
DOI: [10.5772/62289](http://dx.doi.org/10.5772/62289)

This is the published version.

©2016, The Authors

Reproduced by Deakin University under the terms of the [Creative Commons Attribution Licence](http://creativecommons.org/licenses/by/4.0/)

Available from Deakin Research Online:

[http://hdl.handle.net/10536/DRO/DU:30083796](http://hdl.handle.net/10536/DRO/DU:30083796)
Heading Control for a Robotic Dolphin Based on a Self-tuning Fuzzy Strategy

Regular Paper

Zhiqiang Cao, Fei Shen*, Chao Zhou, Nong Gu, Saeid Nahavandi and De Xu

1 State Key Laboratory of Management and Control for Complex Systems, Institute of Automation, Chinese Academy of Sciences, Beijing, China
2 Research Center of Precision Sensing and Control, Institute of Automation, Chinese Academy of Sciences, Beijing, China
3 Centre for Intelligent Systems Research, Deakin University, Australia

*Corresponding author(s) E-mail: fei.shen@ia.ac.cn

Received 03 October 2015; Accepted 22 January 2016

DOI: 10.5772/62289

© 2016 Author(s). Licensee InTech. This is an open access article distributed under the terms of the Creative Commons Attribution License (http://creativecommons.org/licenses/by/3.0), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Abstract

In this paper, a heading controller based on a self-tuning fuzzy strategy for a robotic dolphin is proposed to improve control accuracy and stability. The structure of the robotic dolphin is introduced and the turning motion is analysed. The analytic model indicates that the turning joint angle can be employed for the heading control. This non-linear model prevents the successful application of traditional model-based controllers. A fuzzy controller is proposed to realize the heading control in our work. It should be mentioned that the traditional fuzzy controller suffers from a distinguished steady-state error, due to the fact that the heading range is relatively large and the fuzzy controller’s universe of discourse is fixed. To resolve this problem, a self-tuning mechanism is employed to adjust the input and output scaling factors according to the active working region in pursuit of favourable performance. Experimental results demonstrate the performance of the proposed controller in terms of steady-state error and robustness to interferences.

Keywords Robotic Dolphin, Heading Control, Turning Motion, Self-tuning Fuzzy Control

1. Introduction

In recent decades, extensive efforts have focused on researching unmanned underwater vehicles [1, 2] for marine resource exploitation, ocean exploration and rescue services. The efficient propulsion and good manoeuvrability of marine animals, such as cetaceans and fish, have motivated researchers to take advantage of these properties to create novel advanced underwater vehicles. Against this background, some bio-inspired swimming robots have been developed: robotic fish [3-6], a jellyfish- and butterfly-inspired microrobot [7], and a robotic dolphin. As one kind of cetaceans, dolphins are regarded as one of the best ocean swimmers in that the propulsive efficiency of dolphins can reach up to 0.75-0.9 and the maximum swimming speed is over 11m/s. Moreover, dolphins can also achieve excellent turning manoeuvrability and rotate their bodies with an angular speed of 450°/s with turning radii down to 11-17% of body length (BL) [8].

Attracted by these desired skills, many researchers have focused on developing dolphin-like robots. Fruitful research outcomes have been achieved in the areas of mechatronic design, hydrodynamics analysis and control schemes. Pioneering research in mechatronic design has led to several typical prototypes, including a two-joint robotic dolphin [9, 10], a four-joint pneumatic robotic dolphin [11], a five-joint robotic dolphin with a pair of two-degrees-of-freedom (2-DOF) pectoral fins, a robotic dolphin with a two-motor-driven scotch yoke mechanism [12, 13], and a multi-link robotic dolphin with 3-DOF flippers [14]. Many researchers devote themselves to the...
hydrodynamics analysis of robotic dolphins, which result in improved control performance. Dogangil et al. presented a dynamical behaviour model using the Newton-Euler formulation [11]. Shen et al. developed a propulsive dynamic model based on Kane’s method, which enables the speed optimization of the robotic dolphin [15]. To mimic different swimming motions of dolphins, several control schemes have been implemented. Nakashima et al. focused on the roll and loop-the-loop motion with PD control [10], while Wang et al. aimed to achieve a porpoising motion to save energy at high speed [16]. Shen et al. proposed a fuzzy PID controller, which allows a robotic dolphin to cruise within a given depth [17]. Moreover, two hybrid acrobatic stunts, i.e., front flip and backflip, were implemented in a robotic dolphin by Yu [18]. It should be noted that there is urgent need to maintain a consistent heading direction for a robotic dolphin, which is a prerequisite for long-time navigation. However, heading control is challenging because unavoidable waves and turbulences in open water raise unexpected disturbances. These disturbances will force robotic dolphins to deviate from their desired courses.

