DRAFT
Proceedings of the ASME 2020
International Conference on Ocean, Offshore and Artic Engineering
OMAE2020
June 28 - July 3, 2020, Fort Lauderdale, FL USA

OMAE2020-19276

NONLINEAR AND MACHINE-LEARNING-BASED STATION-KEEPING CONTROL OF AN UNMANNED SURFACE VEHICLE

Armando J. Sinisterra, Alexandrea Barker, Siddhartha Verma, Manhar R. Dhanak

Florida Atlantic University, Boca Raton, FL, USA

ABSTRACT

This study is part of ongoing work on situational awareness and autonomy of a 16' WAM-V USV. The objective of this work is to determine the potential and merits of application of two different station-keeping controllers for a fixed-pose motion control of the USV. The assessment includes performance and power consumption metrics tested under harsh environmental disturbances to evaluate the robustness of the control methods. The first is a nonlinear trajectory-tracking control method based on the sliding-mode control technique, while the second method uses a machine-learning approach based on Deep Reinforcement Learning. Results from both the approaches are compared for various case studies.

Keywords: nonlinear control, deep reinforcement learning, machine-learning, station-keeping, marine robotics, artificial intelligence, unmanned surface vehicles, autonomous surface vehicles.

1. INTRODUCTION

Marine operations involving search-and-rescue, research-related surveys, structural inspections, surveillance and military missions, often pose a significant challenge and potential risks for human operators, especially in remote areas under adverse condition. For this reason the need for use of unmanned surface vehicles (USV) has been expanding recently. Moreover, the level of autonomy of such vessels is improving continually owing to technological advances and novel algorithms. These advances, which have been inherited primarily from technology used in autonomous ground vehicles, allow the USV to model the surrounding environment from data acquired using perception sensors such as LiDARs, RADARs, monocular and stereo cameras, etc., and decide on its response to various surrounding conditions, including obstacles (static and dynamic), environmental forces and illumination conditions among others. These decisions depend on the overarching mission of the vehicle, which can be broken down into a sequence of tasks consisting of processes such as world mapping (using information from the perception system), motion planning, and control of the dynamics of the vehicle. The present study focuses on this last aspect, particularly on station-keeping control, which consists of controlling a fixed pose of the USV using (in this particular case) two transom azimuth thrusters.

Effort is typically made to improve system identification procedures [1] in support of determining more accurate parameters for the estimated dynamic model, and obtain data about environmental conditions that affect the dynamics of the vehicle and incorporate them in the model, such as, for example, current, wind and wave forces [2]. Other situations may also affect the dynamics of the vehicle, especially ones considering time-varying properties associated with mass, and mass distribution, such as when the vehicle is loaded/offloaded. Different approaches to improve the controllability of a USV under the aforementioned situations have been developed; in [3] for example, a nonlinear observer is utilized to estimate the state of environmental forces and include them in the control system. In [4], on the other hand, due to the nature of their application, parameters based on the vehicle's displacement and drag are taken as time-varying variables with associated uncertainty, and thus, an adaptive controller approach is instead applied.

Here we explore two fundamentally different station-keeping controllers under the influence of harsh environmental disturbances, and characterize their performances and power consumption. The first method is a nonlinear sliding-mode trajectory-tracking controller, which is part of what is known as robust control [5]. This method allows the knowledge of the dynamic model to be imprecise in terms of the estimated rigid-body and hydrodynamic parameters associated with it, as well as due to unmodeled dynamics, which may be introduced by environmental disturbances. This imprecise (or simplified) model is then utilized to compute a nominal control law which along with an additional term (which deals with the uncertainty of the model) is able to conduct the USV to the desired state.

This method however, produces the required control forces and moment at a specific point in the USV, which is chosen to be the center of mass. These values need then to be relocated according to the actual geometric configuration of the actuators on the vehicle [6]. To this end, the locally-convex quadratic programming (QP) optimization method is utilized for the control forces and moment allocation, also known as the generalized force vector [7]. The second method is based on the deep reinforcement learning technique, which is a machine learning approach that allows the controller to determine an optimal policy via trial-and-error interactions with the environment. The technique consists of training the agent using the three degrees-of-freedom (3-DOF) equations of motion (EOM), so that it incrementally learns a control policy that maps a given state to an optimal set of actions [8] (throttle and steering of the thrusters) that conduct the vehicle to the next state. The autonomous low-level actions bring the vessel closer to the specified high-level goal, which is ultimately the station-keeping pose.

Both approaches utilize a 3-DOF EOM as described in [9] with some updated parameters. In order to test both control methods against inaccuracies in the dynamic model, two different sets of parameters for the EOM are defined: a true and an estimated set.

The estimated set is used to produce a nominal nonlinear control law and to train the agent, on the conventional and the machine-learning methods, respectively. The values of the parameters of the estimated set deviate from the ones in the true set in order to account for inaccuracies that are inherent to the implementation process, provided that in general the true set is unknown. However, the fact that these true values are known for this simulated experiment, allows us to characterize the response of both type of controllers.

2. WAM-V 16' USV

The wave adaptive modular vehicle (WAM-V) 16' USV is a light-weight catamaran (FIGURE 1) . It consists of a center-top tray that is connected to each pontoon through a set of front and rear articulated bars and a suspension system, which isolates the center tray from the motion induced from incident waves. This property makes the WAM-V ideal as a data acquisition platform, while its payload to weight ratio provides a suitable solution as a battery-powered system.

The WAM-V 16' USV is equipped with two 2 kW electric outboard motors and two corresponding linear actuators providing the vehicle with two-transom azimuth thrusters, each thruster constrained to a rotation of $\alpha = \pm 45°$.

2.1 Equations of Motion

The two control methods studied in this work, utilize the SNAME 1950 convention [10] for both body-fixed and inertial frames of reference. The dynamic model of the USV is provided in terms of the body-fixed coordinates considering only 3-DOF (surge, sway and yaw) as defined in (1) [11], where $\boldsymbol{M}$ in (2) is the inertia matrix corresponding to the rigid-body and added mass combined effects, similarly to $\boldsymbol{C}$ in (3) which is the Coriolis and centripetal terms matrix while matrix $\boldsymbol{D}$ in (4) encompass the drag terms. Vector $\boldsymbol{\tau}$ in (5) correspond to all the external forces, including propulsion and environmental disturbances.

$$\boldsymbol{M}\dot{\boldsymbol{\nu}} + \boldsymbol{C}(\boldsymbol{\nu})\boldsymbol{\nu} + \boldsymbol{D}\boldsymbol{\nu} = \boldsymbol{\tau} \quad \text{Where } \boldsymbol{\nu} = [u\ v\ r]^T \tag{1}$$

$$\boldsymbol{M} = \begin{bmatrix} m - X_{\dot{u}} & 0 & 0 \\ 0 & m - Y_{\dot{v}} & 0 \\ 0 & 0 & I_z - N_r \end{bmatrix} \tag{2}$$

$$\boldsymbol{C}(\boldsymbol{\nu}) = \begin{bmatrix} 0 & 0 & -(m - Y_{\dot{v}})v \\ 0 & 0 & (m - X_{\dot{u}})u \\ (m - Y_{\dot{v}})v & -(m - X_{\dot{u}})u & 0 \end{bmatrix} \tag{3}$$

$$\boldsymbol{D} = -\begin{bmatrix} X_u & 0 & 0 \\ 0 & Y_v & 0 \\ 0 & 0 & N_r \end{bmatrix} \tag{4}$$

$$\boldsymbol{\tau} = \begin{bmatrix} T_x\ T_y\ M_z \end{bmatrix}^T \tag{5}$$

The origin of the body-fixed coordinates is assumed to be at the USV's center of mass and the rigid-body and hydrodynamic parameters have been slightly updated from [9].

