ADP based trajectory-tracking control via backstepping method for underactuated AUV with unknown dynamics

Gaofeng Che (chexuping@163.com)  
Henan Normal University  
Zhen Yu  
Xiamen University

Research Article

Keywords: Adaptive dynamic programming (ADP), Backstepping approach, Trajectory-tracking control, Autonomous underwater vehicle (AUV)

Posted Date: December 8th, 2021

DOI: https://doi.org/10.21203/rs.3.rs-1115471/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License.
Read Full License
ADP based trajectory-tracking control via backstepping method for underactuated AUV with unknown dynamics

Gaofeng Che$^{1*}$ and Zhen Yu$^{2*}$

$^1*$College of Computer and Information Engineering, Henan Normal University, JianShe East Road, Xinxiang, 4530001, Henan, China.
$^2$Department of Automation, Xiamen University, Xiang’an South Road, Xiamen, 361102, Fujian, China.

*Corresponding author(s). E-mail(s): chexuping@163.com; yuzhen20@xmu.edu.cn;
†These authors contributed equally to this work.

Abstract
This paper investigates trajectory-tracking control problem for underactuated autonomous underwater vehicles (AUV) with unknown dynamics. Different from existing adaptive dynamic programming (ADP) schemes, our proposed control scheme can achieve high-level system stability and tracking control accuracy. Firstly, the backstepping approach is introduced into the kinematic model of underactuated AUV and produces a virtual velocity control which is taken as the desired velocity input of the dynamic model of underactuated AUV. Secondly, the error tracking system is constructed according to the dynamic model of underactuated AUV. Thirdly, the critic neural network and the action neural network are employed to transform the trajectory-tracking control problem into optimal control problem based on policy iteration algorithm. At last simulation results are given to verify the effectiveness of the control scheme proposed in this paper.

Keywords: Adaptive dynamic programming (ADP), Backstepping approach, Trajectory-tracking control, Autonomous underwater vehicle (AUV)
1 Introduction

The motion control of autonomous underwater vehicle (AUV) has received more and more attention due to its wide applications. Many control techniques are proposed to solve the motion problems of AUV [1]. Trajectory-tracking control is one fundamental functionality of motion control for AUV [2, 3]. Many control schemes are developed to solve the trajectory-tracking problem for underactuated or fully-actuated AUV, such as output feedback control method [4, 5], adaptive control method [6], fuzzy control method [7], model predictive control method [8], dynamic surface control method [9] and so on.

However, AUV is a complex system and exists the nonlinear dynamics, the unknown disturbances and the faults [10], which make the AUV unstable. In recent years, the neural network is introduced to solve the trajectory-tracking problem for AUV due to its strong nonlinear fitting ability, strong robustness and strong self-learning ability. The adaptive neural network tracking controller [11] constructed with the radial basic function neural network is developed for underwater vehicle with unknown nonlinear function. The deterministic policy gradient [12] using multi Pseudo Q-learning method is proposed to reduce the overestimation of action-value function for underactuated AUV with unknown dynamics and constrained inputs. The neural networks are employed to approximate the uncertain underactuated vessel dynamics and external disturbances [13]. The radial basis function neural network is employed to deal with the uncertain nonlinear dynamics for the underactuated vessel [14]. The neural network is designed to solve the cooperative path following problem for a fleet of underactuated AUV with uncertain nonlinear dynamics [15]. The neural network is employed to estimate the unknown nonlinear dynamics of AUV [16]. A robust adaptive control scheme based on fully-tuned fuzzy neural network is proposed to solve the trajectory and attitude problem for unmanned underwater vehicle with thruster dynamics and unknown disturbances [17].

As an optimization method, adaptive dynamic programming (ADP) scheme has received more attention [19]. ADP scheme has been introduced to compensate for unknown dynamics, such as external disturbances, control input nonlinearities and model uncertainties for nonlinear systems [20, 21]. Policy iteration [22–26] and value iteration [27, 28] are two primary iteration ADP algorithms. In this work, ADP is introduced to solve the trajectory-tracking problem for underactuated AUV with bounded time-varying disturbances and the higher-level tracking accuracy is received. The main contributions of this paper can be stated as below:

(i) The virtual velocity control input is designed based on the kinematic model of underactuated AUV using the backstepping method and Lyapunov stability theorem. It is taken as the reference velocity input of the dynamic model of underactuated AUV to reduce the jitter in the system states of error tracking system.
The error tracking system of underactuated AUV are established by augmented matrix method. The policy iteration ADP scheme based on the critic-action neural networks is employed to transform trajectory-tracking control problem for underactuated AUV into optimal control problem. The ADP structure is different from that is proposed in [29]. The weight values of the critic-action neural networks are updated online.

The unknown dynamics [19, 29] is considered in underactuated AUV. The stability of error tracking system is analyzed based on the Lyapunov stability theorem. In order to verify the effectiveness of the method proposed in this paper, the simulation results of the compared method proposed in [30] are given.

This paper is organized as follows. In Section 2, The kinematic model and the dynamic model of underactuated AUV based on the reference model of AUV in [31] are given. In Section 3, ADP optimal controller is designed via backstepping method. In Section 4, two simulation examples are provided. Section 5 gives the conclusions.

2 Problem formulation and mathematical model of AUV

The mathematical model of underactuated AUV is shown in Fig 1. Two coordinate systems are used that one is the universal frame \( \{O_e - X_eY_eZ_e\} \) and the other is the body-fixed frame \( \{O_b - X_bY_bZ_b\} \).

