Online Approximate Optimal Path-Following for a Mobile Robot

Patrick Walters, Rushikesh Kamalapurkar, Lindsey Andrews, and Warren E. Dixon

***Abstract*—Online approximation of an infinite horizon optimal path-following strategy for a unicycle-type mobile robot is considered. An approximate solution to the optimal control problem is obtained by using an adaptive dynamic programming technique that uses adaptive update laws to estimate the unknown value function. The developed controller overcomes challenges with the approximation of the infinite horizon value function by using an auxiliary function that describes the motion of a virtual target on the desired path. The developed controller guarantees uniformly ultimately bounded (UUB) convergence of the vehicle to a desired path while maintaining a desired speed profile and UUB convergence of the approximate policy to the optimal policy without requiring persistence of excitation.**

I. Introduction

The goal of a mobile robot feedback controller can be classified into three categories: point regulation, trajectory tracking, or path-following. Point regulation refers to the stabilization of a dynamical system about a desired state. Trajectory tracking requires a dynamical system to track a time parametrized reference trajectory. Path-following involves convergence of the system state to a given path at a desired speed profile without temporal constraints. Path-following heuristically yields smoother convergence to the desired path and reduces the risk of control saturation [1]. A path-following control structure can also alleviate difficulties in the control of nonholonomic vehicles [2]. Path-following control is particularly useful for mobile robots with objectives that emphasize path convergence and maintaining a desired speed profile (cf. [3]–[9]).

To improve path-following performance, optimal control techniques have been applied to path-following. The result in [10] combines line-of-sight guidance and model predictive control (MPC) to optimally follow straight line segments. In [11], the MPC structure is used to develop a controller for an omnidirectional robot with dynamics linearized about the desired path. Nonlinear MPC is used in [12] to develop an optimal path-following controller for a general mobile robot model over a finite time-horizon. The survey in [13] cites additional examples of MPC applied to path-following as well as the linear quadratic regulator and dynamic programming control schemes. However, none of these techniques provide online optimal feedback control for path-following while utilizing the system's nonlinear dynamics and guaranteeing stability.

Adaptive dynamic programming-based (ADP-based) techniques have been used to approximate optimal control policies for regulation (cf. [14]–[16]) and trajectory tracking (cf. [17]–[19]). ADP stems from Bellman's principle of optimality where the solution to the Hamilton-Jacobi-Bellman (HJB) equation is approximated using parametric function approximation techniques, and an actor-critic structure is used to estimate the unknown parameters. Various methods have been proposed in [14]–[23] to approximate the solution to the HJB equation. For an infinite horizon regulation problem, function approximation techniques, such as neural networks (NNs), are used to approximate the value function and the optimal policy.

Motivated by the desire to develop an optimal feedback path-following controller, an ADP-based path-following method is developed in this paper for a unicycle-type mobile robot. The path-following technique in this paper generates a virtual target that is tracked by the vehicle. The motion of the virtual target along the given path is described by a predefined state-dependent ordinary differential equation motivated by [1].

For an infinite horizon control problem, the state associated with the virtual target progression is unbounded, which presents several challenges. According to the universal function approximation theorem, a NN is a universal approximator for continuous functions on a compact domain. Since the value function and optimal policy depend on the unbounded path parameter, the domain of the approximation is not compact; hence, to approximate the value function using a NN, an alternate description of the virtual target progression that results in a compact domain for the associated state needs to be developed. In addition, the vehicle requires constant control effort to remain on the path; therefore, any control policy that results in path-following also results in infinite cost, rendering the associated control problem ill-defined.

In this paper, the motion of the virtual target is redefined to remain on a compact domain, and a modified control input is developed. The cost function is formulated in terms of the modified control and redefined virtual target motion, a unique challenge not addressed in previous ADP literature. This formulation admits an admissible control policy, and

Patrick Walters, Rushikesh Kamalapurkar, Lindsey Andrews and Warren E. Dixon are with the Department of Mechanical and Aerospace Engineering, University of Florida, Gainesville, FL, USA. Email: {walters8, rkamalapurkar, landr010, wdixon}@ufl.edu

This research is supported in part by NSF award numbers 1161260, 1217908, ONR grant number N00014-13-1-0151, and a contract with the AFRL Mathematical Modeling and Optimization Institute. Any opinions, findings and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the sponsoring agency.

autonomous value function that can be approximated on a compact domain, facilitating the development of an online approximation to the optimal controller using the ADP framework. A Lyapunov-based stability analysis is presented to establish uniformly ultimately bounded (UUB) convergence of the vehicle to the path while maintaining a desired speed profile and UUB convergence of the approximate policy to the optimal policy.

II. Problem Formulation

This section formulates the path-following problem for a unicycle-type mobile robot. Path-following refers to the problem of converging to a desired path while maintaining a desired speed profile. The desired path is not necessarily parametrized by time, but by some convenient parameter (e.g. path length). The path-following method in this paper utilizes a virtual target that moves along the desired path. The error dynamics are defined kinematically between the virtual target and vehicle. The geometry of the problem is outlined in Figure 1.