In this paper, a multi-link robotic dolphin with a turning joint is designed to realize the turning motion, which paves the foundation for the heading control. Based on a hydrodynamic model, the turning motion analysis is carried out, which indicates that one can employ the yaw angle of the turning joint to control the heading, while the relationship between these two variables is non-linear. Similar to robotic fish [19, 20], it is very difficult to establish a precise mathematical model, since the heading control of a robotic dolphin involves both hydrodynamics of the fluid environment and dynamics of the robot. In this case, the fuzzy controller provides an effective solution. In this paper, a self-tuning fuzzy heading controller is proposed to mitigate the non-linearity and reject the unexpected disturbances mentioned above.

The rest of this paper is organized as follows. Section 2 introduces the structure of a robotic dolphin; the turning motion is also analysed. Section 3 presents a self-tuning fuzzy heading controller. The experimental results are given in Section 4. Finally, the paper is concluded.

2. Structure of a Robotic Dolphin and Turning Motion Analysis

2.1 Structure of a Robotic Dolphin

In nature, dolphins achieve turning manoeuvre either by flexing the anterior body laterally with a special turning unit, or flapping and rotating flippers to produce a yawing moment. Furthermore, they can simultaneously adopt these two modes to achieve a more agile and swift turning [21]. Inspired by this fact, two methods are usually employed to achieve the turning motion of a robotic dolphin. One is to design a turning mechanism between the anterior body and tail part, while the other is to adopt mechanical flippers to realize turning. Since the second method requires a sophisticated seal of pectoral fins, the former one is adopted in this paper.

An experimental robotic dolphin is given in Fig. 1, while Fig. 1(a) demonstrates the structure of the robot. Joints 1-3, which are driven by servos in the rear part of the robotic dolphin, are used to provide the propelling force. A turning joint located in the middle of the robot is designed to achieve the turning motion. The pitch motion of the robotic dolphin is implemented by a barycentre adjustment mechanism. Besides a GPS and a pressure sensor for the robotic dolphin, an inertial sensor, with its y-axis being parallel to the horizontal medial axis of the robotic dolphin, is utilized to sense the heading direction.

Figure 1. An experimental robotic dolphin: (a) The diagram of the robotic dolphin’s structure; (b) the robotic dolphin prototype

2.2 Turning Motion Analysis

As shown in Fig. 2, a plane coordinate frame $O_1x_1z_1$ is established, where $O_1$ is the gravity centre of the robotic dolphin without turning motion, $O_1x_1$ is the horizontal medial axis of the robotic dolphin and $O_1z_1$ is the pitching axis perpendicular to $O_1x_1$ on the horizontal plane. In addition, $B_2$, $B_3$ and $B_4$ are labelled as the gravity centre of the whole robotic dolphin, head part and tail part, respectively, while $J$ is the turning centre of the turning joint. We denote $q_1$, $q_2$ and $q_3$ with the distances between $J$ and $B_2$, $O_1$ and $B_3$ and $O_1$ and $J$, respectively. When the robotic dolphin is not turning, $B_3$ coincides with $O_1$. When the tail of the robotic dolphin deviates at $\theta$ around the turning joint, the coordinate of the gravity centre of the whole robotic dolphin $\Delta x$ changes to $(\Delta x, -\Delta z)$ in $O_1x_1z_1$, where $\Delta x$ and $\Delta z$ are expressed as follows:

$$\Delta x = -\frac{m_1(q_3 + q_1 \cos \theta) - m_2 q_2}{m_1 + m_2}$$

(1)
\[ \Delta z = \frac{m_q q_1 \sin \theta}{m_h + m_t} \]  

where \( m_h \) is the mass of the robotic dolphin’s head part and \( m_t \) is the mass of the robotic dolphin’s tail part, which is from the turning joint to the fluke.

