Proceedings of the 35th Chinese Control Conference
July 27-29, 2016, Chengdu, China

Way-point Tracking Control for a Biomimetic Underwater Vehicle Based on Backstepping

WANG Rui, WANG Shuo, WANG Yu, WEI Qingping
State Key Laboratory of Management and Control for Complex Systems, Institute of Automation, Chinese Academy of Sciences,
Beijing 100190, P. R. China
E-mail: rwang5212@ia.ac.cn, shuo.wang@ia.ac.cn, yu.wang@ia.ac.cn, qingping.wei@ia.ac.cn

Abstract: In this paper, a new approach of way-point tracking control is investigated for a biomimetic underwater vehicle (BUV) propelled by undulatory fins. First, the system design and the model of the BUV are presented. Compared with traditional autonomous underwater vehicles (AUVs) propelled by screw propellers, the concept design enhances the maneuverability and stability of AUVs at low speed. Then the way-point tracking problem of the BUV is formulated. Moreover, a way-point tracking control scheme combining a guidance system of low complexity with backstepping technique is proposed to achieve fast and effective convergence to the desired path composed of a series of given way-points for this BUV. In the end, simulations demonstrate the performance of the proposed way-point tracking controller.

Key Words: Biomimetic underwater vehicle, Undulatory fin, Way-point tracking control, Backstepping, Guidance system

1 Introduction

AUVs are playing a crucial role in ocean exploration and exploitation, such as oceanographic observations, submarine rescue and scientific researches [1–4]. Along with increasing demands for high mobility, robustness and strong anti-disturbance capability, many biomimetic underwater vehicles propelled by undulatory fins have been built [5, 6]. Meanwhile, the ability to accurately maneuver a BUV along a given path is of primary importance for most applications. However, the control issue of the BUV with undulatory fins is very challenging due to the nonlinearity, time-variance dynamics and unpredictable external disturbances. To the authors knowledge, most of researchers focus on undulatory fin control, but seldom consider the use of motion control methods for BUVs, and how to control the BUV precisely has become an open challenging.

The backstepping control algorithms are the most commonly used approach for mobile robot tracking control [7] and have been adopted in the underwater vehicle control systems [8]. Jiang proposed two constructive tracking solutions for the underactuated ship based on Lyapunov's direct method and passivity scheme [10]. The system control for backstepping is quite simple, and the system stability is strictly guaranteed by Lyapunov stability theory. Meanwhile, a popular and effective way to achieve convergence to the desired path is to implement a look-ahead line-of-sight (LOS) guidance law mimicking an experienced sailor. Breivik proposed a guidance-based path following approach to follow straight lines and circles for fully actuated vessels [9]. However, huge computation and complex algorithm are the limitations of this guidance method.

Motivated by the above considerations, this paper aims at designing a fast and effective way-point tracking scheme to force the BUV to track a reference path composed of a series of given way-points. The BUV we consider in this study is named RobCutt-II, which is designed with undulatory long fins on both sides of the vehicle and can perform many motions, including forward/backward swimming, diving/floating motion, and turning maneuver with high mobility. Meanwhile, it is a beneficial attempt to apply the control system combining a guidance system of low complexity with backstepping technique to the way-point tracking control of BUVs. The main advantages of this approach is the simplicity of the controller. Also, fast convergence to the given way-points successively can be achieved.

In the remainder of this paper, the system design of the RobCutt-II are briefly described in Section 2. Way-point tracking control scheme which combines guidance system with backstepping technique is elaborated in Section 3. Simulation results are further provided in Section 4. Finally, the conclusion is presented in Section 5.

2 Overview of the RobCutt-II

Cuttlefish are marine animals swimming by undulations of a pair of latertal fins. They can perform flexible motions in narrow spaces with water turbulence. The biomimetic underwater vehicle named RobCutt-II presented in this paper is designed inspired by this unique propulsion mode. Then the mathematical model of the RobCutt-II is also presented in this section.

