Engineering 4 (2018) 861–868

Contents lists available at ScienceDirect

Engineering

journal homepage: www.elsevier.com/locate/eng

Research
Robotics—Article

Fault-Tolerant Control of a CPG-Governed Robotic Fish

Yueqi Yang [a,b], Jian Wang [a,b], Zhengxing Wu [a], Junzhi Yu [a,*]

[a] State Key Laboratory of Management and Control for Complex Systems, Institute of Automation, Chinese Academy of Sciences, Beijing 100049, China
[b] University of the Chinese Academy of Sciences, Beijing 100049, China

ARTICLE INFO

Article history:
Received 7 February 2018
Revised 4 June 2018
Accepted 20 September 2018
Available online 27 September 2018

Keywords:
Fault-tolerant control
Robotic fish
Motion control
Feedback controller
Feedforward compensator

ABSTRACT

Fault tolerance is essential for the maneuverability of self-propelled biomimetic robotic fish in real-world aquatic applications. This paper explores the fault-tolerance control problem of a free-swimming robotic fish with multiple moving joints and a stuck tail joint. The created control system is composed of two main components: a feedback controller and a feedforward compensator. Specifically, the bio-inspired central pattern generator-based feedback controller is designed to make the robotic fish robust to external disturbances, while the feedforward compensator speeds up the convergence of the overall control system. Simulations are performed for control system analysis and performance validation of the faulty robotic fish. The experimental results demonstrate that the proposed fault-tolerant control method is able to effectively regulate the faulty robotic fish, allowing it to complete the desired motion in the presence of damage and thereby improving both the stability and the lifetime of the real robotic system.

1. Introduction

In recent years, robots have been used to perform complex tasks in many places. Applications include space exploration [1], disaster rescue [2], minesweeping [3], and more. Because of the harsh and dangerous work environments, robots inevitably face the risk of damage. Therefore, many researchers have begun to investigate robot fault-tolerant control [4–8]. Kawata et al. [9] proposed a fault-tolerant adaptive gait-generation method for a multi-limbed robot, Mavrovouniotis et al. [10] developed a fault-tolerant sliding mode control, and Zhang et al. [11] conducted research on reconstruction fault-tolerant control for underwater vehicles. Studies related to fault-tolerant control ensure the stability of robot applications.

With the development of autonomous underwater vehicles (AUVs), more and more researchers have focused on enhancing the maneuverability of underwater vehicles. After a long evolutionary period, fish have acquired sufficient flexibility to swim underwater. Inspired by this intriguing trait, Triantafyllou et al. [12] first created the biomimetic robotic fish in 1994. As a new type of AUV, robotic fish attract considerable attention from researchers due to their superior mobility [13–19]. Zhong et al. [20] designed a novel robotic fish with a wire-driven active body, and Romero et al. [21] proposed a cost-effective intelligent robotic fish. Compared with traditional AUVs, robotic fish have advantages such as low noise, harmlessness to underwater organisms, and more [22]. With these benefits, robotic fish have the potential to be widely used to complete complex underwater missions [23].

Research into the fault-tolerant control of AUVs is mainly divided into two categories: ① sensor fault-tolerant control methods such as those discussed in Ref. [24], and ② fault-tolerant control methods for the motion execution of AUVs. Ahmadzadeh et al. [25] proposed a fault-tolerant control method for AUVs in order to overcome thruster failures, and Rauber et al. [26] presented a fault-tolerant control strategy for the thruster of an AUV. Existing studies on underwater fault-tolerant control seldom deal with robotic fish. However, the high maneuverability behavior exhibited by a robotic fish, such as a C-turn [27], places heavy loads on the actuators and increases the risk of damage to the robotic fish. At the same time, if the configuration of the bio-inspired central pattern generator (CPG)—which is widely applied to generate rhythmic movements for a robot with multiple degrees of freedom [28,29]—is not satisfactory, the joints of the robotic fish may become stuck during motion pattern transition. Since the fish body waves need the coordination of all the joints of the robotic fish, damage to any part of it will strongly influence the motion. In addition, since robotic fish operate in underwater environments, once the fault occurs, it is difficult for a robotic fish to be repaired in time. As a type of underwater vehicle, robotic fish must be

* Corresponding author.
E-mail address: junzhi.yu@ia.ac.cn (J. Yu).

https://doi.org/10.1016/j.eng.2018.09.011

autonomous and reliable. The fault-tolerant control ability of the control system is an important indicator to ensure the stable operation of a robotic fish in a faulty condition. Thus, it is necessary to investigate fault-tolerant control for robotic fish.