\[ F_{t1} = f(v) = \begin{cases} 
0.5 \rho A_{t1} (v(t) \cos \theta)^2 (C_{h1} + C_{d,1} + C_{L,1}) & \alpha \in (-\alpha_{stall}, \alpha_{stall}) \\
0.5 \rho A_{t1} (v(t) \cos \theta)^2 C_{h1} & \alpha \notin (-\alpha_{stall}, \alpha_{stall}) 
\end{cases} \]

where \( A_{t1} \) is the tail fin area of the robotic dolphin, \( C_{h1}, C_{d,1}, \) and \( C_{L,1} \) are the coefficients of lift, induced drag, profile drag and section drag, respectively. \( \alpha \) is the attack angle of the tail fin and \( \alpha_{stall} \) is the stalling angle of attack. The yawing moment \( M \) formed by the above forces is:

\[ M = F_{t1} l_{t1} + F_{t2} l_{t2} \]

where \( l_{t1} \) and \( l_{t2} \) are arms of \( F_{t1} \) and \( F_{t2} \), respectively, and:

\[ l_{t1} = \frac{\Delta z}{\cos \theta} + (\Delta z \tan \theta + q_1) \sin \theta = \frac{m_q (q_2 + q_3)}{m_h + m_t} \sin \theta \]

\[ l_{t2} = q_1 + (\Delta z \tan \theta + q_3) \cos \theta = \frac{m_q}{m_h + m_t} \left[ q_1 + (q_2 + q_3) \cos \theta \right] \]

On the basis of the law of rotation, one may get:

\[ M = I \dot{\omega} \]

where \( I \) is the moment of inertia of the yawing axis and \( \dot{\omega} \) is the yawing angular accelerator of the robotic dolphin. The above equation can be further written as follows:

\[ \dot{\omega} = c_1 [c_4 v^2(t) + c_2 f(v) + c_3 v(t)] (q_1 + (q_2 + q_3) \cos \theta) \sin \theta \sin \theta \]

where \( c_1, c_2, c_3 \) and \( c_4 \) are constants, and \( c_1 = 0.5 \rho C_{h1} A_{t1} m_q \), \( c_2 = m_q (q_2 + q_3) \), \( c_3 = 0.5 \rho C_{d,1} m_h \), \( c_4 = 1 / \left[ I (m_h + m_t) \right] \).

Hence, the real-time heading angle of the robotic dolphin \( \varphi(t) \) is given as: Zhiqiang Cao, Fei Shen, Chao Zhou, Nong Gu, Saeid Nahavandi and De Xu: Heading Control for a Robotic Dolphin Based on a Self-tuning Fuzzy Strategy
where \( \omega_0 \) and \( \varphi_0 \) are the initial yawing angular rate and initial yaw angle of the robotic dolphin, respectively.

One can see from (12) that the robotic dolphin’s heading \( \varphi(t) \) can be controlled by adjusting the turning joint angle \( \theta(t) \) with a swimming velocity \( v(t) \). Moreover, the relationship between \( \varphi(t) \) and \( \theta(t) \) is non-linear. Given the non-linear relationship between the control variable (turning joint angle) and the target variable (the heading of robotic dolphin), as well as the complexity of hydrodynamics, the traditional model-based controllers are not feasible anymore. Since the fuzzy logic control does not require an accurate model, a fuzzy-based approach is presented to solve the heading control problem.

