

# Three-Dimensional Helical Path Following of an Underwater Biomimetic Vehicle-Manipulator System

Rui Wang, Shuo Wang , Yu Wang, Chong Tang, and Min Tan

**Abstract**—This paper aims to investigate the method of 3-D helical path following of an underwater biomimetic vehicle-manipulator system (UBVMS) propelled by undulatory fins (RobCutt-II). By mimicking the unique undulatory propulsion mode of cuttlefish, we hope that RobCutt-II could obtain good low-speed stability, which helps to perform stable and effective underwater operations. For some underwater applications, the UBVMS should autonomously generate and pass through a helical path around specific object for survey and intervention. Therefore, a path-following control scheme is proposed. In particular, the path generator provides a 3-D helical path according to the target position and the specific constraint conditions. To follow the generated path in underwater space, the control architecture, which consists of a guidance system, three active disturbance rejection control (ADRC) subcontrollers, and fuzzy method for control allocation, is proposed. Finally, two experiments conducted on the prototype RobCutt-II illustrate the effectiveness of the proposed control scheme.

**Index Terms**—Active disturbance rejection control (ADRC), guidance system, 3-D helical path following, underwater biomimetic vehicle-manipulator system (UBVMS), undulatory propulsion.

## I. INTRODUCTION

RAPID progress in marine robotics is steadily affording scientists advanced tools for ocean survey and intervention [1], [2]. During this process, many underwater robots were developed [3]. Yamamoto detailed the fishlike swimming robots used for various purposes in scientific research [4]. In addition, some biomimetic underwater vehicles (BUVs) mimicking knifefish, cuttlefish, or stingray, which use flexural undulatory waveforms to swim swiftly, were designed and manufactured

[5]–[8]. This undulatory propulsion mode may provide better swimming stability, higher maneuverability, lower energy consumption, and better environment adaptation, in comparison to traditional jets or axial propellers. On the basis of the studies of wavelike propulsion mechanism and hydrodynamics, it is expected that underwater biomimetic vehicle-manipulator systems (UBVMSs) can complete more complicated tasks autonomously in extreme environments in place of humans. Among these applications, it is often desirable for the UBVMS to surround an object (rare rock, underwater wreck, etc.), and finally to reach the designated location for sample collection, welding, repair, or other underwater operations. The purpose of this paper is to develop a 3-D helical path-following scheme for the ribbon-fins-propelled UBVMS used to perform underwater observation tasks, such that it can generate a reasonable motion path and control the UBVMS to move along the path.

Path following is one of the typical control scenarios in the control literature and it is extremely important for practical applications of an autonomous robot [9], [10]. As for the path-following control of autonomous underwater vehicles (AUVs) or surface ships, worldwide researchers have proposed various control strategies. In a 2-D plane, path following has been well studied [11]–[15]. Breivik *et al.* designed a nonlinear model-based controller for marine surface vessels to follow straight lines and circles [12]. Lekkas *et al.* addressed the problem of planar curved path following of an underwater vehicle using a cascaded controller composed of a modified line-of-sight (LOS) guidance system with a heading controller [14].

In 3-D underwater space, path following is much more challenging, because not only the horizontal position, but also the depth of underwater vehicles has to be considered [16]. Encarnacao *et al.* proposed a path following control law that drives an AUV to desired 3-D straightline and helix [17]. However, the above method has not been applied in the actual systems. Fischer *et al.* developed a nonlinear control design for a fully actuated AUV using a continuous robust integral of the sign of the error control structure. The experimental result showed that the vehicle could track the helical trajectory in a controlled underwater environment [18].

However, there are few studies focused on path following of BUVs propelled by undulatory fins, whose propulsion modes and control inputs are quite different from that of AUVs or surface vessels. Jenkin *et al.* developed a hexapod robot named AQUA that utilized flippers or fins for propulsion and studied its visual and communication system [19]. Unfortunately, closed-loop autonomous motion controls have not been imple-

Manuscript received February 22, 2017; revised July 6, 2017; accepted October 9, 2017. Date of publication November 27, 2017; date of current version April 12, 2018. This work was supported in part by the National Natural Science Foundation of China under Grant 61773378, Grant 61703401, Grant 61233014, and Grant 61333016; and in part by the Foundation for Innovative Research Groups of the National Natural Science Foundation of China under Grant 61421004. (Corresponding author: Shuo Wang.)

Guest Editor: K. Asakawa.

R. Wang, S. Wang, and C. Tang are with the State Key Laboratory of Management and Control for Complex Systems, Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China and also with the University of Chinese Academy of Sciences, Beijing 100049, China (e-mail: rwang5212@ia.ac.cn; shuo.wang@ia.ac.cn; tangchong2014@ia.ac.cn).

Y. Wang and M. Tan are with the State Key Laboratory of Management and Control for Complex Systems, Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China (e-mail: yu.wang@ia.ac.cn; min.tan@ia.ac.cn).

This paper has supplementary downloadable material available at <http://ieeexplore.ieee.org>.

Digital Object Identifier 10.1109/JOE.2017.2762498

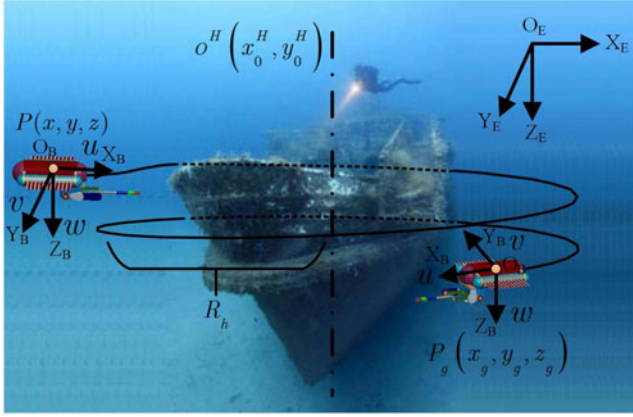


Fig. 1. Diagram of the underwater observation task of RobCutt-II.  $O_E X_E Y_E Z_E$  is the earth-fixed frame,  $O_B X_B Y_B Z_B$  is the body-fixed frame, the current position of RobCutt-II is  $P(x, y, z)$ , the target position is  $P_g(x_g, y_g, z_g)$ ,  $u, v, w$  are the velocities in surge, sway, and heave, the axis of the helix is  $O^H(x_0^H, y_0^H)$  and the radius is  $R_h$ .

mented yet. Bi *et al.* developed a way-point tracking control system for a BUV propelled by wide paired pectoral foils in the horizontal plane [20]. In our previous work [21], a close-loop control law for a BUV propelled by undulatory fins to achieve path following in the underwater space was proposed and the 2-D circular path following was realized in a BUV prototype. Remarkably, as far as we know, the 3-D helical path following of BUVs propelled by undulatory fins has not been considered.

The main contributions of this paper are summarized as follows:

- 1) development of a helical path-following control scheme integrating a helical path generator, a path-following controller, and the fuzzy method for control allocation;
- 2) practical validation of the proposed control strategy on a UBVMs prototype.

In the remainder of this paper, Section II presents the problem description and the control architecture of the 3-D helical path-following control scheme. Section III depicts the modeling of RobCutt-II, including the kinetic model and the allocation of control parameters based on fuzzy inference, which builds the nonlinear relationship between the propulsive force/torque and the control inputs of RobCutt-II. Section IV details the algorithm of path generation. The path-following controller is detailed in Section V. Experimental results, which verify the convergence and effectiveness of the proposed control algorithm, are given in Section VI. Finally, conclusion and future work are summarized in Section VII.

## II. PROBLEM DESCRIPTION AND CONTROL ARCHITECTURE

As shown in Fig. 1, consider a scenario in which RobCutt-II needs to surround an underwater object (here is a sunken ship) for observation task. Wide-angle cameras for inspecting underwater objects and their surrounding environments are mounted on RobCutt-II. Assume that the vehicle moves in a helical trajectory, whose projection in the horizontal plane is a circle with center  $O^H(x_0^H, y_0^H)$  and radius  $R_h$ . The coordinate of the target

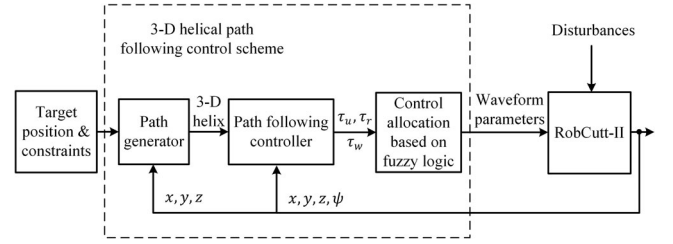


Fig. 2. Block diagram of the 3-D helical path-following control scheme.

point is  $P_g(x_g, y_g, z_g)$ . The integer  $N$  denotes the minimum number of turns of the helical trajectory. The goal of path generation is to give a 3-D helical path that will eventually reach the target point while satisfying the above conditions. The control objective of path following is to ensure that RobCutt-II reaches and follows the generated helical path and finally converges to the target point. Note that when the distance between the vehicle and the target point is less than  $\gamma_g$ , we think that RobCutt-II has reached the target point. The value of  $\gamma_g$  can be determined according to the required accuracy of different tasks.