![Fig. 1 Underactuated AUV model system.](image)

2.1 Kinematic model of AUV

The kinematic model of underactuated AUV is given as follows:

\[
\dot{\eta} = J(\eta)\xi 
\]
where the position vector \( \eta = [x \ y \ z \ \phi \ \theta \ \psi]^T \) consists of vehicle location \( x \), vehicle location \( y \), vehicle location \( z \), roll angle \( \phi \), pitch angle \( \theta \), yaw angle \( \psi \) with respect to the universal frame; the velocity vector \( \xi = [u \ v \ w \ p \ q \ r]^T \) consists of surge \( u \), sway \( v \), heave \( w \), roll rate \( p \), pitch rate \( q \), yaw rate \( r \) with respect to the body-fixed frame; \( J(\eta) \in \mathbb{R}^{6 \times 6} \) is the coordinate transformation matrix.

### 2.2 Dynamic model of AUV

The dynamic model of underactuated AUV is established as follows:

\[
M \ddot{\xi} + C(\xi)\dot{\xi} + D(\xi)\xi + g(\eta) + \tau_e = \tau
\]  

where \( M \in \mathbb{R}^{6 \times 6} \), \( C(\xi) \in \mathbb{R}^{6 \times 6} \), \( D(\xi) \in \mathbb{R}^{6 \times 6} \), \( \tau_e \in \mathbb{R}^{6 \times 1} \), \( \tau \in \mathbb{R}^{6 \times 1} \) and \( g(\eta) \in \mathbb{R}^{6 \times 1} \) are the inertia matrix, the Coriolis and centripetal matrix, the hydrodynamic damping matrix, the force vector introduced by the unknown dynamics, the thrust force vector and the gravity and buoyancy forces vector respectively.

The equation (2) can be transformed into as follows:

\[
\dot{\xi} = M^{-1}(-C(\xi)\dot{\xi} - D(\xi)\xi - g(\eta) - \tau_e + \tau)
\]  

### 2.3 Problem formulation

The position error of underacuated AUV is defined as follows:

\[
e_{\eta} = \eta - \eta_d
\]  

where \( \eta \) is the virtual position control input generated by the backstepping approach.

The time derivative of \( e_{\eta} \) is given as follows:

\[
\dot{e}_{\eta} = \dot{\eta} - \dot{\eta}_d = J(\eta)\dot{\xi} - J(\eta_d)\dot{\xi}_d
\]  

The velocity error of underactuated AUV is defined as follows:

\[
e_{\xi} = \xi - \xi_{bs}
\]  

where \( \xi_{bs} \) is the virtual velocity control input generated by the backstepping approach.

The time derivative of \( e_{\xi} \) is given as follows:

\[
\dot{e}_{\xi} = \dot{\xi} - \dot{\xi}_{bs}
\]  

### 3 ADP optimal control design via backstepping method

The block diagram of the proposed control scheme is show in Figure 2.
3.1 Kinematic control via backstepping method

Considering the kinematic model of underactuated AUV, the objective is to make $e_\eta = 0$ as $t \to \infty$. According to equations (4), (5) and (6), we can get the virtual control input $\xi_{bs}$ that is given as follows:

$$\xi_{bs} = J^{-1}(e_\eta + \eta_d)(J(\eta_d)\xi_d - k_1 e_\eta)$$

where $k_1 > 0$ and $J$ is invertible.

The time derivative of equation (8) is given as follows:

$$\dot{\xi}_{bs} = J^{-1}(e_\eta + \eta_d)(\dot{J}(\eta_d)\xi_d + J(\eta_d)\dot{\xi}_d - \dot{J}(e_\eta + \eta_d)\xi_{bs} - k_1 \dot{e}_\eta)$$

$$= J^{-1}(e_\eta + \eta_d)(\dot{J}(\eta_d)\xi_d + J(\eta_d)\dot{\xi}_d - \dot{J}(e_\eta + \eta_d)J^{-1}(e_\eta + \eta_d)$$

$$(J(\eta_d)\xi_d - k_1 e_\eta) - k_1 \dot{e}_\eta)$$

$$= \Theta_5 + k_1 J^{-1}(e_\eta + \eta_d)\dot{J}(e_\eta + \eta_d)J^{-1}(e_\eta + \eta_d)\dot{e}_\eta - k_1 J^{-1}(e_\eta + \eta_d)\dot{e}_\eta$$

where $\Theta_5 = J^{-1}(e_\eta + \eta_d)(\dot{J}(\eta_d)\xi_d + J(\eta_d)\dot{\xi}_d - \dot{J}(e_\eta + \eta_d)J^{-1}(e_\eta + \eta_d)J(\eta_d)\xi_d)$. (9)

\textbf{Lemma 1} Under the proposed virtual control input $\xi_{bs}$ and the kinematic model of underactuated AUV (1), $e_\eta$ is UUB.

