

Neural Network Model Reference Adaptive Control of a Surface Vessel

Alexander Leonessa and Tannen S. VanZwieten

Abstract—A neural network model reference adaptive controller for trajectory tracking of nonlinear systems is developed. The proposed control algorithm uses a single layer neural network that bypasses the need for information about the system’s dynamic structure and provides portability. Numerical simulations are performed using a three degree of freedom nonlinear dynamic model of a surface vessel. The results demonstrate the controller performance in terms of tuning, robustness and tracking.

I. INTRODUCTION

Research and development on Autonomous Surface Vehicles (ASVs) is motivated by the navigation and communication challenges involved with the use of Autonomous Underwater Vehicles (AUVs). The ASV development objective is to provide real time positioning of and communication with AUVs through the air-sea interface. Guidance and control play an integral part in the ASV’s success, which is a motivating factor for this research.

The overall dynamics for a three degree of freedom ASV are modeled and numerically simulated in [1]. Since the modeling trends and dynamical behavior are similar for different surface vessels, using the model developed in [1] will help to quantify the dynamics and performance capabilities of surface vessels in general and facilitate the testing and tuning of the controllers. Furthermore, the model in [1] allows to study the effect of the model nonlinearities on the controller’s performance.

The dynamical behaviors of a surface vessel can only be partially depicted using current modeling techniques. These dynamics are especially complicated when the vehicle’s motion is in transient, as when considering a tracking problem where the vessel may be required to follow search patterns or its desired path is constantly being modified. Additionally, the ocean environment is characterized by large, unknown perturbations. These features make it desirable to have a control system that is capable of self-tuning.

This paper introduces a neural network model reference adaptive controller which has valuable self-tuning capabilities that allow it to “adapt” to the operating conditions in order to optimize the tracking performance of the closed-loop system. The parameter update mechanism is derived using ultimate boundedness theory, which guarantees that

the tracking error is ultimately bounded when subject to some generalized constraints. The addition of a single layer neural network bypasses the need for information about the system’s dynamic structure and characteristics. The presented control algorithm is tested and tuned using the surface vessel’s nonlinear model presented in [1].

II. MATHEMATICAL PRELIMINARIES

In this section we establish definitions, notation, and a key result used later in the paper. Let \mathbb{R} denote the set of real numbers, let \mathbb{R}^n denote the set of $n \times 1$ real column vectors, let $\mathbb{R}^{n \times m}$ denote the set of real $n \times m$ matrices, and let $(\cdot)^T$ denote transpose. Furthermore, we write $\|\cdot\|$ for the Euclidean vector norm and $A \geq 0$ (resp., $A > 0$) to denote the fact that the Hermitian matrix A is nonnegative (resp., positive) definite.

In this paper we consider nonlinear controlled dynamical systems of the form

$$\dot{x}(t) = F(x(t), u(t)), \quad x(0) = x_0, \quad t \in \mathcal{I}_{x_0}, \quad (1)$$

where $x(t) \in \mathcal{D} \subseteq \mathbb{R}^n$, $t \geq 0$, is the system state vector, $\mathcal{I}_{x_0} \subseteq \mathbb{R}$ is the maximal interval of existence of a solution $x(\cdot)$ of (1), \mathcal{D} is an open set, $0 \in \mathcal{D}$, $u(t) \in \mathcal{U} \subseteq \mathbb{R}^m$, $t \geq 0$, is the control input, \mathcal{U} is the set of all admissible controls such that $u(\cdot)$ is a measurable function with $0 \in \mathcal{U}$, and $F : \mathcal{D} \times \mathcal{U} \rightarrow \mathbb{R}^n$ is continuous on $\mathcal{D} \times \mathcal{U}$.

Following the notation introduced in [2], consider the following nonlinear autonomous dynamical system

$$\dot{x}(t) = f(t, x(t)), \quad x(0) = x_0 \quad t \in \mathcal{I}_{x_0} \quad (2)$$

where $f : \mathcal{I}_{x_0} \times \mathcal{D} \rightarrow \mathbb{R}^n$ is piecewise continuous in t and locally Lipschitz in x .

Definition 2.1: [2] The solutions of (2) are *uniformly ultimately bounded* with ultimate bound $\varepsilon > 0$ if there exists $\gamma > 0$ such that, for every $\delta \in (0, \gamma)$, there exists $T = T(\delta, \varepsilon) > 0$ such that $\|x_0\| < \delta$ implies $\|x(t)\| < \varepsilon$, $t \geq T$.

In the case of autonomous systems, we may drop the word “uniformly”.

To see how Lyapunov analysis can be used to study ultimate boundedness, we introduce the following theorem.

Theorem 2.1: [2] Assume that there exists a continuously differentiable function $V : \mathcal{I}_{x_0} \times \mathcal{D} \rightarrow \mathbb{R}$ such that

$$\alpha(\|x\|) \leq V(t, x) \leq \beta(\|x\|), \quad (t, x) \in \mathcal{I}_{x_0} \times \mathcal{D}, \quad (3)$$

$$\frac{\partial V}{\partial t} + \frac{\partial V}{\partial x} f(t, x) \leq -W(x), \quad \|x\| \geq \mu > 0, \quad (4)$$

Alexander Leonessa is with the Faculty of Mechanical, Materials, and Aerospace Engineering, University of Central Florida, P.O. Box 162450, Orlando, FL 32816 – 2450 aleo@mail.ucf.edu

Tannen S. VanZwieten is a Ph.D. Student in the Department of Mechanical, Materials, and Aerospace Engineering, University of Central Florida, P.O. Box 162450, Orlando, FL 32816 – 2450 ta090098@ucf.edu

where $\alpha(\cdot)$ and $\beta(\cdot)$ are class \mathcal{K} functions, $W : \mathcal{D} \rightarrow \mathbb{R}$ is a continuous positive definite function, and $\mu > 0$ is such that $\mathcal{B}_{\alpha^{-1}(\beta(\mu))}(0) \subset \mathcal{D}$. Then the solutions of (2) are *ultimately bounded* with ultimate bound $\varepsilon \triangleq \alpha^{-1}(\beta(\mu))$.