Let $\mathcal{I}$ denote an inertial frame. Consider the coordinate system i in $\mathcal{I}$ with its origin and basis vectors $i_1 \in \mathbb{R}^3$ and $i_2 \in \mathbb{R}^3$ in the plane of vehicle motion. The basis vector i_3 is defined as coming out of the plane. The point $P \in \mathbb{R}^3$ on the desired path represents the location of the virtual target. The location of the virtual target is determined by the path parameter $s_p \in \mathbb{R}$. It is convenient to select the arc length as the path parameter for the wheeled mobile robot, since the desired speed can be defined as unit length over unit time. Let $\mathcal{F}$ denote a frame fixed to the virtual target with the origin of the coordinate system f fixed in $\mathcal{F}$ at point P. The basis vector $f_1 \in \mathbb{R}^3$ is the unit tangent vector of the path at P, $f_3 \in \mathbb{R}^3$ is defined as coming out of the plane, and $f_2 = f_3 \times f_1$.[1] Let $\mathcal{B}$ denote a frame fixed to the vehicle with the origin of its coordinate system b at the center of mass $Q \in \mathbb{R}^3$. The basis vector $b_1 \in \mathbb{R}^3$ is the unit velocity vector of the vehicle, $b_3 \in \mathbb{R}^3$ is defined as coming out of the plane, and $b_2 = b_3 \times b_1$. For the subsequent development, assume $\{i_1, i_2, i_3\}$, $\{f_1, f_2, f_3\}$, and $\{b_1, b_2, b_3\}$ are standard bases.

Consider the following vector equation from Figure 1,

$$r_{Q/P} = r_Q - r_P,$$

where $r_Q \in \mathbb{R}^3$ and $r_P \in \mathbb{R}^3$ are the position vectors of points Q and P from the origin of the inertial coordinate system, respectively. The rate of change of $r_{Q/P}$ as viewed by an observer in $\mathcal{I}$ and expressed in the coordinate system f is given as

$$v^f_{Q/P} = v^f_Q - v^f_P. \tag{1}$$

The velocity of point P as viewed by an observer in $\mathcal{I}$ and expressed in f is given as

$$v^f_P = \begin{bmatrix} \dot{s}_p & 0 & 0 \end{bmatrix},^T \tag{2}$$

[1]This description of $\mathcal{F}$ is similar to the Frenet-Serret frame and parallel transport frame for this two dimensional problem.

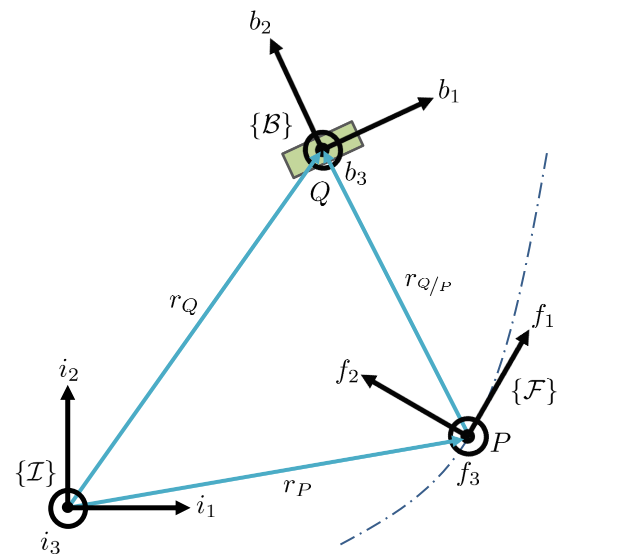

Figure 1. Description of reference frames

where $\dot{s}_p \in \mathbb{R}$ is the velocity of the virtual target along the path. The velocity of point Q as viewed by an observer in $\mathcal{I}$ and expressed in f may be written as

$$v^f_Q = R^f_b v^b_Q, \tag{3}$$

where $R^f_b : \mathbb{R} \to \mathbb{R}^{3\times 3}$ is a transformation from b to f. The velocity of the vehicle as viewed by an observer in $\mathcal{I}$ expressed in b is $v^b_Q = \begin{bmatrix} v & 0 & 0 \end{bmatrix}^T$ where $v \in \mathbb{R}$ is the velocity of the vehicle. The transformation R^f_b is given as

$$R^f_b = \begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix},$$

where $\theta \in \mathbb{R}$ is the angle between f_1 and b_1. The velocity between points P and Q as viewed by an observer in $\mathcal{I}$ and expressed in f is given as

$$v^f_{Q/P} = {}^{\mathcal{F}}\tfrac{d}{dt} r^f_{Q/P} + {}^{\mathcal{I}}\omega^{\mathcal{F}} \times r^f_{Q/P}. \tag{4}$$