\textit{Proof} Choose the Lyapunov function as follows:

$$V_1 = \frac{1}{2} e_\eta^T e_\eta$$

The derivative of $V_1$ is calculated as follows:

$$\dot{V}_1 = e_\eta^T (\dot{\eta} - \dot{\eta}_d)$$

$$= e_\eta^T (J(\eta)\dot{\xi} - J(\eta_d)\dot{\xi}_d)$$

$$= e_\eta^T (J(\eta)\xi_{bs} - J(\eta_d)\xi_d)$$

$$= e_\eta^T (J(\eta)J^{-1}(\eta)(J(\eta_d)\xi_d - k_1 e_\eta) - J(\eta_d)\xi_d)$$

$$= e_\eta^T (-k_1 e_\eta)$$

$$= -k_1 e_\eta^T e_\eta$$

$$\leq 0$$

From equation (11), the time derivative of Lyapunov function is negative semi-definite. It can be concluded that $V_1$ is bounded and $e_\eta$ is UUB. This completes the proof. \qed
According to equations (4)–(9), the dynamic model of underactuated AUV (2) can be transformed as follows:

\[
\tau = M \dot{\xi} - k_1 MJ^{-1}(e_\eta + \eta_d) \dot{e}_\eta + (C(e_\xi + \xi_{bs}) + D(e_\xi + \xi_{bs})) e_\xi \\
+ k_1 (MJ^{-1}(e_\eta + \eta_d) \dot{J}(e_\eta + \eta_d) J^{-1}(e_\eta + \eta_d) - (C(e_\xi + \xi_{bs}) + D(e_\xi + \xi_{bs})) J^{-1}(e_\eta + \eta_d)) e_\eta + \Theta_6 + \tau_e
\]

(12)

where \( \Theta_6 = M \Theta_5 + (C(e_\xi + \xi_{bs}) + D(e_\xi + \xi_{bs})) J^{-1}(e_\eta + \eta_d) \xi_d + g(e_\eta + \eta_d) \).

Let \( Z = [e_\xi \quad e_\eta]^T \), we can get the error tracking system as follows:

\[
[M(Z)] \dot{Z} + [C(Z)] Z + [D(Z)] = \Gamma
\]

(13)

where \( [M(Z)] = \begin{bmatrix} M & -k_1 MJ^{-1}(e_\eta + \eta_d) \\ 0^{6 \times 6} & I_6^{6 \times 6} \end{bmatrix} \); \( \Gamma = \begin{bmatrix} \tau_e \\ 0^{6 \times 1} \end{bmatrix} \);

\[
[D(Z)] = \begin{bmatrix} \Theta_8 \\ J(\eta_d) \xi_d - J(e_\eta + \eta_d) J^{-1}(e_\eta + \eta_d) \xi_d \end{bmatrix}; \quad \Theta_8 = \Theta_6 + \tau_e - M \xi_d - C(\xi_d) \xi_d - D(\xi_d) \xi_d - g(\eta_d);
\]

\[
[C(Z)] = \begin{bmatrix} C(e_\xi + \xi_{bs}) + D(e_\xi + \xi_{bs}) \\ - J(e_\eta + \eta_d) \end{bmatrix}; \quad \Theta_7 = k_1 J(e_\eta + \eta_d) J^{-1}(e_\eta + \eta_d);
\]

\[
k_1 (MJ^{-1}(e_\eta + \eta_d) \dot{J}(e_\eta + \eta_d) J^{-1}(e_\eta + \eta_d) - (C(e_\xi + \xi_{bs}) + D(e_\xi + \xi_{bs})) J^{-1}(e_\eta + \eta_d)); \quad \tau_e = \tau - \tau_d \text{ and } \tau_d = M \xi_d + C(\xi_d) \xi_d + D(\xi_d) \xi_d + g(\eta_d).
\]

### 3.2 ADP optimal control design

The cost function is defined as follows:

\[
V_2(Z(t), \Gamma) = \int_t^\infty e^{\gamma(t-\sigma)} U(Z(\sigma), \Gamma(\sigma)) d\sigma
\]

(14)

where \( \gamma \) is the discount factor and \( 0 \leq \gamma < 1 \); \( U(Z(\sigma), \Gamma(\sigma)) \) is the utility function and \( U(Z(\sigma), \Gamma(\sigma)) = Z^T(\sigma) Q Z(\sigma) + \Gamma^T(\sigma) R \Gamma(\sigma); Q \in \mathbb{R}^{12 \times 12} \) and \( R \in \mathbb{R}^{12 \times 12} \) are the positive definite matrices.

If the cost function (14) is continuously differentiable, then the Hamilton-Jacobi-Bellman (HJB) function can be derived as follows:

\[
H(Z, \Gamma, \nabla V_2(Z)) = U(Z, \Gamma) + (\nabla V_2(Z))^T (-[M(Z)]^{-1}([C(Z)] Z + [D(Z)] - \Gamma)) - \gamma V_2(Z)
\]

(15)

where \( \nabla V_2(Z) = \nabla V_2(Z(t), \Gamma) = \frac{\partial V_2}{\partial Z} \) and \( V_2(0) = 0 \).

According to the Bellman’s optimality principle, the optimal cost function can be represented as follows:

\[
V_2^*(Z) = \min_{\Gamma \in \Omega} \int_t^\infty e^{\gamma(t-\sigma)} U(Z(\sigma), \Gamma(\sigma)) d\sigma
\]

(16)
The optimal cost function (16) satisfies the HJB equation (15) as follows:

$$\min_{\Gamma} H(Z, \Gamma, \nabla V_2^*(Z)) = 0$$

(17)

The optimal control law is

$$\Gamma^*(Z) = -\frac{1}{2}R^{-1}([M(Z)]^{-1})^T \nabla V_2^*(Z)$$

(18)

Considering the optimal controller (18), the HJB equation can be transformed as follows:

$$0 = U(Z, \Gamma^*) + (\nabla V_2^*(Z))^T(-[M(Z)]^{-1}([C(Z)]Z + [D(Z)] - \Gamma^*)) - \gamma V_2^*(Z)$$

$$+ (\nabla V_2^*(Z))^T([C(Z)]Z[D(Z)]) - \gamma V_2^*(Z)$$

(19)

where $V_2^*(0) = 0$.