III. MODELING

In this section we present a mathematical model of an Autonomous Surface Vehicle which will be used for the testing and tuning of the controllers. In particular, our model assumes that pitch, roll and heave motions are negligible and feature only the three degrees of freedom corresponding to surge, sway and yaw motions. Because of this choice we will assume for the remaining of the paper that the state space belongs to \mathbb{R}^6 although the control algorithm can be easily extended to higher dimensions.

A. Equations of Motion

The notation used for the surface vessel's generalized equations of motion follows [3], but is reduced to motion in the horizontal plane. The Earth Fixed Frame (EFF), denoted by x_e , y_e and z_e , is chosen so that the vessel's center of gravity (CG) is at the origin at time $t = 0$. The x_e and y_e axes are directed toward the North and the East, respectively, while the z_e axis points downward. This frame is assumed to be inertial, because when studying the motion of marine vehicles the acceleration at a point on the earth due to the earth's rotation is generally considered negligible. The vessel's configuration in the EFF is

$$\eta(t) \triangleq [x(t), y(t), \psi(t)]^T, \quad t \geq 0, \quad (5)$$

where $x(t) \in \mathbb{R}$ and $y(t) \in \mathbb{R}$ describe the distance traveled along the x_e and y_e directions respectively, and $\psi(t) \in \mathbb{R}$ describes the rotation about the z_e axis (see Fig. 1).

The Body Fixed Frame (BFF) has its origin fixed at the vehicle's center of gravity (CG), the x_b axis points forward, the y_b axis starboard, and the z_b axis downward (see Fig. 1). The vessel's velocity is defined in the BFF as

$$\nu(t) \triangleq [u(t), v(t), r(t)]^T, \quad t \geq 0, \quad (6)$$

where $u(t) \in \mathbb{R}$ and $v(t) \in \mathbb{R}$ are the components of the absolute velocity in the x_b and y_b directions respectively, and $r(t) \in \mathbb{R}$ describes the angular velocity about the z_b axis. The vectors $\eta(t)$ and $\nu(t)$ are related by the kinematic equation [4],

$$\dot{\eta}(t) = J(\eta(t))\nu(t), \quad t \geq 0, \quad (7)$$

where

$$J(\eta) \triangleq \begin{bmatrix} \cos \psi & -\sin \psi & 0 \\ \sin \psi & \cos \psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (8)$$

is the rotation matrix from the BFF to the EFF.

Using the form introduced in [4] and the previous notation, the surface vessel's equation of motion is given by

$$M\dot{\nu}(t) + C(\nu(t))\nu(t) + D(\nu(t))\nu(t) + g(\eta(t)) = \tau(t), \quad t \geq 0, \quad (9)$$

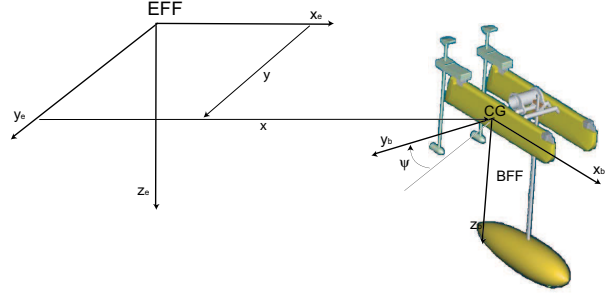


Fig. 1. Coordinate frames

where M is the mass matrix, $C(\nu(t))$ contains Coriolis, centripetal and added-mass terms, $D(\nu(t))$ is the damping matrix, $g(\eta(t))$ is the vector of gravitational forces and moments, and $\tau(t)$ is the input vector. We assume that the vessel is equipped with two motors in the rear which can be rotated independently. This provides an input

$$\tau(t) \triangleq [X(t), Y(t), N(t)]^T, \quad (10)$$

consisting of two forces $X(t), Y(t) \in \mathbb{R}$ along the x_b and y_b axes respectively, as well as a yawing moment $N(t) \in \mathbb{R}$.

While the rigid body inertia, Coriolis, centripetal, and gravitational terms are described in [4], the hydrodynamic terms are much more challenging to model and depend on the particular geometry of the vessel. An example of how to compute them for a specific vessel is provided in [1]. In general, even very thorough hydrodynamic modeling efforts are only able to partially describe the dynamic behavior of the vessel because assumptions made always considerably affect the final result. In light of these considerations we are going to write the vessel dynamics as

$$\dot{\nu}(t) = f(\eta(t), \nu(t)) + B\tau(t), \quad (11)$$

where

$$B \triangleq M^{-1}, \\ f(\eta, \nu) \triangleq -B[C(\nu) + D(\nu)]\nu - Bg(\eta),$$

are *completely* unknown. The only system information utilized is the positive definiteness of B , as the mass matrix always satisfies this assumption.

IV. ADAPTIVE CONTROLLER DESIGN

The nonlinearities and unmodeled dynamics that characterize the surface vessel model introduced in the previous section make it desirable to have a control system that is capable of self-tuning. The adaptive controller developed here is for a fully multi-input/multi-output system which accounts for the strong dynamic coupling which characterizes the surface vessel model and has valuable self-tuning capabilities. The parameter update mechanism derived in this section uses ultimate boundedness theory.

When using model reference adaptive control, a control algorithm is developed so that the system mimics the behavior of a reference system which provides smooth

convergence to the desired trajectory. Choosing the proper reference system allows the vehicle to exhibit less overshoot and oscillatory behavior as well as better tracking performance. Furthermore, the control inputs become more realistic, even when the vehicle is far away from the desired trajectory, diminishing the need to implement input amplitude and rate saturation algorithms.