2.1 Design of the RobCutt-II

The structure of the RobCutt-II can be seen in Fig. 1. The RobCutt-II contains an underwater manipulator system [11]. Two biomimetic underwater propulsors [12] are mounted on both sides of the manipulator system symmetrically. Each propulsor consists of a cylindrical cavity and an undulating long fin, which can be controlled to perform undulating motion or flapping motion. The main parameters of the RobCutt-II are listed in Tables 1. Since the buoyancy is slightly larger than the gravity, the RobCutt-II will float in the water when it is at rest. Moreover, benefiting from the bilateral symmetrical structure and large metacentric height, the RobCutt-II has good static stability. With the coordinated control of the propagating waves on bilateral fins, the propulsors can produce three-dimensional (3-D)

This work was supported in part by the National Key Technology Research and Development Program of China under Grant No.2013BAK03B01, in part by the National Natural Science Foundation of China under Grant 61233014, 51175496, 61421004, 61333016 and in part by the Beijing Natural Science Foundation under Grant 3141002.

thrust vectoring and yaw torque simultaneously, such that the RobCutt-II can perform different swimming modes, including forward/backward swimming, diving/floating motion and turning maneuver with high mobility.

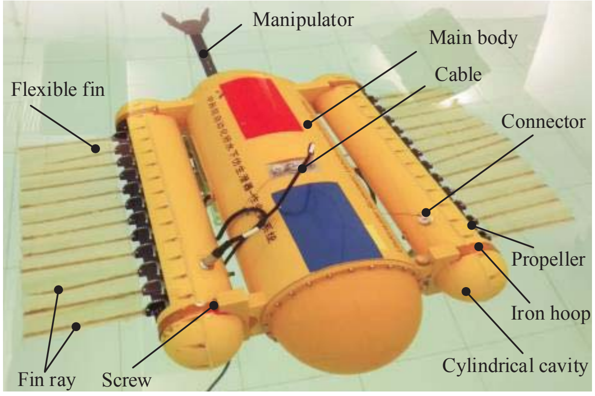


Fig. 1: RobCutt-II prototype

Table 1: Structure Parameters of the RobCutt-II Prototype

Parameters	**value**	**Parameters**	**value**
Mass	51.9 kg	Gravity	508.62 N
Buoyancy	510.58 N	Main body length	760.6 mm
Main body diameter	260 mm	Cavity length	665 mm
Cavity diameter	120 mm	Fin length	460 mm
Fin width	165 mm	Fin thickness	0.82 mm
Number of fin rays	12	Space of fin rays	43 mm
Frequency	$0 \sim 2.5$ Hz	Amplitude	$10^\circ \sim 40^\circ$
Number of waves	$0 \sim 2$	Deflection angle	$-50^\circ \sim 50^\circ$

2.2 Model of the RobCutt-II

In this paper, the way-point tracking issue is addressed in two-dimensional (2-D) plane. While, in 3-D space, we can firstly control the RobCutt-II to swim to the specified depth using the method in [13], and then conduct 2D way-point tracking control. As previously described, the RobCutt-II has good static stability due to large metacentric height, so it's reasonable to neglect the motion in pitch and roll. Furthermore, we assume that the motion in surge and heave are decoupled from the motion in sway and yaw. Therefore, it's sufficient to consider only the 3 degrees-of-freedom (DOF) when designing way-point tracking controller for the RobCutt-II. The 3 DOF kinematic and dynamics can be represented as [14]:

$$\begin{aligned} \dot{\eta} &= J(\psi)\nu \\ M\dot{\nu} &= -C(\nu)\nu - D\nu + \tau + \tau_d \end{aligned} \tag{1}$$

with

$$\begin{aligned} \eta &= [x\ y\ \psi]^{\mathrm{T}}, \\ \nu &= [u\ v\ r]^{\mathrm{T}}, \\ \tau &= [\tau_u\ 0\ \tau_r]^{\mathrm{T}}, \\ \tau_d &= [\tau_{du}\ \tau_{dv}\ \tau_{dr}]^{\mathrm{T}} \end{aligned}$$

where $\eta \in \mathbb{R}^3$ represents the earth-fixed position and course, $J(\psi) \in SO(3)$ is the rotation matrix from the earth-fixed local geographic reference frame to the vehicle-fixed reference frame, $\nu \in \mathbb{R}^3$ represents the vehicle-fixed velocities, M is the vehicle inertia matrix, $C(\nu)$ is the centrifugal and coriolis matrix, D is the hydrodynamic damping matrix, τ is the control input, where τ_u, τ_r are vehicle-fixed propulsion force and moment acting on surge and yaw respectively. $\tau_d \in \mathbb{R}^3$ describes the disturbance forces or moment acting on surge, sway and yaw. In particular, the matrixes M and D are assumed to have the following structure based on the foregoing decoupling assumption:

$$M \triangleq \begin{bmatrix} m_{11} & 0 & 0 \\ 0 & m_{22} & 0 \\ 0 & 0 & m_{33} \end{bmatrix}, D \triangleq \begin{bmatrix} d_{11} & 0 & 0 \\ 0 & d_{22} & 0 \\ 0 & 0 & d_{33} \end{bmatrix} \tag{2}$$

with

$$\begin{aligned} m_{11} &= m - X_{\dot{u}}, \\ m_{22} &= m - Y_{\dot{v}}, \\ m_{33} &= I_z - N_{\dot{r}}, \\ d_{11} &= -X_u, \\ d_{22} &= -Y_v, \\ d_{33} &= -N_r \end{aligned}$$

where m is the mass of the RobCutt-II. The detailed description about the definition of $X_{\dot{u}}, Y_{\dot{v}}, N_{\dot{r}}, X_u, Y_v$ and N_r can be found in [15]. Thus it is omitted.

With the particular structure of the inertia matrix M given in Eq. (2), the centripetal and coriolis matrix $C(\nu)$ is parameterized as in Eq. (3).

$$C(\nu) \triangleq \begin{bmatrix} 0 & 0 & -m_{22}v \\ 0 & 0 & m_{11}u \\ m_{22}v & -m_{11}u & 0 \end{bmatrix} \tag{3}$$

3 Control System Design

This section presents the control algorithm to solve the way-point tracking problem of the RobCutt-II. The problem statement is firstly described in this section. Then the way-point tracking control scheme which combines guidance system with backstepping technique is presented. Finally, the algorithms are introduced in detail in the rest of this section.

3.1 Problem Statement

As shown in Fig. 2, two reference coordinate frames are established, where $O_E X_E Y_E$ is the inertial frame, i.e. the earth-fixed frame, and $O_B X_B Y_B$ is the vehicle-fixed reference frame. Meanwhile, denote the inertial position and the course angle of the RobCutt-II by $p(x, y)$ and ψ respectively. In way-point tracking, the primary objective concerning a vehicle is to make its position converge to and follow a desired geometric path composed of a series of given way-points. In this study, the desired path is composed of a series of given way-points, i.e. $WP = \{p_1, p_2, ..., p_n\}, p_i = (x_i, y_i) \in \mathbb{R}^2, i = 1, 2, ..., n$. For each way-point $p_i, i = 1, 2, ..., n$, there is an adjacent round $B_{\gamma_i}(p_i)$ with center p_i and radius $\gamma_i > 0$ associated with it, i.e. $B_{\gamma_i}(p_i) = \left\{p \in \mathbb{R}^2 : \|p - p_i\| \leq \gamma_i, \gamma_i > 0\right\}$. The control objective is to develop a proper feedback control law to force the RobCutt-II to visit the adjacent round of way-points $p_i, i = 1, 2, ..., n$ sequently and converge to the last way-point p_n in the end.

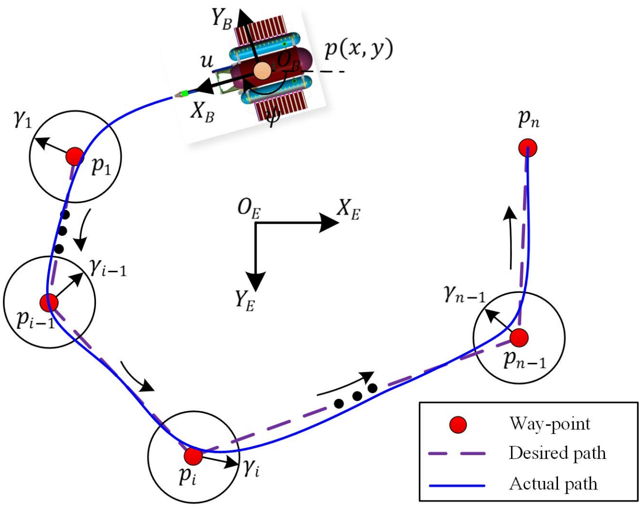


Fig. 2: The schematic diagram of the way-point tracking problem.