### 3.3 Critic neural network design

The critic neural network is designed to approximate $V_2^*(Z)$ as follows:

$$V_2^*(Z) = W_1^T \varpi_1(Z) + \delta_1(Z)$$

(20)

where $W_1$ is the unknown ideal constant weights of the critic neural network; $\varpi_1(Z)$ is the activation function vector of the critic neural network; and $\delta_1(Z)$ is the approximate error of the critic neural network.

The derivative of equation (20) is represented as follows:

$$\nabla V_2^*(Z) = \nabla \varpi_1^T(Z)W_1 + \nabla \delta_1(Z)$$

(21)

where $\nabla \varpi_1(Z) \triangleq \frac{\partial \varpi_1(Z)}{\partial Z}$ and $\nabla \delta_1(Z) \triangleq \frac{\partial \delta_1(Z)}{\partial Z}$.

Let $\hat{W}_1$ be the approximation of $W_1$, the approximation of $V_2^*(Z)$ can be represented as follows:

$$\hat{V}_2(Z) = \hat{W}_1^T \varpi_1(Z)$$

(22)

Then, the HJB function can be derived as follows:

$$H(Z, \hat{\Gamma}, \hat{W}_1) = \hat{W}_1^T \nabla \varpi_1(Z) (-[M(Z)]^{-1}([C(Z)]Z + [D(Z)] - \hat{\Gamma})) + U(Z, \hat{\Gamma})$$

$$- \gamma \hat{W}_1^T \varpi_1(Z)$$

(23)

The square residual error $E_1(\hat{W}_1)$ is defined as follows:

$$E_1(\hat{W}_1) = \frac{1}{2}e_{w_1}^T e_{w_1}$$

(24)
Given any admissible control policy $\hat{\Gamma}$, it is desired to select $\hat{W}_1$ to minimize $E_1(\hat{W}_1)$. The weight updating law based on the gradient descent algorithm is given as follows:

$$\dot{\hat{W}}_1 = -\alpha_1 \varrho_1 (\varrho_1^T \hat{W}_1 + Z^T Q Z + \hat{\Gamma}^T R \hat{\Gamma})$$

(25)

where $\alpha_1$ is the adaptive gain of the critic neural network and $\varrho_1 = \frac{\varrho}{\varrho + 1}$ and $\varrho = \nabla \varphi_1 [-M(Z)]^{-1}([C(Z)]Z + [D(Z)] - \hat{\Gamma}) - \gamma \varphi_1(Z)$.

According to the definition of $\varrho_1$, there exist positive constants $\varrho_{1M} > 1$ and $\varrho_{1m} > 0$ such that $\varrho_{1m} \leq \|\varrho_1\| \leq \varrho_{1M}$.

The equation (25) can be transformed as follows:

$$\dot{\tilde{W}}_1 = \dot{\hat{W}}_1 - \dot{\tilde{W}}_1$$

$$= -\alpha_1 \varrho_1 (\varrho_1^T \tilde{W}_1 - \nabla \delta_1(Z)(-[M(Z)]^{-1}([C(Z)]Z + [D(Z)] - \hat{\Gamma})))$$

(26)

where $\varepsilon_{w_1} = -\nabla \delta_1(Z)(-[M(Z)]^{-1}([C(Z)]Z + [D(Z)] - \hat{\Gamma}))$.

### 3.4 Action neural network design

The action neural network is designed to approximate $\Gamma^*$ as follows:

$$\Gamma^* = W^T_2 \varphi_2(Z) + \delta_2(Z)$$

(27)

where $W_2$ is the unknown ideal constant weights of the action neural network; $\varphi_2(Z)$ is the activation function vector of the action neural network; $\delta_2(Z)$ is the approximation error of the action neural network.

Let $\hat{W}_2$ be the approximation of $W_2$, the actual output can be expressed as follows:

$$\hat{\Gamma} = \hat{W}_2^T \varphi_2(Z)$$

(28)

According to equations (18) and (28), the feedback error is defined as follows:

$$e_{w_2} = \hat{W}_2^T \varphi_2 + \frac{1}{2} R^{-1}([M(Z)]^{-1})^T \nabla \varphi_1^T(Z) \hat{W}_1$$

(29)

The square residual error $E_2(\hat{W}_2)$ is defined as follows:

$$E_2(\hat{W}_2) = \frac{1}{2} e_{w_2}^T e_{w_2}$$

(30)

It is desired to select $\hat{W}_2$ to minimize the objective function. The weight updating law based on the gradient descent algorithm is given as follows:

$$\dot{\hat{W}}_2 = -\alpha_2 \varphi_2(Z)(\varphi_2^T(Z)\hat{W}_2 + \frac{1}{2} R^{-1}[M(Z)]^{-1} \nabla \varphi_1^T(Z) \hat{W}_1)^T$$

