first step is to represent (4) in state space format. As a result, we have:

$$\begin{aligned} \dot{V} &= -M_{\mathrm{s}}^{-1}\big(C_{\mathrm{s}}(V)V + D_{\mathrm{s}}(V)V - \tau_{\mathrm{s}}\big) \\ \dot{\eta} &= T(\eta_{\mathrm{s}})V. \end{aligned} \tag{18}$$

Equation (18) can be rewritten in the following form:

$$\dot{x_{\mathrm{s}}} = f_{\mathrm{s}}(x_{\mathrm{s}}) + \begin{bmatrix} M_{\mathrm{s}}^{-1} \\ 0_{3\times 3} \end{bmatrix} \tau_s, \tag{19}$$

where $x_{\mathrm{s}} = \begin{bmatrix} v^T & \eta_{\mathrm{s}}^T \end{bmatrix}^T$ is the vector of states, $f : \mathbb{R}^6 \to \mathbb{R}^6$ is a nonlinear function. By combining (13) and (19) we obtain:

$$\dot{x_s} = f_s(x_s) + \begin{bmatrix} M_s^{-1}T \\ 0_{3\times m} \end{bmatrix} \begin{bmatrix} g_1(n_1) \\ \vdots \\ g_m(n_m) \end{bmatrix}, \tag{20}$$

$$\dot{x_s} = f_s(x_s) + g_s(u_s). \tag{21}$$

where $g_s : \mathbb{R}^m \to \mathbb{R}^6$ is a nonlinear function that contains the influence of input variables to the system and $u_{\mathrm{s}} = [n_1, n_2, ..., n_m]^T$ is the vector of actuators shaft speeds. In the remainder of this paper, (21) is considered as the dynamics of the ASV, a Multi-Input Multi-Output (MIMO) non-affine in control system.

3.2. Problem Formulation

Consider the following class of non-affine systems:

$$\dot{x}(t) = f(x(t)) + g(u(t)) + \omega(t), \tag{22}$$

where $x(t) \in \mathbb{R}^n$ is the state of the system, $u(t) \in \mathbb{R}^m$ is the system input, $\omega(t) \in \mathbb{R}^n$ is the disturbance applied to the system, $f : \mathbb{R}^n \to \mathbb{R}^n$ is a Lipschitz continuous nonlinear function and $g : \mathbb{R}^m \to \mathbb{R}^n$ is a nonlinear continuously differentiable function with $g_1(0) = 0, \cdots, g_n(0) = 0$. In the context of this paper, it is assumed that the function $g(\cdot)$ is unknown but satisfies the following assumption:

Assumption 1. *There exists a lower bound and an upper bound* γ_l, $\gamma_u \in \mathbb{R}$, *such that*

$$0 < \gamma_l < \left| J\Big(g\big(u(t)\big)\Big) \right| < \gamma_u \tag{23}$$

for all $t \geq 0$.

Using the *Implicit Function Theorem* and assumptions on $g(\cdot)$, the existence of $g^{-1}(\cdot)$ can be demonstrated [22]. The above assumptions on the system dynamics are moderately mild and can be concluded for broad classes of nonlinear systems [22, 38].

Assumption 2. *The overall disturbance acting upon the system is bounded, i.e., there exists* $\omega_M > 0$ *such that* $\|\omega(t)\| \leq \omega_M$ *for all* $t \geq 0$.

Suppose $x_R(t)$ is the desired trajectory of the system. Then, one can write the trajectory tracking error of the system as:

$$e(t) = x_R(t) - x(t). \tag{24}$$

The objective is to design an adaptive controller that adopts state feedback to ensure that $x(t)$ follows $x_R(t)$ for all $t > 0$.

4. The Adaptive Control Strategy

In this section, the proposed control strategy for the aforementioned class of non-affine systems is explained and the stability analysis and the proof of correctness are carried out.

4.1. Proposed Control Strategy

The control strategy is based on transforming the non-affine system to an affine nonlinear system and then, keeping $e(t)$ in a residual set by approximating $g^{-1}(.)$ and adopting a proper control law. Let

$$U(t) = g\big(u(t)\big), \tag{25}$$

where $U \in \mathbb{R}^n$ is treated as the control signal for the affine-in-control system, i.e.,

$$\dot{x}(t) = f(x) + U(t) + \omega(t). \tag{26}$$

Similarly as earlier research works in adaptive control (such as [19, 39]), we define the following control law for the above system:

$$U(t) = ke(t) - f(x), \tag{27}$$

where k is the controller gain, which will be determined below using a Lyapunov technique. By adopting the above control rule, it can be shown that (26) can follow the desired trajectory $x_R(t)$. However, in the problem considered in this paper, one of the main challenges is that $U(t)$ is not recognizable for the non-affine system (22), i.e., generally, $u(t)$ cannot be computed using $U(t)$. Therefore, the objective is to estimate $u(t)$ using the trajectory tracking error of the system and a well-tuned controller gain.

Based on the results in the literature [29] and similar to the methodology used in [21], feed-forward NNs with one hidden layer are capable of approximating any continuous function on a compact set, regardless of the nature of NN activation functions and input space dimensions. Assume $g^{-1}(\cdot)$ as the inverse of $g(\cdot)$. Let us define $g^{-1}(\cdot)$ as:

$$g^{-1}(U) = diag^{-1}\big(W^T\psi(U)\big) + \epsilon, \tag{28}$$

where $\psi(U) \in \mathbb{R}^{N\times n}$ is known as the vector of NN activation functions, $W \in \mathbb{R}^{N\times n}$ is the ideal approximation weight vector, ϵ is the approximation error and N is the number of neurons. In the presented methodology, the controller updates its set of weights $\hat{W}$ based on the tracking error $e(t)$ to approximate $g^{-1}(\cdot)$. As a result, at each time $t \geq 0$, the estimation of $g^{-1}(\cdot)$ can be written as:

$$\hat{g}^{-1}\big(U\big) = diag^{-1}\big(\hat{W}^T\psi(U)\big), \tag{29}$$

where $\hat{g}^{-1}(.)$ and $\hat{W}$ are estimates of $g^{-1}(\cdot)$ and W, respectively. The $diag(.)$

operator is defined as:

$$diag\big(A\big) = \begin{bmatrix} a_1 & 0 & ... \\ 0 & a_2 & ... \\ 0 & 0 & \ddots \end{bmatrix}$$

where $A = [a_1, a_2, ...]^T$ and $diag^{-1}\big(diag(A)\big) = A$. The error in the estimation of $g^{-1}(\cdot)$ can be defined as:

$$\tilde{g}^{-1}(U) = g^{-1}(U) - \hat{g}^{-1}(U) = diag\big(\tilde{W}^T\psi(U)\big) + \epsilon, \tag{30}$$

where

$$\tilde{W} = W - \hat{W} \tag{31}$$

is the weight approximation error. Furthermore, using (29), the error dynamics of system (22) can be determined as:

$$\dot{e}(t) = -\Big(f\big(x(t)\big) + g\big(diag^{-1}\big(\hat{W}^T\psi(U)\big)\big) + \omega(t) - \dot{x}_R(t)\Big). \tag{32}$$

Consider the following update rule for $\hat{W}$:

$$\dot{\hat{W}} = -\Gamma\psi(U)diag\big(e(t)\big) - \mu\Gamma\hat{W}, \tag{33}$$

where Γ is a diagonal $N \times N$ matrix with positive diagonal elements and $\mu \in \mathbb{R}$ is the NN tuning gain. The complete proposed adaptive control algorithm for the non-affine system (22) is described in Algorithm 1.

4.2. Stability Analysis and the Algorithm Design

In this section, the stability analysis of the algorithm is carried out. By employing uniform ultimate boundedness, it is shown that the error $e(t)$ converges to a residual set and states stay bounded for all $t \geq 0$.