The angular velocity of $\mathcal{F}$ as viewed by an observer in $\mathcal{I}$ expressed in f is given as ${}^{\mathcal{I}}\omega^{\mathcal{F}} = \begin{bmatrix} 0 & 0 & \kappa\dot{s}_p \end{bmatrix}^T$ where $\kappa \in \mathbb{R}$ is the path curvature, and the relative position of the vehicle with respect to the virtual target expressed in f is given as $r^f_{Q/P} = \begin{bmatrix} x & y & 0 \end{bmatrix}^T$. Substituting (2), (3), and (4) into (1) the planar positional error dynamics are given as

$$\begin{aligned} \dot{x} &= (\kappa y - 1)\,\dot{s}_p + v\cos\theta \\ \dot{y} &= -\kappa x \dot{s}_p + v\sin\theta. \end{aligned}$$

The angular velocity of $\mathcal{B}$ as viewed by an observer in $\mathcal{F}$ is given as

$${}^{\mathcal{F}}\omega^{\mathcal{B}} = {}^{\mathcal{F}}\omega^{\mathcal{I}} + {}^{\mathcal{I}}\omega^{\mathcal{B}}. \tag{5}$$

From (5), the planar rotational error dynamic expressed in f is given as

$$\dot{\theta} = -\kappa \dot{s}_p + w,$$

where $w \in \mathbb{R}$ is the angular velocity of the vehicle. The full vehicle error dynamics are given by

$$\begin{aligned} \dot{x} &= \dot{s}_p(\kappa y - 1) + v\cos\theta \\ \dot{y} &= -x\kappa\dot{s}_p + v\sin\theta \\ \dot{\theta} &= w - \kappa\dot{s}_p. \end{aligned} \tag{6}$$

Assumption 1. The path curvature κ is assumed to be bounded, i.e. the desired path is C^2 continuous.

Motivated by the development in [1] the location of the virtual target is determined by

$$\dot{s}_p \triangleq v_{des}\cos\theta + k_1 x, \tag{7}$$

where $v_{des} \in \mathbb{R}$ is a desired positive, bounded and time-invariant speed profile, and $k_1 \in \mathbb{R}$ is an adjustable positive gain.

To facilitate the subsequent control development, we define an auxiliary function $\phi : \mathbb{R} \to (-1, 1)$ as

$$\phi \triangleq \tanh(k_2 s_p), \tag{8}$$

where $k_2 \in \mathbb{R}$ is a positive gain. From (7) and (8), the time derivative of ϕ is

$$\dot{\phi} = k_2\left(1 - \phi^2\right)(v_{des}\cos\theta + k_1 x). \tag{9}$$

Note that the path curvature and desired speed profile can be written as a function of ϕ.

Based on (6) and (7), auxiliary control inputs v_e, $w_e \in \mathbb{R}$ are designed as

$$\begin{aligned} v_e &\triangleq v - v_{ss}, \\ w_e &\triangleq w - w_{ss}, \end{aligned} \tag{10}$$

where $w_{ss} \triangleq \kappa v_{des}$ and $v_{ss} \triangleq v_{des}$ based on the control input required to remain on the path.

Substituting (7) and (10) into (6), and augmenting the system state with (9), the closed-loop system is

$$\begin{aligned} \dot{x} &= \kappa y v_{des}\cos\theta + k_1\kappa x y - k_1 x + v_e\cos\theta \\ \dot{y} &= v_{des}\sin\theta - \kappa x v_{des}\cos\theta - k_1\kappa x^2 + v_e\sin\theta \\ \dot{\theta} &= \kappa v_{des} - \kappa(v_{des}\cos\theta + k_1 x) + w_e \\ \dot{\phi} &= k_2\left(1 - \phi^2\right)(v_{des}\cos\theta + k_1 x). \end{aligned} \tag{11}$$

The closed-loop system in (11) can be rewritten in the following control affine form

$$\dot{\zeta} = f(\zeta) + g(\zeta)u, \tag{12}$$

where $\zeta = \begin{bmatrix} x & y & \theta & \phi \end{bmatrix}^T \in \mathbb{R}^4$ is the state vector, $u = \begin{bmatrix} v_e & w_e \end{bmatrix}^T \in \mathbb{R}^2$ is the control vector, and the locally Lipschitz functions $f : \mathbb{R}^4 \to \mathbb{R}^4$ and $g : \mathbb{R}^4 \to \mathbb{R}^{4\times 2}$ are given by

$$f(\zeta) = \begin{bmatrix} \kappa y v_{des}\cos\theta + k_1\kappa x y - k_1 x \\ v_{des}\sin\theta - \kappa x v_{des}\cos\theta - k_1\kappa x^2 \\ \kappa v_{des} - \kappa(v_{des}\cos\theta + k_1 x) \\ k_2\left(1 - \phi^2\right)(v_{des}\cos\theta + k_1 x) \end{bmatrix}, \tag{13}$$

$$g(\zeta) = \begin{bmatrix} \cos(\theta) & 0 \\ \sin(\theta) & 0 \\ 0 & 1 \\ 0 & 0 \end{bmatrix}.$$

To facilitate the subsequent stability analysis, a subset of the state denoted by $e \in \mathbb{R}^3$ is defined as $e \triangleq \begin{bmatrix} x & y & \theta \end{bmatrix}^T \in \mathbb{R}^3$.