The main objective of this paper is to propose a fault-torrent control method to deal with the problem of a robotic fish with a stuck tail joint. In order to ensure stable motion performance for a robotic fish, a CPG-governed feedback controller was developed. We conducted dynamic modeling for a fault situation in which one of the tail joints of the robotic fish is stuck [30]. Based on the modeling, we developed feedforward compensation in order to speed up convergence. Together, these two parts form the feedforward-feedback fault-tolerant control system.

Simulations and experiments were performed in order to verify the effectiveness of the proposed control system. Existing studies on underwater fault-tolerant control are mainly aimed at underwater vehicles using a traditional propeller. Unlike previous works, this paper focuses on the problem of motion correction for a faulty robotic fish whose propulsion is generated by a multi-jointed tail. Based on an analysis of the unique motion and dynamic models of the robotic fish, a fault-tolerant control method for a multi-jointed robotic fish is proposed for the first time. Our method improves the robustness of the control system and thus lays a foundation for the practical application of robotic fish in complex environments.

The rest of this paper is organized as follows: The mechanical structure of the robotic fish is introduced in Section 2. Section 3 describes the feedback controller based on the CPG model and the feedforward compensation based on the dynamic model, and then discusses the development of the overall control system. The experiments and corresponding results are described in Section 4. Finally, Section 5 provides conclusions and looks forward to future work.

2. Overview of the multi-jointed robotic fish

The controlled object used in this study is a self-propelled multi-jointed robotic fish [31]. Fig. 1 shows the mechanical structure of the robotic fish, which follows the structure of *Esox lucius*. The robotic fish has a self-propulsive body with a four-joint tail and a caudal fin. The four tail joints of the robotic fish are driven by servomotors. In addition, a rigid head with two movable pectoral fins is designed to meet the requirements of three-dimensional motion. In this paper, we only discuss the tolerance of a robotic fish moving in a two-dimensional plane, so the pectoral fins are always kept still. The controller of the robotic fish is embedded in its head. With the coordination of the servomotors, the robotic fish has a high peak steering speed of 670° per second and an average steering speed of 460° per second. If the tail joint of the robotic fish gets stuck, the motion direction of the robotic fish deviates severely. To determine the deviation, an inertial measurement unit (IMU) is installed in the head of the robotic fish to measure the yaw angle.

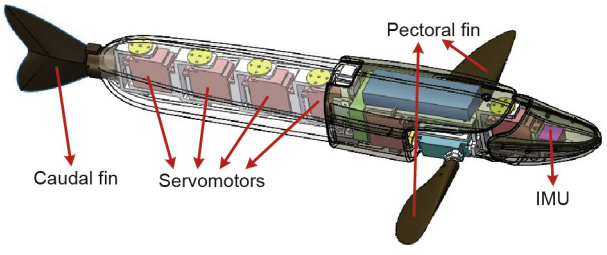


Fig. 1. Mechanical structure of the self-propelled multi-jointed robotic fish.

Fig. 2 shows the appearance of the robotic fish. The whole robotic fish is covered by a waterproof skin made of an emulsion. Above the skin of the head, red and yellow color markers made of tape are placed on the outside of the rigid shell in order to help the measuring device track the head position and the posture of the robotic fish during its movements.

3. Fault-tolerant controller design

This section proposes a fault-tolerant control method for the robotic fish. Specifically, a feedback controller based on CPG and a feedforward compensator based on a mathematical model are designed.