2.2 Estimated and True Parameters

In order to emulate an actual implementation of the methods developed in this work, two different sets of parameters are defined, as shown in TABLE I. The first is the estimated set of parameters, which is calculated by means of a system identification procedure [12] or/and the use of analytic techniques such as strip theory [13]. These values deviate from the real ones, which in general are unknown. However, for the purpose of this work, a second set called the true set of parameters is generated from the first set, by sampling from a normal distribution with a standard deviation of 30% for each corresponding estimated parameters. The second set define the true dynamics of the USV in this simulated scenario, while the first set is used to train the agent when developing the RL approach, as well as to define a nominal control law for the nonlinear controller. Both methods are tested against the real dynamics, that is, the true set of parameters in order to characterize the response of each controller.

3. NONLINEAR SLIDING STATION-KEEPING CONTROL

This approach consists of defining a control law which drives all system trajectories to converge to a time-varying sliding surface ($\boldsymbol{S}$) in finite time, also known as the reach time t_{reach}. Once on this surface, the system trajectories will remain there. This sliding surface can be interpreted as the tracking error of the system's state vector with respect to a desired state, and is defined in terms of the output of interest and its corresponding derivatives. The control law then has to be defined such that it ensures that the time derivative of the

Lyapunov function $V = 1/2\,\boldsymbol{S}^{\boldsymbol{T}}\boldsymbol{S}$ is negative in order to asymptotically reach stability of the control system.

TABLE I: RIGID BODY AND HYDRODYNAMIC (ESTIMATED AND TRUE) SET OF PARAMETERS

Parameters	m	I_z	$x_{\dot{u}}$	$y_{\dot{v}}$	$N_{\dot{r}}$	X_u	Y_v	N_r
Units	$\boldsymbol{kg}$	$\boldsymbol{kg.m^2}$	$\boldsymbol{kg}$	$\boldsymbol{kg}$	$\boldsymbol{kg.m^2}$	$\frac{\boldsymbol{kg}}{\boldsymbol{s}}$	$\frac{\boldsymbol{kg}}{\boldsymbol{s}}$	$\frac{\boldsymbol{kg.m^2}}{\boldsymbol{s}}$
Estimated parameters	280	239.37	-3.75	-33.52	-24.54	-20	-150	-0.30
True parameters	325.16	277.98	-3.14	-28.11	-20.58	-16.77	-125.81	-0.25

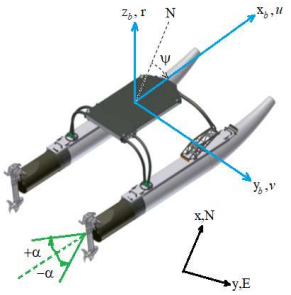

FIGURE 1: WAM-V BODY-FIXED COORDINATES AND THRUST CONFIGURATION

3.1 Sliding Surface

A sliding surface is defined according to (6) as:

$$\boldsymbol{S} = \left(\frac{d}{dt} + \boldsymbol{\Lambda}\right)^2 \int_0^t \boldsymbol{\eta_t} dt$$
$$\boldsymbol{S} = \dot{\boldsymbol{\eta}}_t + 2\boldsymbol{\Lambda}\boldsymbol{\eta_t} + \boldsymbol{\Lambda}^2 \int_0^t \boldsymbol{\eta_t} dt \quad (6)$$

Where $\boldsymbol{\eta_t}$ corresponds to the pose error of the USV with respect to a desired pose $\boldsymbol{\eta_d}$, that is $\boldsymbol{\eta_t} = \boldsymbol{\eta} - \boldsymbol{\eta_d}$ relative to the inertial frame of reference, and $\boldsymbol{\Lambda}$ is a design diagonal matrix with elements corresponding to time constants regarding exponential convergence of the state trajectory once it reaches the sliding surface.

3.2 EOM Relative to Inertial Frame

Provided the sliding surface function $\boldsymbol{S}$ has been defined with respect to the inertial frame of reference, the body-fixed EOM in (1) are redefined accordingly:

$$\boldsymbol{MJ^T\ddot{\eta} + CJ^T\dot{\eta} + DJ^T\dot{\eta} = \tau} \quad (7)$$

Where the expression $\boldsymbol{\nu = J^T(\eta)\dot{\eta}}$ that transforms the velocity vector from body-fixed to inertial coordinates, has been used along with the rotation matrix defined as:

$$\boldsymbol{J}(\psi) = \begin{bmatrix} cos\psi & -sin\psi & 0 \\ sin\psi & cos\psi & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Equation (7) can be arrange as:

$$\ddot{\boldsymbol{\eta}} = (\boldsymbol{J^{-T}M^{-1}})(\boldsymbol{\tau - EJ^T\dot{\eta}}) \quad (8)$$

Where $\boldsymbol{E = C + D}$. One can also rewrite (8) replacing $\boldsymbol{\eta}$ for $\boldsymbol{\eta_t}$ provided that in the case of station-keeping control $\dot{\boldsymbol{\eta}}_{\boldsymbol{d}} = \ddot{\boldsymbol{\eta}}_{\boldsymbol{d}} = \boldsymbol{0}$, as follows:

$$\ddot{\boldsymbol{\eta}}_{\boldsymbol{t}} = (\boldsymbol{J^{-T}M^{-1}})(\boldsymbol{\tau - EJ^T\dot{\eta}_t}) \quad (9)$$

3.3 Nominal Control Law

The expression "Once the system trajectory gets to the sliding surface it remains there" is equivalent to equating the time derivative of equation (6) to zero, that is:

$$\dot{\boldsymbol{S}} = \ddot{\boldsymbol{\eta}}_{\boldsymbol{t}} + 2\boldsymbol{\Lambda}\dot{\boldsymbol{\eta}}_{\boldsymbol{t}} + \boldsymbol{\Lambda^2\eta_t} = \boldsymbol{0} \quad (10)$$

Substituting equation (9) in (10) leads to:

$$\dot{\boldsymbol{S}} = \boldsymbol{J^{-T}M^{-1}\tau} + \boldsymbol{(2\Lambda - J^{-T}M^{-1}EJ^T)\dot{\eta}_t + \Lambda^2\eta_t} = \boldsymbol{0} \quad (11)$$