III. Formulation of optimal control problem

The cost functional for the optimal control problem is defined as

$$J(\zeta, u) \triangleq \int_t^\infty r(\zeta(\tau), u(\tau))\, d\tau, \tag{14}$$

where $r : \mathbb{R}^4 \to [0, \infty)$ is the local cost defined as

$$r(\zeta, u) \triangleq \zeta^T \bar{Q}\zeta + u^T R u.$$

In (14), $R \in \mathbb{R}^{2\times 2}$ is a symmetric positive definite matrix, and $\bar{Q} \in \mathbb{R}^{4\times 4}$ is defined as

$$\bar{Q} \triangleq \begin{bmatrix} Q & 0_{3\times 1} \\ 0_{1\times 3} & 0 \end{bmatrix},$$

where $Q \in \mathbb{R}^{3\times 3}$ is a positive definite matrix such that $\underline{q}\left\|\xi_q\right\|^2 \le \xi_q^T Q \xi_q \le \overline{q}\left\|\xi_q\right\|^2, \forall \xi_q \in \mathbb{R}^3$ where $\underline{q}$ and $\overline{q}$ are positive constants. The infinite-time scalar value function $V : \mathbb{R}^4 \to [0, \infty)$ is written as

$$V(\zeta_0) = \min_{u\in\mathcal{U}} \int_t^\infty r(\zeta(\tau), u(\tau))\, d\tau, \tag{15}$$

where $\mathcal{U}$ is the set of admissible control policies and $\zeta_0 = \zeta(t)$.

The objective of the optimal control problem is to determine the optimal policy $u^* : \mathbb{R}^4 \to \mathbb{R}^2$ such that the controller $u = u^*(\zeta)$ minimizes the cost functional in (14) subject to the constraints in (12). Assuming a minimizing policy exists and the value function is continuously differentiable, the Hamiltonian is defined as

$$H \triangleq r(\zeta, u^*) + \frac{\partial V}{\partial \zeta}(f + g u^*). \tag{16}$$

The Hamilton-Jacobi-Bellman (HJB) equation is given as [24]

$$\frac{\partial V}{\partial t} + H = 0, \tag{17}$$

where $\frac{\partial V}{\partial t} = 0$, since there exists no explicit dependence on time.

A closed form solution for the optimal policy is obtained by satisfying a necessary condition of the minimum principal

$$\frac{\partial H}{\partial u^*} = 0$$

and the corresponding sufficient condition

$$\frac{\partial^2 H}{\partial u^{*2}} > 0.$$

The policy is given as

$$u^* = -\frac{1}{2} R^{-1} g^T \left(\frac{\partial V}{\partial \zeta}\right)^T. \tag{18}$$

The analytical expression for the optimal policy in (18) requires knowledge of the value function, which is the solution to the HJB. Given the kinematics in (13), it is unclear how to determine an analytical solution to (17), as is generally the case, since (17) is a nonlinear differential equation; hence, the subsequent development focuses on the development of an approximate solution.

IV. Approximate Solution

The subsequent development is based on a neural network (NN) approximation of the value function and optimal policy, and follows a similar structure to [16]. The development is included here for completeness. Over any compact domain $\chi \subset \mathbb{R}^4$, the value function $V : \mathbb{R}^4 \to [0, \infty)$ can be represented by a single-layer NN with L neurons as

$$V(\zeta) = W^T \sigma(\zeta) + \epsilon(\zeta), \tag{19}$$

where $W \in \mathbb{R}^L$ is the ideal weight vector bounded above by a known positive constant, $\sigma : \mathbb{R}^4 \to \mathbb{R}^L$ is a bounded, continuously differentiable activation function, and $\epsilon : \mathbb{R}^4 \to \mathbb{R}$ is the bounded, continuously differentiable function reconstruction error.

From (18) and (19), the optimal policy can be represented as

$$u^* = -\frac{1}{2} R^{-1} g^T \left(\sigma'^T W + \epsilon'^T\right) \tag{20}$$

where $\sigma' : \mathbb{R}^4 \to \mathbb{R}^{L \times 4}$ and $\epsilon' : \mathbb{R}^4 \to \mathbb{R}^4$ are derivatives with respect to the state. Based on (19) and (20), the value function and optimal policy NN approximations are defined as

$$\hat{V} = \hat{W}_c^T \sigma, \tag{21}$$

$$\hat{u} = -\frac{1}{2} R^{-1} g^T \sigma'^T \hat{W}_a, \tag{22}$$

where $\hat{W}_c, \hat{W}_a \in \mathbb{R}^L$ are estimates of the ideal weight vector W. The weight estimation errors are defined as $\tilde{W}_c \triangleq W - \hat{W}_c$ and $\tilde{W}_a \triangleq W - \hat{W}_a$. The NN approximation of the Hamiltonian is given as

$$\hat{H} = r(\zeta, \hat{u}) + \frac{\partial \hat{V}}{\partial \zeta} (f + g\hat{u}) \tag{23}$$

by substituting (21) and (22) into (16). The Bellman error $\delta \in \mathbb{R}$ is defined as the error between the optimal and approximate Hamiltonian and is given as

$$\delta \triangleq \hat{H} - H, \tag{24}$$

where $H = 0$. Therefore, the Bellman error can be written in a measurable form as

$$\delta = r(\zeta, \hat{u}) + \hat{W}_c{}^T \omega,$$

where $\omega \triangleq \sigma' (f + g\hat{u}) \in \mathbb{R}^L$.