RobCutt-II can be considered an underactuated system, because it cannot generate lateral motion and pitching motion voluntarily. As explained in [22], underactuated systems have fewer inputs than degrees of freedom. While many interesting techniques and results have been presented for underactuated systems, the tracking problems for underactuated vehicles remain challenging because most of these systems are not fully feedback linearizable and exhibit nonholonomic constraints [23]. Moreover, the UBVMs propelled by undulatory fins is a multi-variable, nonlinear, and strong-coupling system. Thus, the key issue in the UBVMs control is how to make it feasible to achieve a fast convergence to the desired path without a precise mathematical model. As explained earlier, RobCutt-II cannot perform sway motion voluntarily. All the lateral motion generated by waves, wind, etc., on RobCutt-II can be considered as a perturbation, and should be corrected, which is the primary motivation of the closed-loop control.

Motivated by the above considerations, we design the 3-D helical path-following control scheme of RobCutt-II as presented in Fig. 2. The path generator provides a 3-D helical trajectory according to the target position and the specific constraint conditions. The main advantage of this approach is the simplicity of the algorithm and there is no need of complex calculations to attain the helical path. In addition, we design a path following controller by decoupling the multivariable system to steer the surge speed, the heading, and the depth, respectively, so that the number of degrees of freedom to be controlled is the same as the number of available control inputs (i.e., propulsive force  $\tau_u$ , moment  $\tau_r$ , and heave force  $\tau_w$ ). As demonstrated in [8], the ADRC controller yields better transient performance in the control of underwater vehicle propelled by undulatory fins than the commonly used proportional-integral-derivative (PID) controller. Thus, the ADRC technique is used in the development of the surge speed controller, the heading controller, and the depth controller, which are designed independently to force RobCutt-II to track the speed assignment, the target heading, and the

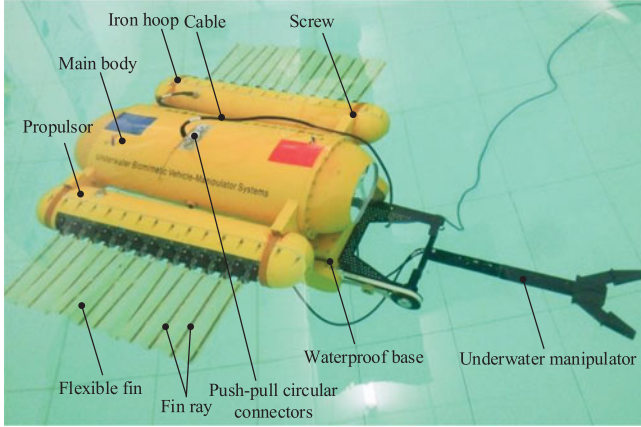


Fig. 3. RobCutt-II prototype.

target depth in the presence of disturbances. The basic concept of this method is total disturbance estimation and compensation in real time, which will be detailed in Section V.

### III. MODELING OF THE ROB CUTT-II

The UBVMs named RobCutt-II (ROBotic CUTTlefish) is shown in Fig. 3. RobCutt-II mimics the unique undulatory propulsion mode of cuttlefish, which helps to obtain good low-speed stability for stable and effective underwater survey and intervention tasks. Readers may refer to [24] and [25] for more details about the mechanical design of RobCutt-II. This section describes the modeling of RobCutt-II swimming in the 3-D underwater space. The kinetic model of underwater vehicles is first introduced. Then, the allocation of control parameters is presented.

#### A. Kinetic Model of Underwater Vehicles

The position and orientation of underwater vehicles are described in the earth-fixed frame  $O_E X_E Y_E Z_E$ , while the linear and angular velocities and the control force/torque are described in the body-fixed frame  $O_B X_B Y_B Z_B$ . As explained in [24], RobCutt-II has good static stability due to large metacentric height, so we neglect the dynamics associated with the motion in roll and pitch when modeling the vehicle. Furthermore, we assume that the motion in heave is decoupled from the motion in surge, in sway, and in yaw. Moreover, we do not include the nonlinear damping since the linear damping is more significant than the nonlinear damping for an underwater vehicle moving at low speed. In this case, the vehicle dynamics can be described by (see [26])

$$\begin{aligned}\dot{x} &= u \cos(\psi) - v \sin(\psi) \\ \dot{y} &= u \sin(\psi) + v \cos(\psi) \\ \dot{z} &= w \\ \dot{\psi} &= r \\ M\dot{\nu} &= -C(\nu)\nu - D\nu + \tau + \tau_d\end{aligned}\quad (1)$$

with

$$M = \begin{bmatrix} m_{11} & 0 & 0 & 0 \\ 0 & m_{22} & 0 & m_{24} \\ 0 & 0 & m_{33} & 0 \\ 0 & m_{24} & 0 & m_{44} \end{bmatrix}, D = \begin{bmatrix} d_{11} & 0 & 0 & 0 \\ 0 & d_{22} & 0 & d_{24} \\ 0 & 0 & d_{33} & 0 \\ 0 & d_{42} & 0 & d_{44} \end{bmatrix}$$

$$C = \begin{bmatrix} 0 & 0 & 0 & -m_{22}v - m_{24}r \\ 0 & 0 & 0 & m_{11}u \\ 0 & 0 & 0 & 0 \\ m_{22}v + m_{24}r & -m_{11}u & 0 & 0 \end{bmatrix}$$

where  $u, v, w$ , and  $r$  are the velocities in surge, sway, heave, and yaw, respectively,  $x, y, z, \psi$  denote the position and orientation of the vehicle,  $M$  is the inertia matrix including hydrodynamic added inertia,  $D$  is the linear damping matrix, and  $C$  is the coriolis and centripetal matrix. The parameters  $m_{11}, m_{22}, m_{24}, m_{33}, m_{44}$  are given by the vehicle inertia and added mass effects. The parameters  $d_{11}, d_{22}, d_{24}, d_{42}, d_{33}, d_{44}$  are given by the linear hydrodynamic damping. The estimation for these parameters can be found in [27].  $\tau_{du}, \tau_{dv}, \tau_{dw}, \tau_{dr}$  describe the disturbance force and moment acting on surge, sway, heave, and yaw. The available controls are the surge force  $\tau_u$ , the heave force  $\tau_w$ , and the yaw moment  $\tau_r$ .

#### B. Allocation of Control Parameters

The controls described in the previous section are propulsive force and torque, while the actual control inputs of undulatory propulsion are the parameters of propagating waves on bilateral fins. Specifically, they include the left fin frequency, the right fin frequency, the amplitude of waves, the phase difference, and the deflection angle, which are denoted by  $F_L, F_R, A, \phi$ , and  $\theta_B$ . Hydrodynamic modeling of the undulatory fin has been considered by several researchers with the goal of demonstrating the quantitative effect of waveform parameters on the fin's thrust production [28]–[31]. Curet *et al.* developed a bio-inspired robotic knifefish with an undulatory propulsor and examined the basic properties of undulatory swimming [28]. Wei *et al.* built the analytical expressions between the generated thrust and the oscillating frequency, amplitude, wavelength, and wave number of the long-fin using the finite element method [30]. It should be noted that they made some idealistic assumptions to simplify the calculation. Epstein *et al.* plotted the relationship between the propulsive wave's frequency, amplitude, and length and the robotic fin's surge thrust through physical experiments [31]. We introduce fuzzy inference to build the nonlinear relationship between the propulsive force/torque and the control inputs of the RobCutt-II. In this way, the parameters of propagating waves can be more precisely determined than the simple linear logic. Fig. 4 describes that the proposed fuzzy method is composed of fuzzification, an inference engine, rule base, and defuzzification.

In particular, standard triangular membership functions are utilized in the fuzzification. Fuzzy rules are determined according to the evaluation of the experimental results of motion control. Four key principles are summarized as follows.



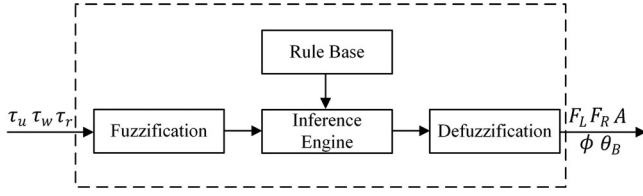


Fig. 4. Block diagram of parameter mapping based on fuzzy inference.

- 1) Heave force is closely related to the deflection angle of ribbon-fin. Heave force becomes large as the deflection angle becomes large within a certain range.
- 2) When required yaw torque is small while surge force is large, the directions of waves along the two fins should be consistent but the wave frequency should become different to output large surge force and small yaw torque simultaneously.
- 3) The priority of the heading control and the depth control are higher than the priority of the surge control. Namely, when yaw torque, heave force, and surge force are all large, large yaw torque and heave force are guaranteed preferentially.
- 4) When surge force and yaw torque are both small, the phase difference between two adjacent fin rays becomes small, while the frequency of bilateral fins, the amplitudes of waves, and the deflection angles are determined by the heave force. Namely, when required heave force is small, the frequency of bilateral fins, the amplitudes of waves, and the deflection angles become small and vice versa.