Definition 1 (Uniform Ultimate Boundedness). *The solution to system (22) is Uniformly Ultimately Bounded with the ultimate bound $b \in \mathbb{R}$, if there exists a positive constant $c \in \mathbb{R}$, independent of $t_0 \geq 0$, and if for all $a \in (0, c)$, there is $\tau = \tau(a, b)$ such that:*

$$\|x(t_0)\| \leq a \Rightarrow \|x(t)\| \leq b, \forall t \geq t_0 + \tau.$$

Algorithm 1 *Adaptive Control Algorithm for Non-Affine Systems:*

Initialization: Obtain $x(0)$ and $x_R(t)$. Assign initial values to the elements in the vector of weights.

1: Calculate $e(t)$ using (24).

2: Compute U using (27) at each time t.

3: Estimate u by adopting (29).

4: Apply u to the system.

5: Update the vector of weights based on (33).

6: Obtain the state of the system and go to 1.

If the above statement holds for arbitrarily large a then the slution is Globally Uniformly Ultimately Bounded.

The above definition can be extended also to the trajectory tracking error $e(t)$. Indeed, our intention is to show that the error is uniformly ultimately bounded and that the state $x(t)$ is contained for all $t \geq 0$. Therefore, considering the boundedness of ϵ, i.e., there exists a positive real value ϵ_M such that $\epsilon(t) < \epsilon_M$ for all $t \geq 0$ [30], there exists a vector of activation functions $\psi(.)$ and a set of weights, both with dimension $N \times 1$, such that as $N \to \infty$, ϵ converges to zero [21, 29, 30].

Before presenting the main result of the paper, the following assumptions must be considered, in order to prove the correctness of Theorem 1.

Assumption 3. *The desired trajectory $x_R(t)$ and its derivative $\dot{x}_R(t)$ are bounded, i.e., there exists $x_M \in \mathbb{R}$ such that $\max\{|x_R(t)|, |\dot{x}_R(t)|\} \leq x_M$, for all $t \geq 0$.*

Assumption 4. *The elements in the vector of ideal weights W are bounded, i.e., there exists $W_M \in \mathbb{R}$ such that $\|W\| \leq W_M$.*

Assumption 5. *The NN activation functions are bounded. As a result, there is a positive real value ψ_M such that $\|\psi(.)\| \leq \psi_M$.*

It is worthy to mention that for designing the controller, having the knowledge over the bounds discussed in Assumptions 1-4 is not required.

Next, we analyze the stability of the proposed method and demonstrate the feasibility of the choices for control law (27) and the update rule for the NN weights (33).

Theorem 1. *Suppose the control and the NN weight update laws are:*

$$U = ke(t) - f(x(t))$$
$$\dot{\hat{W}} = \Gamma\psi(U)diag\big(e(t)\big) + \mu\Gamma\hat{W}.$$

If

$$k > \frac{\frac{1}{4}(M+1)^2\psi_M^2}{\mu}, \tag{34}$$

where M is the Lipschitz constant of $g(\cdot)$, then the trajectory tracking error $e(t)$ and NN weights estimation error $\tilde{W}$ are UUB and there exists a set of NN activation functions and a vector of weights with which the nonlinearities of $g^{-1}(.)$ can be approximated.

Proof. Consider the following Lyapunov function:

$$V = \frac{1}{2}e^Te + \frac{1}{2}Tr\Big(\tilde{W}^T\Gamma^{-1}\tilde{W}\Big) \tag{35}$$

with $Tr(\cdot)$ as the trace operator. Then, the derivative of V is:

$$\dot{V} = \dot{e}^Te + Tr\Big(\dot{\tilde{W}}^T\Gamma^{-1}\tilde{W}\Big). \tag{36}$$

From (32),

$$\dot{V} = -\Big(f\big(x(t)\big) + g\big(diag^{-1}\big(\hat{W}^T\psi(U)\big)\big) + \omega(t) - \dot{x}_R(t)\Big)^T e + Tr\Big(\dot{\tilde{W}}^T\Gamma^{-1}\tilde{W}\Big). \tag{37}$$

Since, in this paper, it is assumed that $f(x)$ is determined, using (25) and (27),

the above equation can be rewritten as:

$$\begin{aligned}\dot{V} =& -\Big(ke - g(u) + g\big(diag^{-1}\big(\hat{W}^T\psi(U)\big)\big) + \omega(t) - \dot{x}_R(t)\Big)^T e \\ &+ Tr\Big(\dot{\tilde{W}}^T\Gamma^{-1}\tilde{W}\Big) \\ =& -ke^Te + \Big(g(u) - g\big(diag^{-1}\big(\hat{W}^T\psi(U)\big)\big)\Big)^T e + \omega(t)^T e \\ &- \dot{x}_R(t)^T e + Tr\Big(\dot{\tilde{W}}^T\Gamma^{-1}\tilde{W}\Big).\end{aligned}$$

Using (33), we have:

$$\begin{aligned}\dot{V} =& -ke^Te + \Big(g(u) - g\big(diag^{-1}\big(\hat{W}^T\psi(U)\big)\big)\Big)^T e + \omega(t)^T e \\ &-\dot{x}_R(t)^T e + Tr\Big(\big(\psi(U)diag\big(e(t)\big) + \mu\hat{W}\big)^T\tilde{W}\Big)\end{aligned}$$

and by adopting (31),

$$\begin{aligned}\dot{V} =& -ke^Te + \Big(g(u) - g\big(diag^{-1}\big(\hat{W}^T\psi(U)\big)\big)\Big)e + \omega(t)^T e \\ &-\dot{x}_R(t)^T e + Tr\Big(\big(\psi(U)diag\big(e(t)\big) + \mu(W - \tilde{W})\big)^T\tilde{W}\Big).\end{aligned}$$

Taking into account the smoothness of $g(.)$ which indicates its Lipschitz continuity and Assumptions 2, 4 and 5, it can be concluded that

$$\begin{aligned}\dot{V} \leq -k\left\|e\right\|^2 + M\left\|e\right\|\left\|u - \hat{W}^T\psi(U)\right\|_F + \omega_M\left\|e\right\| + x_{\mathrm{M}}\left\|e\right\| \\ + \psi_M\left\|e\right\|\left\|\tilde{W}\right\|_F + \mu W_M\left\|\tilde{W}\right\|_F - \mu\left\|\tilde{W}\right\|_F^2,\end{aligned} \quad (38)$$

where M is the Lipschitz constant and $\|.\|_F$ is the Frobenius norm operator. Considering (28), one can rewrite the above equation as:

$$\begin{aligned}\dot{V} \leq& -k\left\|e\right\|^2 + M\left\|e\right\|\left\|W^T\psi(U) + \epsilon - \hat{W}^T\psi(U)\right\|_F \\ &+ \omega_M\left\|e\right\| + x_{\mathrm{M}}\left\|e\right\| + \psi_M\left\|e\right\|\left\|\tilde{W}\right\|_F \\ &+ \mu W_M\left\|\tilde{W}\right\|_F - \mu\left\|\tilde{W}\right\|_F^2\end{aligned} \quad (39)$$

$$\begin{aligned}\dot{V} \leq& -k\left\|e\right\|^2 + M\psi_M\left\|e\right\|\left\|\tilde{W}\right\|_F + M\epsilon_M\left\|e\right\| + \omega_M\left\|e\right\| \\ &+ x_{\mathrm{M}}\left\|e\right\| + \psi_M\left\|e\right\|\left\|\tilde{W}\right\|_F + \mu W_M\left\|\tilde{W}\right\|_F - \mu\left\|\tilde{W}\right\|_F^2.\end{aligned} \quad (40)$$