A. Reference System

The reference system can be written as

$$\dot{x}_r(t) = A_r x_r(t) + B_r \hat{r}(t), \quad t \geq 0, \quad (12)$$

where $x_r(t) \in \mathbb{R}^6, t \geq 0$ is the reference state, $A_r \in \mathbb{R}^{6 \times 6}$, $B_r \in \mathbb{R}^{6 \times 3}$ are constant matrixes and $\hat{r}(t) \in \mathbb{R}^3, t \geq 0$ is the reference input.

The reference system considered here is composed of three uncoupled second order oscillators. Each oscillator is characterized by a damping coefficient $\zeta_i > 0, i = 1, 2, 3$, and a natural frequency $\omega_{0i} > 0, i = 1, 2, 3$. This choice was mostly motivated by the simplicity of the corresponding reference dynamics. The dynamics of the i^{th} oscillator are given by

$$\ddot{x}_{ri}(t) + 2\zeta_i \omega_{0i} \dot{x}_{ri}(t) + \omega_{0i}^2 x_{ri}(t) = \omega_{0i}^2 \hat{r}_i(t), \quad t \geq 0, \quad i = 1, 2, 3. \quad (13)$$

Thus, the reference system can be rewritten as

$$\begin{bmatrix} \dot{x}_{1r}(t) \\ \dot{x}_{2r}(t) \end{bmatrix} = \begin{bmatrix} 0_3 & I_3 \\ -\omega_0^2 & -A_{r3} \end{bmatrix} \begin{bmatrix} x_{1r}(t) \\ x_{2r}(t) \end{bmatrix} + \begin{bmatrix} 0_3 \\ \omega_0^2 \end{bmatrix} \hat{r}(t), \quad t \geq 0, \quad (14)$$

where

$$x_{1r}(t) \triangleq \begin{bmatrix} x_{r1}(t) & x_{r2}(t) & x_{r3}(t) \end{bmatrix}^T, \quad (15)$$

$$x_{2r}(t) \triangleq \begin{bmatrix} \dot{x}_{r1}(t) & \dot{x}_{r2}(t) & \dot{x}_{r3}(t) \end{bmatrix}^T, \quad (16)$$

and

$$A_{r3} \triangleq \text{diag}(2\zeta_1 \omega_{01}, 2\zeta_2 \omega_{02}, 2\zeta_3 \omega_{03}), \quad (17)$$

$$\omega_0 \triangleq \text{diag}(\omega_{01}, \omega_{02}, \omega_{03}). \quad (18)$$

Finally, the desired trajectory for the surface vessel can be written as

$$\tilde{x}_d(t) = \begin{bmatrix} x_d(t) & y_d(t) & \psi_d(t) \end{bmatrix}^T. \quad (19)$$

By choosing

$$\hat{r}(t) = \omega_0^{-2} (\ddot{x}_d(t) + A_{r3} \dot{x}_d(t) + \omega_0^2 \tilde{x}_d(t)), \quad (20)$$

we find that

$$\begin{bmatrix} \dot{x}_{1r}(t) - \dot{\tilde{x}}_d(t) \\ \dot{x}_{2r}(t) - \dot{\tilde{x}}_d(t) \end{bmatrix} = A_r \begin{bmatrix} x_{1r}(t) - \tilde{x}_d(t) \\ x_{2r}(t) - \tilde{x}_d(t) \end{bmatrix}. \quad (21)$$

Since

$$A_r \triangleq \begin{bmatrix} 0_3 & I_3 \\ -\omega_0^2 & -A_{r3} \end{bmatrix}, \quad (22)$$

is a stable matrix, it follows that $x_{1r}(t) - \tilde{x}_d(t) \rightarrow 0$ and $x_{2r}(t) - \dot{\tilde{x}}_d(t) \rightarrow 0$, i.e. the reference state converges to the desired trajectory. The remaining problem is to design a control command, $\tau(t)$, such that $[\eta^T(t) \quad \nu^T(t)]^T -$

$x_r(t) \rightarrow 0$ (i.e., the system state converges to the reference state). Considering that the mass, Coriolis/centrifugal and damping matrices of the real system contain unknown parameters and unknown terms, a control signal which accounts for these uncertainties needs to be considered.

1) *Tracking Error Dynamics*: The position and velocity tracking errors are defined as $e_1(t) \triangleq \eta(t) - x_{1r}(t)$ and $e_2(t) \triangleq \dot{\eta}(t) - \dot{x}_{1r}(t)$, respectively. The corresponding error dynamics are given by

$$\dot{e}_1(t) = \dot{\eta}(t) - \dot{x}_{1r}(t) = e_2(t), \quad (23)$$

$$\dot{e}_2(t) = \frac{d(J(\eta(t))\nu(t))}{dt} + \omega_0^2 x_{1r}(t) + A_{r3} x_{2r}(t) - \omega_0^2 \hat{r}(t). \quad (24)$$

Substituting (7) and (11) into (24) yields

$$\begin{aligned} \dot{e}_2(t) = & -\omega_0^2 e_1(t) - A_{r3} e_2(t) + J(\eta(t)) [h(\eta(t), \nu(t)) \\ & + f(\eta(t), \nu(t)) + B\tau(t) + \omega_0^2 J^{-1}(\eta(t))(\eta(t) - \hat{r}(t))], \end{aligned} \quad (25)$$

where

$$h(\eta, \nu) \triangleq J^{-1}(\eta) \left(\frac{\partial(J(\eta)\nu)}{\partial \eta} + A_{r3} \right) J(\eta)\nu. \quad (26)$$