The commonly used Mamdani's Max–Min method rule is used to achieve fuzzy inference and weight average method is applied for defuzzification. Readers may refer to [32] for the detailed description of the fuzzy method.

#### IV. THREE-DIMENSIONAL HELICAL PATH GENERATION

Path generation can be viewed as an act in which a feasible or even optimal path between two given points is to be found under some predefined constraints. For some observation applications, helical trajectory around the object is commonly used. In this section, we will introduce the method of 3-D helical path generation. The main idea is to set the starting point of the helical path on the perpendicular line of the initial position of the vehicle to the rotation axis, then generate a 3-D helical path according to the number of turns  $N$  and the descent height  $h^H$ . Without loss of generality, we assume that  $h^H > 0$ . As illustrated in Fig. 5(a), the initial and target position are represented as  $P_s(x_s, y_s, z_s)$  and  $P_g(x_g, y_g, z_g)$ , respectively. The feasible path generation algorithm is described in Algorithm 1.

Specifically, we first calculate the starting point  $S^H$  of the helical path. Fig. 5(b) shows that the coordinate  $S^H(x^H, y^H)$  can be obtained by solving (2) and (3)

$$\frac{x_s - x_0^H}{y_s - y_0^H} = \frac{x^H - x_0^H}{y^H - y_0^H} \quad (2)$$

$$(x^H - x_0^H)^2 + (y^H - y_0^H)^2 = R_h^2 \quad (3)$$

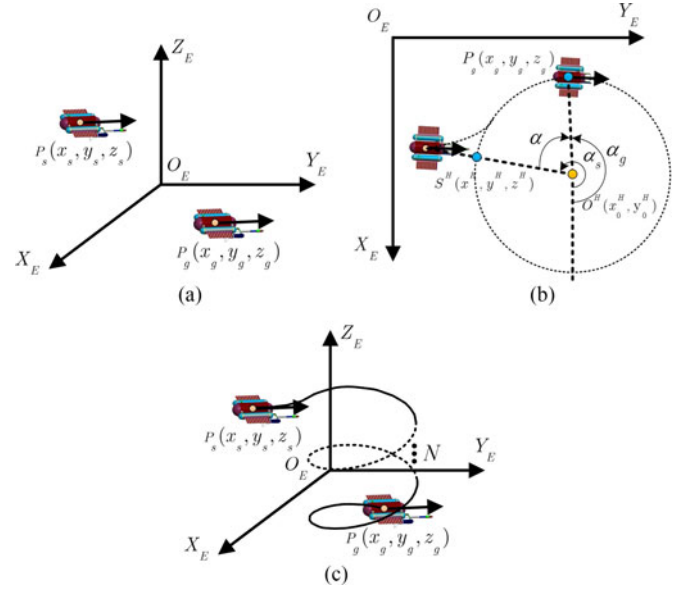


Fig. 5. Three-dimensional helical path generation. (a) Initial position and target position. (b) Projection of the helical path on the  $O_E X_E Y_E$  plane. (c) Generated 3-D helical path. The initial position of RobCutt-II is  $P_s(x_s, y_s, z_s)$ , the target position is  $P_g(x_g, y_g, z_g)$ , and the starting point and the axis of helical path are  $S^H(x^H, y^H, z^H)$  and  $O^H(x_0^H, y_0^H)$ , respectively.

#### Algorithm 1: The 3-D Helical Path Generation.

- 1: Input the initial and target position and constraints.
- 2: Calculate the starting point  $S^H$  of the helical path.
- 3: Calculate the angles  $\alpha_s$  and  $\alpha_g$ .
- 4:  $\alpha_{temp} = \alpha_g - \alpha_s$ .
- 5: **if** (right-handed helix is required) **then**
- 6:    $(\alpha_{temp} \geq 0)? \alpha = 2\pi - \alpha_{temp} : \alpha = -\alpha_{temp}$ .
- 7: **else**
- 8:    $(\alpha_{temp} \geq 0)? \alpha = \alpha_{temp} : \alpha = \alpha_{temp} + 2\pi$ .
- 9: **end if**
- 10: Derive the helical path equation.

In addition, we set  $z^H = z_s$ .

Then, the angles between the lines  $\overrightarrow{O^H S^H}$ ,  $\overrightarrow{O^H P_g}$ , and the  $O_E X_E Y_E$  axis on the projection plane  $O_E X_E Y_E$  can be calculated, respectively. Take  $\alpha_s$  as an example

$$\alpha_s = \begin{cases} \pi + \frac{1}{2} \text{ssgn}(y_{se}) \cdot \pi, & x_{se} = 0 \\ \arctan\left(\frac{y_{se}}{x_{se}}\right) + \pi + \frac{1}{2} \text{ssgn}(y_{se}) \cdot \pi \\ -\frac{1}{2} \text{ssgn}(x_{se}) \text{ssgn}(y_{se}) \cdot \pi, & x_{se} \neq 0 \end{cases} \quad (4)$$

where  $x_{se} = x^H - x_0^H$ ,  $y_{se} = y^H - y_0^H$ .  $\text{ssgn}(\cdot)$  denotes a sign function defined as follows:  $\text{ssgn}(0) = 1$ ;  $\text{ssgn}(x) = 1, x > 0$ ;  $\text{ssgn}(x) = -1, x < 0$ .

Furthermore, the rotation angle  $\alpha$  can be derived.  $\alpha$  is the angle from  $\overrightarrow{O^H S^H}$  to  $\overrightarrow{O^H P_g}$ . It should be noted that helices can be either right handed or left handed. With the LOS along

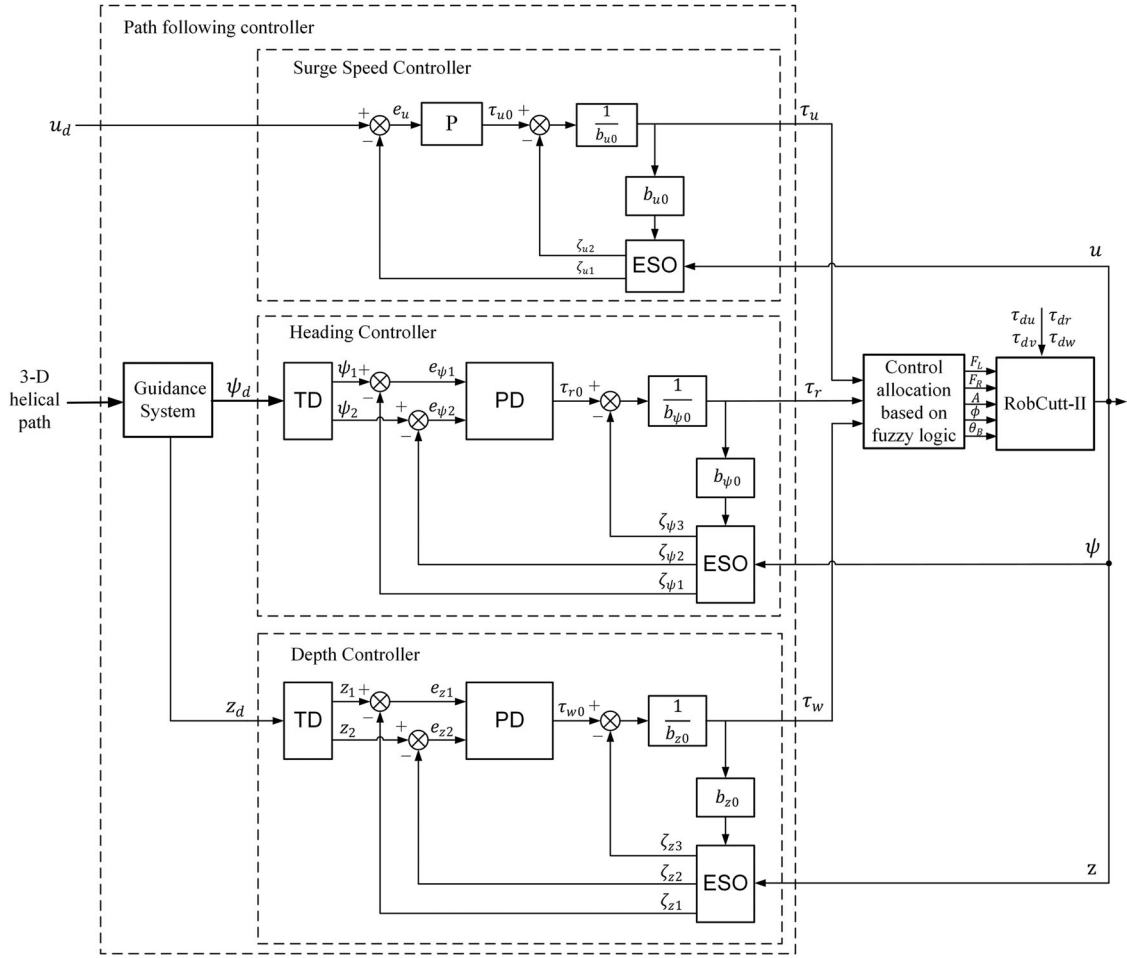


Fig. 6. Block diagram of the path-following controller.

the helix's axis, if a clockwise screwing motion moves the helix away from the observer, then it is called a right-handed helix, otherwise, it is a left-handed helix.