(31)

where $\alpha_2$ is the adaptive gain of the action neural network and $\alpha_2 > 0$. 
The equation (31) can be rewritten as follows:

\[
\begin{aligned}
\dot{W}_2 &= \dot{W}_2 - \dot{\hat{W}}_2 \\
&= -\alpha_2 \varpi_2(Z)(\varpi_2^T(Z)W_2) + \frac{1}{2} R_1^{-1} [M(Z)]^{-1}\nabla \varpi_1^T(Z) \hat{W}_1 - \delta_2(Z) \\
&+ R_1^{-1} [M(Z)]^{-1} \nabla \delta_1(Z) \\
&= -\alpha_2 \varpi_2(Z)(\varpi_2^T(Z)W_2) + \frac{1}{2} R_1^{-1} [M(Z)]^{-1}\nabla \varpi_1^T(Z) \hat{W}_1 + \varepsilon_{w_2}
\end{aligned}
\]

(32)

where \(\varepsilon_{w_2} = -(\delta_2(Z) - R_1^{-1}[M(Z)]^{-1}\nabla \delta_1(Z)\)).

### 3.5 Stability analysis

According to the optimal control \(\hat{\Gamma}\) (28) and the error tracking system (13), we have

\[
[M(Z)]\dot{Z} + [C(Z)]Z + [D(Z)] = \hat{W}_2^T \varpi_2(Z)
\]

(33)

The error tracking system (33) can be transformed using equation (27) as follows:

\[
[M(Z)]\dot{Z} + [C(Z)]Z + [D(Z)] = \hat{W}_2^T \varpi_2(Z) + \Gamma^* - \delta(Z)
\]

(34)

**Assumption 2** The unknown ideal constant weights \(W_1\) and \(W_2\) satisfy that \(\|W_1\| \leq W_{1M}\) and \(\|W_2\| \leq W_{2M}\) respectively. \(W_{1M}\) and \(W_{2M}\) are the positive constants.

**Assumption 3** The approximation errors of the neural networks \(\delta_1\) and \(\delta_2\) satisfy that \(\delta_{1m} \leq \|\delta_1\| \leq \delta_{1M}\) and \(\delta_{2m} \leq \|\delta_2\| \leq \delta_{2M}\) respectively. \(\delta_{1m}, \delta_{1M}, \delta_{2m}\) and \(\delta_{2M}\) are the positive constants.

**Assumption 4** The activation function vectors \(\varpi_1\) and \(\varpi_2\) satisfy that \(\varpi_{1m} \leq \|\varpi_1\| \leq \varpi_{1M}\) and \(\varpi_{2m} \leq \|\varpi_2\| \leq \varpi_{2M}\) respectively. \(\varpi_{1m}, \varpi_{1M}, \varpi_{2m}\) and \(\varpi_{2M}\) are the positive constants.

**Assumption 5** The gradient of activation function vector of critic neural network \(\nabla \varpi_1\) satisfies that \(\varpi_{1m} \leq \|\nabla \varpi_1\| \leq \varpi_{1M}\). \(\varpi_{1m}\) and \(\varpi_{1M}\) are the positive constants.

**Theorem 6** For the error tracking system (13), Assumptions 2-5 hold. The optimal cost function and the optimal control law are provided by (16) and (18). The weight updating laws of the critic neural network and the action neural network are given by (25) and (31). Then the tracking error \(Z\) and the weight estimate error \(\hat{W}_1\) and \(\hat{W}_2\) are asymptotically stable.

**Proof** Let us choose a Lyapunov function candidate composed of three positive terms

\[
L(t) = L_1(t) + L_2(t) + L_3(t)
\]

(35)

where

\[
L_1(t) = \frac{1}{2\alpha_1} tr\{\hat{W}_1^T\hat{W}_1\}; \quad L_2(t) = \frac{1}{2\alpha_2} tr\{\hat{W}_2^T\hat{W}_2\}; \quad L_3(t) = Z^T Z + V_2(Z, \Gamma).
\]
The derivative of the Lyapunov function candidate (35) along the trajectories of the error tracking system (34) is given as follows:

$$\dot{L}(t) = \dot{L}_1(t) + \dot{L}_2(t) + \dot{L}_3(t)$$ (36)

According to equation (26), $\dot{L}_1$ can be represented as follows:

$$\dot{L}_1(t) = \frac{1}{\alpha^1} \text{tr}\{\dot{W}_1^T \dot{W}_1\}$$

$$= \frac{1}{\alpha^1} \text{tr}\{\dot{W}_1^T (\text{\text{-}1}_1 g_1(\text{\text{\text{-}1}}^T \text{\text{-}1}_w + \varepsilon_{w1}))\}$$

$$= \frac{1}{\alpha^1} \text{tr}\{-\text{\text{-}1}_1 g_1^T \text{\text{-}1} W^T_1 + \dot{W}_1^T \varepsilon_{w1}\}$$

$$\leq -\frac{3}{\alpha^1} \|g_1\|^2 \|\dot{W}_1\|^2 + \|\varepsilon_{w1}\|^2$$

$$\leq -\frac{\alpha^1}{\|w\|^2} \|\dot{W}_1\|^2 + \|\varepsilon_{w1}\|^2$$

Hence, $\dot{L}_1 < 0$ if $\|\dot{W}_1\| > \frac{2\|\varepsilon_{w1}\|}{\sqrt{\alpha^1}}$.