Solving for $\boldsymbol{\tau}$ one obtains the expression for the nominal control law:

$$\hat{\boldsymbol{\tau}} = -\boldsymbol{(\widehat{M}J^T2\Lambda - \widehat{E}J^T)\dot{\eta}_t - \widehat{M}J^T\Lambda^2\eta_t} \quad (12)$$

Where matrices $\widehat{\boldsymbol{M}}$ and $\widehat{\boldsymbol{E}}$ are formed using equations (2) to (4) and the estimated set of parameters in TABLE I. The total control law is then given by:

$$\boldsymbol{\tau = \hat{\tau} - ksat(S\Phi^{-1})} \quad (13)$$

Where the saturation function is defined as follows:

$$sat(\boldsymbol{S\Phi^{-1}}) = \begin{Bmatrix} \frac{S_i}{\Phi_i} & if\ S_i < \Phi_i \\ sgn(S_i)\ if\ S_i > \Phi_i \end{Bmatrix} \quad (14)$$

And thus, the term $\boldsymbol{ksat(S\Phi^{-1})}$ is defined as a vector with components $k_i * S_i/\Phi_i$ or $k_i sgn(S_i)$, depending if the state trajectory is inside or outside the bounded region defined by Φ_i.

3.4 Lyapunov Function and *k* Computation

Looking at equation (11) it results convenient to define the Lyapunov function as follows:

$$V = \frac{1}{2}\boldsymbol{S^T(J^{-T}M^{-1})^{-1}S} \quad (15)$$

Taking the time derivative of (15) and considering the *sliding condition* criteria in its vectorial form, one gets:

$$\dot{V} = \boldsymbol{S^T(J^{-T}M^{-1})^{-1}\dot{S}} \leq -\boldsymbol{|S^T|(J^{-T}\widehat{M}^{-1})^{-1}\Gamma} \quad (16)$$

Where $\boldsymbol{\Gamma}$ is a 3-by-1 vector with strictly positive constant elements of small magnitude.

Substituting (11) in (16), and considering the system trajectories out of the bounded region defined by $\boldsymbol{\Phi}$ one gets:

$$\dot{V} = \boldsymbol{S^T\{\hat{\tau} - Ksgn(S) + (MJ^T2\Lambda - EJ^T)\dot{\eta}_t + MJ^T\Lambda^2\eta_t\}} \leq -\boldsymbol{|S^T|(J^{-T}\widehat{M}^{-1})^{-1}\Gamma} \quad (17)$$

Substituting (12) in (17) and solving one gets:

$$\dot{V} = \boldsymbol{S^T\{(M - \widehat{M})J^T2\Lambda\dot{\eta}_t + (\widehat{E} - E)J^T\dot{\eta}_t + (M - \widehat{M})J^T\Lambda^2\eta_t - ksgn(S)\}} \leq -\boldsymbol{|S^T|(J^{-T}\widehat{M}^{-1})^{-1}\Gamma} \quad (18)$$

Where hat-values correspond to estimated quantities (with intrinsic inaccuracies with respect to the true values). Now we solve for $\boldsymbol{k}$ knowing that the product of $\boldsymbol{S^T k sgn(S)} = \boldsymbol{|S^T|k}$:

$$-|\boldsymbol{S^T}|\boldsymbol{k} \leq -|\boldsymbol{S^T}|\left(\boldsymbol{J^{-T}}\widehat{\boldsymbol{M}}^{-1}\right)^{-1}\boldsymbol{\Gamma} - \boldsymbol{S^T}\{(\boldsymbol{M}-\widehat{\boldsymbol{M}})\boldsymbol{J^T}\boldsymbol{2\Lambda}\left(\dot{\boldsymbol{\eta}}_t + \frac{\boldsymbol{\Lambda}}{\boldsymbol{2}}\boldsymbol{\eta}_t\right) + \left(\widehat{\boldsymbol{E}}-\boldsymbol{E}\right)\boldsymbol{J^T}\dot{\boldsymbol{\eta}}_t\} \quad (19)$$

We now solve for $\boldsymbol{k}$ by taking the absolute value of the second term at the right-hand side of (19), and then multiplying the entire inequality by $|\boldsymbol{S^T}|^{-1}$:

$$\boldsymbol{k} \geq \widehat{\boldsymbol{M}}\boldsymbol{J^T}\boldsymbol{\Gamma} + \boldsymbol{\mathcal{M}}|\boldsymbol{J^T}|2\boldsymbol{\Lambda}\left|\dot{\boldsymbol{\eta}}_t + \frac{\boldsymbol{\Lambda}}{\boldsymbol{2}}\boldsymbol{\eta}_t\right| + \boldsymbol{\mathcal{E}}|\boldsymbol{J^T}\dot{\boldsymbol{\eta}}_t| \quad (20)$$

Where $\boldsymbol{\mathcal{M}} = \left|\boldsymbol{M}-\widehat{\boldsymbol{M}}\right|$ and $\boldsymbol{\mathcal{E}} = |\widehat{\boldsymbol{E}}-\boldsymbol{E}|$. Provided that in this simulated scenario the true parameters of the EOM have been generated from the estimated set of parameters by sampling from a normal distribution with a standard deviation of 30% the values of the estimated parameters, the values for matrices $\boldsymbol{\mathcal{M}}$ and $\boldsymbol{\mathcal{E}}$ are precisely defined as the absolute value of the corresponding estimated matrices $\widehat{\boldsymbol{M}}$ and $\widehat{\boldsymbol{E}}$, multiplied by 0.3.

3.5 Stability Analysis

The total control law in (13) is derived based on the sliding condition in (16). Integration of the latter verifies that the trajectory reaches the corresponding sliding surface in (6), in finite time when $S_i > 0$ as shown below:

$$\left(\boldsymbol{J^{-T}}\boldsymbol{M^{-1}}\right)^{-1}\dot{\boldsymbol{S}} \leq -\left(\boldsymbol{J^{-T}}\widehat{\boldsymbol{M}}^{-1}\right)^{-1}\boldsymbol{\Gamma}$$

$$\boldsymbol{M}\int_{\boldsymbol{0}}^{tr}\dot{\boldsymbol{S}}dt \leq -\widehat{\boldsymbol{M}}\int_{0}^{tr}\boldsymbol{\Gamma}dt$$

Solving the inequality for the reach time one gets:

$$\frac{\boldsymbol{\Gamma}}{|\boldsymbol{\Gamma}|^2}\widehat{\boldsymbol{M}}^{-1}\boldsymbol{MS}(0) \geq \boldsymbol{t_r}$$

Similarly when $S_i < 0$ one obtains:

$$-\frac{\boldsymbol{\Gamma}}{|\boldsymbol{\Gamma}|^2}\widehat{\boldsymbol{M}}^{-1}\boldsymbol{MS}(0) \geq \boldsymbol{t_r}$$

Therefore, one can rewrite the final expression for the reach time as:

$$\left|\frac{\boldsymbol{\Gamma}}{|\boldsymbol{\Gamma}|^2}\widehat{\boldsymbol{M}}^{-1}\boldsymbol{MS}(0)\right| \geq \boldsymbol{t_r}$$

3.6 Constrained Nonlinear Iterative Control Allocation Using Quadratic Programming

The generalized force vector computed according to (13) is located at the center of mass of the USV, which in general differs from the actual location of the thrusters in a marine craft. The goal with control allocation is to produce a set of thruster actions (forces) in order to obtain, as close as possible, an equivalent effect in the dynamics of the vehicle.