3.2 Way-point Tracking Control Scheme

In order to solve aforementioned problem, a way-point tracking control scheme is designed and the block diagram is illustrated in Fig. 3. The way-point tracking controller is mainly composed of two components, i.e. the guidance system and the backstepping controller.

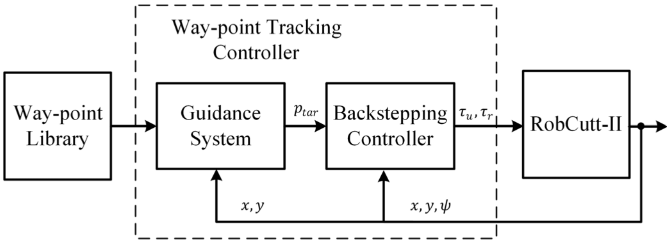


Fig. 3: The control block diagram of way-point tracking control.

The guidance system is used to give the current target point $P_{tar} = (x_d, y_d)$ that RobCutt-II needs to track based on the position feedback and the way-point library. It should be mentioned that some middle points are added between the way-points to reduce cross-track error during way-point tracking.

The system control for backstepping is quite simple, and the system stability can be strictly guaranteed by Lyapunov stability theory. For that reason, we employ backstepping controller to output propulsion force τ_u and moment τ_r based on position and course feedback, which force the RobCutt-II to converge to the given way-points. Next, the two components of the way-point tracking controller are further described in detail.

3.3 Guidance System

A guidance system based on the finite state machine (FSM) is implemented in the way-point tracking control scheme. The FSM is designed to directly give the target point in the next state according to the current state of the RobCutt-II. Algorithm complexity is low, which is conducive to real-time motion control. The way-point tracking process consists of four states, i.e. the initial state, the path switching state, the line segment following state and the final state. Fig. 4 shows the switching condition of each state.

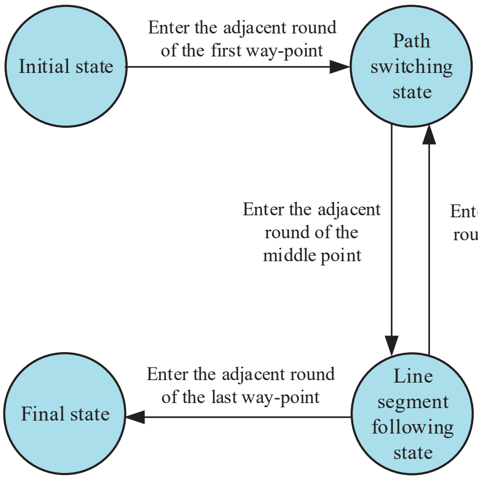


Fig. 4: The finite state machine of the way-point tracking process.

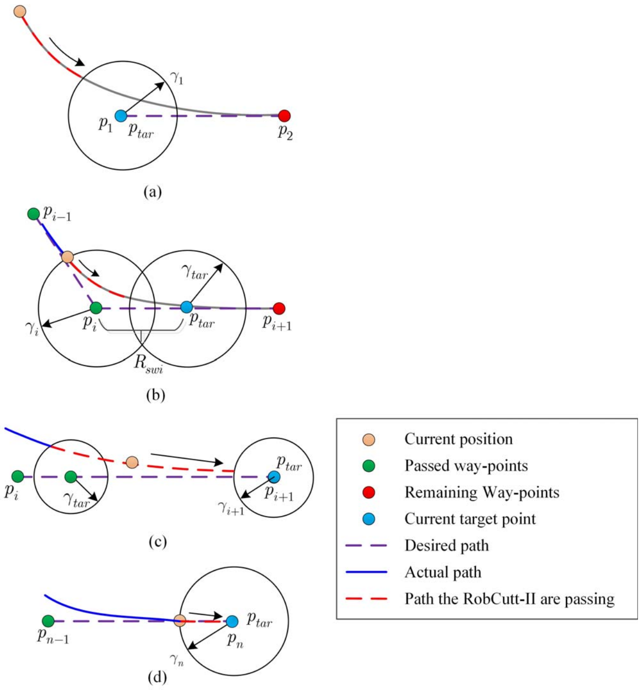


Fig. 5: The definition of each state. (a) Initial state. (b) Path switching state. (c) Line segment following state. (d) Final state.