At last, denote a point on the helical path by  $P(x, y, z)$ , then the helical path [see Fig. 5(c)] equation can be obtained by

$$\begin{cases} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} x_0^H \\ y_0^H \end{bmatrix} + R_h \begin{bmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{bmatrix} \begin{bmatrix} q_{0x}^H \\ q_{0y}^H \end{bmatrix} \\ z = z^H + |\theta| \cdot \frac{h^H}{\alpha + 2\pi N} \end{cases} \quad (5)$$

where  $(q_{0x}^H, q_{0y}^H)^T$  is the normalized unit vector of  $(x^H - x_0^H, y^H - y_0^H)$ ,  $h^H = z_g - z_s$  is the height of the helical path, and  $|\theta| \in [0, \alpha + 2\pi N]$  is the rotation angle from the initial position  $S^H(x^H, y^H, z^H)$  to the position  $P(x, y, z)$  on the projection plane  $O_E X_E Y_E$ . If the right-handed helix is required,  $\theta \leq 0$ , otherwise,  $\theta > 0$ .

## V. PATH-FOLLOWING CONTROL

This section presents the control algorithm to solve the path following problem of RobCutt-II. Fig. 6 depicts the design of the path-following controller. Specifically, a guidance system based on the LOS principle [11] is implemented to deduce the target heading  $\psi_d$  and the target depth  $z_d$  based on position

feedback. Moreover, three ADRC controllers, i.e., the surge speed controller, the heading controller, and the depth controller are designed independently to output the required propulsive force and moment for RobCutt-II to follow the generated 3-D helical path in the presence of disturbances. Note that only the guidance system needs to be redesigned for other 3-D trajectories.

### A. Guidance System

The guidance system plans the motion of the RobCutt-II. As for a right-hand helical path, some notations of the proposed guidance system are given in Fig. 7.

- 1)  $P(x, y, z)$  and  $\psi$  are the 3-D position and the heading angle of RobCutt-II, respectively.
- 2)  $\gamma$  is the radius of the virtual circle associated with RobCutt-II.
- 3)  $\psi_d$  is the target heading angle.
- 4)  $P_{\text{los}}(x_d, y_d, z_d)$  defines the foresight point.
- 5)  $P'_{\text{los}}$  represents the other intersection.

The algorithm of the guidance system for determining the target heading  $\psi_d$  and the target depth  $z_d$  is described in Algorithm 2.

In particular,  $\alpha_{\text{total}}$  is the total rotation angle of the vehicle around the rotation axis on the projection plane  $O_E X_E Y_E$ ,

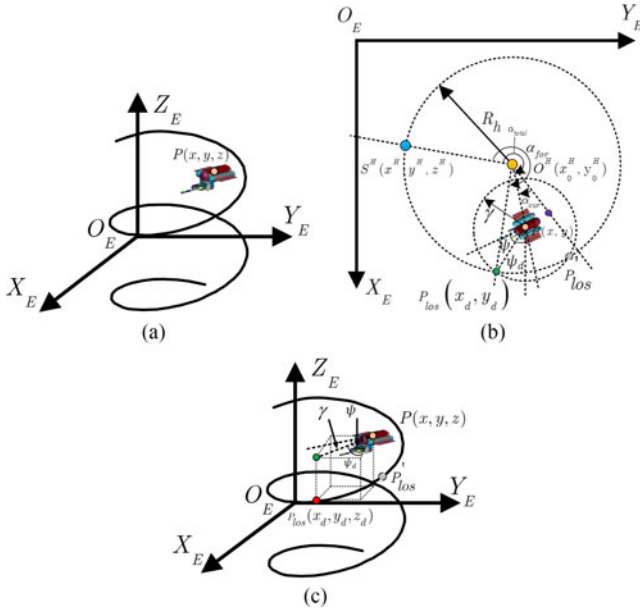


Fig. 7. Guidance system. (a) Current position of the vehicle. (b) Foresight point on the  $O_E X_E Y_E$  plane. (c) Foresight point in 3-D space.  $P(x, y, z)$  is the current position of RobCutt-II,  $P_{los}(x_d, y_d, z_d)$  is the foresight point.

---

**Algorithm 2:** The Determination of the Target Heading and the Target Depth.

---

- 1: **repeat**
  - 2:   Get the position and attitude feedback data.
  - 3:   Calculate the foresight point  $p_{los}(x_d, y_d)$  on the projection plane.
  - 4:   Calculate the angles  $\alpha_{cur}$  and  $\alpha_{for}$ .
  - 5:    $\alpha_{total} = \alpha_{total} + \alpha_{cur}$ .
  - 6:   **while**  $(\alpha_{for} < \alpha_{total} - \pi/4)$  **do**
  - 7:      $\alpha_{for} = \alpha_{for} + 2 \cdot \pi$ .
  - 8:   **end while**
  - 9:    $z_d = z^H + \alpha_{for} \cdot \frac{h^H}{\alpha + 2\pi N}$ .
  - 10:   Compute the target heading  $\psi_d$  according to  $p_{los}(x_d, y_d, z_d)$ .
  - 11: **until** the distance between the RobCutt-II and the target point is less than  $\gamma_g$
- 

$\alpha_{cur}$  denotes the rotation angle of the vehicle around the center of rotation in the current control period, and  $\alpha_{for}$  represents the corresponding rotation angle of the foresight point on the projection plane  $O_E X_E Y_E$ .

As shown in Fig. 7(a), suppose that the current position of RobCutt-II is  $P(x, y, z)$  (yellow point), we first calculate the foresight point on the  $O_E X_E Y_E$  plane. As illustrated in Fig. 7(b), RobCutt-II and the helical path are projected onto the  $O_E X_E Y_E$  plane, the purple point represents the position of RobCutt-II in the last control period. To find the foresight point that is  $\gamma$  apart from the barycenter of the vehicle on the projection circle, assume that a virtual circle centered at the barycenter of the vehicle with a radius equal to  $\gamma$  is associated with the vehicle. We set the foresight point to be the intersection of the virtual circle and the projection circle of the helical path.

Particularly, when there are two intersection points, the intersection point further forward along the helical path is selected as the current foresight point (green point), whose position can be derived by solving

$$(x - x_d)^2 + (y - y_d)^2 = \gamma^2 \quad (6)$$

$$(x_d - x_0^H)^2 + (y_d - y_0^H)^2 = R_h^2. \quad (7)$$

Otherwise, if there is no intersection, the point, which is both located on the projection circle and closest to the virtual circle, is chosen as the foresight point. In this case, the foresight point can be deduced from

$$(x - x_0^H)(y_d - y_0^H) - (y - y_0^H)(x_d - x_0^H) = 0 \quad (8)$$

$$(x_d - x_0^H)^2 + (y_d - y_0^H)^2 = R_h^2. \quad (9)$$

Furthermore, the 3-D position of the foresight point on the helical path is determined. In the case of  $N > 1$ , multiple corresponding foresight points can be found on the helical path. However, depending on the total rotation angle of the vehicle around the rotation axis, a single foresight point  $P_{los}(x_d, y_d, z_d)$  (red point) is determined as shown in Fig. 7(c).

Then, the target heading  $\psi_d$  is determined according to the foresight point

$$\psi_d = \begin{cases} \pi + \frac{1}{2} \text{ssgn}(y_e) \cdot \pi, & x_e = 0 \\ \arctan\left(\frac{y_e}{x_e}\right) + \pi + \frac{1}{2} \text{ssgn}(y_e) \cdot \pi & \\ -\frac{1}{2} \text{ssgn}(x_e) \text{ssgn}(y_e) \cdot \pi, & x_e \neq 0 \end{cases} \quad (10)$$

where  $x_e = x - x_d$  and  $y_e = y - y_d$ . The target depth is selected as the depth of the foresight point.

Next, we introduce the detailed control design of the surge speed controller, the heading controller, and the depth controller. It should be remarked that aforementioned kinetic model of RobCutt-II in Section III-A is a rough model with large model uncertainties due to the assumption of decoupling and symmetry and other simplifications. In addition, it is difficult to identify the model parameters accurately. Furthermore, the hydrodynamic analysis of undulatory propulsion is approximated by the fuzzy logic model. All of these uncertainties impose higher requirements on the design of three subcontrollers. Fortunately, the ADRC law is not predicated on an accurate and detailed dynamic model of the plant and is extremely tolerant of uncertainties and simple to use [34]. Therefore, we design three subcontrollers based on the ADRC technique.

### B. Surge Speed Controller

Based on the model of RobCutt-II, the dynamic equation of surge speed can be written as

$$\dot{u} = f_u(\nu, \tau_d, t) + b_u \tau_u \quad (11)$$

where  $f_u(\nu, \tau_d, t) = -(d_{11}/m_{11})u + (m_{22}/m_{11})vr + (m_{24}/m_{11})r^2 + (1/m_{11})\tau_{du}$  is the general disturbance for surge speed dynamics,  $b_u = (1/m_{11})$  is the control input gain.