Control allocation of azimuth thrusters in marine crafts imposes a nonconvex optimization problem which can be reformulated as a locally convex QP optimization problem, as defined in equation (21), similarly to as in [7].

Equation (21) represents a cost function which is to be minimized by optimizing the values of the variables in vector $[\Delta\boldsymbol{f}, \Delta\boldsymbol{\alpha}, \boldsymbol{s}]$, where $\boldsymbol{f} = \boldsymbol{f}_o + \Delta\boldsymbol{f}$ corresponds to the current force vector which includes all active thrusters in the vehicle's configuration, and is defined as the sum of the last force $\boldsymbol{f}_o$ and the increment $\Delta\boldsymbol{f}$ computed after optimization. Similarly, the current azimuth angles are defined according to $\boldsymbol{\alpha} = \boldsymbol{\alpha_o} + \Delta\boldsymbol{\alpha}$, where $\boldsymbol{\alpha_o}$ corresponds to the previous sample while $\Delta\boldsymbol{\alpha}$ ensures the azimuth angles do not deviate more than what the actuator can achieve within a sample time. This electromechanical limitation is addressed according to constraint (25). Finally, vector $\boldsymbol{S}$ (which is different from $\boldsymbol{S}$ in Section 3), is just a slack variable which accounts for the difference (error) between the goal (generalized force vector) and the control forces and moment achieved after control allocation. An entire description of the control allocation implemented for this particular vehicle and thrust configuration is found in [14].

$$C = \min_{\Delta\boldsymbol{f}, \Delta\boldsymbol{\alpha}, \boldsymbol{s}}\left\{\Delta\boldsymbol{f}^T\boldsymbol{P}\Delta\boldsymbol{f} + \boldsymbol{s}^T\boldsymbol{Q}\boldsymbol{s} + \Delta\boldsymbol{\alpha}^T\boldsymbol{\Omega}\Delta\boldsymbol{\alpha} + \mathrm{f}_o^{\mathrm{T}}2\boldsymbol{P}\Delta\boldsymbol{f} + \frac{\partial}{\partial\boldsymbol{\alpha}}\left(\frac{\varrho}{\varepsilon + \det(\boldsymbol{T}(\boldsymbol{\alpha})\boldsymbol{T}^T(\boldsymbol{\alpha}))}\right)_{\boldsymbol{\alpha}_o}\Delta\boldsymbol{\alpha}\right\} \quad (21)$$

Subject to:

$$\boldsymbol{s} + \boldsymbol{T}(\boldsymbol{\alpha}_o)\Delta\boldsymbol{f} + \frac{\partial}{\partial\boldsymbol{\alpha}}(\boldsymbol{T}(\boldsymbol{\alpha}_o)\boldsymbol{f})|_{\boldsymbol{\alpha}_o\boldsymbol{f}_o}\Delta\boldsymbol{\alpha} = \boldsymbol{\tau} - \boldsymbol{T}(\boldsymbol{\alpha}_o)\boldsymbol{f}_o \quad (22)$$

$$\boldsymbol{f}_{min} - \boldsymbol{f}_o \leq \Delta\boldsymbol{f} \leq \boldsymbol{f}_{max} - \boldsymbol{f}_o \quad (23)$$

$$\boldsymbol{\alpha}_{min} - \boldsymbol{\alpha}_o \leq \Delta\boldsymbol{\alpha} \leq \boldsymbol{\alpha}_{max} - \boldsymbol{\alpha}_o \quad (24)$$

$$\Delta\boldsymbol{\alpha}_{min} \leq \Delta\boldsymbol{\alpha} \leq \Delta\boldsymbol{\alpha}_{max} \quad (25)$$

4. DEEP REINFORCEMENT LEARNING

4.1 Deep Reinforcement Learning

Reinforcement learning (RL) [8] is a process by which an 'agent' (in this case, an autonomous USV) learns to earn rewards through trial-and-error interactions with an 'environment'. This trial-and-error nature allows the agent to operate with virtually complete autonomy, and to adapt effectively to unforeseen circumstances. This characteristic is vital when operating in unfamiliar surroundings, or in continually changing environmental conditions.

The 'environment' in the present study is represented by the set of ordinary differential equations in (1) that describe the 3-DOF of the USV, in addition to any imposed external disturbances. The agent is provided with a high-level goal, which in our case is to hold a specified goal pose (station-keeping), and it determines optimal control-responses at any given time based on its 'state'. In the present study, the state vector consists of 10 quantities of interest: radial distance from target coordinates; deviation from the desired orientation; the yaw angle in the inertial reference frame (ψ); the longitudinal, lateral, and rotational velocities in the inertial frame ($\dot{x}, \dot{y}, \dot{\psi}$); the port and starboard thrust values (Tport, Tstarboard); and the port and starboard motor azimuth angles (δport, δstarboard). This set of variables allows us to comprehensively describe the state of the USV within its environment (i.e., a testing arena, or the open ocean).

The control-responses determined by the RL algorithm are referred to as 'actions', and consist of 4 variables that control the maneuvering of the USV, namely, 2 thrust values for the port and starboard motors, and their respective azimuth angles. Whenever the agent performs an action by selecting appropriate values for these 4 variables, it transitions to a new state, and receives a 'reward' based on whether it is closer to achieving its specified high-level objective. We note that when designing a problem using RL, shaping the reward function has the most significant influence on the overall behavior of the agent, especially since we do not manage the low-level decision-making process of the agent directly. The mapping between the agent's states and the optimal actions can be achieved using either a tabulated formulation, or with the help of Artificial Neural Networks (ANN). Further details regarding the algorithm and the training procedure are provided below.

4.2 The Bellman Equation

The trial-and-error nature of RL implies that initially the agent is unable to make useful decisions. As the agent starts interacting with its environment in a random fashion, it will happen upon certain actions that improve its chances of collecting a high long-term reward. Hundreds to thousands of successive experiments are then run, which allows the agent to improve its decision-making capabilities continually using a combination of exploration and exploitation. The training procedure depends on the Bellman equation, which aims to maximize the total cumulative reward received throughout an experiment:

$$V^{\pi^*}(s_t) = \max_{a_t} \left(r(s_t, a_t) + \gamma \sum_{s_{t+1}} P(s_{t+1}|a_t, s_t)\, V^{\pi^*}(s_{t+1}) \right)$$

The training is assumed to be complete when V^{π^*} converges, i.e., it no longer changes with further training. Here, π^* represents the optimal 'policy' that encodes the behavior of the agent, and γ represents the discount factor, which emphasizes long-term rewards over short-term benefits. The emphasis on long-term reward is an important distinguishing feature of RL, since it allows the agent the freedom to choose actions that may be detrimental in the immediate future, but which may prove to be the best possible choice for maximizing the reward received over the long term. In the present work, the value $\gamma=0.99$ is used for all training runs.