3. Heading Control of a Robotic Dolphin Based on a Self-tuning Fuzzy Strategy

3.1 Heading Control System

Fig. 3 illustrates the developed control system. This system mainly includes a fuzzy controller, a self-tuning mechanism, a servo and an inertial sensor, wherein the sensor is employed to measure the heading \( \varphi \) of the robotic dolphin in real time. The input of the self-tuning fuzzy controller is the heading error, while its output is fed into the servo to change the turning angle \( \theta \) of the turning joint. To mitigate the interference in the environment, the proposed fuzzy controller is designed to produce a fast response and compensate the big error. Nevertheless, when the input is small, the working region is confined to the region of ZE fuzzy set, which causes the controller output to remain almost unchanged. Thus, the steady-state error is produced as the heading range is relatively large and the fuzzy controller’s universe of discourse is fixed. One solution is to further refine the fuzzy membership functions to improve the sensitivity of the controller, but it will inevitably increase the complexity of the controller. Since the input and output scaling factors are respectively used to transfer the controller input and output into the universe of discourse and the actual control variable range, a self-tuning mechanism for tuning the input and output scaling factors is designed to improve the sensitivity of the controller, as well as the steady-state accuracy of the system.

3.2 Design of the Self-tuning Controller

The inputs of the fuzzy controller are heading error \( e(k) \) and differential error \( ec(k) \), which are defined as \( e(k) = \varphi^*(k) - \varphi(k) \) and \( ec(k) = e(k) - e(k-1) \), respectively. \( \varphi^*(k) \) is the target heading, \( \varphi(k) \) is the feedback heading and \( k \) is the sampling index. It should be noted that the output of the inertial sensor \( \varphi_i(k) \) is in the range of \( 0^\circ \sim 360^\circ \). If the value of \( \varphi(k) \) is directly taken from \( \varphi_i(k) \) and it switches between the first quadrant \( 0^\circ \sim 90^\circ \) and the fourth quadrant \( 270^\circ \sim 360^\circ \), the error \( e(k) \) will experience a sudden change. Thus, to ensure \( e(k) \) continuous change in the interval \( (-180^\circ \sim 180^\circ) \), \( \varphi(k) \) should be given as

\[
\varphi(k) = \begin{cases} 
\varphi_i(k) + 360 & \text{if } (0^\circ \leq \varphi_i(k) < 180^\circ) \text{ and } (180^\circ \leq \varphi_i(k) < 360^\circ) \\
\varphi_i(k) - 360 & \text{if } (0^\circ \leq \varphi_i(k) < 180^\circ) \text{ and } (360^\circ \leq \varphi_i(k) < 270^\circ) \\
\varphi_i(k) & \text{if } \varphi_i(k) \in [0^\circ, 180^\circ] \\
\text{others}
\end{cases}
\]

3.2.1 Membership Function Design

Triangular membership functions with a 50% overlap on neighbours are used for \( e \), \( ec \) and \( u \), as shown in Fig. 4. For the input \( e \) and \( ec \) as well as the output \( u \), labelled as E, EC and U, respectively, seven linguistic values are adopted as follows:

\[
\begin{align*}
E &= \{NB, NM, NS, ZE, PS, PM, PB\} \\
EC &= \{NB, NM, NS, ZE, PS, PM, PB\} \\
U &= \{NB, NM, NS, ZE, PS, PM, PB\}
\end{align*}
\]

Figure 3. The heading controller of a robotic dolphin based on a self-tuning fuzzy strategy
where NB, NM, NS, ZE, PS, PM and PB are linguistic values, which denote negative large, negative middle, negative small, zero, positive small, positive middle and positive large, respectively.

### 3.2.2 Fuzzy Rule Base Design

The rule base reflecting the intelligence of the fuzzy control is responsible for specifying the actions that shall be taken under different situations. Taking advantage of the experimental experience of the robotic dolphin’s turning motion, the rule base is obtained as shown in Table 1. Each rule in the rule base has the IF-THEN form. Since position control type is adopted for the fuzzy control, the rule base is established by obeying the following principles:

i. When the heading error is positive (negative) large and the error change is positive (negative) large, it indicates that the heading is deviated largely from the target heading and the heading error is rapidly increasing. Thus, the control variable \( U \) should be positive (negative) large to eliminate the error as soon as possible.

ii. When the heading error is positive (negative) small or zero and the error change is negative (positive) large, it indicates that, although the current heading is close to the target, the heading error still has the increasing trend because it is reduced excessively. Thus, the control variable \( U \) should be negative (positive) small or negative (positive) large, respectively, to prevent overshoot.