The block diagram of the surge speed controller is given in Fig. 6. An extended state observer (ESO) provides the estimates of the internal dynamics of RobCutt-II and the external disturbances, which include the environmental disturbances and the unknown measurement error based on control signal  $\tau_u$  and system outputs  $u$  in real time. Once the ESO is designed and well-tuned, its outputs  $\zeta_{u1}$  will track  $u$  and  $\zeta_{u2} \approx f_u(\cdot) + (b_u - b_{u0})\tau_u$ , where  $b_{u0}$  denotes the estimation of the control gain  $b_u$ . By canceling the effect of  $f_u$  using  $\zeta_{u2}$ , the ADRC actively compensates for  $f_u$  in real time. The ADRC control law for plant (11) is given by

$$\tau_u = \frac{\tau_{u0} - \zeta_{u2}}{b_{u0}} \quad (12)$$

where  $\tau_{u0}$  will be specified later. By reporting the control signal given by (12) into the control input of plant (11), the plant (11) is simplified as

$$\dot{u} \approx \tau_{u0}. \quad (13)$$

Notice that the plant (11) is reduced to an integral system. Then, a proportional (P) controller is sufficient to control it

$$\tau_{u0} = k_{up}e_u \quad (14)$$

where  $k_{up}$  is the proportional gain and  $e_u = u_d - \zeta_{u1}$  denotes the state error.

The linear discrete ADRC controller can be expressed as (see [33])

$$\begin{cases} e_u = \zeta_{u1}(k) - u(k) \\ \zeta_{u1}(k+1) = \zeta_{u1}(k) + h_u[\zeta_{u2}(k) - 2\omega_{ou}e_u + b_{u0}\tau_u] \\ \zeta_{u2}(k+1) = \zeta_{u2}(k) - h_u\omega_{ou}^2e_u \\ \tau_{u0} = k_{up}[u_d - \zeta_{u1}(k+1)] \\ \tau_u = \frac{\tau_{u0} - \zeta_{u2}(k+1)}{b_{u0}} \end{cases} \quad (15)$$

where  $k$  denotes the  $k$ th sample constant,  $u_d$  is the target surge speed,  $u$  denotes the real-time surge speed of RobCutt-II,  $\zeta_{u1}$  is the estimation of surge speed,  $\zeta_{u2}$  is the estimation of the general disturbance,  $\omega_{ou}$  is the ESO bandwidth. Readers may refer to [34] for more details about the definitions of the ADRC parameters.

### C. Heading Controller

Analogously, the dynamic equation of yaw motion is obtained based on (1) as

$$\begin{cases} \dot{\psi} = r \\ \dot{r} = f_\psi(\nu, \tau_d, t) + b_\psi\tau_r \end{cases} \quad (16)$$

where  $f_\psi(\nu, \tau_d, t) = -(d_{42}/m_{44})v - (d_{44}/m_{44})r + (m_{11} - m_{22}/m_{44})uv - (m_{24}/m_{44})ur - (m_{24}/m_{44})\dot{v} + (1/m_{44})\tau_{dr}$  is a multivariable function of system states, external disturbances, and time, and  $b_\psi = (1/m_{44})$  denotes the control gain. Compared with the surge speed dynamics, the heading dynamics is a second-order system instead of a first-order system.

Specifically, Fig. 6 shows that a tracking differentiator is used to obtain the differential signal and the tracking signal of the setpoint. An ESO outputs the estimations (i.e.,  $\zeta_{\psi1}$ ,  $\zeta_{\psi2}$ ,  $\zeta_{\psi3}$ ) of the system states  $\psi$ ,  $\dot{\psi}$ , and the general disturbances  $f_\psi(\cdot) + (b_\psi - b_{\psi0})\tau_r$  based on heading angle feedback. By actively compensating  $f_\psi$  using  $\zeta_{\psi3}$ , the control law for plant (16) is given by

$$\tau_r = \frac{\tau_{r0} - \zeta_{\psi3}}{b_{\psi0}}. \quad (17)$$

Therefore, the plant (16) is reduced to a double integrator

$$\ddot{\psi} \approx \tau_{r0}. \quad (18)$$

We use a proportional-derivative controller to control it

$$\tau_{r0} = k_{\psi p}e_{\psi1} + k_{\psi d}e_{\psi2} \quad (19)$$

where  $k_{\psi p}$  and  $k_{\psi d}$  are the proportional gain and derivative gain, respectively,  $e_{\psi1} = \psi_1 - \zeta_{\psi1}$  and  $e_{\psi2} = \psi_2 - \zeta_{\psi2}$  are states errors,  $\psi_1$  is the tracking signal of  $\psi_d$  and  $\psi_2$  is the differential signal of  $\psi_1$  subject to the acceleration limit of  $\delta_\psi$ . The linear discrete ADRC controller can be listed as

$$\begin{cases} \psi_1(k+1) = \psi_1(k) + h_\psi\psi_2(k) \\ \psi_2(k+1) = \psi_2(k) + h_\psi\text{fhan}(\psi_1(k) - \psi_d, \psi_2(k), \delta_\psi, h_{\psi0}) \\ e_\psi = \zeta_{\psi1}(k) - \psi(k) \\ \zeta_{\psi1}(k+1) = \zeta_{\psi1}(k) + h_\psi[\zeta_{\psi2}(k) - 3\omega_{o\psi}e_\psi] \\ \zeta_{\psi2}(k+1) = \zeta_{\psi2}(k) + h_\psi[\zeta_{\psi3}(k) - 3\omega_{o\psi}^2e_\psi + b_{\psi0}\tau_r] \\ \zeta_{\psi3}(k+1) = \zeta_{\psi3}(k) - h\omega_o^3e_\psi \\ e_{\psi1} = \psi_1(k+1) - \zeta_{\psi1}(k+1) \\ e_{\psi2} = \psi_2(k+1) - \zeta_{\psi2}(k+1) \\ \tau_{r0} = k_{\psi p}e_{\psi1} + k_{\psi d}e_{\psi2} \\ \tau_r = \frac{\tau_{r0} - \zeta_{\psi3}(k+1)}{b_{\psi0}} \end{cases} \quad (20)$$

where  $h_\psi$  is the heading control period,  $h_{\psi0}$  denotes the filtering factor related to the suppression of noise,  $\omega_{o\psi}$  refers to the desired natural frequency of the closed-loop system,  $\text{fhan}(\cdot)$  is the optimal control synthesis function of the discrete-time system [34].

### D. Depth Controller

The dynamic model of heave motion deduced from (1) can be described as

$$\begin{cases} \dot{z} = w \\ \dot{w} = f_z(\nu, \tau_d, t) + b_z\tau_w \end{cases} \quad (21)$$

where  $b_z = (1/m_{33})$ ,  $f_z$  represents the general disturbance including the external disturbance and internal dynamics. Specially,  $f_z(\nu, \tau_d, t) = -(d_{33}/m_{33})w + (1/m_{33})\tau_{dw}$ .

Since the dynamic equation represented by (21) is similar to (16), an analogous ADRC controller to force RobCutt-II to track the target depth is implemented. The detailed expressions are omitted here.



TABLE I  
CONTROLLER PARAMETERS

Surge	$\delta_u$	$h_{u0}$	$\omega_{ou}$	$b_{u0}$	$k_{up}$
	0.009	0.1	3.5	0.02	6.25
Heading	$\delta_\psi$	$h_{\psi0}$	$\omega_{o\psi}$	$b_{\psi0}$	$k_{\psi p}$
	0.08	0.1	4	0.28	16
Depth	$\delta_z$	$h_{z0}$	$\omega_{oz}$	$b_{z0}$	$k_{zp}$
	0.02	0.1	5	0.28	16

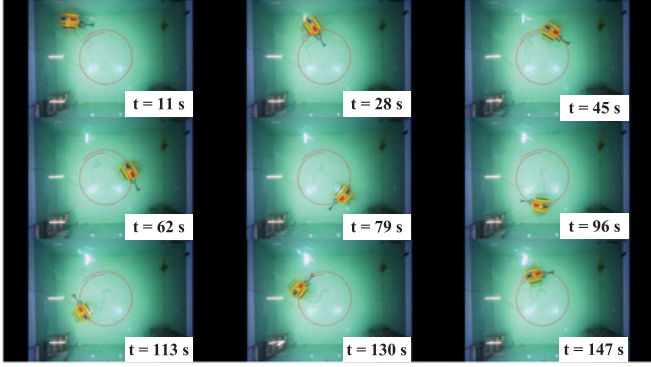


Fig. 8. Image sequences of RobCutt-II during path following of a helical path.

## VI. EXPERIMENTAL RESULTS

The control scheme for 3-D helical path following of RobCutt-II was tested in an indoor pool with dimensions of 5 m  $\times$  4 m  $\times$  1.1 m (length  $\times$  width  $\times$  depth).