Assumption 2. There exists a set of sampled data points $\{\zeta_j \in \chi | j = 1, 2, \ldots, N\}$ such that $\forall t \in [0, \infty)$,

$$\text{rank}\left(\sum_{j=1}^{N} \frac{\omega_j \omega_j^T}{p_j}\right) = L, \tag{25}$$

where $p_j \triangleq \sqrt{1 + \omega_j^T \omega_j}$ denotes the normalization constant, and ω_j is evaluated at the specified data point ζ_j.

In general, the rank condition in (25) cannot be guaranteed to hold a priori. However, heuristically, the condition can be met by sampling redundant data, i.e., $N \gg L$. Based on Assumption 2, it can be shown that $\sum_{j=1}^{N} \frac{\omega_j \omega_j^T}{p_j} > 0$ such that

$$\underline{c} \left\|\xi_c\right\|^2 \leq \xi_c^T \left(\sum_{j=1}^{n} \frac{\omega_j \omega_j^T}{p_j}\right) \xi_c \leq \overline{c} \left\|\xi_c\right\|^2, \ \forall \xi_c \in \mathbb{R}^4$$

even in the absence of persistent excitation [25], [26].

The adaptive update law for $\hat{W}_c$ in (21) is given by

$$\dot{\hat{W}}_c = -\Gamma \left(\eta_{c1} \frac{\partial \delta}{\partial \hat{W}_c} \frac{\delta}{p} + \frac{\eta_{c2}}{N} \sum_{j=1}^{N} \frac{\partial \delta_j}{\partial \hat{W}_c} \frac{\delta_j}{p_j}\right), \tag{26}$$

where $\eta_{c1}, \eta_{c2} \in \mathbb{R}^{L \times L}$ are positive adaptation gains, $\Gamma \in \mathbb{R}^{L \times L}$ is a constant positive diagonal weighting matrix, $\frac{\partial \delta}{\partial \hat{W}_c} = \omega$ is the regressor matrix, and $p \triangleq \sqrt{1 + \omega^T \omega}$ is a normalization constant. The update law for $\hat{W}_a$ in (22) is given by

$$\dot{\hat{W}}_a = \text{proj}\left\{-\eta_a \left(\hat{W}_a - \hat{W}_c\right)\right\}, \tag{27}$$

where $\eta_a \in \mathbb{R}$ is a positive gain, and $\text{proj}\{\cdot\}$ is a smooth projection operator (see Remark 3.7 in [27]). Using the properties of the projection operator, the policy NN weight estimation errors are bounded above by positive constants.

V. Stability Analysis

To facilitate the subsequent stability analysis, an unmeasurable form of the Bellman error can be written using (16), (23), and (24), as

$$\delta = -\tilde{W}_c^T \omega - \epsilon' f + \frac{1}{2} \epsilon' G \sigma'^T W + \frac{1}{4} \tilde{W}_a^T G_\sigma \tilde{W}_a + \frac{1}{4} \epsilon' G \epsilon'^T, \tag{28}$$

where $G \triangleq gR^{-1}g^T \in \mathbb{R}^{4\times4}$ and $G_\sigma \triangleq \sigma' G \sigma'^T \in \mathbb{R}^{L\times L}$ are symmetric, positive semi-definite matrices. Similarly, at the sampled points the Bellman error can be written as

$$\delta_j = -\tilde{W}_c^T \omega_j + \frac{1}{4}\tilde{W}_a^T G_{\sigma j} \tilde{W}_a + E_j, \tag{29}$$

where $E_j \triangleq \frac{1}{2}\epsilon'_j G_j \sigma_j'^T W + \frac{1}{4}\epsilon'_j G_j \epsilon_j'^T - \epsilon'_j f_j \in \mathbb{R}$.

The function f on any compact set $\chi \subset \mathbb{R}^4$ is Lipschitz continuous, and therefore bounded by

$$\|f(\zeta)\| \leq L_f \|\zeta\|, \ \forall \zeta \in \chi,$$

where L_f is the positive Lipschitz constant, and the normalized regressor in (26) is upper bounded by $\left\|\frac{\omega}{p}\right\| \leq 1$.