To determine the optimal policy π^* during training, the agent observes the state of the environment s_t at every new turn, and performs an action a_t. $P(s_{t+1}|a_t, s_t)$ denotes the probability that this particular action will cause the agent to a transition to a new state s_{t+1}. The action thus influences both the transition to the next state and the reward received r_{t+1}. In basic RL algorithms such as Q-Learning or DQN (Deep Q Networks), the agent's goal is to learn the optimal control policy $a_t = \pi^*(s_t)$ that maximizes the Value function $V^{\pi^*}(s_t)$. More recent RL algorithms rely on two independent networks for encoding the optimal actions and the Value functions, and are referred to as actor-critic methods [15]. The specific RL algorithm used for training in the present work is referred to as 'RACER', and the relevant details may be found in ref. [16].

4.3 The Training Procedure

One of the primary tasks in RL is to determine the optimal policy π^* that guides the agent's actions. More specifically, the policy can be thought of as a multivariate function, where the state-variables serve as inputs, and the actions are the outputs. Thus, the task of training an agent involves determining a suitable functional-approximator for the policy, which relates the input state-values to the most appropriate action-values. When considering relatively simple problems, trained policies may take the form of 'tables', where the best action value for every possible combination of states-actions is tabulated in a grid. This approach is feasible when dealing with a small number of states and actions, and especially when working with discrete values of these variables. However, such tables quickly become unwieldy when considering a large number of action- and state-variables, or when non-discrete (continuous) values are used for these variables. In the current scenario, we are concerned with 10 distinct state-variables and 4 action-variables, which vary continuously between specified physical bounds. Thus, ANNs are more suitable to use as functional approximators instead of a tabulated approach. The ANN used in this study consists of an input layer, an output layer, and 3 hidden-layers, with each hidden-layer being comprised of 128 nodes.

For training the agent, the USV is initialized with randomized initial position, velocity, and yaw angle within a square box of size 40m centered on the target station-keeping point. The initial yaw angle is sampled from a uniform distribution in $[-\pi, \pi]$. The randomized initialization of the state-variables is vital to prevent 'overfitting' of the ANN, and to ensure that the learned policy is sufficiently general to be effective when the agent starts with different initial conditions. Furthermore, randomly fluctuating environmental forces are imposed in both the North and East directions during training, with the force components sampled from a Gaussian distribution with mean 0 *N* and standard deviation 300 *N*. This allows the agent to learn how to adapt effectively to unforeseen environmental disturbances.

During training, both the state vector and the corresponding reward are communicated from the environment to the agent at pre-determined time-intervals (0.1s in our case). The agent then uses this information to update the parameters of the ANN (the weights and the biases), such that the mean-squared error based on the Bellman equation is minimized. The ANN also outputs the 4 action values which are communicated to the environment such that the simulation may proceed forward in time by integrating the equations of motion (1), and the agent ends up in a new state. This process repeats continually until the training terminates. At each communication step, the agent evaluates the reward it receives, which may be a combination of the various objectives that the craft must attain. For instance, in our case, the craft's objective is to navigate to the station-holding target point, assume the specified heading, and minimize power consumption in the process. To achieve the first two goals, the agent is

assigned an increasingly negative award the more it deviates from the target coordinates and angle. More specifically, the agent is allotted a reward of $-100(\Delta r + \Delta\theta)$ at every turn, where Δr represents the radial distance from target, and $\Delta\theta$ represents the heading error. Furthermore, high rotation rates are punished by allocating a reward of $-100|\dot{\psi}|$ at evert turn. To minimize power consumption, a stepwise reward of -2/3P(T) (where P(T) is described in Eq. (26)) is given to the agent. Once the agent manages to reach within a radius of 0.1L (10% of the craft's length) from the target point, we keep a record of the radial and heading errors, which may fluctuate in time due to the presence of environmental disturbances. At the end of each episode, the mean square of these errors is assigned as a negative terminal reward, which encourages the agent to minimize long-term oscillations in the desired parameters. The duration of each simulation episode is 70*s*.

We note that in order to speed up training, we also limit the maximum radial distance that the agent can traverse; if the agent exceeds a radial distance of 30 meters from the target point, the training-simulation terminates and the agent receives a large negative terminal reward (-4e5). This large punishment discourages the agent from exploring regions that are far away from the target point, and results in shorter training durations. The general training approach adopted here implies that the optimal policy produced from a single training campaign can be used for all of the case studies described in the Results section.

5. METHODOLOGY

This section describes all the simulation and electromechanical parameters, as well as the environmental forces affecting the response of both type of controllers in every test run. The simulation time step has been set to 0.01 seconds for the nonlinear controller, which is equivalent as to having a data acquisition system streaming sensor data (such as position and heading of the USV) at a frequency of 100 Hz, which corresponds to the capacity of our actual equipment to be used in future implementations. The time step size used for advancing the equations of motion in the RL simulations is 0.001*s*, but the control interval is 0.1*s*.

The electromechanical constraints of each of the transom thrusters are given by its thrust limits [-250, 350] *N* (according to bollard-pull test in [17], and azimuth rotation limits [-45°, 45°]. The thrust rate is modelled by low-pass filtering the thrust values in order to obtain a more realistic propulsive response, as described in [14]. The azimuth rotation rate, on the other hand, is limited to rotate at a maximum of 0.45° per sample, that is equivalent to $0.079rad/0.01s = 0.79rad/s$, this is according to experimental data performed over the actual thrusters. This value per sample of 0.079 rad is then used to define $\Delta\boldsymbol{\alpha}_{min}$ and $\Delta\boldsymbol{\alpha}_{max}$ in equation (25).

The propulsive power consumption is computed at each sample according to the follow expression in [18]:

$$P(T) = (P_{max} - P_{min})\left(\frac{|T|}{T_{max}}\right)^{\eta} + P_{min} \tag{26}$$

Where,
$P_{max} = 1120\ W$, max. propulsive power per motor
$P_{min} = 0\ W$;
$T_{max} = 350\ N;\ T_{min} = -250\ N$;
$1.3 \leq \eta \leq 1.7$, typically.

A quadratic polynomial fit of equation (26) is performed in order to compute the entries for the two dimensional matrix $\boldsymbol{P}$ in (21).