In the first experiment, RobCutt-II started with initial position and orientation  $(x_0, y_0, z_0, \psi_0) = (0.498 \text{ m}, 0.592 \text{ m}, 0.348 \text{ m}, 6.172 \text{ rad})$ . The target position was (2.300 m, 0.900 m, 0.600 m). We set the rotation axis to (2.300 m, 2.000 m) with radius 1.1 m. RobCutt-II revolved around the rotation axis and should complete at least  $N = 1$  circle. The control period was 0.05 s and the ADRC parameters of the path-following controller are tabulated in Table I. Notice that these parameters can be determined according to the previous heading keeping and depth control experiments [8], [24]. Remarkably, compared with the PID controller, the parameters of the ADRC controller have a wider adaptive range, which helps to quickly determine the appropriate parameters of the controllers. In addition, the target surge speed was  $u_d = 0.08 \text{ m/s}$ ,  $\gamma_g$  was set to 0.1 m, and the radius  $\gamma$  of the virtual circle of RobCutt-II was 0.3 m.

The experimental results based on these conditions are presented in Figs. 8–12. The image sequences of RobCutt-II during path following are shown in Fig. 8. The reference helical path is presented by a red dotted line. Fig. 9 shows that the path generator first planned a helical path satisfying the constraint conditions. RobCutt-II was initially off the path, with an initial distance error of about 1.2 m. Then, RobCutt-II could catch up and land on the desired path smoothly. Fig. 10 shows the time evolution of the tracking error, which was calculated by the following three steps:

- 1) found the point (denoted as  $P_1$ ) on the projection circle of the helical path nearest the projection point of RobCutt-II on the projection plane  $O_E X_E Y_E$ ;

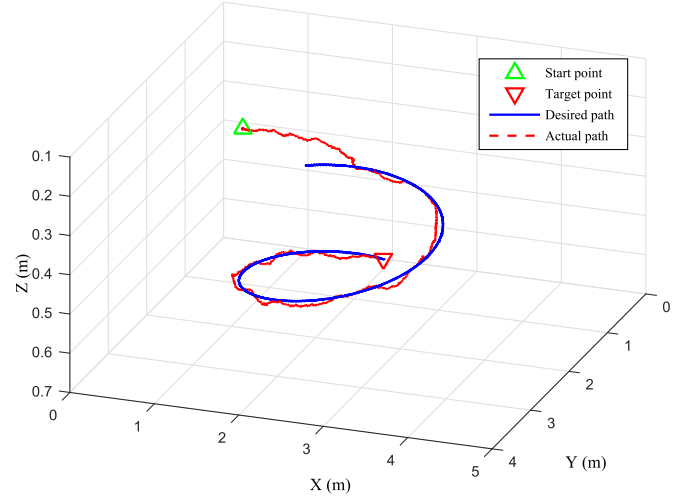


Fig. 9. Desired path and actual path of RobCutt-II.

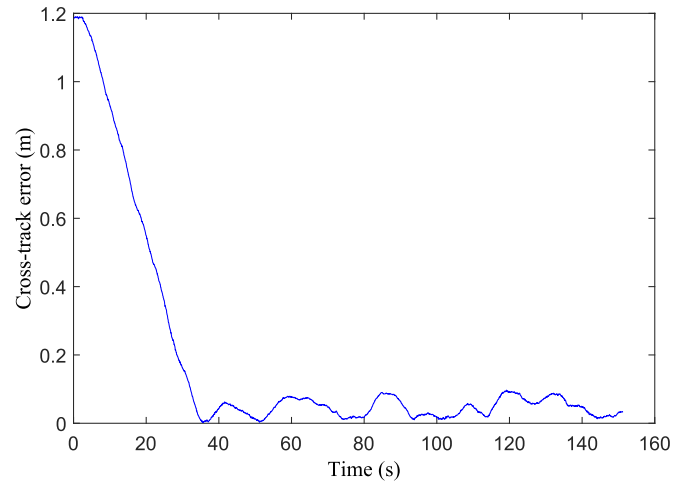


Fig. 10. Time evolution of the path-following error.

- 2) found the point  $P_2$  on the helical path corresponding to  $P_1$ ;
- 3) the 3-D distance between the barycenter of RobCutt-II to  $P_2$  was calculated as the current tracking error.

In Fig. 10, the convergence of the path-following error states that path following was achieved. The maximum following error after RobCutt-II converged to the helical path is 0.1 m, which is about one-eighth of the length of the main body of RobCutt-II. Fig. 11 gives the time evolution of the body-fixed velocity. The red dotted line indicates the desired surge speed, and the blue solid lines are the actual velocities. Note that the velocity fluctuation of RobCutt-II is observed. The reason may be that the position given by the global visual tracking system contains error imbedded with the true value due to accidental disturbances. The differential calculation caused more serious fluctuation of the body-fixed velocity. But overall the average of the surge speed is close to the desired surge speed. The control parameters of propagating waves on bilateral fins are shown in Fig. 12. Note that the turning motion of RobCutt-II was realized



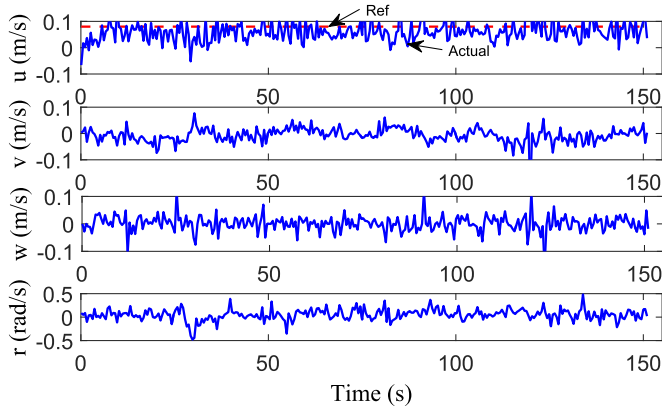


Fig. 11. Time evolution of the body-fixed velocity.

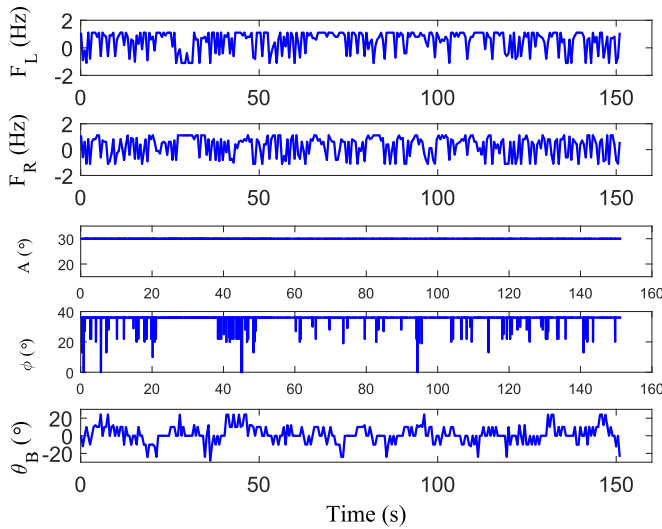


Fig. 12. Time history of the control parameters of propagating waves on bilateral fins.

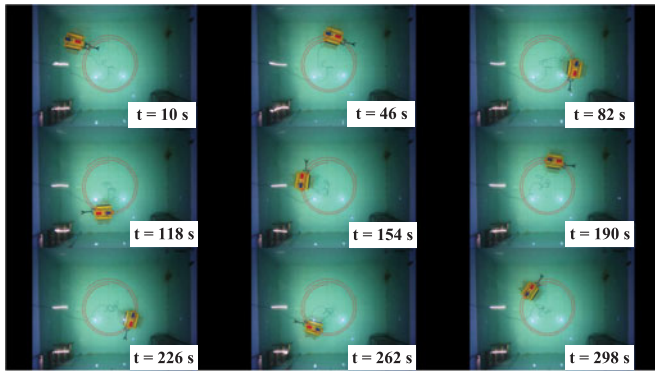
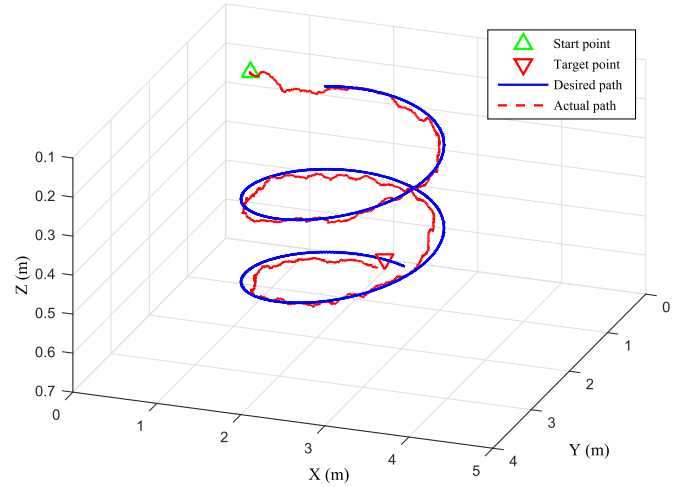
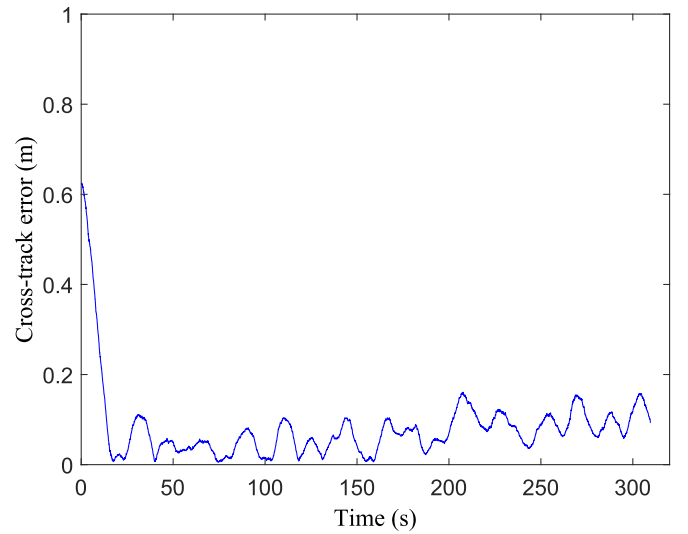