In the initial state, the first way-point p_1 is chosen as the target point and the RobCutt-II moves toward the adjacent round of the first way-point from its initial position. As shown in Fig. 5(a), when the RobCutt-II enters the adjacent round of the target point, i.e. $E_{pp_1} < \gamma_1, E_{pp_1} = \sqrt{(x-x_1)^2+(y-y_1)^2}$, the initial state is over. Then in the path switching state, the middle point located on the straight line segment p_ip_{i+1} is set as the current target point, which are R_{swi} away from p_i as seen in Fig. 5(b). Specifically, middle points are used to reduce cross-track error during way-point tracking. Fig. 5(c) shows that after entering the adjacent round of the middle point, i.e. $E_{pp_{tar}} < \gamma_{tar}$, the RobCutt-II switches to the line segment following state, in which way-point p_{i+1} is selected as the target point. If way-point p_{i+1} is not the last way-point, the RobCutt-II returns to path switching state. Otherwise, the way-point tracking process enters the final state, in which the control objective is to force the RobCutt-II to reach the last way-point as close as possible (See Fig. 5(d)).

3.4 Backstepping Controller

To force the RobCutt-II to track the target point, a backstepping controller is designed in this section. Assume that the current tracking target point is $p_{tar} = (x_d, y_d)$, then the tracking error can be written as

$$\begin{aligned} x_e &= x - x_d \\ y_e &= y - y_d \\ e_{xy} &= \sqrt{{x_e}^2 + {y_e}^2} \\ \psi_e &= \psi - \psi_d \end{aligned} \tag{4}$$

Combined with Eq. (1), the differential equation of the RobCutt-II can be concluded as

$$\begin{aligned} \dot{e_{xy}} &= -u cos(\psi_e) + v sin(\psi_e) \\ \dot{\psi_e} &= r + \frac{sin(\psi_e)}{e_{xy}} u + \frac{cos(\psi_e)}{e_{xy}} v \end{aligned} \tag{5}$$

Feedback control laws for force τ_u and moment τ_r need to be derived to make the equilibrium state $(0, \gamma)$ of (e_{xy}, ψ_e) uniformly asymptotically stable, i.e. $e_{xy} \to 0, \psi_e \to \gamma, \gamma > 0$. Notice that $e_{xy} = 0$ is a singular point of Eq. (5), so γ should be a positive number.

With the formulation developed previously, kinematic control is firstly designed:

$$\begin{aligned} u &= k_1(e_{xy} - \gamma) cos^n(\psi_e) \\ r &= -\frac{cos(\psi_e)}{e_{xy}} v - k_3 \psi_e \end{aligned} \tag{6}$$

where γ can be an arbitrarily small positive number, n is a natural number, and $k_3 > k_1 > 0$. Such that the error vector of the RobCutt-II satisfies $(e_{xy}, \psi_e) \to (0, \gamma)$.

Then the kinematic control is extended to the dynamic case, where the suitable forces/moments to drive the RobCutt-II to follow a desired geometric path are derived from the velocity control input.

From Eq. (1), the differential equation of the reference forward and angular velocities can be described as

$$\begin{aligned} \dot{u} &= \frac{1}{m_{11}}(\tau_u + m_{22} v r - d_{11} u) \\ \dot{r} &= \frac{1}{m_{44}}(\tau_r + (m_{11} - m_{22}) u v - d_{44} r) \end{aligned} \tag{7}$$

Define auxiliary velocity errors

$$\beta_1 = u - \alpha_1, \beta_3 = r - \alpha_3 \tag{8}$$

Then the force τ_u and moment τ_r are designed based on Eq. (7) as

$$\begin{aligned} \tau_u &= -m_{22} v r + d_{11} u + m_{11} \dot{\alpha_1} - m_{11} \frac{sin(\psi_e)}{e_{xy}} \psi_e - k_4 \beta_1 \\ \tau_r &= (m_{22} - m_{11}) u v + d_{44} r + m_{44} \dot{\alpha_3} - m_{44} \psi_e - k_6 \beta_3 \end{aligned} \tag{9}$$

where $k_4 > 0, k_6 > 0$, such that $\beta_1, \beta_3 \to 0$ as $t \to \infty$. Namely the velocity of the RobCutt-II would converge to the desired velocity. By considering the Lyapunov function candidates $V_1 = \frac{1}{2}{\psi_e}^2 + \frac{1}{2}{\beta_1}^2 + \frac{1}{2}{\beta_3}^2$ and $V_2 = \frac{1}{2}(e_{xy} - \gamma)^2$, then we can prove that $\dot{V_1} \le 0$ and $\dot{V_2} \le 0$. Therefore the systems stability is guaranteed by Lyapunov stability theory.