A constant wind force of +50 *N* in the y direction is added throughout the entire test run, and superimposed during the duration of an additional wind gust. This wind gust is introduced as an environmental perturbation at time 40 *s*, and it will last for the following 10 *s*, according to the expression below:
$wind\_gust = |\ A * \sin(2\pi/T * time)\ |$,
where A = 300*N* and T = 20*s*. Moreover, the wind gusts are applied at an angle of 60° according to the *NED* coordinates convention, leading to the most critical condition in the *y*-direction (east).

6. RESULTS

Characterization of the time response of both control systems is done considering three different case scenarios, which differ not only with respect to the distance from the initial to the goal pose, but also on the kind of behavior expected from the controllers. Below is a description of each of the cases as pictured in FIGURE 2:

a) Near-field station keeping control: The initial and goal poses are given by $(0, 0, 5\pi/4)$ and $(3, 1, \pi)$, respectively. This case is intended to evaluate the performance of each of the two controllers at close proximity to the goal. The USV is expected to rotate in a counter-clockwise direction in order to achieve the goal, since it leads to lower power consumption.

b) Mid-field station keeping control: The initial and goal poses are given by $(0, 0, \pi)$ and $(10, 0, \pi)$, respectively. This case is intended not only to evaluate the performance of each controller at a medium distance from the goal, but also to propel a particular response which conducts the USV in a full-reverse motion, without altering the heading along the entire trajectory (the heading will only be affected by the wind perturbations).

c) Far-field station keeping control: The initial and goal poses are given by $(0, 0, \pi)$ and $(19, 5, \pi/4)$, respectively. This case is intended to evaluate the performance of each controller at a large distance from the goal. The USV is expected to rotate in a counter-clockwise direction in order to achieve the goal, since it leads to lower power consumption.

A set of time-domain specifications are used in order to characterize the response of both controllers, explained as follows. The rise time t_r, is considered as the time the controller takes to conduct the USV to 90% the desired value of a particular control variable, that is x, y or ψ. The rise value is computed according to the following expression:
$V_{rise} = (V_{desired} - V_{initial}) * 0.9 + V_{initial}$

The peak time t_p and the peak overshoot M_p are considered at the maximum value (or minimum when it's a negative

overshoot) of a particular control variable, before the wind gust appears. The peak overshoot M_p is the magnitude of this deviation with respect to the desired value of a control variable.

The settling time t_s is considered to be after the initial peak overshoot (for underdamped cases) whenever the value of the control variable is first within a 5% deviation with respect to the desired value, remaining in this region from that point on. This criteria however sometimes needed to be adjusted provided the response of the control system settled at a value beyond the 5% tolerance, which happened for example for the *y* variable in the near-field case (TABLE II), were 7% was used instead.

The wind gust peak M_g is the magnitude of the maximum deviation with respect to the desired value of a control variable, caused by the wind gust.

6.1 Nonlinear Station Keeping Control

This robust control based approach allows for uncertainty in the dynamic model of the USV and its associated physical parameter. In this particular case, the rigid-body and hydrodynamic parameters are allowed to deviate from the true values by up to 30%, as explained in Section 3.4, relative to TABLE I. Parameters for the total control law in (13) were defined as follows: $\boldsymbol{\Lambda} = \mathrm{diag}(1.8, 1.8, 1.8)$, $\phi_i = 0.05$ for all i in (14) and $\boldsymbol{\Gamma} = [0.02, 0.02, 0.02]^T$ in (20).

Inspecting at FIGURE 2 and TABLE II, we observe, in general, a smooth response of the control system with some overshooting and rather small oscillations for all case scenarios. The rise time reveals a swift reaction of the controller in order to conduct the USV to its desired state, while the settling time t_s is achieved just a few seconds later. Also, the steady-state error *SSE* is low for all control variables, except perhaps for the *y* coordinate in the far-field case where this error is around 8.7 cm, which is still very low. The highest deviations with respect to desired values are produced at the initial transient response of the control system, defined by the value of M_p, but is then swiftly decreased from that point forward, even in the presence of high wind gusts, as confirmed by the values of M_g. This means that the controller successfully rejects the harsh environmental disturbances.

FIGURE 3 shows the response of the thrusters over time, for the last and most critical case scenario in terms of thruster activity and saturation of the motors and actuators, which happens in the far-field run, as can be verified by its cumulative power value in (fourth column). It is easy to see from this figure, a high level of thruster activity (in terms of thrust values and azimuth rotations) at the initial transient response and when the wind gust appears (at 40 seconds), and correspondingly, high propulsive power consumption.

One can also verify from FIGURE 2 the expected counter-clockwise rotations for cases a) and c), and the full reverse motion in case b), as was expected in the first place, in order to economize the power consumption of the USV.

6.2 RL-Based Control

The results from the machine-learning based approach are compared in this section against the performance of the non-linear controller. Just as in the previous case, the rigid-body and hydrodynamic parameters differ from the true values by up to 30. In FIGURE 4 and TABLE IV, we observe that an initial overshoot and gradual oscillation are present for all the scenarios tested, as confirmed by the peak overshoot values M_p. The rise times (t_r) for the three cases are under 10 seconds, including the critical far-field case. With the constraint of a 5 percent error the settling times t_s , were large or nonexistent in all cases; however, the settling times recorded in TABLE IV use the larger criteria of 10 percent for the *x* and ψ, the variables which can then be seen to settle within 12 seconds . We note that there is a significant offset in the *y* direction which indicates that the autonomous controller has difficulty in maintaining the specified pose. The source of this error was the constant force applied in the *y*-direction during prediction runs, which was not included when training the RL agent. Moreover, the oscillatory behavior of the solution can also be attributed to the application of the large constant force. We confirmed both of these occurrences using prediction runs with no constant forcing applied, which yielded minimal steady-state error and no oscillations. The steady-state error *SSE* was estimated from the difference of the desired value and mean of values between the peak time t_p and the introduction of the wind gust. From this estimation, the errors as given in TABLE IV are generally greater than desired, particularly for the *y* coordinate, as discussed. At the critical instance of the wind gust, the system response deviates from the desired values, as evidenced by the wind gust peak values M_g.

The port, starboard and cumulative propulsive power for each of the cases are shown in TABLE V. Notably the far-field case has the lowest power consumption despite the greatest initial distance to traverse. The time-response of the thrusters for the far-field case is shown in FIGURE 5, with respect to azimuth angle, thrust, and power consumption. The thrust and power have large values during the initial displacement and the wind gust, but both these values and the azimuth angle display oscillatory behavior over the entire time period. This can again be attributed to the presence of the constant force that was not included during training runs, and causes the controller to keep correcting continually, but inefficiently. We expect that the performance will improve substantially when the training parameters are expanded to include scenarios involving constant currents.