The augmented equations of motion in (11) present a unique challenge with respect to the value function V which is utilized as a Lyapunov function in the stability analysis. To prevent penalizing the vehicle progression along the path, the path parameter ϕ is removed from the cost function with the introduction of a positive semi-definite state weighting matrix $\bar{Q}$. However, since $\bar{Q}$ is positive semi-definite, efforts are required to ensure the value function is positive definite. To address this challenge, the fact that the value function can be interpreted as a time-invariant map $V : \mathbb{R}^4 \to [0,\infty)$ or a time-varying map $V_t : \mathbb{R}^3 \times [0,\infty) \to [0,\infty)$ is exploited. Lemma 2 in [19] is used to show that the time-varying map is a positive definite and decrescent function for use as a Lyapunov function. Hence, on any compact set χ the optimal value function $V_t : \mathbb{R}^3 \times [0,\infty) \to \mathbb{R}$ satisfies the following properties

$$V_t(0,t) = 0,$$

$$\underline{\upsilon}(\|e\|) \leq V_t(e,t) \leq \overline{\upsilon}(\|e\|), \tag{30}$$

$\forall t \in [0,\infty)$ and $\forall e \subset \chi$ where $\underline{\upsilon} : [0,\infty] \to [0,\infty)$ and $\overline{\upsilon} : [0,\infty] \to [0,\infty)$ are class $\mathcal{K}$ functions.

To facilitate the subsequent stability analysis, consider the candidate Lyapunov function $V_L : \mathbb{R}^{3+2L} \times [0,\infty) \to [0,\infty)$ given as

$$V_L(Z,t) = V_t(e,t) + \frac{1}{2}\tilde{W}_c^T \Gamma^{-1} \tilde{W}_c + \frac{1}{2}\tilde{W}_a^T \tilde{W}_a.$$

Using (30), the candidate Lyapunov function can be bounded by

$$\underline{\upsilon_L}(\|Z\|) \leq V_L \leq \overline{\upsilon_L}(\|Z\|), \tag{31}$$

where $\underline{\upsilon_L}, \overline{\upsilon_L} : [0,\infty) \to [0,\infty)$ are class $\mathcal{K}$ functions and $Z \triangleq \begin{bmatrix} e^T & \tilde{W}_c^T & \tilde{W}_a^T \end{bmatrix}^T \in \chi \cup \mathbb{R}^{2L}$. To facilitate the subsequent stability analysis, let $\beta \subset \chi \cup \mathbb{R}^{2L}$ be a compact set, and φ_e, φ_c, φ_a, ι_c, ι_a, $\iota \in \mathbb{R}$ denote positive constants defined as

$$\varphi_e \triangleq \underline{q} - \frac{\eta_{c1} \sup_{Z\in\beta} \|\epsilon'\| L_f}{2},$$

$$\varphi_c \triangleq \frac{\eta_{c2}}{N}\underline{c} - \frac{\eta_a}{2} - \frac{\eta_{c1} \sup_{Z\in\beta} \|\epsilon'\| L_f}{2},$$

$$\varphi_a \triangleq \frac{\eta_a}{2},$$

$$\begin{aligned}\iota_c \triangleq \sup_{Z\in\beta} \Bigg\| &\frac{\eta_{c2}}{4N}\sum_{j=1}^{N} \tilde{W}_a^T G_{\sigma j} \tilde{W}_a + \frac{\eta_{c1}}{4}\tilde{W}_a^T G_\sigma \tilde{W}_a \\ &+ \frac{\eta_{c1}}{2}\epsilon' G \sigma'^T W + \frac{\eta_{c1}}{4}\epsilon' G \epsilon'^T \\ &+ \frac{\eta_{c2}}{N}\sum_{j=1}^{N} E_j + \eta_{c1}\epsilon' L_f \Bigg\|,\end{aligned}$$

$$\iota_a \triangleq \sup_{Z\in\beta} \left\| \frac{1}{2} G_\sigma W + \frac{1}{2}\sigma' G \epsilon'^T \right\|,$$

$$\iota \triangleq \sup_{Z\in\beta} \left\| \frac{1}{4}\epsilon' G \epsilon'^T \right\|.$$

When Assumption 2 and the sufficient gain conditions

$$\underline{q} > \frac{\eta_{c1}\overline{\|\epsilon'\|}L_f}{2}, \tag{32}$$

$$\underline{c} > \frac{N\eta_a}{2\eta_{c2}} + \frac{N\eta_{c1}\overline{\|\epsilon'\|}L_f}{2\eta_{c2}} \tag{33}$$

are satisfied, the constant $K \in \mathbb{R}$ is positive and defined as

$$K \triangleq \sqrt{\frac{\iota_c^2}{2\alpha\varphi_c} + \frac{\iota_a^2}{2\alpha\varphi_a} + \frac{\iota}{\alpha}}$$

where $\alpha \triangleq \frac{1}{2}\min\left\{\varphi_e, \frac{\varphi_c}{2}, \frac{\varphi_a}{2}\right\}$.