4 Simulation Results

In this section, the proposed control scheme will be applied to the RobCutt-II model shown in Section 2 for way-point tracking problem. In this study, a reference path consisting of four way-points is chosen

$$p_1 = (4, 1), p_2 = (4, 3), p_3 = (1, 3), p_4 = (1, 1)$$

The model parameters of the RobCutt-II used in the simulations are listed in Tables 2. The initial position of the RobCutt-II is $p(x, y) = (0.5, 2)$, and the course is $\frac{\pi}{2}$. Tables 3 tabulates the parameters of the way-point tracking controller, where $L_{p_i p_{i+1}}$ represents the length of the straight line segment $p_i p_{i+1}$.

Table 2: Model parameters of the RobCutt-II

m_{11}	m_{22}	m_{33}	d_{11}	d_{22}	d_{33}
57.5	61.3	1.15	26	29	1.5

Table 3: Controller parameters

γ_1	γ_2	γ_3	γ_4	γ_{tar}
0.1	0.1	0.1	0.05	0.1
k_1	k_2	k_3	k_4	R_{swi}
0.1	0.5	50	20	$0.1 L_{p_i p_{i+1}}$

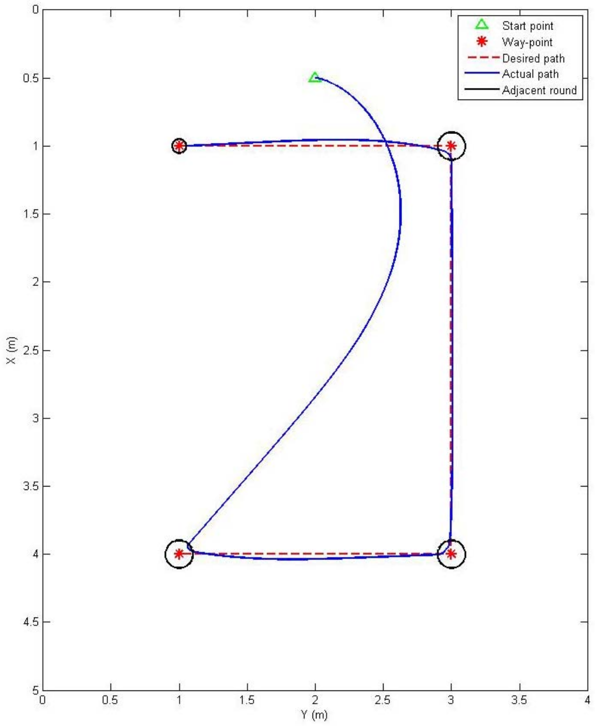


Fig. 6: Simulation results: tracking path.

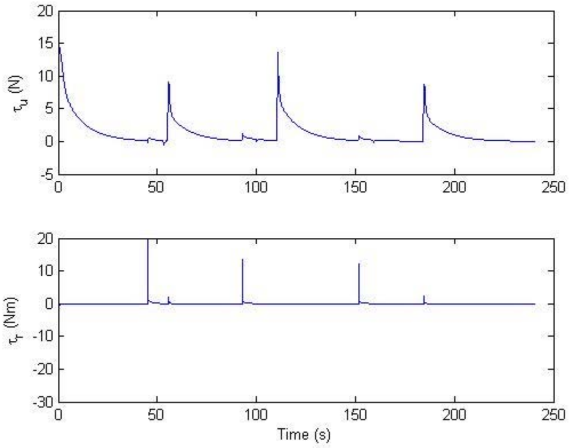


Fig. 7: Simulation results: time evolution of control signals.