7. CONCLUSIONS

Inspecting the performance of both type of controllers in Figures 2 and 4, it is noticeable their potential to control the dynamics of the WAM-V 16' USV for the station-keeping operation. They both showed a quick response to conduct the vehicle near the desired state, in less than 10s according to t_r, for all case scenarios. The nonlinear controller, however, proved to be more stable, showing very little oscillations, as opposed to the RL controller which showed pronounced oscillations, especially for the *x* variable in the near- and middle- field trajectories. This behavior of the RL controller can be explained due to the fact

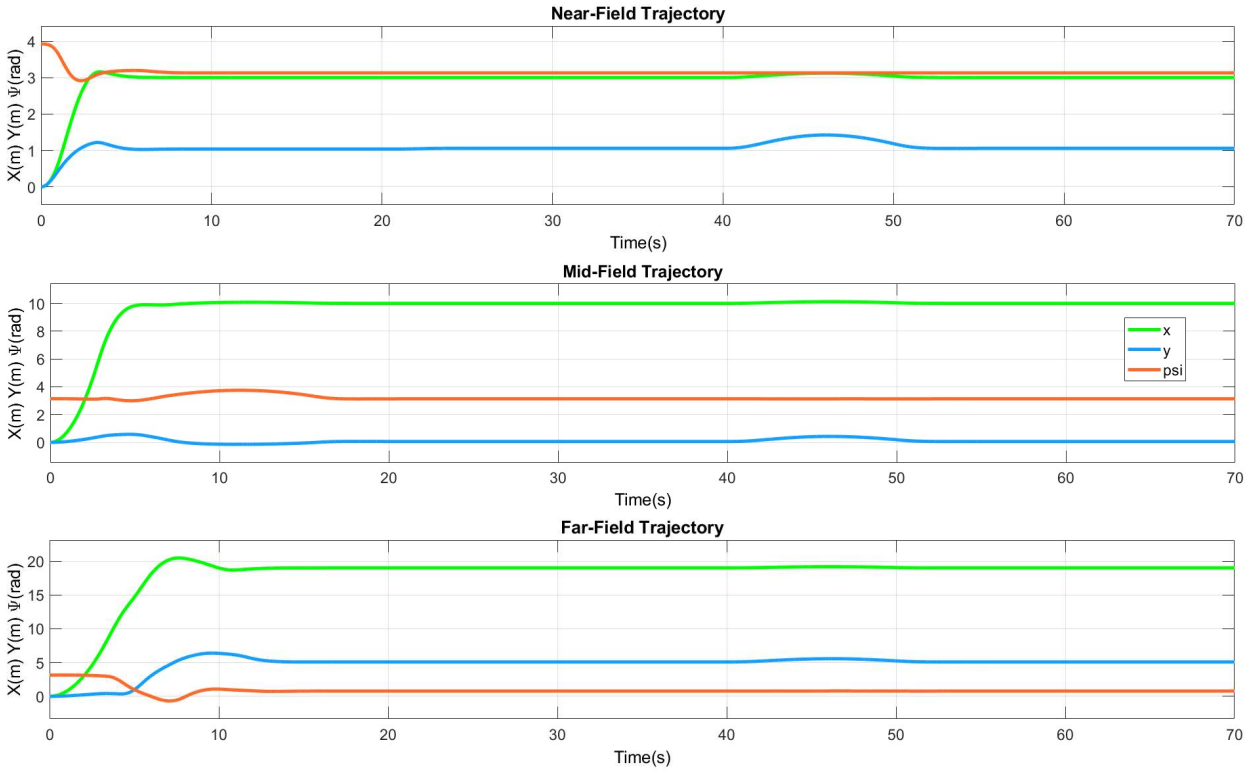

FIGURE 2: NONLINEAR STATION KEEPING CONTROL RESPONSE OVER TIME IN TERMS OF THE POSE OF THE USV (x, y, ψ). FROM HIGH TO LOW: NEAR-FIELD, MID-FIELD AND FAR-FIELD CASE SCENARIOS.

TABLE II: TIME DOMAIN SPECIFICATIONS FOR THE NONLINEAR STATION KEEPING CONTROLLER RESPONSE FOR DIFFERENT CASE SCENARIOS

Variable	Units	t_r [s]	val(t_r)	t_p [s]	val(t_p)	t_s [s]	val(t_s)	M_p	M_g	SSE
Near-field										
x	m	2.49	2.72	3.45	3.16	3.67	3.15	0.16	0.13	0.005
y	m	1.88	0.91	3.34	1.22	4.75	1.07	0.22	0.43	0.06
ψ	rad	1.53	3.22	2.38	2.92	2.85	2.99	0.22	0.0	0.008
Mid-field										
x	m	4.01	9.03	X	X	4.41	9.51	X	0.13	0.0
y	m	X	X	4.67	0.58	X	X	0.58	0.43	0.06
ψ	rad	X	X	11.19	3.750	15.74	3.29	0.608	0.0	0.008
Far field										
x	m	5.71	17.17	7.63	20.48	8.78	19.95	1.48	0.17	0.01
y	m	6.96	4.52	9.59	6.40	12.88	5.25	1.40	0.57	0.087
ψ	rad	4.99	1.016	7.09	-0.689	11.96	0.821	1.47	0.002	0.008

TABLE III: PROPULSIVE POWER CONSUMPTION FOR THE THREE CASE SCENARIOS. (ALL UNITS IN WATTS)

Thruster	Power Case (a)	Power Case (b)	Power Case (c)	Cumulative Power
Port	1,135,332	1,310,280	1,465,549	3,911,161
Starboard	686,235	821,672	791,194	2,299,101
Cumulative Power	1,821,567	2,131,952	2,256,743	6,210,262

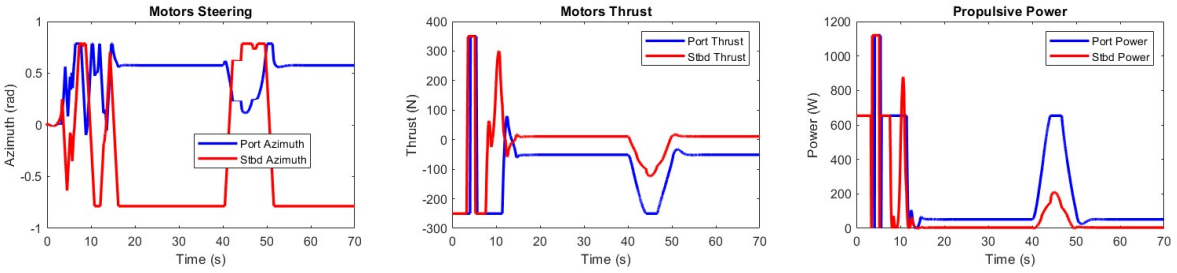

FIGURE 3: THRUSTERS RESPONSE OVER TIME (BLUE FOR PORT).

FIGURE 4: MACHINE-LEARNING STATION KEEPING CONTROL RESPONSE OVER TIME IN TERMS OF THE POSE OF THE USV (x, y, ψ).

TABLE IV: TIME DOMAIN SPECIFICATIONS FOR THE MACHINE-LEARNING STATION-KEEPING CONTROLLER RESPONSE FOR DIFFERENT CASE SCENARIOS.