According to equation (32), $\dot{L}_2$ can be represented as follows:

$$\dot{L}_2(t) = \frac{1}{\alpha^2} \text{tr}\{\dot{W}_2^T \dot{W}_2\}$$

$$= \frac{1}{\alpha^2} \text{tr}\{\dot{W}_2^T (\text{\text{-}2}_2 \dot{w}_2(Z) \dot{w}_2(Z)^T \dot{W}_2 + \frac{1}{\alpha^2} R^{-1}[M(Z)]^{-1} \nabla \dot{w}_1^T (Z)_\dot{W}_1$$

$$+ \varepsilon_{w2})\}$$

$$= -\text{tr}\{\dot{W}_2^T \dot{w}_2(Z) \dot{w}_2(Z)^T \dot{W}_2 + \frac{1}{\alpha^2} \dot{W}_2^T \dot{w}_2(Z) \dot{w}_2(Z)^T \dot{W}_2$$

$$+ \varepsilon_{w2} 2\}$$

$$\leq -\|\dot{w}_2\|^2 \|\dot{W}_2\|^2 + 2\|\dot{W}_2\|^2 + \|R\|^{-2} \|M(Z)\|^{-2} \|\nabla \dot{w}_1(Z)\| \|\dot{W}_1\|^2$$

$$+ \frac{\alpha^2}{\|w\|^2} \|\varepsilon_{w2}\|^2$$

$$\leq -\left(\frac{\alpha^2}{\|w\|^2} \right) \|\dot{W}_2\|^2 + \frac{\alpha^2}{\|w\|^2} \|\dot{W}_1\|^2 + \frac{\alpha^2}{\|w\|^2} \|\varepsilon_{w2}\|^2$$

Hence, $\dot{L}_2 < 0$ if $\|\dot{W}_2\| > \sqrt{\frac{\alpha^2}{\|w\|^2}}$ and $\dot{W}_2 > \sqrt{\alpha^2}$.

$\dot{L}_3$ is given as follows:

$$\dot{L}_3(t) = Z^T \ddot{Z} - Z^T QZ - \Gamma^* R \Gamma^*$$

$$= Z^T ([M(Z)]^{-1} \text{\text{-}1}[C(Z)Z - [D(Z)] + \dot{W}_2^T \dot{w}_2(Z) + \Gamma^* - \delta_2(Z)$$

$$- Z^T QZ - \Gamma^* R \Gamma^*$$

$$= -Z^T [M(Z)]^{-1} \text{\text{-}1}[C(Z)Z - [D(Z)] + \dot{W}_2^T \dot{w}_2(Z) + \Gamma^* - \delta_2(Z)$$

$$- Z^T QZ - \Gamma^* R \Gamma^*$$

$$\leq -\|\|M(Z)\|^{-1}\|C(Z)\|\|Z\|^2 + 4\|Z\|^2 + \frac{1}{\alpha^2} \|M(Z)\|^2 \|D(Z)\|^2$$

$$+\frac{1}{\alpha^2} \|M(Z)\|^2 \|D(Z)\|^2$$

$$+ \frac{1}{\alpha^2} \|M(Z)\|^2 \|D(Z)\|^2$$

$$\leq -\left(\frac{\alpha^2}{\|w\|^2} \right) \|\dot{W}_2\|^2 + \frac{\alpha^2}{\|w\|^2} \|\dot{W}_1\|^2 + \frac{\alpha^2}{\|w\|^2} \|\varepsilon_{w2}\|^2$$

Hence, $\dot{L}_3 < 0$ if $\|\dot{W}_2\| > \sqrt{\frac{\alpha^2}{\|w\|^2}}$ and $\dot{W}_2 > \sqrt{\alpha^2}$.

Then, we can get

$$\dot{L}(t) = \dot{L}_1 + \dot{L}_2 + \dot{L}_3$$

$$< 0$$

Therefore, it can be concluded that the tracking error $Z$ and the neural network estimation error $\dot{W}_1$ and $\dot{W}_2$ are UUB. This completes the proof.
4 Simulation

In order to verify the effectiveness of the proposed control technique, two simulation examples without unknown dynamics and with unknown dynamics are performed compared with the traditional ADP scheme without backstepping method respectively. Because underactuated AUV does not have independent actuators in the sway and heave axes, the available controls are the surge force, pitch moment and the yaw moment. The simulations are conducted using 5-DOF kinematic and dynamic models that $\phi = 0$ and $p = 0$.

According to the kinematic model of underactuated AUV (1), the coordinate transformation matrix is given as follows:

$$J' = \begin{bmatrix}
c\theta c\psi & -s\psi & s\theta c\psi & 0 & 0 \\
c\theta s\psi & c\psi & s\theta s\psi & 0 & 0 \\
-s\theta & 0 & c\theta & 0 & 0 \\
0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & \sec \theta
\end{bmatrix}$$

(41)

where $c\theta = \cos \theta$; $s\theta = \sin \theta$; $s\psi = \sin \psi$; $c\psi = \cos \psi$.