Fig. 13. Image sequences of RobCutt-II during path following of a helical path with two laps.

by regulating different frequency of bilateral ribbon-fins. Specially, the sign of fin frequency represents the direction of the propagating waves. If the undulatory frequency was positive, the traveling wave propagated from the head of RobCutt-II to the tail. Otherwise, negative undulatory frequency indicates that

Fig. 14. Desired path and actual path of RobCutt-II when  $N = 2$ .Fig. 15. Time evolution of the path-following error when  $N = 2$ .

the traveling wave propagated from the tail of the vehicle to the head. The amplitudes of the bilateral ribbon-fins kept at  $30^\circ$ . In addition, the deflection angles of bilateral fins were regulated to produce the heave force. In particular, the deflection angles of fins fluctuated between  $-30^\circ$  and  $30^\circ$  in the helical path following experiment. When the deflection angle was greater than zero, a positive heave force was produced and vice versa. Thus, its depth control can be realized by controlling fin surfaces.

Then, another case to follow a helical path with two laps (namely,  $N = 2$ ) was studied. The initial position and orientation of RobCutt-II was (0.806 m, 1.138 m, 0.140 m), and the target position was (2.300 m, 0.900 m, 0.600 m). We set  $\gamma = 0.36$  m,  $u_d = 0.06$  m/s. Note that we fine-tuned the parameters to achieve better path-following results. Other settings were the same as those of the previous experiment.

Figs. 13–17 show the experimental results of path following. It is seen that similar results were obtained in this experiment.

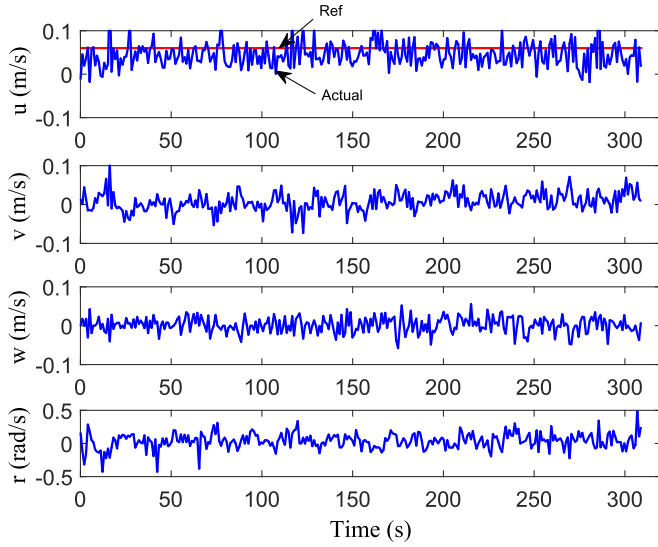


Fig. 16. Time evolution of the body-fixed velocity when  $N = 2$ .

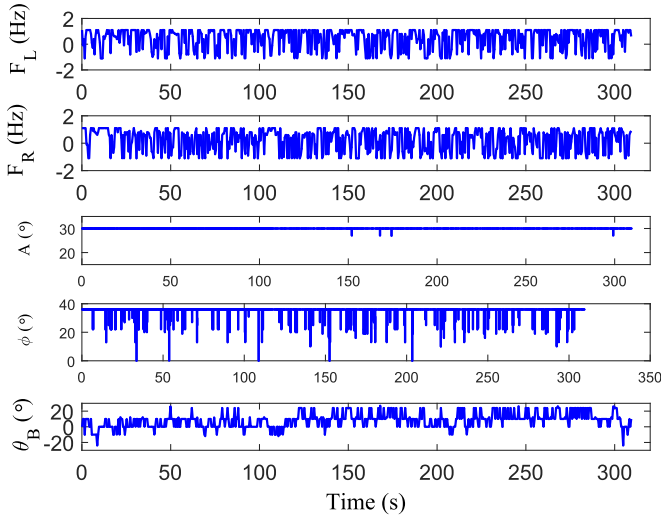


Fig. 17. Time history of the control parameters of propagating waves on bilateral fins when  $N = 2$ .

The generated helical curve equation is

$$\begin{cases} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} 2.3 \\ 2.0 \end{bmatrix} + \begin{bmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{bmatrix} \begin{bmatrix} -0.953 \\ -0.550 \end{bmatrix} \\ z = 0.14 + 0.034\theta \end{cases} \quad (22)$$

where  $\theta \in [0, 13.613]$ . This proves the feasibility of the proposed path generation algorithm.

Fig. 13 shows the image sequences of RobCutt-II during path following. Trajectories of the barycenter of RobCutt-II are traced in Fig. 14. Fig. 15 shows the time evolution of the path-following error. Fig. 16 shows the time evolution of the body-fixed velocity. Fig. 17 shows the time evolution of the control parameters of propagating waves on bilateral fins. By virtue of the path-following control scheme, RobCutt-II converged to the helical path and moved along the planned path. It is observed from Fig. 15 that the tracking error is larger than that of  $N = 1$ . Specially, the path-following error is increased on the second lap

of path following. The reason may be that the indoor pool was not large enough to remove the effects of the reflective waves, so RobCutt-II was subjected to external disturbances when it swam in the pool for a long time. Nevertheless, the maximum cross-track error after RobCutt-II landed on the helical path is only 0.16 m, which is also far less than the size of RobCutt-II. Thus, it can be concluded that the proposed path-following controller makes the RobCutt-II trajectory robust enough to respond to the disturbances and to achieve a satisfactory recovery.

At this point, the control scheme for 3-D helical path following of RobCutt-II in the 3-D underwater space is validated. To the best of our knowledge, the closed-loop control for 3-D helical path following of an underwater vehicle-manipulator system propelled by undulatory fins is realized for the first time.

## VII. CONCLUSION AND FUTURE WORK

The effectiveness of a 3-D helical path-following control scheme has been experimentally investigated on a UBVMs (RobCutt-II). RobCutt-II mimics the unique undulatory propulsion mode of cuttlefish. The 3-D pose estimation including the position and orientation of RobCutt-II is implemented based on the real-time tracking on the images taken by a global visual tracking system. The control scheme consists of a 3-D helical path generator, a guidance system based on LOS principle, three ADRC subcontrollers, and the fuzzy method for control allocation. Two 3-D helical path-following experiments were performed. The experimental results show that RobCutt-II achieves closed-loop 3-D helical path following.

In future work, we will study the waveform of propagating waves on bilateral fins. In particular, the counter propagating waves may be used to enhance the maneuverability and stability of the robotic ribbon-fin propulsion. Moreover, the hovering control and the coordinated motion control of the vehicle-manipulator system will be studied. Our final aim is naturally to achieve future applications of the designed UBVMs for underwater autonomous movements and operations.

## REFERENCES

- [1] I. Neveln *et al.*, "Biomimetic and bio-inspired robotics in electric fish research," *J. Exp. Biol.*, vol. 216, no. 13, pp. 2501–2514, 2013.
- [2] B. Ferreira, A. Matos, N. Cruz, and A. Moreira, "Coordination of marine robots under tracking errors and communication constraints," *IEEE J. Ocean. Eng.*, vol. 41, no. 1, pp. 27–39, Jan. 2016.
- [3] G. Antonelli, *Underwater Robots*. Berlin, Germany: Springer-Verlag, Ch. 1, 2014.
- [4] I. Yamamoto, *Practical Robotics and Mechatronics: Marine, Space and Medical Applications*. Stevenage, U.K.: Inst. Eng. Technol., Ch. 1, 2016.
- [5] M. Maciver, E. Fontaine, and J. Burdick, "Designing future underwater vehicles: Principles and mechanisms of the weakly electric fish," *IEEE J. Ocean. Eng.*, vol. 29, no. 3, pp. 651–659, Jul. 2004.
- [6] R. Torres, O. Curet, G. Lauder, and M. MacIver, "Kinematics of the ribbon fin in hovering and swimming of the electric ghost knifefish," *J. Exp. Biol.*, vol. 216, no. 5, pp. 823–834, 2013.
- [7] C. Zhou and K. Low, "Design and locomotion control of a biomimetic underwater vehicle with fin propulsion," *IEEE/ASME Trans. Mechatron.*, vol. 17, no. 1, pp. 25–35, Feb. 2012.
- [8] Q. Wei, S. Wang, Y. Wang, C. Zhou, and M. Tan, "Course and depth control for a biomimetic underwater vehicle—RobCutt-I," *Int. J. Offshore Polar Eng.*, vol. 25, no. 2, pp. 81–87, 2015.
- [9] L. Lapiere and B. Jouvencel, "Robust nonlinear path-following control of an AUV," *IEEE J. Ocean. Eng.*, vol. 33, no. 2, pp. 89–102, Apr. 2008.