#### 3.2.3 Defuzzification

Here, the commonly used centre of gravity defuzzification is adopted:

\[
\hat{u} = \frac{\sum_{i=1}^{m} h_i \mu_i(E_i, EC_i, U_i, e, ec)}{\sum_{i=1}^{m} \mu_i(E_i, EC_i, U_i, e, ec)}
\]

where \( \hat{u} \) is the defuzzification output of the fuzzy controller, \( m \) is the number of the enabled fuzzy rules in the rule base, \( h_i \) is membership function centre of the fuzzy output language variable, and \( \mu_i(E_i, EC_i, U_i, e, ec) \) is the membership value of the \( i \)th fuzzy rule obtained according to fuzzy reasoning, which is calculated by:

\[
\mu_i(E_i, EC_i, U_i, e, ec) = \mu_{e_i}(e) \cap \mu_{EC_i}(ec) \cap \mu_{U_i}(u)
\]

#### 3.2.4 Self-tuning Mechanism

Since the target heading is in a large range (0°~360°), the fuzzy controller input, including heading error and error change, will vary over a large range. For the regular fuzzy controller, the input and output scaling factors are set according to the entire range and remain unchanged in the whole process. When the heading error is large, the fuzzy controller is able to respond quickly to reduce the error. When the heading error is small, however, the working region is confined to the region of the ZE fuzzy set, which will cause a steady-state error. In this case, a self-tuning mechanism is designed to tune the input scaling factors to expand the working region for improving the input resolution [22]. Accordingly, the output of our controller is changed by tuning the output scaling factor to improve the performance of the controller. Based on these ideas, the self-tuning mechanism adopts the piecewise way to realize the

<table>
<thead>
<tr>
<th>EC</th>
<th>U</th>
<th>NB</th>
<th>NM</th>
<th>NS</th>
<th>ZE</th>
<th>PS</th>
<th>PM</th>
<th>PB</th>
</tr>
</thead>
<tbody>
<tr>
<td>NB</td>
<td>NB</td>
<td>NB</td>
<td>NM</td>
<td>NM</td>
<td>NB</td>
<td>NS</td>
<td>PS</td>
<td>PM</td>
</tr>
<tr>
<td>NM</td>
<td>NB</td>
<td>NB</td>
<td>NB</td>
<td>NM</td>
<td>NM</td>
<td>NS</td>
<td>PM</td>
<td>PB</td>
</tr>
<tr>
<td>NS</td>
<td>NB</td>
<td>NM</td>
<td>NS</td>
<td>NS</td>
<td>ZE</td>
<td>PS</td>
<td>PM</td>
<td>PB</td>
</tr>
<tr>
<td>ZE</td>
<td>NB</td>
<td>NB</td>
<td>NM</td>
<td>NS</td>
<td>ZE</td>
<td>PS</td>
<td>PM</td>
<td>PB</td>
</tr>
<tr>
<td>PS</td>
<td>NB</td>
<td>NM</td>
<td>ZE</td>
<td>PS</td>
<td>PM</td>
<td>FB</td>
<td>PB</td>
<td></td>
</tr>
<tr>
<td>PM</td>
<td>NB</td>
<td>NM</td>
<td>PS</td>
<td>PM</td>
<td>PM</td>
<td>PB</td>
<td>PB</td>
<td></td>
</tr>
<tr>
<td>PB</td>
<td>NM</td>
<td>NS</td>
<td>PS</td>
<td>PB</td>
<td>PM</td>
<td>PB</td>
<td>PB</td>
<td></td>
</tr>
</tbody>
</table>

Table 1. Rule base of fuzzy controller
tuning according to the range of the heading error. When the heading error is in a large range, fixed scaling factors are adopted. When the heading error is in a small area, variable scaling factors are designed as follows:

\[ K_e = \begin{cases} \frac{1}{e_{\text{max}}}, & |e| > e_{\text{ih}} \\ \frac{1}{e_{\text{max}}}, & |e| \leq e_{\text{ih}} \end{cases} \]

where \(K_e, K_c\), and \(K_u\) are scaling factors of heading error \(e\), error change \(\dot{e}\) and output \(u\), respectively. \(e_{\text{ih}}\) is the threshold of the heading error. \(e_{\text{max}}, e_{\text{max}}\), and \(u_{\text{max}}\) are respectively the maximum value of heading error, error change and the output. \(e_{\text{max}}\) and \(e_{\text{max}}\) are the maximum value of heading error and error change in the self-tuning period. \(T, \beta\) is the regulating factor for amplifying the output of the controller appropriately.