According to the dynamic model of underactuated AUV (2), the matrices $M'$, $C'(\xi)$, $D'(\xi)$ and the vector $g'(\eta)$ are described as follows:

$$M' = \begin{bmatrix}
215 & 0 & 0 & 0 & 0 \\
0 & 265 & 0 & 0 & 0 \\
0 & 0 & 265 & 0 & 0 \\
0 & 0 & 0 & 80 & 0 \\
0 & 0 & 0 & 0 & 80
\end{bmatrix}$$

(42)

$$C'(\xi) = \begin{bmatrix}
0 & 0 & 0 & 265w & -265v \\
0 & 0 & 0 & 215u & 0 \\
0 & 0 & 0 & -215u & 0 \\
-50w & 0 & 0 & 0 & 0 \\
50v & 0 & 0 & 0 & 0
\end{bmatrix}$$

(43)

$$D'(\xi) = \begin{bmatrix}
D'_1 & 0 & 0 & 0 & 0 \\
0 & D'_2 & 0 & 0 & 0 \\
0 & 0 & D'_3 & 0 & 0 \\
0 & 0 & 0 & D'_4 & 0 \\
0 & 0 & 0 & 0 & D'_5
\end{bmatrix}$$

(44)

$$g'(\eta) = [0 \quad 0 \quad -(1822.25 - B) \cos \theta \quad (18.2225 - 0.01B) \sin \theta \quad 0]^T$$

(45)

where $B$ is the buoyancy force; $D'_1 = 70 + 100u$; $D'_2 = 100 + 200v$; $D'_3 = 100 + 200w$; $D'_4 = 50 + 100q$; $D'_5 = 50 + 100r$.

4.1 Example one without unknown dynamics

The numerical values of the parameters used in the simulations are given as follows:
\[ B = 1822.25, \quad \tau_d' = [500 \quad 0 \quad 0 \quad 100 \quad 10]^T, \quad \gamma = 0.9, \quad \alpha_1 = 0.01, \quad \alpha_2 = 0.01, \quad k_1 = 0.01. \]

Figure 3 and Figure 4 show the tracking errors of desired velocity. Tracking errors of desired position are shown in Figure 5 and Figure 6. From above simulation results, the states of error tracking system (34) converge to zero with the method proposed in this paper and the tracking error of desired velocity with ADP scheme without backstepping method is bounded. The time response of underactuated AUV clearly shows that the proposed control method guarantees a higher-level performance.

The tracking of desired velocity of underactuated AUV with two different control schemes is given in Figure 7. Figure 8 shows the tracking of desired position of underactuated AUV with two different control schemes. Figure 9 illustrates the spatial trajectories of underactuated AUV with different control schemes. The actual position and the actual velocity and the actual spatial
Fig. 5  Tracking error of desired position with method proposed in this paper.

Fig. 6  Tracking error of desired position with ADP without backstepping method.

Fig. 7  Tracking of desired velocity.

trajectory with the method proposed in this paper is more closed to the desired spatial trajectory.
4.2 Example two with unknown dynamics

In this section the unknown dynamics is considered.

Figure 10 and Figure 11 show the tracking errors of desired velocity. Tracking errors of desired position are shown in Figure 12 and Figure 13. The states of error tracking system (34) converge to zero with the method proposed in this paper. Because the simulation example exists bounded time-varying disturbances, there exists small bounded chatter in the process of convergence. The time response of the underactuated AUV clearly shows that the proposed control method guarantees a higher-level performance.

The tracking of desired velocity of underactuated AUV with two different control schemes is given in Figure 14. Figure 15 shows the tracking of desired position of underactuated AUV with two different control schemes. Figure 16 illustrates the spatial trajectories of underactuated AUV with different control schemes. The actual position and the actual velocity and the actual spatial trajectory with the method proposed in this paper is more closed to the desired spatial trajectory.
Fig. 10 Tracking error of desired velocity with method proposed in this paper.

Fig. 11 Tracking error of desired velocity with ADP without backstepping method.

Fig. 12 Tracking error of desired position with method proposed in this paper.
5 Conclusions

The stability of error tracking system (34) is guaranteed based on the Lyapunov stability criteria. The simulation results have shown excellent convergence of the error tracking systems (34) compared with the ADP scheme.
without backstepping method. The proposed control scheme achieves good tracking performance. In the future work, we will consider the single critic neural network based ADP scheme to solve the trajectory-tracking control problem of underactuated AUV with actuator faults.

References

[1] G. Che, L. Liu, Z. Yu, Nonlinear trajectory-tracking control for autonomous underwater vehicle based on iterative adaptive dynamic programming, J. Intell. Fuzzy Syst. 37 (3) (2019a) 4205–4215.

[2] G. Che, L. Liu, Z. Yu, An improved ant colony optimization algorithm based on particle swarm optimization algorithm for path planning of autonomous underwater vehicle, J. Amb. Intel. Hum. Comp. 11 (2019b) 3349-3354.

[3] C. Shen, Y. Shi, B. Buckham, Trajectory tracking control of an autonomous underwater vehicle using Lyapunov-based model predictive control, IEEE Trans. Ind. Electron. 65 (7) (2018) 5796–5805.

[4] L.-Y. Hao, Y. Yu, H. Li, Fault tolerant control of UMV based on sliding mode output feedback, Appl. Math. Comput. 359 (2019) 433–455.

[5] Z. Peng, J. Wang, Output-feedback path-following control of autonomous underwater vehicles based on an extended state observer and projection neural networks, IEEE Trans. Syst. Man, Cybern. Syst. 48 (4) (2018) 535–544.

[6] D.A. Smallwood, L.L. Whitcomb, Model-based dynamic positioning of underwater robotic vehicles: theory and experiment, IEEE J. Oceanic Eng. 29 (1) (2004) 169–186.
[7] J.-S. Wang, C.S.G. Lee, Self-adaptive recurrent neuro-fuzzy control of an autonomous underwater vehicle, IEEE Trans. Robot. Automat. 19 (2) (2003) 283–295.