2) *Control Command*: The next goal is to define a control signal, $\tau(t)$, which guarantees that the tracking error is ultimately bounded. Consider the controller

$$\tau(t) = \Theta^* w(\eta(t), \nu(t), \hat{r}(t)) + \delta^*(\eta(t), \nu(t)), \quad (27)$$

where

$$w(\eta, \nu, \hat{r}) \triangleq \omega_0^2 J^{-1}(\eta)(\eta - \hat{r}) + h(\eta, \nu) + J^{-1}(\eta)(K_1 e_1 + K_2 e_2), \quad (28)$$

and $\Theta^* \in \mathbb{R}^{3 \times 3}$ and $\delta^*(\cdot, \cdot) \in \mathbb{R}^3$. The arbitrary gains $K_1 \in \mathbb{R}^{3 \times 3}$ and $K_2 \in \mathbb{R}^{3 \times 3}$ are part of our set of tuning parameters. By substituting (27) into (23) and (25), the following closed loop error dynamics are obtained

$$\begin{aligned} \dot{e}_1(t) &= e_2(t), \\ \dot{e}_2(t) &= -\tilde{\omega}_0^2 e_1(t) - \tilde{A}_{r3} e_2(t) + J(\eta(t)) [(B\Theta^* + I) \\ & w(\eta(t), \nu(t), \hat{r}(t)) + B\delta^*(\eta(t), \nu(t)) \\ & + f(\eta(t), \nu(t))], \end{aligned} \quad (29)$$

where $\tilde{\omega}_0^2 \triangleq \omega_0^2 + K_1$ and $\tilde{A}_{r3} \triangleq A_{r3} + K_2$. Finally, by choosing $\Theta^* = -B^{-1} = -M$ and $\delta^*(\eta(t), \nu(t)) = -B^{-1} f(\eta(t), \nu(t))$, we obtain

$$\begin{bmatrix} \dot{e}_1(t) \\ \dot{e}_2(t) \end{bmatrix} = \tilde{A}_r \begin{bmatrix} e_1(t) \\ e_2(t) \end{bmatrix}, \quad (30)$$

where

$$\tilde{A}_r = \begin{bmatrix} 0 & I_3 \\ -\tilde{\omega}_0^2 & -\tilde{A}_{r3} \end{bmatrix}, \quad (31)$$

which, when K_1 and K_2 are properly chosen, correspond to a stable matrix, providing asymptotic stability of the closed loop error dynamics. Next, since Θ^* and $\delta^*(\eta(t), \nu(t))$ are unknown, their estimates need to be introduced. In particular, Θ^* in (27) will be replaced with its estimate $\Theta(t)$ such that $\Theta(t) = \Theta^* + \hat{\Theta}(t)$, where $\hat{\Theta}(t)$ represents

the estimation error. Next, following the approach described in [5], it will be assumed that, for a given $\varepsilon^* > 0$, the vector function $\delta^*(\eta, \nu)$ can be approximated over a compact set $\mathcal{D} \subset \mathbb{R}^3 \times \mathbb{R}^3$ by a linear parameterized neural network with a maximum approximation error given by ε^* . Hence, there exists $\varepsilon(\eta, \nu)$ such that $\|\varepsilon(\eta, \nu)\| < \varepsilon^*$ for all $(\eta, \nu) \in \mathcal{D}$, and

$$\delta^*(\eta, \nu) = W^* \sigma(\eta, \nu) + \varepsilon(\eta, \nu), \quad (32)$$

where $W^* \in \mathbb{R}^{3 \times q}$ is the matrix of optimal *unknown* (constant) weights that minimize the approximation error over \mathcal{D} , $\sigma : \mathbb{R}^3 \times \mathbb{R}^3 \rightarrow \mathbb{R}^q$ is a vector of basis functions such that each component of $\sigma(\cdot, \cdot)$ takes values in $[0, 1]$, $\varepsilon(\cdot, \cdot)$ is the vector of approximation errors, and $\|W^*\| \leq w^*$, where w^* is a bound for the optimal weight matrix. Now, following the procedure described in [6], $\delta^*(\eta(t), \nu(t))$ is replaced in (27) with $W(t)\sigma(\eta(t), \nu(t))$ where $W(t) \in \mathbb{R}^{3 \times q}$ is the estimate of the optimal weights such that $W(t) = W^* + \tilde{W}(t)$, with $\tilde{W}(t)$ representing the estimation error.

When replacing the controller (27) with the following

$$\tau(t) = \Theta(t)w(\eta(t), \nu(t), \hat{r}(t)) + W(t)\sigma(\eta(t), \nu(t)), \quad (33)$$

the corresponding error dynamics are given by

$$\begin{bmatrix} \dot{e}_1(t) \\ \dot{e}_2(t) \end{bmatrix} = \tilde{A}_r \begin{bmatrix} e_1(t) \\ e_2(t) \end{bmatrix} + \begin{bmatrix} 0 \\ J(\eta(t))\gamma(t) \end{bmatrix}, \quad (34)$$

where

$$\gamma(t) \triangleq B \begin{bmatrix} \tilde{\Theta}(t)w(\eta(t), \nu(t), \hat{r}(t)) + \tilde{W}(t)\sigma(\eta(t), \nu(t)) \\ -\varepsilon(\eta(t), \nu(t)) \end{bmatrix}. \quad (35)$$

The stability properties of the error dynamics (34) are analyzed in the next section.