The above non-equality can be represented in matrix form, i.e.,

$$\dot{V} \leq -\begin{bmatrix} \|e\| \\ \left\|\tilde{W}\right\|_F \end{bmatrix}^T \begin{bmatrix} k & -\frac{1}{2}(M+1)\psi_M \\ -\frac{1}{2}(M+1)\psi_M & \mu \end{bmatrix} \begin{bmatrix} \|e\| \\ \left\|\tilde{W}\right\|_F \end{bmatrix} + \begin{bmatrix} M\psi_M + \omega_M + x_{\mathrm{M}} & \mu W_M \end{bmatrix} \begin{bmatrix} \|e\| \\ \left\|\tilde{W}\right\|_F \end{bmatrix} \quad (41)$$

which can be rewritten as:

$$\dot{V} \leq -z^T Q z + P z. \quad (42)$$

The necessary and sufficient conditions for correctness of $\dot{V} \leq 0$ are Q to be positive definite and

$$\|z\| > \frac{\|P\|}{\sigma_m(Q)} \quad (43)$$

where $\sigma_m(Q)$ is the minimum singular value of Q. For positive definiteness of Q,

$$k > \frac{\frac{1}{4}(M+1)^2\psi_M^2}{\mu}.$$

The minimum singular value Q can be calculated as:

$$\sigma_m(Q) = \frac{\sqrt{S_1 - S_2}}{2},$$

where

$$S_1 = k^2 + \frac{1}{2}(M+1)^2\psi_M^2 + \mu^2$$
$$S_2 = \sqrt{(k^2-\mu^2)^2 + (k+\mu)^2(M+1)^2\psi_M^2}.$$

For ease of calculation, take $\mu = k$. Then,

$$\sigma_m(Q) = k + \frac{1}{2}(M+1)\psi_M. \quad (44)$$

From (44) and (43),

$$\|z\| > \frac{M\psi_M + \omega_M + x_{\mathrm{M}} + \mu W_M}{k + \frac{1}{2}(M+1)\psi_M}. \quad (45)$$

Therefore, if

$$\left\|\tilde{W}\right\|_F > \frac{M\psi_M + \omega_M + x_{\mathrm{M}} + \mu W_M}{k + \frac{1}{2}(M+1)\psi_M} \quad (46)$$

or

$$\|e\| > \frac{M\psi_M + \omega_M + x_{\mathrm{M}} + \mu W_M}{k + \frac{1}{2}(M+1)\psi_M} \tag{47}$$

then (45) holds. (47) and (46) specify that e and/or $\tilde{W}$ will always converge to a residual set if (34) holds. Moreover, the size of the residual set can be decreased by increasing k. The above result indicates that e and $\tilde{W}$ are uniformly ultimately bounded. Therefore, it can be concluded that the state is bounded for all $t \geq 0$. Based on the results in [29, 30] there exists a set of activation functions and a vector of weights that can approximate the nonlinearities of $g^{-1}(\cdot)$.

Theorem 1 implies the correctness of the method and shows that trajectory tracking and weight estimation errors will converge to the set provided in (45) for all $t \geq 0$. Moreover, the size of the set can be reduced by increasing the controller gain.

There are several cases where the input function is partially known, i.e., it consists of a known part with an explicit inverse and an unknown part, such as vessels where the unknown part appears, mostly, during turns. As a result, the inverse of $g(.)$ can be written as:

$$g^{-1}(U) = g'^{-1}(U) + diag^{-1}\big(\hat{W}^T\psi(U)\big) \tag{48}$$

where $g'^{-1}(.)$ is the inverse of the known part of $g(.)$ and U is calculated using (27). It can be shown that Theorem 1 is extendable to this case.

Corollary 1. *With the control law and the NN weights update rule defined in Theorem 1, the unknown part of $g^{-1}(.)$ in (48) can be estimated and the trajectory tracking and the weight estimation errors are uniformly ultimately bounded.*

Proof. Taking into account the Lipschitz continuity of $g(.)$ and by combining (48) and (38), (39) can be concluded. The remainder of the proof is similar to the proof of Theorem 1.

Remark 1. *Since the weight matrix $\hat{W}$ is being updated online, the presented algorithm is capable of handling the possible changes that might happen in $g^{-1}(.)$*

during the ship operation. As a result, the algorithm can be used for both fixed pitch propellers and controllable pitch propellers.

4.3. The Case of State Dependent Uncertainty

In the previous sections, it is assumed that the knowledge over $f(x)$ is certain and there is no state dependent uncertainty in the system. However, in many applications, this is not the case, as the ship hydrodynamical model may face some degrees of uncertainty during sailing. Moreover, hyrodynamical modeling of ships for maneuvering purposes is a laborious process. In this section, it is shown that using the same strategy and by making a small change in the previously presented algorithm, state dependent uncertainties can be handled as well. This is also proved by presenting a theorem. For this purpose let us rewrite the governing equation of the system (22) as follows:

$$\dot{x}(t) = \hat{f}(x(t)) + g(u(t)) + \omega(t) + \omega_f\big(x(t)\big), \tag{49}$$

where $\hat{f}$ is an estimate of f (which is known) and ω_f is the state dependent uncertainty (that is unknown). It can be concluded that:

$$f(x(t)) = \hat{f}(x(t)) + \omega_f(x(t)). \tag{50}$$

The above equation indicates that $\omega_f(x(t))$ is also Lipschitz continuous. Similar to g^{-1}, let us introduce an approximation method for f, i.e.,

$$f(x(t)) = diag^{-1}\big({W_f}^T\psi(x(t))\big) + \epsilon_f \tag{51}$$

and

$$\hat{f}(x(t)) = diag^{-1}\big(\hat{W}_f^T\psi(x(t))\big), \tag{52}$$

where W_f is the approximation weight matrix, $\hat{W}_f$ is its estimate and ϵ_f is the estimation error. Then, similar as in the previous section, it can be deduced that,

$$\omega_f(x(t)) = diag\big(\tilde{W}_f^T\psi(x(t))\big), \tag{53}$$

where $\tilde{W}_f$ is the weight approximation error. As a result, the error dynamics are:

$$\dot{e}(t) = -\Big(f\big(x(t)\big) + g\big(diag^{-1}\big(\hat{W}_f^T\psi(x(t))\big)\big) + \omega_f\big(x(t)\big) + \omega(t) - \dot{x}_R(t)\Big). \quad (54)$$

Theorem 2. *Suppose that the adaptive control law for system (49) is:*

$$U = ke(t) - \hat{f}(x(t)). \quad (55)$$

Using the NN weights update rule (33) and the following update rule for $\hat{W}_f$*:*

$$\dot{\hat{W}}_f = -\Gamma_f\psi(x(t))diag\big(e(t)\big) - \mu_f\Gamma_f\hat{W}_f, \quad (56)$$

if

$$k > \frac{\frac{1}{4}(M+1)^2\psi_M^2}{\mu} + \frac{\psi_M^2}{\mu_f},$$

then the trajectory tracking error $e(t)$ *is UUB.*

Proof. Let us consider the following Lyapunov function:

$$V = \frac{1}{2}e^Te + \frac{1}{2}Tr\Big(\tilde{W}^T\Gamma^{-1}\tilde{W}\Big) + \frac{1}{2}Tr\Big(\tilde{W}_f^T\Gamma_f^{-1}\tilde{W}_f\Big) \quad (57)$$

After derivation we obtain:

$$\begin{aligned}\dot{V} = &-\Big(\hat{f}\big(x(t)\big) + g\big(diag^{-1}\big(\hat{W}_f^T\psi(x(t))\big)\big) + \omega_f(x(t)) \\ &+ \omega(t) - \dot{x}_R(t)\Big)^T e + Tr\Big(\dot{\tilde{W}}^T\Gamma^{-1}\tilde{W}\Big) \\ &+ Tr\Big(\dot{\tilde{W}}_f^T\Gamma_f^{-1}\tilde{W}_f\Big).\end{aligned} \quad (58)$$