[8] C. Shen and Y. Shi, Distributed implementation of nonlinear model predictive control for AUV trajectory tracking, Automatica (2019) doi:https://doi.org/10.1016/j.automatica.2020.108863.

[9] K.D. von Ellenrieder, Dynamic surface control of trajectory tracking marine vehicles with actuator magnitude and rate limits, Automatica 105 (2019) 433–442.

[10] G. Liu, G. Chen, J. Jiao, R. Jiang, Dynamic modeling and control simulation of an autonomous underwater vehicle, J. Costal Res. 73 (2015) 741–746.

[11] B. Miao, T. Li and W. Luo, A DSC and MLP based robust adaptive NN tracking control for underwater vehicle, Neurocomputing 111 (2013) 184–189.

[12] W. Shi, S. Song, C. Wu and C.L.P. Chen, Multi Pseudo Q-learning-based deterministic policy gradient for tracking control of autonomous underwater vehicle, IEEE Trans. Neural Netw. Learn. Syst. 30 (12) (2019) 3524–3546.

[13] S.-L. Dai, S. He, M. Wang and C. Yuan, Adaptive neural control of underactuated surface vessels with prescribed performance guarantees, IEEE Trans. Neural Netw. Learn. Syst. 30 (12) (2019) 3686–3698.

[14] M. Li, T. Li, X. Gao, Q. Shan, C.L.P. Chen and Y. Xiao, Adaptive NN event-triggered control for path following of underactuated vessels with finite-time convergence, Neurocomputing 379 (2020) 203–213, 2020.

[15] H. Wang, K. Liu and S. Li, Command filter based globally stable adaptive neural control for cooperative path following of multiple underactuated autonomous underwater vehicles with partial knowledge of the reference speed, Neurocomputing 275 (2018) 1478–1489.

[16] Y. Guo, H. Qin, B. Xu, Y. Han, Q.-Y. Fan and P. Zhang, Composite learning adaptive sliding mode control for AUV target tracking, Neurocomputing 351 (2019) 180–1486.

[17] Y.-C. Liu, S.-Y. Liu and N. Wang, Fully-tuned fuzzy neural network based robust adaptive tracking control of unmanned underwater vehicle with thruster dynamics, Neurocomputing 196 (2016) 1–13.

[18] G. Che, Z. Yu, Neural-network estimators based fault-tolerant tracking control for AUV via ADP with rudders faults and ocean current disturbance,
Neurocomputing 411 (2020) 442–454.

[19] Z. Li, S. Xiao, S. Sam Ge, H. Su, Constrained multiegged robot system modeling and fuzzy control with uncertain kinematics and dynamics incorporating foot force optimization, IEEE Trans. Syst. Man, Cybern. 46 (1) (2016) 1–15.

[20] Y.-J. Liu, S. Tong, Optimal control-based adaptive NN design for a class of nonlinear discrete-time block triangular systems, IEEE Trans. Cybern. 46 (11) (2016) 2670–2680.

[21] R. Cui, C. Yang, Y. Li and S. Sharma, Adaptive neural network control of AUVs with control input nonlinearities using reinforcement learning, IEEE Trans. Syst. Man, Cybern., Syst. 47 (6) (2017).1019–1029.

[22] R. Song, F.L. Lewis, Q. Wei, H. Zhang, Off-policy actor-critic structure for optimal control of unknown system with disturbances, IEEE Trans. Cybern. 46 (5)(2016) 1041–1050.

[23] H. Zhang, J. Zhang, G. -H. Yang, Y. Luo, Leader-based optimal coordination control for the consensus problem of multiagent differential games via fuzzy adaptive dynamic programming, IEEE Trans. Fuzzy Syst. 23 (1) (2016) 152–163.

[24] R. Song, W. Xiao, H. Zhang, C. Sun, Adaptive dynamic programming for a class of complex-valued nonlinear systems, IEEE Trans. Neural Netw. Learn. Syst. 25 (9) (2014) 1733–1739.

[25] D. Liu, Q. Wei, Policy iteration adaptive dynamic programming algorithm for discrete-time nonlinear systems, IEEE Trans. Neural. Netw. Learn. Syst. 25 (3) (2014) 621–634.

[26] G. Che, Z. Yu, Neural-network estimators based fault-tolerant tracking control for AUV via ADP with rudders faults and ocean current disturbance, Neurocomputing 411 (2020) 442–454.

[27] C. Mu, D. Wang, H. He, Novel iterative neural dynamic programming for data-based approximate optimal control design, Automatica 81 (2017) 240–252.

[28] W. Gao, Y. Jiang, Z.-P. Jiang, T. Chai, Output-feedback adaptive optimal control of interconnected systems based on robust adaptive dynamic programming, Automatica 72 (2016) 37–45.

[29] C. Qin, H. Zhang, Y. Luo, Online optimal tracking control of continuous-time linear systems with unknown dynamics by using adaptive dynamic programming, Int. J. Control 87 (5) (2014) 1000–1009.
[30] Q. Lin, W. Wei, D. Liu, A novel tracking control scheme for a class of discrete-time nonlinear systems using generalised policy iteration adaptive dynamic programming algorithm, Int. J. Syst. Sci. 48 (3) (2017) 525–534.

[31] A. J. Healey, D. Lienard, Multivariable sliding mode control for autonomous diving and steering of unmanned underwater vehicles, IEEE J. Oceanic Eng. 18 (3) (1993) 327–339.