3) *Closed Loop Error Dynamics*: To show stability of the closed loop error dynamics (34), the following Lyapunov function candidate is considered

$$V(e_1, e_2, \tilde{\Theta}, \tilde{W}) = \frac{1}{2} \begin{bmatrix} e_1^T & e_2^T \end{bmatrix} P \begin{bmatrix} e_1 \\ e_2 \end{bmatrix} + \frac{1}{2} \text{tr} \left(B \tilde{\Theta} \Gamma_1^{-1} \tilde{\Theta}^T \right) + \frac{1}{2} \text{tr} \left(B \tilde{W} \Gamma_2^{-1} \tilde{W}^T \right), \quad (36)$$

where $P \triangleq \begin{bmatrix} P_1 & P_{12} \\ P_{12}^T & P_2 \end{bmatrix} > 0$, $\Gamma_1 > 0$, and $\Gamma_2 > 0$ are design constant matrices. Note that $B = M^{-1}$ is positive definite. The corresponding Lyapunov derivative is given by

$$\begin{aligned} \dot{V}(t) &= \frac{1}{2} \begin{bmatrix} \dot{e}_1^T(t) & \dot{e}_2^T(t) \end{bmatrix} \left(\tilde{A}_r^T P + P \tilde{A}_r \right) \begin{bmatrix} e_1(t) \\ e_2(t) \end{bmatrix} \\ &+ e^T(t) J(\eta(t)) \gamma(t) + \text{tr} \left(B \tilde{\Theta}(t) \Gamma_1^{-1} \dot{\tilde{\Theta}}^T(t) \right) \\ &+ \text{tr} \left(B \tilde{W}(t) \Gamma_2^{-1} \dot{\tilde{W}}^T(t) \right), \end{aligned}$$

where $\dot{\Theta}(t) = \dot{\tilde{\Theta}}(t)$, $\dot{W}(t) = \dot{\tilde{W}}(t)$ and $e(t) \triangleq P_{12}^T e_1(t) + P_2 e_2(t)$. By choosing P so that the following Lyapunov equation is satisfied

$$\tilde{A}_r^T P + P \tilde{A}_r + \begin{bmatrix} Q_1 & 0_3 \\ 0_3 & Q_2 \end{bmatrix} = 0, \quad (37)$$

where $Q_1, Q_2 \in \mathbb{R}^{3 \times 3}$ are positive definite matrices, we obtain

$$\begin{aligned} \dot{V}(t) &= -\frac{1}{2} e_1^T(t) Q_1 e_1(t) - \frac{1}{2} e_2^T(t) Q_2 e_2(t) \\ &- e^T(t) J(\eta(t)) B \varepsilon(\eta(t), \nu(t)) \\ &+ \text{tr} \left(B \tilde{\Theta}(t) w(\eta(t), \nu(t), \hat{r}(t)) e^T(t) J(\eta(t)) + B \tilde{\Theta}(t) \Gamma_1^{-1} \dot{\tilde{\Theta}}^T(t) \right) \\ &+ \text{tr} \left(B \tilde{W}(t) \sigma(\eta(t), \nu(t)) e^T(t) J(\eta(t)) + B \tilde{W}(t) \Gamma_2^{-1} \dot{\tilde{W}}^T(t) \right). \end{aligned}$$

Finally, the update laws

$$\dot{\Theta}(t) = -J^T(\eta(t)) e(t) w^T(\eta(t), \nu(t), \hat{r}(t)) \Gamma_1 - \sigma_1 \Theta(t), \quad (38)$$

$$\dot{W}(t) = -J^T(\eta(t)) e(t) \sigma^T(\eta(t), \nu(t)) \Gamma_2 - \sigma_2 W(t), \quad (39)$$

where $\sigma_1 > 0$ and $\sigma_2 > 0$, provide the following bound for the Lyapunov derivative

$$\begin{aligned} \dot{V}(t) &\leq -\frac{1}{2} \left\| Q_1^{\frac{1}{2}} e_1(t) \right\| \left(\left\| Q_1^{\frac{1}{2}} e_1(t) \right\| \right. \\ &\quad \left. - 2 \left\| Q_1^{-\frac{1}{2}} P_{12} J(\eta(t)) B \varepsilon(\eta(t), \nu(t)) \right\| \right) \\ &- \frac{1}{2} \left\| Q_2^{\frac{1}{2}} e_2(t) \right\| \left(\left\| Q_2^{\frac{1}{2}} e_2(t) \right\| \right. \\ &\quad \left. - 2 \left\| Q_2^{-\frac{1}{2}} P_2 J(\eta(t)) B \varepsilon(\eta(t), \nu(t)) \right\| \right), \\ &- \frac{\sigma_1}{2} \text{tr} \left(B \tilde{\Theta}(t) \Gamma_1^{-1} \tilde{\Theta}^T(t) \right) + \frac{\sigma_1}{2} \text{tr} \left(B \Theta^* \Gamma_1^{-1} \Theta^{*T} \right) \\ &- \frac{\sigma_2}{2} \text{tr} \left(B \tilde{W}(t) \Gamma_2^{-1} \tilde{W}^T(t) \right) + \frac{\sigma_2}{2} \text{tr} \left(B W^* \Gamma_2^{-1} W^{*T} \right). \end{aligned} \quad (40)$$

Now, we define the domain

$$\begin{aligned} \mathcal{D}_w &\triangleq \{ (e_1, e_2, \tilde{\Theta}, \tilde{W}) : \\ &\quad \left\| Q_1^{\frac{1}{2}} e_1 \right\| \geq 2 \left\| Q_1^{-\frac{1}{2}} \right\| \left\| P_{12} \right\| \left\| B \right\| \varepsilon^* \\ &\quad \left\| Q_2^{\frac{1}{2}} e_2 \right\| \geq 2 \left\| Q_2^{-\frac{1}{2}} \right\| \left\| P_2 \right\| \left\| B \right\| \varepsilon^* \\ &\quad \text{tr} \left(B \tilde{\Theta} \Gamma_1^{-1} \tilde{\Theta}^T \right) \geq \text{tr} \left(B \Theta^* \Gamma_1^{-1} \Theta^{*T} \right) \\ &\quad \text{tr} \left(B \tilde{W} \Gamma_2^{-1} \tilde{W}^T \right) \geq \text{tr} \left(B W^* \Gamma_2^{-1} W^{*T} \right) \} \end{aligned} \quad (41)$$

which is such that

$$\dot{V}(t) < 0, \quad (e_1(t), e_2(t), \tilde{\Theta}(t), \tilde{W}(t)) \in \mathcal{D}_w. \quad (42)$$

It follows from Theorem 2.1 that the solutions of (34), (38), and (39) are ultimately bounded with ultimate bound

$$\varepsilon \triangleq \max_{(e_1, e_2, \tilde{\Theta}, \tilde{W}) \in \partial \mathcal{D}_w} V(e_1, e_2, \tilde{\Theta}, \tilde{W}), \quad (43)$$

We conclude that the closed loop error dynamics are ultimately bounded, hence the tracking error will enter a neighborhood \mathcal{D}_T of the origin in a finite time $T > 0$ and it will not exit such a neighborhood at any time $t \geq T$. Note that the size of \mathcal{D}_T is proportional to the bound ε^* , hence by improving the approximation for $\delta^*(\eta(t), \nu(t))$, $(\eta(t), \nu(t)) \in \mathcal{D}$, the residual tracking error for $t > T$ is reduced.