Using a similar approach as for the proof of Theorem 1 and by adopting (53) and (55), the following relationship can be obtained:

$$\begin{aligned}\dot{V} \leq &-k\left\|e\right\|^2 + M\psi_M\left\|e\right\|\left\|\tilde{W}\right\|_F + \left(M\epsilon_M + \omega_M\right)\left\|e\right\| \\ &+ x_{\mathrm{M}}\left\|e\right\| + 2\psi_M\left\|e\right\|\left\|\tilde{W}_f\right\|_F + \psi_M\left\|e\right\|\left\|\tilde{W}\right\|_F \\ &+ \mu W_M\left\|\tilde{W}\right\|_F - \mu\left\|\tilde{W}\right\|_F^2 + \mu_f W_{f_M}\left\|\tilde{W}_f\right\|_F - \mu_f\left\|\tilde{W}_f\right\|_F^2.\end{aligned} \quad (59)$$

By representing the above inequality in matrix form, we have:

$$\dot{V} \leq -\begin{bmatrix} \|e\| \\ \left\|\tilde{W}\right\|_F \\ \left\|\tilde{W}_f\right\|_F \end{bmatrix}^T \begin{bmatrix} k & -\frac{1}{2}(M+1)\psi_M & -\psi_M \\ -\frac{1}{2}(M+1)\psi_M & \mu & 0 \\ -\psi_M & 0 & \mu_f \end{bmatrix} \begin{bmatrix} \|e\| \\ \left\|\tilde{W}\right\|_F \\ \left\|\tilde{W}_f\right\|_F \end{bmatrix} + \begin{bmatrix} M\epsilon_M + \omega_M + x_{\mathrm{M}} \\ \mu W_M \\ \mu_f W_{f_M} \end{bmatrix}^T \begin{bmatrix} \|e\| \\ \left\|\tilde{W}\right\|_F \\ \left\|\tilde{W}_f\right\|_F \end{bmatrix}. \quad (60)$$

which can be rewritten in the following form:

$$\dot{V} \leq -z_f^T Q_f z_f + P_f z_f. \quad (61)$$

If matrix Q_f is positive definite then $\dot{V} \leq 0$ holds. As a result,

$$k > \frac{\frac{1}{4}(M+1)^2\psi_M^2}{\mu} + \frac{\psi_M^2}{\mu_f}.$$

The remainder of the proof can be carried out with the same approach as used in the proof of Theorem 1.

It can be concluded from the above theorem that the overall system can be uncertain and that with a small change in Algorithm 1, using the same strategy, the state dependent uncertainties can also be handled. The proposed methodology for this case is represented in Algorithm 2. In the next section, the presented algorithm is applied to an ASV with unknown actuator dynamics and state dependent uncertainties.

4.4. Application to ASVs

In this part, the proposed adaptive control strategy is presented for control of ASVs with uncertainty in the maneuvering model and unknown propellers dynamics.

Suppose the desired trajectory, the initial position of the vessel, and its initial speed in 3DoF are denoted by $\eta_d(t)$, $\eta_s(0)$ and $V(0)$, respectively. If Δt is the

Algorithm 2 *Adaptive Control Algorithm for Non-Affine Systems with State Dependent Uncertainties:*

Initialization: Obtain $x(0)$ and $x_R(t)$. Assign initial values to the elements in the vector of weights.

1: Calculate $e(t)$ using (24).

2: Compute U using (55) at each time t.

3: Estimate u by adopting (29) and f by (52).

4: Apply u to the system.

5: Update the matrices of weights based on (33) and (56).

6: Obtain the state of the system and go to 1.

duration from one time step to the next, then the preferred speed of the vessel in its body-fixed coordinates can be calculated as:

$$V_d(t) = \frac{1}{\Delta t} R^{-1}(\eta_s)\big(\eta_d(t) - \eta_s(t)\big). \tag{62}$$

Using this, the speed error vector is found as:

$$e_{\mathrm{s}}(t) = V_d(t) - V(t). \tag{63}$$

By adopting (27) and (18), the control law is established as:

$$\tau_s = k e_{\mathrm{s}}(t) + M_s^{-1}\big(C_s(V)V + D_s(V)V\big). \tag{64}$$

If the system model contains state dependent uncertainty then:

$$\tau_s = k e_{\mathrm{s}}(t) - \hat{f}(x(t)). \tag{65}$$

The thrust allocation problem is solved using (17) with which the vector of desired forces generated by actuators is found, denoted by τ_d. Based on the length of the NN, the matrix of squashing functions $\psi(\tau_d)$ is computed. Note that the NN weight matrix $\hat{W}$ and $\psi(\tau_d)$ have similar sizes, i.e., $N \times m$, where m is the number of actuators. The estimated actuators shaft speeds are found

Algorithm 3 *Adaptive Control Algorithm for ASVs:*

Initialization: Obtain $\eta_s(0)$, $V(0)$ and $\eta_d(t)$. Initialize NN weight matrix.

1: Compute $V_d(t)$ using (62).

2: Calculate $e(t)$ by adopting (63).

3: By use of (64) and (65), compute the ship control law.

4: By exploiting (66), estimate the required actuators shaft speeds and apply them to the system.

5: Update the NN weights matrix.

6: Obtain the system states and go to 1.

as follows:

$$n = \begin{bmatrix} n_1 \\ \vdots \\ n_m \end{bmatrix} = diag^{-1}\big(\hat{W}^T \psi(\tau_d)\big). \tag{66}$$

After this step, the NN weight matrices are updated. The NN weight matrix update rules are regulated as below:

$$\begin{aligned} \dot{W} &= -\Gamma \psi(U) diag\big(T^T (TT^{-1})^{-1} e_{\mathrm{s}}(t)\big) - \mu \Gamma \hat{W} \\ \dot{\hat{W}}_f &= -\Gamma_f \psi(x(t)) diag\big(e_{\mathrm{s}}(t)\big) - \mu_f \Gamma_f \end{aligned} \tag{67}$$

The overall algorithm for the adaptive control of ASVs is presented in Algorithm 3.

Remark 2. *The proposed control approach in this paper requires position and speed information. However, this is not a disadvantage as speed information can be derived by taking the difference quotient of position information.*

5. Simulation Experiments and Evaluation Results

The chosen ASV for evaluating the performance of the algorithm is Cybership II from [7], which is a 1:70 scale replica of an Offshore Support Vessel. It

is assumed that the ASV has four actuators: two propellers, one stern thruster and a bow thruster as illustrated in Figure 3. As a result,

$$\tau_s = T_{3\times 4} \begin{bmatrix} K_{\mathrm{T}_1}\rho D_1^4 |n_{\mathrm{p}_1}| n_{\mathrm{p}_1} \\ K_{\mathrm{T}_2}\rho D_2^4 |n_{\mathrm{p}_2}| n_{\mathrm{p}_2} \\ K_{\mathrm{T}_3}\rho D_3^4 |n_{\mathrm{p}_3}| n_{\mathrm{p}_3} \\ K_{\mathrm{T}_4}\rho D_4^4 |n_{\mathrm{p}_4}| n_{\mathrm{p}_4} \end{bmatrix} \tag{68}$$

and

$$T = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ -0.1 & 0.1 & 0.2 & 0.5 \end{bmatrix}. \tag{69}$$

Note that the vector of actuator dynamics in (68) is unknown to the controller. Moreover, it is supposed that the knowledge over inertial mass, Coriolis and centrifugal and damping matrices are uncertain. The parameters of the model vessel are summarized in Table 1.

To assess the performance of the algorithm, three simulation scenarios are considered. The first and second scenarios are trajectory tracking scenarios and the third scenario is a dynamic positioning case. For these experiments the length of the NN is opted to be $N = 300$ and the chosen activation function with which the matrix of activation functions $\psi(.)$ is constructed, is as the following:

$$y = 0.05\left(\frac{1 - e^{-x}}{1 + e^{-x}}\right). \tag{70}$$

For all experiments, based on (34), $k = 500$ is considered for the control law and $\mu = 0.1$ and Γ is chosen to be an identity matrix. Simulations are carried out with a computer which has a core i7 2.6 GHz CPU and 8 GB of RAM.