- [10] G. Zhang and X. Zhang, "Concise robust adaptive path-following control of underactuated ships using DSC and MLP," *IEEE J. Ocean. Eng.*, vol. 39, no. 4, pp. 685–694, Oct. 2014.
- [11] T. Fossen, K. Pettersen, and R. Galeazzi, "Line-of-sight path following for Dubins paths with adaptive sideslip compensation of drift forces," *IEEE Trans. Control Syst. Technol.*, vol. 23, no. 2, pp. 820–827, Mar. 2015.
- [12] M. Breivik and T. Fossen, "Path following of straight lines and circles for marine surface vessels," in *Proc. IFAC Conf. Control Appl. Marine Syst.*, Ancona, Italy, 2004, pp. 65–70.
- [13] H. Katayama and H. Aoki, "Straight-line trajectory tracking control for sampled-data underactuated ships," *IEEE Trans. Control Syst. Technol.*, vol. 22, no. 4, pp. 1638–1645, Jul. 2014.
- [14] A. Lekkas and T. Fossen, "Integral LOS path following for curved paths based on a monotone cubic Hermite spline parametrization," *IEEE Trans. Control Syst. Technol.*, vol. 22, no. 6, pp. 2287–2301, Nov. 2014.
- [15] M. Serrano, G. Scaglia, S. Godoy, V. Mut, and O. Ortiz, "Trajectory tracking of underactuated surface vessels: a linear algebra approach," *IEEE Trans. Control Syst. Technol.*, vol. 22, no. 3, pp. 1103–1111, May 2014.
- [16] H. Huang, Q. Tang, Y. Li, L. Wan, and Y. Pang, "Dynamic control and disturbance estimation of 3D path following for the observation class underwater remotely operated vehicle," *Adv. Mech. Eng.*, vol. 2013, no. 604393, pp. 1–16, 2013.
- [17] P. Encarnacao and A. Pascoal, "3D path following for autonomous underwater vehicle," in *Proc. 39th IEEE Conf. Decis. Control*, Sydney, NSW, Australia, Dec. 2000, pp. 2977–2982.
- [18] N. Fischer, D. Hughes, P. Walters, E. Schwartz, and W. Dixon, "Nonlinear RISE-based control of an autonomous underwater vehicle," *IEEE Trans. Robot.*, vol. 30, no. 4, pp. 845–852, Aug. 2014.
- [19] B. Verzijlbergen and M. Jenkin, "Swimming with robots: Human robot communication at depth," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Taipei, Taiwan, Oct. 2010, pp. 4023–4028.
- [20] S. Bi, C. Niu, Y. Cai, L. Zhang, and H. Zhang, "A waypoint-tracking controller for a bionic autonomous underwater vehicle with two pectoral fins," *Adv. Robot.*, vol. 28, no. 10, pp. 673–681, 2014.
- [21] R. Wang, S. Wang, Y. Wang, and C. Tang, "Path following for a biomimetic underwater vehicle based on ADRC," in *Proc. IEEE Int. Conf. Robot. Autom.*, Singapore, May 2017, pp. 3519–3524.
- [22] M. Reyhanoglu, A. van der Schaft, N. McClamroch, and I. Kolmanovsky, "Dynamics and control of a class of underactuated mechanical systems," *IEEE Trans. Autom. Control*, vol. 44, no. 9, pp. 1663–1671, Sep. 1999.
- [23] 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, Aug. 2007.
- [24] R. Wang, S. Wang, and Y. Wang, "A hybrid heading control scheme for a biomimetic underwater vehicle," in *Proc. 26th Int. Ocean Polar Eng. Conf.*, Rhodes, Greece, Jun. 2016, pp. 619–625.
- [25] 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 Trans. Mechatron.*, vol. 21, no. 2, pp. 815–824, Apr. 2016.
- [26] T. Fossen, *Guidance and Control of Ocean Vehicles*. New York, NY, USA: Wiley, Ch. 2, 1994.
- [27] T. Fossen, *Marine Control Systems: Guidance, Navigation and Control of Ships, Rigs and Underwater Vehicles*. Tiller, Norway: Marine Cybern., Ch. 2, 2002.
- [28] O. Curet, N. Patankar, G. Lauder, and M. MacIver, "Mechanical properties of a bio-inspired robotic knife-fish with an undulatory propulsor," *Bioinspiration Biomimetics*, vol. 6, no. 2, 2011, Art. no. 026004.
- [29] H. Liu, B. Taylor, and O. Curet, "Fin ray stiffness and fin morphology control ribbon-fin-based propulsion," *Soft Robot.*, vol. 4, no. 2, pp. 103–116, 2017.
- [30] Q. Wei, S. Wang, X. Dong, L. Shang, and M. Tan, "Design and kinetic analysis of a biomimetic underwater vehicle with two undulating long-fins," *Acta Automatica Sinica*, vol. 39, no. 8, pp. 1330–1338, 2013.
- [31] M. Epstein, J. Colgate, and M. MacIver, "Generating thrust with a biologically inspired robotic ribbon fin," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Beijing, China, Oct. 2006, pp. 2412–2417.
- [32] R. Wang, S. Wang, Y. Wang, M. Tan, and J. Yu, "A paradigm for path following control of a ribbon-fin propelled biomimetic underwater vehicle," *IEEE Trans. Syst., Man Cybern., Syst.*, 2017, DOI: [10.1109/TSMC.2017.2705340](https://doi.org/10.1109/TSMC.2017.2705340).
- [33] Z. Gao, "Scaling and bandwidth-parameterization based controller tuning," in *Proc. Amer. Control Conf.*, 2003, pp. 4989–4996.
- [34] J. Han, *Active Disturbance Rejection Control Technique—The Technique for Estimating and Compensating the Uncertainties*. Beijing, China: Nat. Defense Ind. Press, Ch. 2–5, 2008.



**Rui Wang** received the B.E. degree in automation from Beijing Institute of Technology, Beijing, China, in July 2013. He is currently working toward the Ph.D. degree in control theory and control engineering in the State Key Laboratory of Management and Control for Complex Systems, Institute of Automation, Chinese Academy of Sciences, Beijing, China.

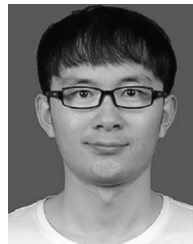
He is currently also at the University of Chinese Academy of Sciences, Beijing, China. His research interests include intelligent control, robotics, and biomimetic robots.



**Shuo Wang** received the B.E. degree in electrical engineering from the Shenyang Architecture and Civil Engineering Institute, Shenyang, China, in 1995, the M.E. degree in industrial automation from Northeastern University, Shenyang, China, in 1998, and the Ph.D. degree in control theory and control engineering from the Institute of Automation, Chinese Academy of Sciences, Beijing, China, in 2001.

He is currently a Professor in the State Key Laboratory of Management and Control for Complex Systems, Institute of Automation, Chinese Academy of

Sciences. His research interests include biomimetic robot, underwater robot, and multirobot systems.



**Yu Wang** received the B.E. degree in automation from Beijing Institute of Technology, Beijing, China, in 2011 and the Ph.D. degree in control theory and control engineering from the Institute of Automation, Chinese Academy of Sciences, Beijing, China, in 2016.

He is currently a Research Assistant in the State Key Laboratory of Management and Control for Complex Systems, Institute of Automation, Chinese Academy of Sciences. His research interests include intelligent control, robotics, and biomimetic robots.



**Chong Tang** received the B.E. degree in automation from Northwestern Polytechnical University, Xi'an, China, in July 2014. He is currently working toward the Ph.D. degree in control theory and control engineering in the State Key Laboratory of Management and Control for Complex Systems, Institute of Automation, Chinese Academy of Sciences, Beijing, China, and University of Chinese Academy of Sciences, Beijing, China.

His research interests include intelligent control, robotics, and underwater biomimetic robots, underwater computer vision.



**Min Tan** received the B.E. degree in automation from Tsinghua University, Beijing, China, in 1986 and the Ph.D. degree in control theory and control engineering from the Institute of Automation, Chinese Academy of Sciences, Beijing, China, in 1990.

He is currently a Professor in the State Key Laboratory of Management and Control for Complex Systems, Institute of Automation, Chinese Academy of Sciences, Beijing, China. His research interests include advanced robot control, biomimetic robot, multirobot system.