Theorem 1. *If Assumptions 1 and 2 hold, and the sufficient conditions (32), (33), and*

$$K < \overline{\upsilon_L}^{-1}\left(\underline{\upsilon_L}(r)\right) \tag{34}$$

are satisfied, where $r \in \mathbb{R}$ is the radius of a selected compact set β, then the policy in (22) with the update laws in (26) and (27) guarantee UUB regulation of vehicle to the virtual target and UUB convergence of the approximate policy to the optimal policy.[2]

Proof: The time derivative of the candidate Lyapunov function is

$$\dot{V}_L = \frac{\partial V}{\partial \zeta} f + \frac{\partial V}{\partial \zeta} g\hat{u} - \tilde{W}_c^T \Gamma^{-1} \dot{\hat{W}}_c - \tilde{W}_a^T \dot{\hat{W}}_a.$$

Substituting (17), (26), and (27) yields

$$\begin{aligned}\dot{V}_L = & -e^T Q e - u^* R u^* + \frac{\partial V}{\partial \zeta} g\hat{u} - \frac{\partial V}{\partial \zeta} g u^* \\ & + \tilde{W}_c^T \left[\eta_{c1}\frac{\omega^T}{p}\delta + \frac{\eta_{c2}}{N}\sum_{j=1}^{N} \frac{\omega_j}{p_j}\delta_j \right] \\ & + \tilde{W}_a^T \eta_a \left(\hat{W}_a - \hat{W}_c\right).\end{aligned}$$

[2]The sufficient condition in (34) requires the compact set β to be large enough based on the constant K. The constant K for a given β can be reduced to satisfy the sufficient condition by reducing the function approximation error in (19) and (20). The function approximation error can be decreased by increasing the number of neurons in the neural network.

Using Young's inequality, (19), (20), (22), (28), and (29) yields

$$\dot{V}_L \leq -\varphi_e \|e\|^2 - \varphi_c \left\|\tilde{W}_c\right\|^2 - \varphi_a \left\|\tilde{W}_a\right\|^2 + \iota_c \left\|\tilde{W}_c\right\| + \iota_a \left\|\tilde{W}_{a1}\right\| + \iota. \quad (35)$$

By completing the squares, (35) can be upper bounded as

$$\dot{V}_L \leq -\varphi_e \|e\|^2 - \frac{\varphi_c}{2} \left\|\tilde{W}_c\right\|^2 - \frac{\varphi_a}{2} \left\|\tilde{W}_a\right\|^2 + \frac{\iota_c^2}{2\varphi_c} + \frac{\iota_a^2}{2\varphi_a} + \iota,$$

which can be further upper bounded as

$$\dot{V}_L \leq -\alpha \|Z\|^2, \forall \|Z\| \geq K > 0, \quad (36)$$

for all $Z \in \beta$.

Using (31), (34), and (36) Theorem 4.18 in [28] is invoked to conclude that Z is UUB, and that $\|Z(t)\|$ will decrease until $\|Z(t)\| \leq \underline{v_L}^{-1}(\overline{v_L}(K))$ where $\|Z(t)\|$ will remain. Therefore, the system state will remain in the compact set β. Based on the definition of Z, and the inequalities in (31) and (36), e, $\tilde{W}_c$, $\tilde{W}_a \in \mathcal{L}_\infty$. Since $\phi \in \mathcal{L}_\infty$ by definition in (9), then $\zeta \in \mathcal{L}_\infty$. $\hat{W}_c$, $\hat{W}_a \in \mathcal{L}_\infty$ follows from the definition of W. From (21) and (22), $\hat{V}$, $\hat{u} \in \mathcal{L}_\infty$. From (12), $\dot{\zeta} \in \mathcal{L}_\infty$. By the definition in (24), $\delta \in \mathcal{L}_\infty$. From (26) and (27), $\dot{\hat{W}}_a$, $\dot{\hat{W}}_c \in \mathcal{L}_\infty$.

■

VI. Conclusion

An online approximation of an optimal path-following controller is developed for a mobile robot. Adaptive dynamic programming is used to approximate the solution to the HJB equation without the need for persistence of excitation. A gradient descent adaptive update law approximates the value function. A Lyapunov-based stability analysis proves UUB convergence of the vehicle to the desired path while maintaining the desired speed profile, and UUB convergence of the approximate policy to the optimal policy.

References

[1] L. Lapierre, D. Soetanto, and A. Pascoal, "Non-singular path-following control of a unicycle in the presence of parametric modeling uncertainties," *Int. J. Robust Nonlinear Control*, vol. 16, pp. 485–503, 2003.

[2] P. Morin and C. Samson, "Motion control of wheeled mobile robots," in *Springer Handbook of Robotics*. Springer Berlin Heidelberg, 2008, pp. 799–826.

[3] L. Lapierre and D. Soetanto, "Nonlinear path-following control of an AUV," *Ocean Eng.*, vol. 34, pp. 1734–1744, 2007.

[4] P. Encarnacao and A. Pascoal, "3D path following for autonomous underwater vehicle," in *Proc. IEEE Conf. Decis. Control*, 2000.

[5] L. Lapierre and B. Jouvencel, "Robust nonlinear path-following control of an AUV," *IEEE J. Ocean. Eng.*, vol. 33, no. 2, pp. 89–102, 2008.

[6] A. Morro, A. Sgorbissa, and R. Zaccaria, "Path following for unicycle robots with an arbitrary path curvature," *IEEE Trans. Robot.*, vol. 27, no. 5, pp. 1016–1023, 2011.