Experiment I: Circular Trajectory Tracking

For the first experiment, the considered trajectory is assumed to be circular with the following specifications:

$$\eta_d(t) = \begin{bmatrix} \eta_{d_x}(t) \\ \eta_{d_y}(t) \\ atan2(\dot{\eta}_{d_x}, \dot{\eta}_{d_y}) \end{bmatrix} \tag{71}$$

Table 1: The model ASV parameters.

Parameter	Value	Parameter	Value
m	23.8	$Y_{\|r\|v}$	-0.805
x_g	0.046	$N_{\|r\|v}$	0.13
I_z	1.76	Y_r	-7.25
$X_{\dot{u}}$	-2	N_r	-1.9
$Y_{\dot{v}}$	-10	$Y_{\|v\|r}$	-0.845
$Y_{\dot{r}}$	0	$N_{\|v\|r}$	0.08
$N_{\dot{v}}$	0	$Y_{\|r\|r}$	-3.45
X_u	-0.722	$N_{\|r\|r}$	-0.75
Y_v	-0.889	K_{T_1}	0.08
$X_{\|u\|u}$	-1.327	K_{T_2}	0.08
$Y_{\|v\|v}$	-36.472	K_{T_3}	0.07
X_{uuu}	-5.866	K_{T_4}	0.07
N_v	0.03130	D_{p_1}	0.08
$N_{\|v\|v}$	3.956	D_{p_2}	0.08
ρ	1024	D_{p_3}	0.05
		D_{p_4}	0.05

$$\eta_{d_x}(t) = \alpha\cos(\tfrac{\beta t}{\alpha}), \quad \eta_{d_y}(t) = \alpha\sin(\tfrac{\beta t}{\alpha}) \tag{72}$$

where α and β are the radius of the circular trajectory and traveling speed, respectively. It is assumed that $V(0) = \begin{bmatrix}0, & 0, & 0\end{bmatrix}^T$, $\eta_s(0) = \begin{bmatrix}10, & 0, & 1.57\end{bmatrix}^T$, $\alpha = 10$ and $\beta = 0.2\ m/s$. Note that in this experiment the reference speed is constant.

The results for the circular trajectory tracking case are shown in Figure 4. It can be inferred from the figures that after the transient and training time of the NN that take few seconds, the ship can smoothly follow the reference trajectory and actuators generated thrust as well as ship speed converge to steady state values.

In this experiment, the proposed algorithm is compared with MIMO nonlinear PID control scheme [3] where the control law is:

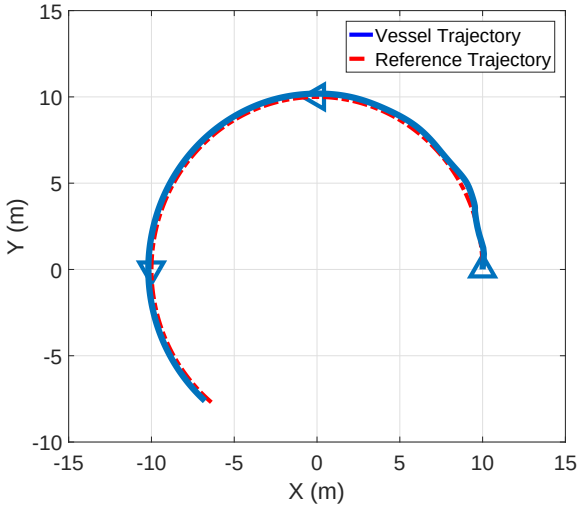
$$\tau = -K_{\mathrm{m}}\dot{V} + R^{-1}\big(\eta_s(t)\big)\tau_{\mathrm{PID}} \tag{73}$$

and

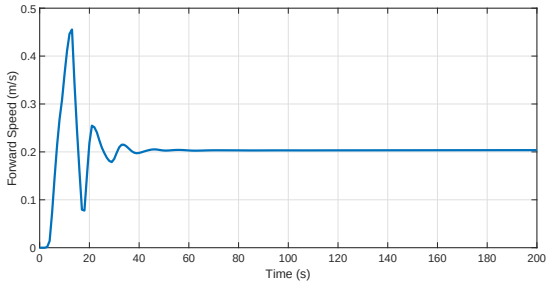
$$\tau_{\mathrm{PID}} = -K_{\mathrm{p}}(\eta_d - \eta) - K_{\mathrm{d}}\dot{\eta} - K_{\mathrm{i}}\int_0^t (\eta_d - \eta)d\tau. \tag{74}$$

Parameter K_{m} is the acceleration feedback. As suggested in [3], $K_{\mathrm{i}} = 0$. Other parameters are chosen as $K_{\mathrm{p}} = 0.8$, $K_{\mathrm{d}} = 1$ and $K_{\mathrm{m}} = 4$. As explained in Section 2, it is assumed that the precise knowledge over actuators model is not available during the operation. As a result, thrust coefficients are presumed to be $K_{T_1} = K_{T_2} = 0.12$ and $K_{T_3} = K_{T_4} = 0.1$. On the other hand, for the adaptive control simulations, it is assumed no knowledge about the model exist. The experiment results are shown in Figure 5. Simulation results are represented in terms of Root-Square Error (RSE). It can be inferred that by using the proposed methodology the ship can stay closed to the reference trajectory.

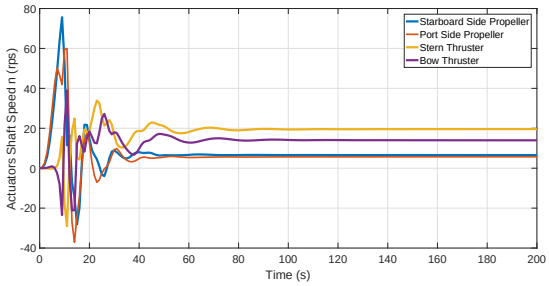
As mentioned in the previous section, as k increases the size of the residual sets (45) and (46) decreases which leads to the decrease in error. Figure 6 shows the value of RSE for different k values. It is seen that as k increases, the bounds of error decreases.



(a) The ship trajectory vs the reference trajectory.

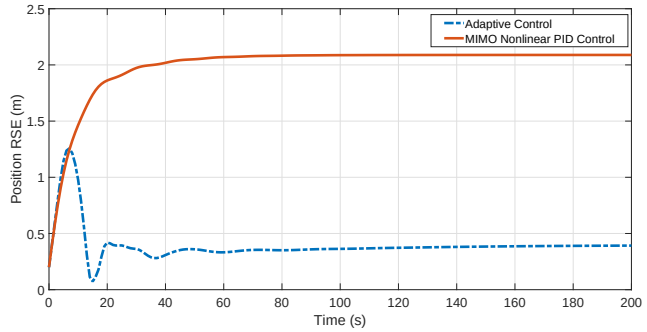


(b) Forward speed of the ship $(u(t))$.

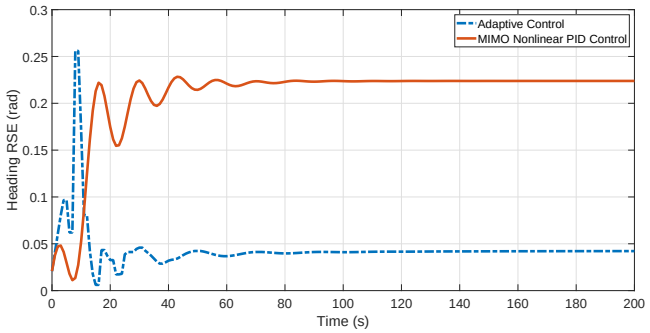


(c) Shaft speed of actuators $\big((n(t)\big)$.

Figure 4: Results of Experiment 2.



(a) Position RSE during the maneuver.



(b) Heading RSE during the maneuver.

Figure 5: Performance comparison of the proposed algorithm vs a conventional control scheme.