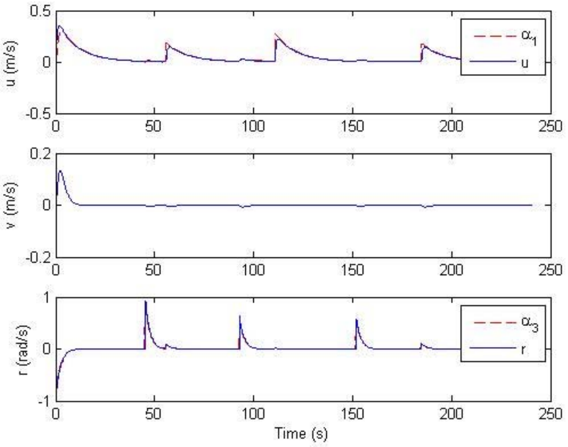


Fig. 8: Simulation results: time evolution of vehicle-fixed velocities.

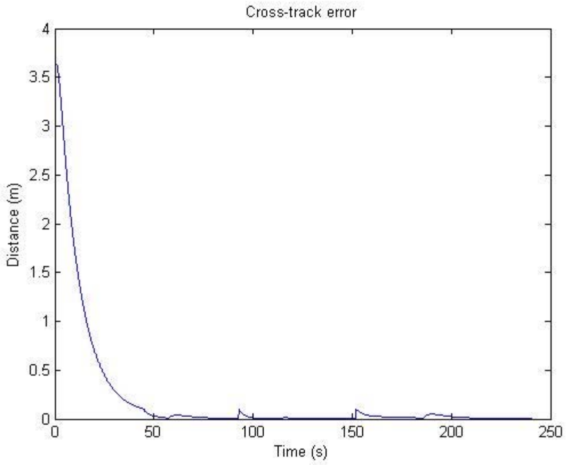
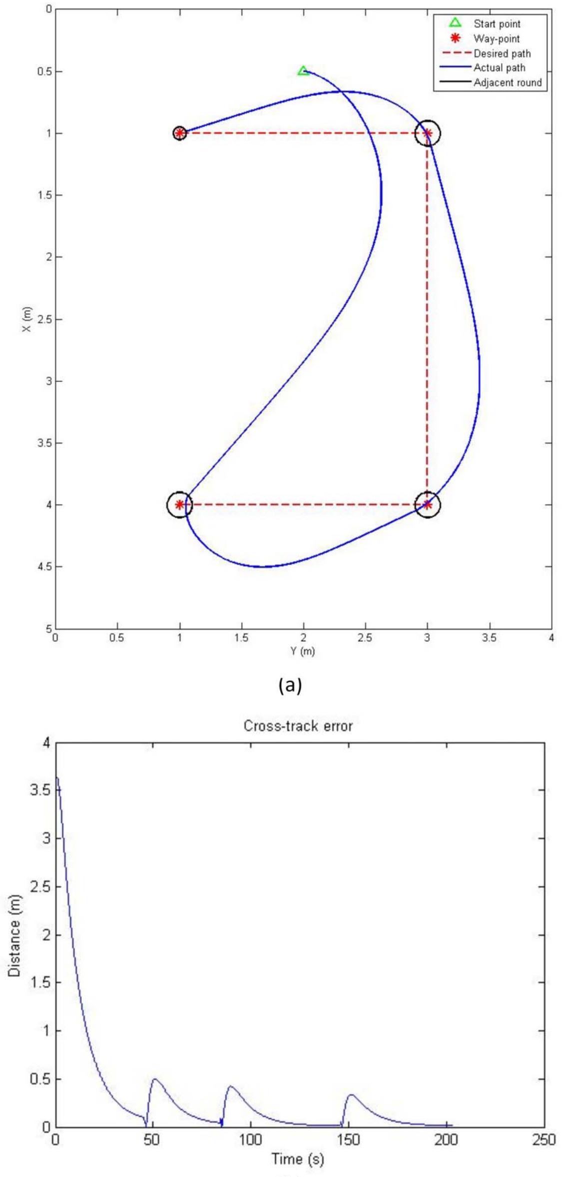


Fig. 9: Simulation results: cross-track error.