[7] D. Dacic, D. Nesic, and P. Kokotovic, "Path-following for nonlinear systems with unstable zero dynamics," *IEEE Trans. Autom. Control*, vol. 52, no. 3, pp. 481–487, 2007.

[8] A. Aguiar and J. Hespanha, "Trajectory-tracking and path-following of underactuated autonomous vehicles with parametric modeling uncertainty," *IEEE Trans. Autom. Control*, vol. 52, no. 8, pp. 1362–1379, 2007.

[9] N. Sarkar, X. Yun, and V. Kumar, "Dynamic path following: a new control algorithm for mobile robots," in *Proc. IEEE Conf. Decis. Control*, 1993, pp. 2670–2675 vol.3.

[10] E. Borhaug, K. Pettersen, and A. Pavlov, "An optimal guidance scheme for cross-track control of underactuated underwater vehicles," in *Proc. 14th Mediterr. Conf. Control and Autom.*, 2006, pp. 1–5.

[11] K. Kanjanawanishkul and A. Zell, "Path following for an omnidirectional mobile robot based on model predictive control," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2009, pp. 3341–3346.

[12] T. Faulwasser and R. Findeisen, "Nonlinear model predictive path-following control," in *Nonlinear Model Predictive Control*, L. Magni, D. Raimondo, and F. Allgöwer, Eds. Springer, 2009, vol. 384, pp. 335–343.

[13] P. Sujit, S. Saripalli, and J. Borges Sousa, "Unmanned aerial vehicle path following: A survey and analysis of algorithms for fixed-wing unmanned aerial vehicles," *IEEE Control Syst. Mag.*, vol. 34, no. 1, pp. 42–59, Feb 2014.

[14] K. Vamvoudakis and F. Lewis, "Online actor-critic algorithm to solve the continuous-time infinite horizon optimal control problem," *Automatica*, vol. 46, no. 5, pp. 878–888, 2010.

[15] S. Bhasin, R. Kamalapurkar, M. Johnson, K. Vamvoudakis, F. L. Lewis, and W. Dixon, "A novel actor-critic-identifier architecture for approximate optimal control of uncertain nonlinear systems," *Automatica*, vol. 49, no. 1, pp. 89–92, 2013.

[16] R. Kamalapurkar, P. Walters, and W. E. Dixon, "Concurrent learning-based approximate optimal regulation," in *Proc. IEEE Conf. Decis. Control*, Florence, IT, Dec. 2013, pp. 6256–6261.

[17] T. Dierks and S. Jagannathan, "Output feedback control of a quadrotor UAV using neural networks," *IEEE Trans. Neural Netw.*, vol. 21, pp. 50–66, 2010.

[18] H. Zhang, L. Cui, X. Zhang, and Y. Luo, "Data-driven robust approximate optimal tracking control for unknown general nonlinear systems using adaptive dynamic programming method," *IEEE Trans. Neural Netw.*, vol. 22, no. 12, pp. 2226–2236, 2011.

[19] R. Kamalapurkar, H. Dinh, S. Bhasin, and W. E. Dixon, "Approximate optimal trajectory tracking for continuous time nonlinear systems," *Automatica*, to appear (see also arXiv:1301.7664).

[20] R. Beard, G. Saridis, and J. Wen, "Galerkin approximations of the generalized Hamilton-Jacobi-Bellman equation," *Automatica*, vol. 33, pp. 2159–2178, 1997.

[21] M. Abu-Khalaf and F. Lewis, "Nearly optimal HJB solution for constrained input systems using a neural network least-squares approach," in *Proc. IEEE Conf. Decis. Control*, Las Vegas, NV, 2002, pp. 943–948.

[22] D. Vrabie and F. Lewis, "Neural network approach to continuous-time direct adaptive optimal control for partially unknown nonlinear systems," *Neural Netw.*, vol. 22, no. 3, pp. 237 – 246, 2009.

[23] K. Vamvoudakis and F. Lewis, "Online synchronous policy iteration method for optimal control," in *Recent Advances in Intelligent Control Systems*, W. Yu, Ed. Springer, 2009, pp. 357–374.

[24] D. Kirk, *Optimal Control Theory: An Introduction*. Dover, 2004.

[25] G. V. Chowdhary and E. N. Johnson, "Theory and flight-test validation of a concurrent-learning adaptive controller," *J. Guid. Control Dynam.*, vol. 34, no. 2, pp. 592–607, March 2011.

[26] G. Chowdhary, T. Yucelen, M. Mühlegg, and E. N. Johnson, "Concurrent learning adaptive control of linear systems with exponentially convergent bounds," *Int. J. Adapt. Control Signal Process.*, vol. 27, no. 4, pp. 280–301, 2013.

[27] W. E. Dixon, A. Behal, D. M. Dawson, and S. Nagarkatti, *Nonlinear Control of Engineering Systems: A Lyapunov-Based Approach*. Birkhauser: Boston, 2003.

[28] H. K. Khalil, *Nonlinear Systems*, 3rd ed. Prentice Hall, 2002.