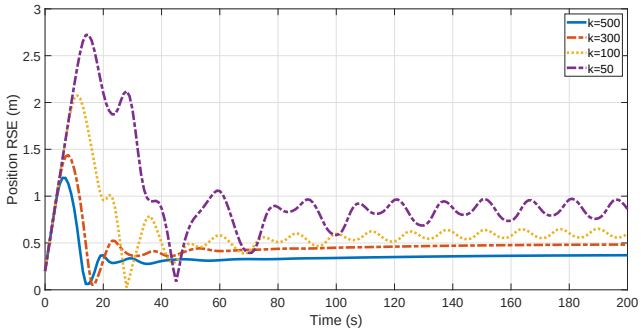


Figure 6: The effect of k on error bounds.

Experiment II: Dynamic Positioning

The second experiment is a dynamic positioning scenario where the ship has to maintain its position at $\eta_d(t) = \begin{bmatrix} 0, & 0, & 1.57 \end{bmatrix}^T$. Furthermore, it is assumed that there exists a current in the environment with the inertial velocities $V_c(t) = \begin{bmatrix} 0.1, & 0.1, & 0 \end{bmatrix}^T$.

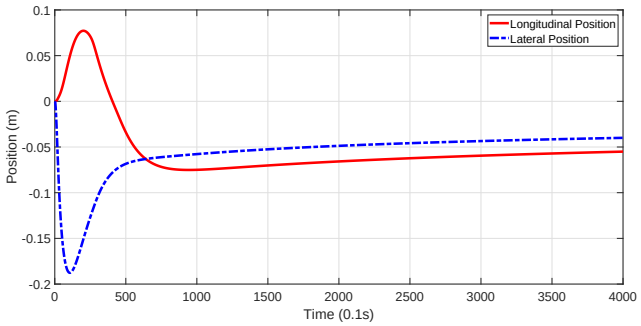
Figure 7 shows the experiment results. Similar to the previous case, the position of the ship is stabilized and actuators shaft speeds converge after the transient time and the training time of NN. This indicates that the NN-based adaptive controller succeeded in handling the uncertainties within propellers dynamics and the ship model.

Experiment III: Trajectory Tracking in The Port of Rotterdam

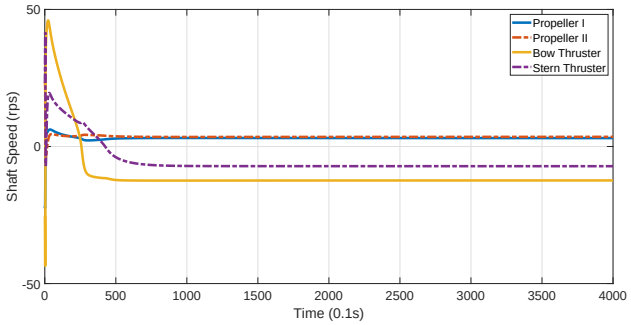
In the third experiment, the real trajectory of a vessel is considered in Oude Maas river in port of Rotterdam using AIS data received from the Port of Rotterdam authority. The considered path is the trajectory of an inland tanker vessel during two hours of voyage. Using Froude scaling the trajectory is scaled down to be aligned with the dimentions of the model ship with $C_{\text{Froude}} = 70$. During this voyage, the ship should sail with different course speeds. In simulations, it is also assumed that there is a stream in the river which applies force to the replica model ship hull. This force is considered to be $\tau_{\text{c}} = [0.1, -0.1, 0]^T$ in global reference frame. The trajectory of the ship is depicted in Figure 8.

The experiment results are shown in Figure 10. The trajectory tracking performance of the vessel is depicted in Figure 10a and the course speed of the vessel is compared with the scaled reference speed of the ship in Figure 10b. The applied thrust by the actuators are represented in Figure 10c. It is seen that after the transient and NN training time the ship can follow the planned trajectory.

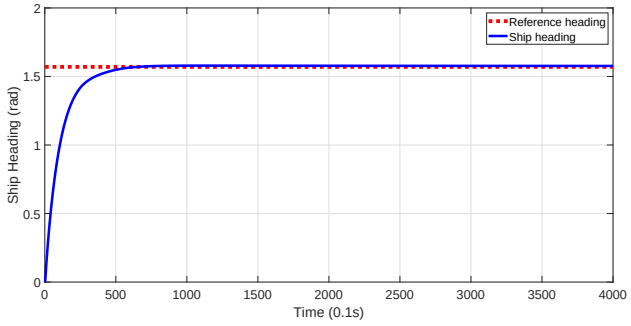
One of the main concerns regarding novel methods for trajectory tracking control of ships is the applicability of these algorithms to real ships and the interaction of the on-board power and propulsion system with the trajectory tracking algorithm. In this regard, the power and propulsion system should be



(a) Ship position keeping performance.



(b) Actuators shaft speeds $\big(n(t)\big)$.



(c) The ship heading $\big(\eta_\theta(t)\big)$.

Figure 7: Dynamic positioning performance of the ship.

able to generate requested thrust by the controller with a rough approximation. To examine this issue, a model of a power and propulsion system has been adopted. The applied thrust in Figure 10c is scaled up using Froude scaling to be fitting for a real size vessel and then, it is used as the reference thrust for the power and propulsion system. The propellers and thrusters of the on-board propulsion system should be able to follow the reference thrust roughly.

The architecture of the considered power and propulsion system is presented in Figure 9. The prime movers are connected to a DC-link through converters. The electric motors that rotate the actuators are fed and controlled by motor inverter-controllers. The reader is referred to [31] for more information regarding configuration and modeling of the power system. In this model, the propulsion drive-train specifications are as follows:

Port side and starboard side propellers: $K_{\mathrm{T}} = 0.8$, $K_{\mathrm{Q}} = 0.08$, $D = 2$m, 1.8 MW, 60 Hz, 460 v.

Bow and stern thrusters: $K_{\mathrm{T}} = 0.8$, $K_{\mathrm{Q}} = 0.08$, $D = 1$m, 500 kW, 60 Hz, 460 v.

Matlab Simscape toolbox is partially used for the modeling. Due to highly demanding data logging of this toolbox, the simulation can not be done for the whole voyage time which is approximately 6400 seconds. As a result, the focus is on period which fastest transients with highest peaks happen and in this case, this period is at the beginning of the simulation.

The simulation results are shown in Figure 11. Figures 11a to 11d show the generated thrust by the actuators vs the requested thrust by the controllers. The angular speed of electric motors is shown in Figure 12. The results suggest that the transients are traceable by the propulsion system and it can generate the requested thrust. Therefore, the algorithm is potentially applicable to real-size vessels.

Figure 8: The considered trajectory in the port of Rotterdam waterways.

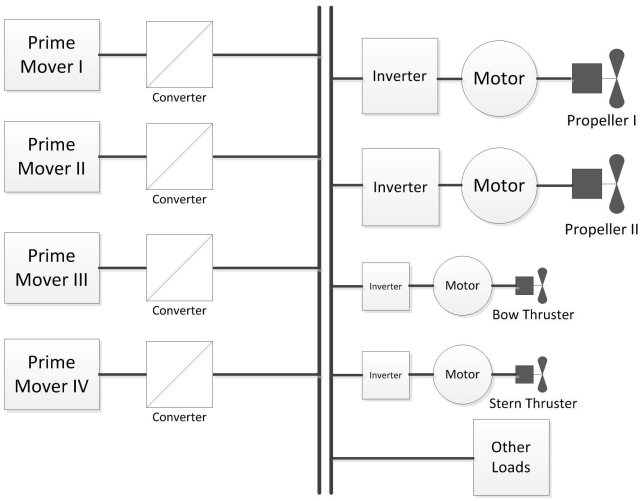
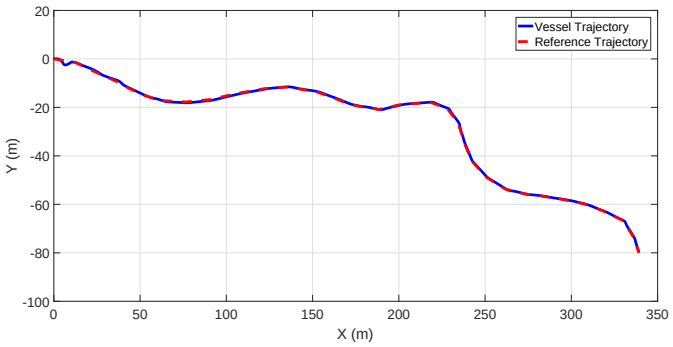
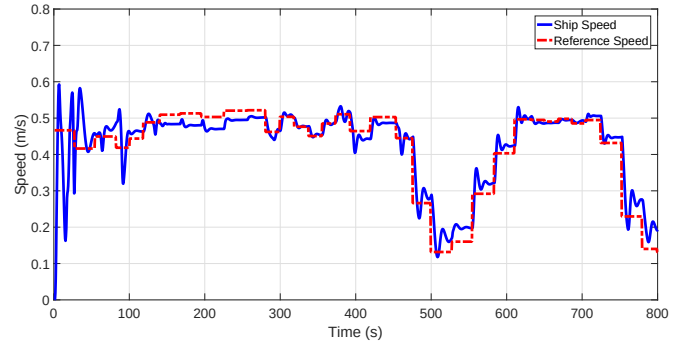