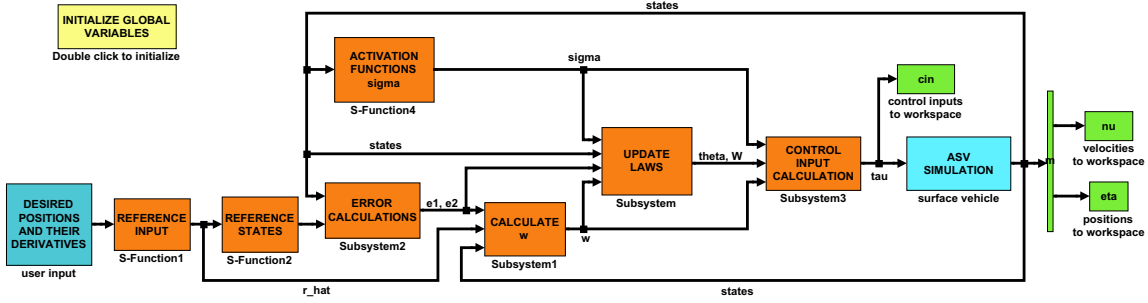


Fig. 2. Adaptive control Simulink diagram

V. NUMERICAL SIMULATIONS

In this section different maneuvers will be performed in order to test the controller performance capabilities. Results include the desired, actual and reference trajectories and orientation so that the behavior of the vessel is completely shown. The parameter estimates, $\Theta(t)$ and $W(t)$, $t \geq 0$, will also be plotted to show the system adaptation throughout the simulation. The complete set of control inputs, $\tau(t)$, $t \geq 0$, is also shown; so that it may be determined if the necessary forces and torque can be actuated.

The diagram shown in Fig. 2 shows the simulation code. Note that the “ASV simulation” block consists of the dynamic equations (7) and (11) where B and $f(\eta(t), \nu(t))$ have been chosen as described in [1].

The reference system, as described in the previous section, consists of three uncoupled second order differential equations. The constants describing the reference system dynamics are

$$\omega_0 = 0.2 I_3, \quad \zeta = \begin{bmatrix} 0.7 & 0 & 0 \\ 0 & 0.7 & 0 \\ 0 & 0 & 0.45 \end{bmatrix}. \quad (44)$$

Other constants chosen for the controller include $\Gamma_1 = 10 I_3$, $\Gamma_2 = 10 I_{12}$, $K_1 = K_2 = 45 I_3$, $\sigma_1 = \sigma_2 = 0.01$, and

$$Q_1 = Q_2 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 5 \end{bmatrix}. \quad (45)$$

The following results were obtained for the adaptive controller implemented with these characteristics.

A. Circular Trajectory Results

The first maneuver performed using the adaptive controller is a circular trajectory with radius $A = 10$ m and angular velocity $\omega = \frac{2\pi}{75}$ rad/s. The behavior of the vehicle is simulated for 75 s, or one complete cycle. The corresponding trajectory is shown in Fig. 3. The vehicle converges quickly to the desired trajectory while staying relatively close to the reference system. However, during the transient stage of the vehicle the reference system requires the vehicle to proceed toward the perimeter using a large

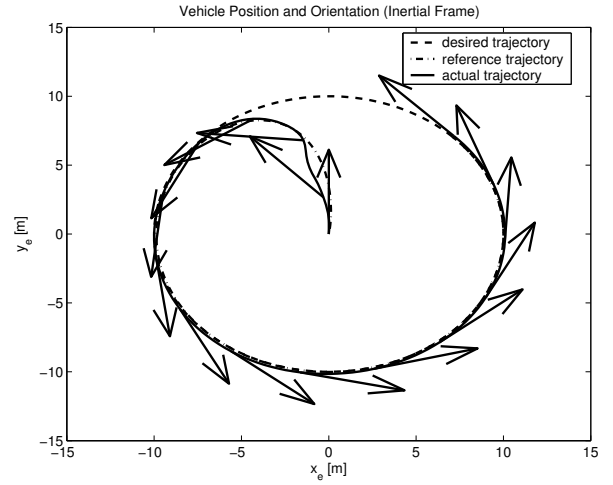


Fig. 3. Circular trajectory

sway force. This is because the second order oscillator reference system for the orientation is decoupled from the surge and sway reference systems. Therefore, instead of forcing the reference orientation to face the desired location, the vehicle turns tangent to the perimeter before it reaches it. Not only is this inefficient, but it causes undesirable fluctuations in the control input because of the coupling between sway and yaw, which can be seen for the first 10 s of the simulation, as shown in Fig. 4. The orientation error, $e_\psi(t)$, $t \geq 0$, on the left side of this figure shows similar oscillating behavior. Although the reference system error converges slowly and smoothly to the origin, the real system jumps around as it follows this path. Clearly the reference dynamics do not properly address the coupled system dynamics. After the vehicle reaches the desired trajectory, the error decreases and control input oscillations disappear because this effect is no longer relevant.

The control gains as a function of time are shown in Fig. 5. The transient behavior of the $\Theta(t)$, $t \geq 0$, converges after 25 s. The $W(t)$, $t \geq 0$, gains, on the other hand, take longer to converge.