Using the above scaling factors, the regularized heading error, error change and output, defined in [-1, 1], can be obtained as:

\[ e_s = K_e e, \quad \dot{e}_s = K_c \dot{e}, \quad u_s = K_u u \]  \hspace{1cm} (18)

4. Experimental Results

To verify the rationality of the proposed controller, a series of experiments was conducted in a lake including comparison experiment, anti-interference experiment and heading switching experiment. The parameters of input and output scaling factors and other parameters of the controller are given as: \(e_{\text{max}}=90, e_{\text{max}}=40, u_{\text{max}}=60, e_{\text{ih}}=5, T=1.8s\) and \(\beta=2.5\). In addition, to maintain the propulsion, the robotic dolphin oscillates its tail at a frequency of 2.3Hz and the control period of the system was 0.62s. The static accuracy and dynamic accuracy of the adopted inertial sensor are 0.3° and 0.7°, respectively.

4.1 Heading Control Experiment Based on a Self-tuning Fuzzy Controller

Fig. 5 depicts the experimental results of heading control with an initial heading of 273° and a target heading of 140°, and the curves of heading and heading error, \(K_e, K_c\), and \(K_u\) are demonstrated.

From the heading curve, one can see that the controller responds quickly to reduce the heading error, and the heading approaches the target direction after 7s. The curves of heading error, \(K_e, K_c\), and \(K_u\) are given after convergence. As shown in the curves of \(K_e, K_c\), and \(K_u\) the input and output scaling factors are adjusted in each self-tuning period. The heading error curve proves that our controller achieves an effective heading control.

4.2 Contrast Experiment with Regular Fuzzy Control

To further validate the performance of the self-tuning fuzzy controller, the comparative experiment with a regular fuzzy controller was carried out. Since the input and output scaling factors of a regular fuzzy controller is fixed throughout the control process, the self-tuning fuzzy controller can quickly eliminate large errors. However, a distinguished steady-state error within a range of 3° to 8° exists in the regular fuzzy controller (see Fig. 6(a)). This steady-state error cannot be eliminated due to fixed scaling factors of the controller. In contrast, as shown in Fig. 6(b), a better convergence performance can be obtained by our controller due to the self-tuning mechanism.

4.3 Anti-interference Experiment

Fig. 7 shows the result of an anti-interference experiment with the target heading of 321°. At first, the heading of the robotic dolphin changes rapidly from an initial heading of 144° to the target heading. After 10s, the heading begins to maintain itself in the vicinity of the target heading. A man-made interference is then suddenly imposed on the robotic dolphin, which results in a deviation of about 50° from the
target heading. After that, the heading of the robotic dolphin is re-stabilized around the target heading, which indicates the robustness of the proposed self-tuning fuzzy controller.

4.4 Heading Switching Experiment

The experimental result of heading switching is provided in Fig. 8. In the initial stage, the robotic dolphin swims with a target heading of 330°. After the target heading is reset to 150°, the robotic dolphin responds quickly and steady swimming with the new target heading is achieved after 10s.

5. Conclusion

In this paper, we have developed a fuzzy control method to achieve a robust heading control for a multilink robotic dolphin. The turning motion of a robotic dolphin is analysed. Considering the severe non-linearity in heading control and disturbances due to water waves and turbulence, a self-tuning fuzzy controller is presented to improve the rapid response ability and the steady-state accuracy of the control system. The effectiveness of the proposed method is verified by experiments. In the future, we will conduct hydrodynamics validation based on the deduced turning motion analysis model.

6. Acknowledgements

This work was supported in part by the National Natural Science Foundation of China, under Grants 61421004 and 61473295, and in part by the Beijing Natural Science Foundation, under Grant 4152054. Fei Shen is the corresponding author.

7. References