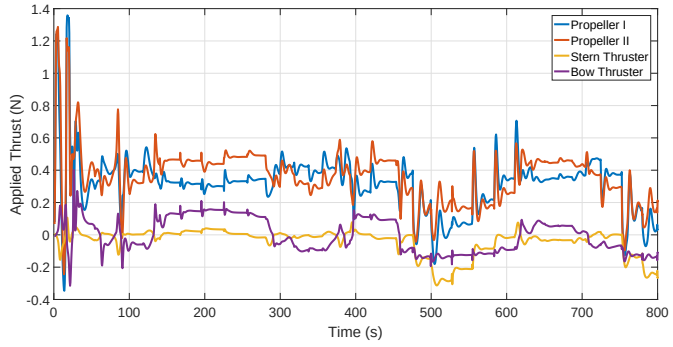
Figure 9: Architecture of the considered power system [31].



(a) Trajectory tracking result.

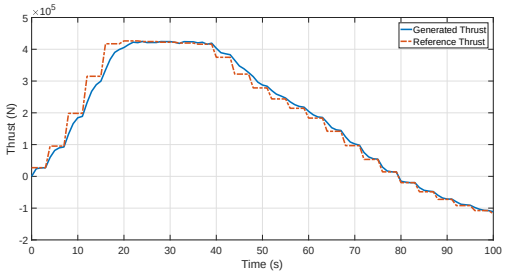


(b) Forward speed of the vessel ($u(t)$).

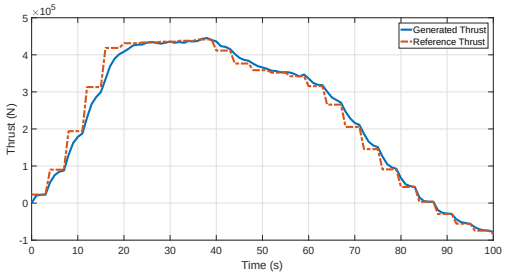


(c) Applied thrust by actuators ($\tau_{\text{act}}(t)$).

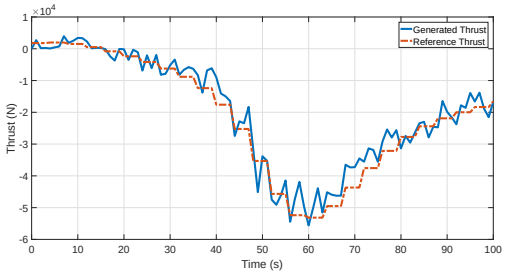
Figure 10: Simulation results of Experiment 1.



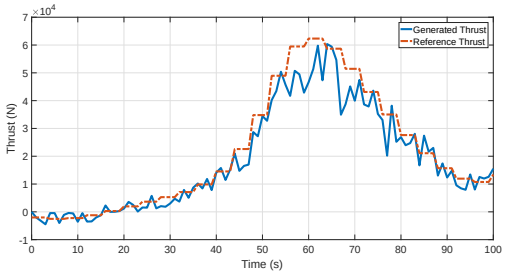
(a) Port side propeller: generated thrust vs requested thrust.



(b) Starboard side propeller: generated thrust vs requested thrust.



(c) Stern thruster: generated thrust vs requested thrust.



(d) Bow thruster: generated thrust vs requested thrust.

Figure 11: Performance of the power and propulsion system.

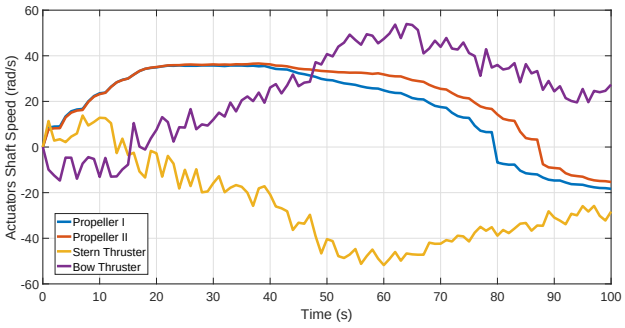


Figure 12: Angular speed of propellers and thrusters.

6. Conclusions

The propellers dynamics and hydrodynamical specifications of ASVs undergo sever uncertainties during maneuvering which makes the position and speed control of ASVs challenging. In this paper, a novel NN-based adaptive control algorithm has been proposed for motion and position keeping control of ASVs with unknown actuators dynamics and state dependent uncertainties. For the correctness proof of the algorithm, uniform ultimate boundedness, a Lyapunov technique and Weierstrass approximation theorem have been adopted. For the numerical analysis, three cases have been considered; trajectory following and dynamic positioning. It has been illustrated that the algorithm is successful in terms of keeping the overall system stable and fulfilling the objective of the operation.

Acquiring knowledge about the future state of the vessel is useful and favorable for the control of ASVs. The benefits of predictive control techniques for the control of ASVs are discussed in [31], [40], [41] where the prediction of future power demand would be advantageous for the power and propulsion system. Using this prediction, not only the energy conservation issues can be addressed [31] but also, it can lead to increased stability of the power and propulsion system. In this regard, the objective for the future researches is to combine this algorithm with receding horizon techniques in order to gain more accurate

predictions for the control of ASVs.

Acknowledgment

This research is supported by the project ShipDrive: A Novel Methodology for Integrated Modelling, Control, and Optimization of Hybrid Ship Systems (project 13276) of the Netherlands Organization for Scientific Research (NWO), domain Applied and Engineering Sciences (TTW).

The authors would like to thank the Port of Rotterdam Authority for delivering the AIS data and Daan de Boer for categorizing and arranging the data.

References

[1] Rolls-Royce, "Rolls-royce publishes vision of the future of remote and autonomous shipping. Accessed: January 2017," [Online]. Available: `http://www.rolls-royce.com/media/press-releases`

[2] H. Zheng, R. R. Negenborn, and G. Lodewijks, "Predictive path following with arrival time awareness for waterborne {AGVs}," *Transportation Research Part C: Emerging Technologies*, vol. 70, pp. 214 – 237, 2016.

[3] T. I. Fossen, *Handbook of Marine Craft Hydrodynamics and Motion Control*. Wiley, West Sussex, UK, 2011.

[4] H. Zheng, R. R. Negenborn, and G. Lodewijks, "Fast admm for distributed model predictive control of cooperative waterborne agvs," *IEEE Transactions on Control Systems Technology*, vol. PP, no. 99, pp. 1–8, 2016.

[5] Z. Peng, J. Wang and D. Wang, "Distributed Maneuvering of Autonomous Surface Vehicles Based on Neurodynamic Optimization and Fuzzy Approximation," in IEEE Transactions on Control Systems Technology, vol. 26, no. 3, pp. 1083-1090, May 2018.