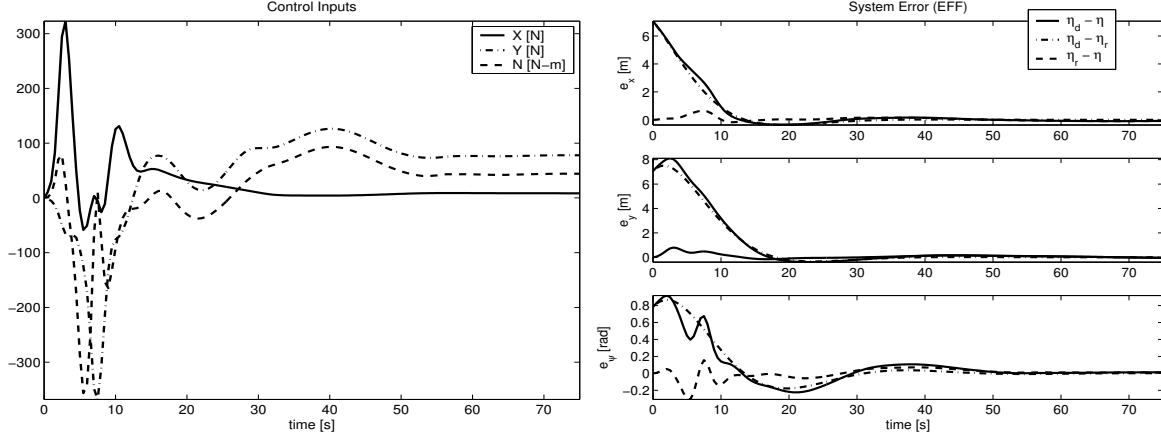


Fig. 4. Inputs, $\tau(t)$, and system error, $e(t)$, for a circular trajectory

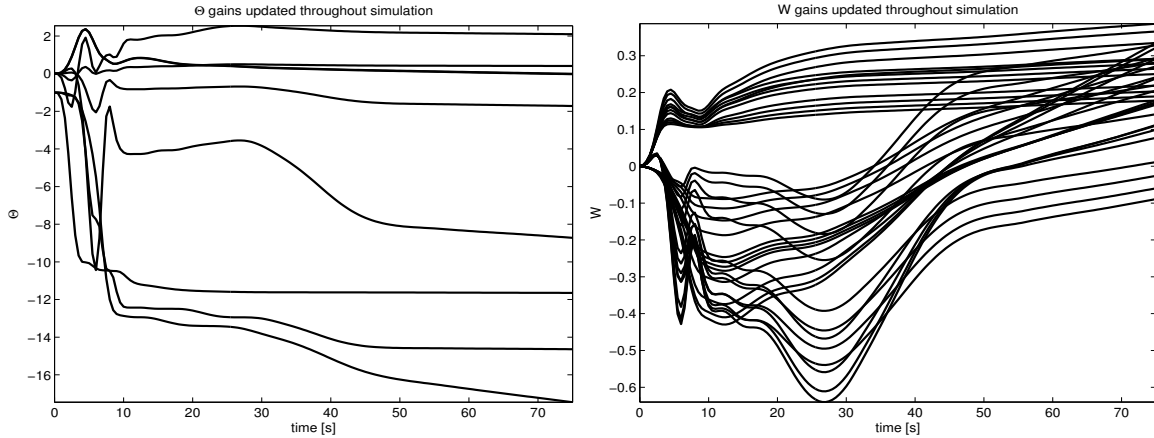


Fig. 5. Adaptive gains, $\Theta(t)$ and $W(t)$, for a circular trajectory

B. Octomorphic Trajectory Results

The octomorphic trajectory is given by the following parametric equations,

$$x_d(t) = 2A \sin\left(\frac{\omega t}{2}\right), \quad x_d(0) = 0, \quad t \geq 0, \quad (46)$$

$$y_d(t) = A \sin(\omega t), \quad y_d(0) = 0. \quad (47)$$

The tangential angle for this curve, corresponding to the desired orientation, is given by

$$\psi_d(t) = \tan^{-1}\left(\frac{\dot{y}_e(t)}{\dot{x}_e(t)}\right) = \tan^{-1}\left(\frac{\cos(\omega t)}{\cos\left(\frac{\omega t}{2}\right)}\right).$$

This maneuver is more complex than the circular trajectory that was tracked in the previous section. Tracking a circular trajectory involves a single turn with a constant turning radius. The octomorphic trajectory involves both left and right turns as well as straight lines.

A similar behavior can be observed for the trajectory tracking in Fig. 6 as for the circular trajectory. During the transient state there were errors and oscillations in the control input as the vehicle attempts to use its sway force to

converge to the desired trajectory. All three control inputs quickly show peak input values, but the oscillating behavior lasts for the first 20 s (see Fig. 7). The sway force and yaw torque reach magnitudes of 1750 N and 1300 N respectively. This goes well outside the achievable range. The uncoupled reference system dynamics that require the vehicle to make unreasonable maneuvers are again to blame for the large control inputs and their unachievable oscillating behavior. The $\Theta(t)$ gains converge by $t = 25$ s, but the $W(t)$ gains continue to modify themselves throughout the entire simulation (see Fig. 8). Also, the $\Theta(t)$ converges to larger values (up to a magnitude of 40), while $W(t)$ remains small (rarely greater than unity). This result is similar to that seen for the circular maneuver.

VI. CONCLUSION

A neural network based model reference adaptive control algorithm was developed which guarantees ultimately bounded tracking error. The addition of the single layer neural network in the control algorithm eliminates the need to know any of the system dynamics, including its structure. This controller was implemented considering a reference

system of three independent second order oscillators. Numerical simulations were performed and showed excellent tracking performance in spite of a strong dependence on the reference system dynamics during the transient region. Improvements could be made through optimization of the reference system dynamics and implementation of actuators' amplitude and rate saturation constraints.