The simulation results are depicted in Fig. 6-9. Fig. 6 shows how the RobCutt-II tends to the reference path. It is observed that the proposed way-point tracking controller is able to force the RobCutt-I to track the given way-points sequently and to converge to the last way-point with almost no overshoot. The time evolution of the vehicle-fixed velocities are given in Fig. 7. The actual velocities well follow the desired velocities given by the kinematic design. Furthermore, the graph reflecting time history of control signals $\tau_u(t), \tau_r(t)$ is depicted in Fig. 8. Fig. 9 illustrates that the cross-track error converges exponentially to zero.

In addition, a comparison study is performed to assess the effect of middle points in the way-point tracking control. Fig. 10 shows the simulation result of the way-point tracking control without middle points, which is performed with the same control parameters. The simulation results demonstrate that middle points added between the way-points can markedly reduce cross-track error during way-point tracking.



(a)

(b)

Fig. 10: Results of the simulation without middle points. (a) Tracking path. (b) Cross-track error.

5 Conclusion

In this paper, a control scheme combining guidance system with backstepping technique has been presented for the

RobCutt-II way-point tracking control. The low-complexity guidance system is used to give the target point that the RobCutt-II currently needs to track. Then, the backstepping controller further outputs control force and moment based on the target point and state feedback. Simulation result shows a fast convergence to the given way-points by the action of the proposed way-point tracking controller. In addition, a comparison study demonstrates the performance of the guidance system.

References

[1] D. Blidberg, The development of autonomous underwater vehicles (AUV); a brief summary, in *Proceedings of 2001 International Conference on Robotics and Automation*, 2001.

[2] B. Wang, C. Wu, T. Ge, Control system of a novel underwater vehicle for global ocean science to the deepest ocean, in *Proceedings of 32nd Chinese Control Conference*, 2013: 7268-7272.

[3] Y. Wang, S. Wang, and M. Tan, Path generation of autonomous approach to a moving ship for unmanned vehicles, *IEEE Transactions on Industrial Electronics*, 62(9): 5619-5629, 2015.

[4] Y. Wang, S. Wang, M. Tan, C. Zhou, and Q. Wei, Real-Time Dynamic Dubins-Helix Method for 3-D Trajectory Smoothing, *IEEE Transactions on Control Systems Technology*, 23(2): 730-736, 2015.

[5] I. Neveln, Y. Bai, J. Snyder, J. Solberg, O. Curet, K. Lynch, and M. Maclver, Biomimetic and bio-inspired robotics in electric fish research, *Journal of experimental Biology*, 216(13): 2501-2514, 2013.

[6] R. Wang, S. Wang, and Q. Wei, Research development and analysis of biomimetic underwater vehicle propelled by undulatory fins, *Automation Panorama*, (7): 70-74, 2015.

[7] C. Tsai, and K. Song, Visual tracking control of a wheeled mobile robot with system model and velocity quantization robustness, *IEEE Transactions on Control Systems Technology*, 17(3): 520-527, 2009.

[8] K. Do, Z. Jiang, and J. Pan, Robust adaptive path following of underactuated ships, *Automatica*, 40(6): 929-944, 2004.

[9] M. Breivik, and T. Fossen, Path following of straight lines and circles for marine surface vessels, in *Proceedings of 2004 IFAC Conference on Control Applications in Marine Systems*, 2004.

[10] Z. Jiang, Global tracking control of underactuated ships by Lyapunov's direct method, *Automatica*, 38(2): 301-309, 2002.

[11] Y. Wang, S. Wang, Q. Wei, M. Tan, C. Zhou, and J. Yu, Development of an underwater manipulator and its free-floating autonomous operation, *IEEE/ASME Transactions on Mechatronics*, (99), 2016.

[12] R. Wang, and S. Wang, Design and implementation of a biomimetic underwater propeller with a undulating long fin, *Journal of Huazhong University of Science and Technology (Nature Science Edition)*, 43(s1): 408–411, 2015.

[13] Q. Wei, S. Wang, Y. Wang, C. Zhou, and M. Tan, Course and depth control for a biomimetic underwater vehicle - RobCutt-I, *International Journal of Offshore and Polar Engineering*, 25(2): 81–87, 2015.

[14] T. Fossen, Marine control systems: guidance, navigation and control of ships, rigs and underwater vehicles, *Marine Cybernetics*, 2002.

[15] SNAME, Nomenclature for treating the motion of a submerged body through a fluid JR, *Technical and research bulletin*, New York: Society of Naval Architects and Marine Engineers, 1952: 1-5.