[6] A. Haseltalab, R. R. Negenborn, and G. Lodewijks, "Multi-level predictive control for energy management of hybrid ships in the presence of uncertainty and environmental disturbances," vol. 49, no. 3, pp. 90 – 95, 2016, 14th IFAC Symposium on Control in Transportation Systems, Istanbul, Turkey, May, 2016.

[7] R. Skjetne, y. N. Smogeli, and T. I. Fossen, "A Nonlinear Ship Manoeuvering Model: Identification and adaptive control with experiments for a model ship," *Modeling, Identification and Control*, vol. 25, no. 1, pp. 3–27, 2004.

[8] Z. Zhao, W. He, and S. S. Ge, "Adaptive neural network control of a fully actuated marine surface vessel with multiple output constraints," *IEEE Transactions on Control Systems Technology*, vol. 22, no. 4, pp. 1536–1543, July 2014.

[9] S. L. Dai, M. Wang, and C. Wang, "Neural learning control of marine surface vessels with guaranteed transient tracking performance," *IEEE Transactions on Industrial Electronics*, vol. 63, no. 3, pp. 1717–1727, March 2016.

[10] M. Chen, S. S. Ge, B. V. E. How, and Y. S. Choo, "Robust adaptive position mooring control for marine vessels," *IEEE Transactions on Control Systems Technology*, vol. 21, no. 2, pp. 395–409, March 2013.

[11] J. Velagić, Z. Vukić and E. Omerdić, Adaptive fuzzy ship autopilot for track-keeping Control Engineering Practice, vol. 11, pp. 433-443, 2001.

[12] M. E. N. Sørensen and M. Breivik, "Comparing nonlinear adaptive motion controllers for marine surface vessels," vol. 48, no. 16, pp. 291 – 298, 2015.

[13] J. Du, X. Hu, M. Krstić and Y. Sun, Dynamic positioning of ships with unknown parameters and disturbances, Control Engineering Practice, vol. 76, pp. 22-30, 2018.

[14] J. Du, X. Hu, M. Krstić and Y. Sun, Robust dynamic positioning of ships with disturbances under input saturation, Automatica, vol. 73, pp. 207-214, 2016.

[15] R. D. Geertsma, K. Visser and R. R. Negenborn, "Adaptive pitch control for ships with diesel mechanical and hybrid propulsion Applied Energy", vol. 228, pp. 2490 - 2509, 2018.

[16] L. Liu, D. Wang, Z. Peng, C. L. P. Chen and T. Li, "Bounded Neural Network Control for Target Tracking of Underactuated Autonomous Surface Vehicles in the Presence of Uncertain Target Dynamics," in IEEE Transactions on Neural Networks and Learning Systems, vol. 30, no. 4, pp. 1241-1249, April 2019.

[17] P. Schulten, "The interaction between diesel engines, ship and propellers during manoeuvering," Ph.D. dissertation, Delft University of Technology, 2005.

[18] G. A. Rovithakis and M. A. Christodoulou, *Adaptive Control with Recurrent High-order Neural Networks.* Bristol, PA, USA: Springer-Verlag London, 2000.

[19] A. Das and F. L. Lewis, "Distributed adaptive control for synchronization of unknown nonlinear networked systems," Automatica, vol. 46, no. 12, pp. 2014 – 2021, 2010.

[20] A. Yesildirek and F. Lewis, "Feedback linearization using neural networks," Automatica, vol. 31, no. 11, pp. 1659 – 1664, 1995.

[21] F. L. Lewis, A. Yesildirak, and S. Jagannathan, *Neural Network Control of Robot Manipulators and Nonlinear Systems*, Taylor & Francis, Inc., Bristol, PA, USA, 1998.

[22] H. Nijmeijer and A. van der Schaft, *Nonlinear Dynamical Control Systems*, Springer-Verlag, New York, NY, USA, 1990.

[23] S. H. Lane and R. F. Stengel, "Flight control design using non-linear inverse dynamics," Automatica, vol. 24, no. 4, pp. 471 – 483, 1988.

[24] S. S. Ge and J. Zhang, "Neural-network control of nonaffine nonlinear system with zero dynamics by state and output feedback," *IEEE Transactions on Neural Networks*, vol. 14, no. 4, pp. 900–918, July 2003.

[25] A. J. Calise, N. Hovakimyan, and M. Idan, "Adaptive output feedback control of nonlinear systems using neural networks," Automatica, vol. 37, no. 8, pp. 1201 – 1211, 2001.

[26] N. Hovakimyan, E. Lavretsky, and A. J. Sasane, "Dynamic inversion for nonaffine-in-control systems via time-scale separation: part i," in *American Control Conference, 2005. Proceedings of the 2005*, June 2005, pp. 3542–3547 vol. 5.

[27] E. Lavretsky and N. Hovakimyan, "Adaptive dynamic inversion for nonaffine-in-control systems via time-scale separation: part ii," in proceedings of the *American Control Conference*, pp. 3548–3553 vol. 5, June 2005.

[28] A. Haseltalab and R. R. Negenborn, "Adaptive control for a class of partially unknown non-affine systems: Applied to autonomous surface vessels", In proceedings of IFAC world congress, vol. 50, no. 1, pp. 4252 – 4257, 2017, Toulouse, France, 2017.

[29] K. Hornik, M. Stinchcombe, and H. White, "Multilayer feedforward networks are universal approximators," *Neural Netw.*, vol. 2, no. 5, pp. 359–366, Jul. 1989.

[30] M. H. Stone, "The generalized weierstrass approximation theorem," Mathematics Magazine, vol. 21, no. 4, pp. 167–184, 1948.

[31] A. Haseltalab and R. R. Negenborn, "Predictive on-board power management for all-electric ships with dc distribution architecture," in proceedings of *Oceans Conference*, pp. 1–8, June 2017, Aberdeen, Scotland.

[32] R. Izadi-Zamanabadi and M. Blanke, "A ship propulsion system as a benchmark for fault-tolerant control," Control Engineering Practice, vol. 7, no. 2, pp. 227 – 239, 1999.

[33] D. K. P. Barnitsas, M.M.; Ray, "Kt, Kq and efficiency curves for the wageningen b-series propellers," Department of Naval Architecture and Marine Engineering, University of Michigan, Ann Arbor, Tech. Rep., 1981.

[34] P. van Oossanen and M. Oosterveld, "Further computer-analyzed data of the wageningen b-screw series," *International Shipbuilding Progress*, vol. Vol. 22, No. 251, 1975.

[35] H. K. Woud and D. Stapersma, Design of Propulsion and Electric Power Generation Systems, IMAREST, 2002.

[36] F. Gutsche, The Study of Ships Propellers in Oblique Flow,Volume 4306 of DRIC translation, Defence Research Information Centre, 1975.

[37] G. Kuiper, M. Grimm, B. McNeice, B. Noble, and M. Krikke, "Propeller inflow at full scale during a maneuver," in *24th Symposium on Naval Hydrodynamics* Fukuoka, Japan, 2002.

[38] J. D. Boskovic, L. Chen, and R. K. Mehra, "Multivariable adaptive controller design for a class of non-affine models arising in flight control," in proceedings of the 40th IEEE Conference on Decision and Control, vol. 3, 2001, pp. 2442–2447 vol.3.

[39] B. J. Yang and A. J. Calise, "Adaptive control of a class of nonaffine systems using neural networks," *IEEE Transactions on Neural Networks*, vol. 18, no. 4, pp. 1149–1159, July 2007.

[40] R. D. Geertsma, R. R. Negenborn, K. Visser, and J. Hopman, "Design and control of hybrid power and propulsion systems for smart ships: A review of developments," *Applied Energy*, vol. 194, pp. 30 – 54, 2017.

[41] H. Zheng, R. R. Negenborn and G. Lodewijks, "Predictive path following with arrival time awareness for waterborne AGVs," Transportation Research Part C: Emerging Technologies, vol. 70, pp. 214-237, 2016.