REFERENCES

- [1] T. S. VanZwieten, "Dynamic Simulation and Control of an Autonomous Surface Vehicle," Master's thesis, Florida Atlantic University, Department of Ocean Engineering, December 2003.
- [2] H. K. Khalil, *Nonlinear Systems*. Upper Saddle River, NJ: Prentice-Hall, Inc., 2nd ed., 1996.
- [3] "Nomenclature for Treating the Motion of a Submerged Body Through a Fluid," *The Society of Naval Architects and Marine Engineers*, no. 1-5, 1950.
- [4] T. L. Fossen, *Guidance and Control of Ocean Vehicles*. Chichester, England: John Wiley & Sons Ltd., 1999.
- [5] T. Hayakawa, W. M. Haddad, N. Hovakimyan, and V. Chellaboina, "Neural Network Adaptive Control for Nonlinear Nonnegative Dynamical Systems," in *Proc. Amer. Contr. Conf.*, (Denver, CO), pp. 561-566, June 2003.
- [6] F. L. Lewis, A. Yesildirek, and K. Liu, *Neural Network Control of Robot Manipulators and Nonlinear Systems*. London, UK: The Society of Naval Architects and Marine Engineers, 1999.

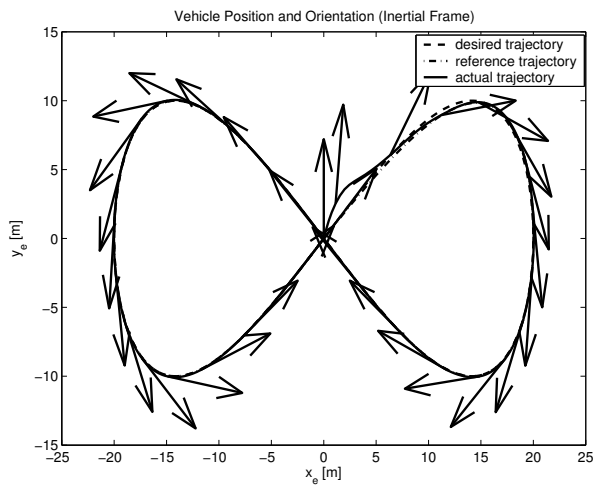


Fig. 6. Octomorph trajectory

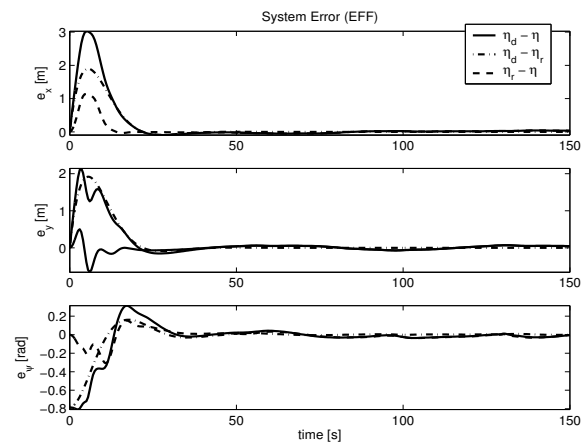
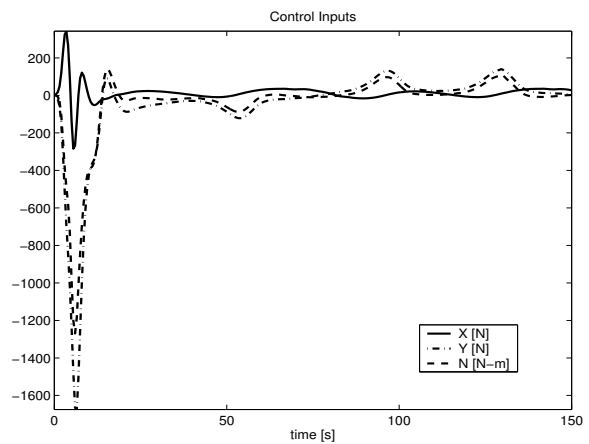


Fig. 7. Inputs, $\tau(t)$, and system error, $e(t)$, for an octomorph trajectory

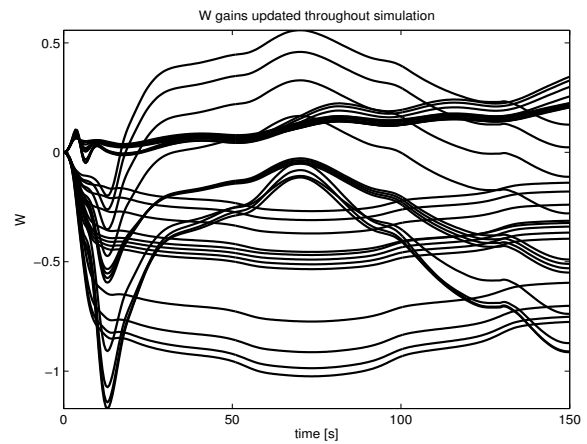
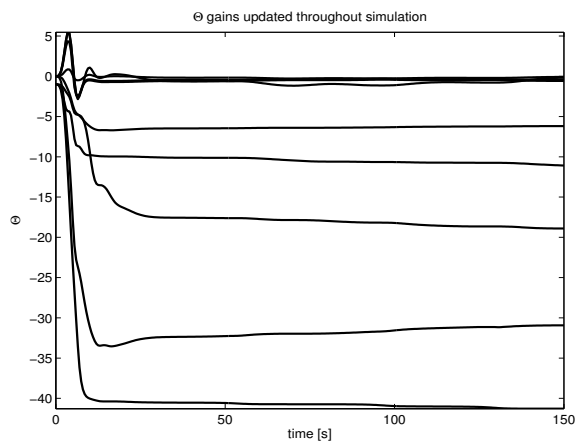


Fig. 8. Adaptive gains, $\Theta(t)$ and $W(t)$, for an octomorph trajectory