Variable	Units	t_r [s]	val(t_r)	t_p [s]	val(t_p)	t_s [s]	val(t_s)	M_p	M_g	SSE
Near-field										
x	m	3.54	2.70	5.10	3.61	6.05	3.30	0.61	1.71	0.16
y	m	1.87	0.90	5.08	2.41	X	X	1.42	5.08	-0.94
ψ	rad	2.53	3.23	4.88	3.32	2.53	3.23	0.18	0.85	.025
Mid-field										
x	m	4.12	9.0	6.61	11.84	7.91	11.00	1.84	1.58	0.62
y	m	4.04	0.0	38.05	1.03	X	X	1.03	4.92	-0.79
ψ	rad	X	X	4.94	2.45	6.10	2.83	0.69	0.89	0.09
Far field										
x	m	7.25	17.1	10.44	20.55	7.25	10.44	1.55	1.48	0.49
y	m	9.47	4.5	34.90	6.05	X	X	1.05	2.72	0.29
ψ	rad	2.24	1.02	3.86	-.067	11.76	.706	0.85	.35	-0.02

TABLE V: PROPULSIVE POWER CONSUMPTION FOR THE THREE CASE SCENARIOS. (ALL UNITS IN WATTS)

Thruster	Power Case (a)	Power Case (b)	Power Case (c)	Cumulative Power
Port	12,224,500	14,116,100	12,659,900	39,000,500
Starboard	9,750,690	11,369,900	6,968,660	28,089,300
Cumulative Power	21,975,200	25,486,100	19,628,600	67,089,800

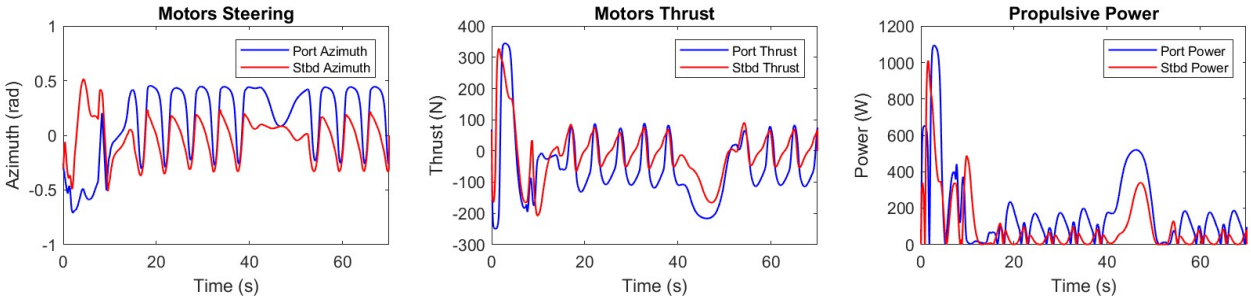

FIGURE 5: THRUSTERS RESPONSE OVER TIME. FROM LEFT TO RIGHT: AZIMUTH ROTATIONS, THRUST AND PROPULSIVE POWER.

that the controller was not trained against randomly constant wind force values, and thus, further training is required in order to improve this condition. The nonlinear controller also showed the best performance in terms of holding the desired pose against strong wind gusts (according to M_g values) and steady-state error values (*SSE)*. Furthermore, Tables III and V also revealed the superiority of the nonlinear controller to optimize for power consumption, by a factor of almost 11.

ACKNOWLEDGEMENTS

This work is supported by the Office of Naval Research under grants N000141512724 and N00014-18-1-2212 (Program Manager: Kelly Cooper).

REFERENCES

[1] J. G. Marquardt, J. Alvarez and K. D. von Ellenrieder, "Characterization and System Identification of an Unmanned Amphibious Tracked Vehicle," *IEEE Journal of Oceanic Engineering,* vol. 39, no. 4, pp. 641-661, 2014.

[2] H. Qu, E. I. Sarda, I. R. Bertaska and K. D. von Ellenrieder, "Wind Feed-forward Control of a USV," in *MTS/IEEE Oceans 2015*, Genova, 2015.

[3] M. A. Diddans, M. R. Dhanak and A. J. Sinisterra, "A Low-Level USV Controller Incorporating an Environmental Disturbance Observer," in *MTS/IEEE Oceans*, Seattle, 2019.

[4] W. B. Klinger, I. R. Bertaska and K. D. von Ellenrieder, "Experimental testing of an adaptive controller for USVs with uncertain displacement and drag," in *MTS/IEEE Oceans*, St. John's, 2014.

[5] J.-J. E. Slotine and W. Li, Applied Nonlinear Control, Upper Saddle River, New Jersey: Prentice-Hall, Inc., 1991.

[6] T. I. Fossen, Handbook of Marine Craft Hydrodynamics and Motion Control, The Atrium, Southern Gate, Chichester, West Sussex, PO19 8SQ, United Kingdom: John Wiley & Sons, Ltd., 2011.

[7] T. A. Johansen, T. I. Fossen and S. P. Berge, "Constrained Nonlinear Control Allocation With Singularity Avoidance Using Sequential Quadratic Programming," *IEEE Transactions On Control Systems Technology,* vol. 12, no. 1, pp. 211-216, 2004.

[8] R. S. Sutton and A. G. Barto, Reinforcement learning: An introduction, Cambridge,MA, USA: The MIT press, 1998.

[9] E. I. Sarda, I. R. Bertaska, A. Qu and K. D. Ellenrieder, "Development of a USV station-keeping controller," in *MTS/IEEE OCEANS 2015*, Genova, 2015.

[10] The Society of Naval Architects and Marine Engineers, "Nomenclature for Treating the Motion of a Submerged Body through a Fluid," in *Technical and Research Bulletin* , 1950, pp. 1-15.

[11] T. I. Fossen, Guidance and Control of Ocean Vehicles, Chichester: John Wiley & Sons, 1994.

[12] International Maritime Organization, Explanatory Notes to the Standards for Ship Manoeuvrability, London, 2002.

[13] J. N. Newman, Marine Hydrodynamics, Massachusetts Institute of Technology, 1977.

[14] A. J. Sinisterra, S. Verma and M. R. Dhanak, "Performance characterization and comparison of conventional and machine-learning-based techniques for control of a USV," in *MTS/IEEE Oceans*, Seattle, 2019.

[15] V. R. Konda and J. N. Tsitsiklis, "On Actor-Critic Algorithms," *SIAM J. CONTROL OPTIM.,* vol. 42, no. 4, p. 1143–1166, 2003.

[16] G. Novati and P. Koumoutsakos, "Remember and Forget for Experience Replay," in *36th International Conference on Machine Learning*, Long Beach, California, USA, 2019.

[17] M. A. Diddams, Control of an Unmanned Surface Vehicle Using An Environmental Disturbance Observer, Boca Raton, Florida, USA: MSc. Thesis, Florida Atlantic University, 2018.

[18] C. De Wit, "Optimal Thrust Allocation Methods for Dynamic Positioning of Ships," MSc Thesis, Delft University of Technology, Delft, the Netherlands, 2009.