3.1. Feedback controller based on CPG

In general, there are two types of motion control methods for robotic fish: fish body waves curve fitting and the CPG-based method. The latter method was adopted for the robotic fish in this paper. Vertebrates can generate rhythmic signals without a central nerve through biological CPG networks. Inspired by this, the CPG model can generate control signals through the interaction of neuron oscillators in order to control the motion of a robotic fish [32]. The CPG model has the advantages of strong robustness and smoothness in motion mode switching.

The control method used in this paper is a Hopf oscillator-based CPG model with phase differences [33]. The CPG model consists of multiple neuron oscillators, as shown in Fig. 3.

The details of the CPG model are given below [31]:

$$\begin{cases} \dot{x}_n = -w_n(y_n - b_n) + x_n\left[r_n^2 - x_n^2 - (y_n - b_n)^2\right] \\ \qquad +h_1\left[x_{n-1}\cos\varphi_n + (y_{n-1} - b_{n-1})\sin\varphi_n\right] \\ \dot{y}_n = w_n x_n + (y_n - b_n)\left[r_n^2 - x_n^2 - (y_n - b_n)^2\right] \\ \qquad +h_2\left[x_{n+1}\sin\varphi_i + (y_{n+1} - b_{n+1})\cos\varphi_n\right] \\ z_n = c_n y_n \end{cases} \tag{1}$$

where n represents the nth oscillator $(n = 1, \ldots, N)$ and N denotes the number of the neurons in the CPG network; x_n and y_n represent the state variables of the nth oscillator, and $\dot{x}_n$ and $\dot{y}_n$ represent the

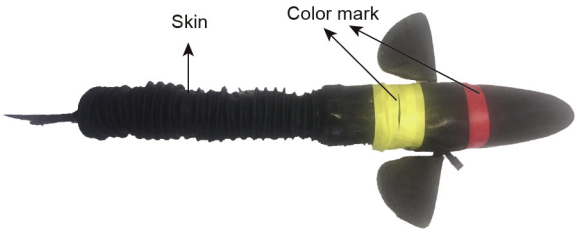


Fig. 2. Appearance of the robotic fish with color markings.

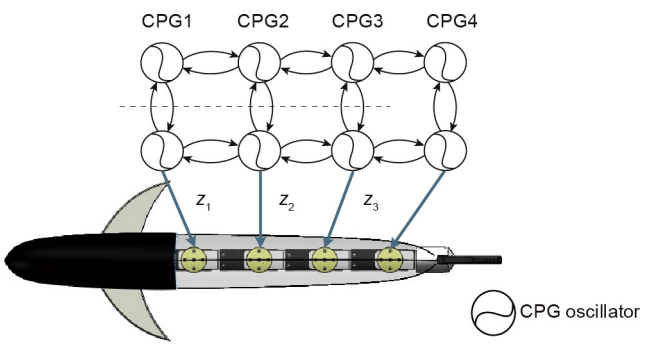


Fig. 3. Topological structure of the CPG model.

derivative of x_n and y_n, respectively; w_n and r_n denote the natural frequency and amplitude of the nth oscillator, respectively; φ_n represents the phase difference between adjacent neurons; b_n denotes the deflection factor of the nth oscillator; h_1 and h_2 represent the coupling coefficients; z_n is the output signal of the nth oscillator; and c_n is a constant coefficient.

Through appropriate adjustment of the CPG parameters, the motions of the robotic fish can be controlled. The swimming speed can be regulated by changing w_n and r_n, and backward swimming can be realized by setting φ_n. $w_n = w$ and $r_n = r$ were employed for all the oscillators in this paper, and φ was set appropriately according to Ref. [33]. When b_i is adjusted, the CPG model generates an asymmetric bias output signal to change the swimming direction of the robotic fish. As mentioned previously, if the tail joint gets stuck, this fault will affect the yaw of the robotic fish. Therefore, it is feasible to control b_i using the yaw angle as a feedback, to ensure that the robotic fish can maintain good motion performance in the event of faults. As the head of the robotic fish always swings when swimming because of its unique motion method, it is necessary for the determined yaw angle to be properly filtered. Based on the above analysis, a proportional integral (PI) feedback controller was designed as follows:

$$\begin{cases} b_i(t) = K_P^i\, e(t) + K_I^i \int_0^t e(u)\mathrm{d}u \\ e(t) = \begin{cases} \hat{\gamma} - \mathrm{f}\gamma(t), & |\hat{\gamma} - \mathrm{f}\gamma(t)| \geqslant T \\ 0, & |\hat{\gamma} - \mathrm{f}\gamma(t)| < T \end{cases} \end{cases} \tag{2}$$

where t denotes the moment, $b_i(t)$ denotes the bias of the ith oscillator of the robotic fish at time t, $\mathrm{f}\gamma(t)$ represents the filtering result of the yaw angle at time t, $\hat{\gamma}$ is the target yaw, $e(t)$ denotes the PI controller input at time t, T represents the threshold and is set as $T = 0.17$, and K_P^i and K_I^i are the feedback gains of the ith joint, K_P^i is the proportional controller coefficient and K_I^i is the integral controller coefficient.

The swing of the head of the robotic fish is caused by the periodic signal output of the CPG. The swing period is the same as that of the CPG oscillator. Therefore, a mean filter with a threshold is designed to deal with the effect of the yaw swing. The window time of the mean filter is set to the period of the CPG oscillator. When the parameters of the CPG model change, it will take some time for its outputs to reach a steady state. Therefore, the frequency of the change control will affect the stability of the system. In order to improve the stability of the system and reduce the burden on the servo, a threshold near the target yaw angle is set. A corresponding block diagram of the feedback control system is illustrated in Fig. 4.

In our previous work [30], a data-driven dynamic model was built for the robotic fish in that study. Using the same model as the real robotic fish, we can obtain the output of the feedback control through a simulation, as shown in Fig. 5. In the simulation in the current paper, the third joint of the tail of the robotic fish is set to be stuck at $\pi/6$, and the target yaw is set to 0. Fig. 5(a) shows the straight swimming behavior of the robotic fish in the event of a stuck tail joint. When the fault occurs, the swimming direction of the robotic fish is greatly affected. The feedback control system can effectively reduce the impact of the stuck joint on the yaw, and eventually causes the output yaw angle to converge to the target yaw angle, as shown in Fig. 5(b).

3.2. Feedforward compensator based on dynamic analysis

The feedback controller proposed in Section 3.1 still has many problems, including a long convergence time, which is due to the delay caused by the filter. To tackle this problem, this section proposes a feedforward compensator based on the dynamic model.

According to Ref. [32], the dynamic model of the robotic fish is specified as follows:

$$\begin{cases} \dot{\boldsymbol{V}}_0 = -M^{-1}\boldsymbol{\Gamma} \\ \boldsymbol{\Gamma} = \sum_{i=0}^{n} {}^{i}H_0^{\mathrm{T}} f_{\mathrm{w},i} + f_{\mathrm{r}} \end{cases} \tag{3}$$

where $\dot{\boldsymbol{V}}_0$ is the velocity vector of the robotic fish head. M represents the whole inertial matrix of the robotic fish in the head coordinate system, $\boldsymbol{\Gamma}$ is the (6×1) force vector of the robotic fish head,

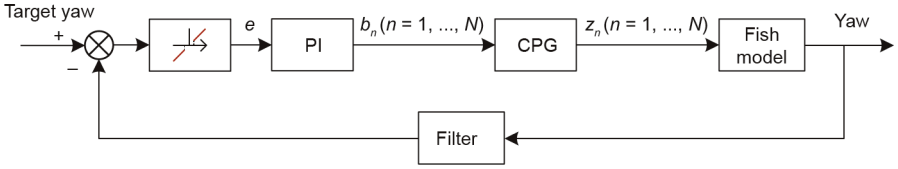


Fig. 4. Block diagram of the feedback control system.

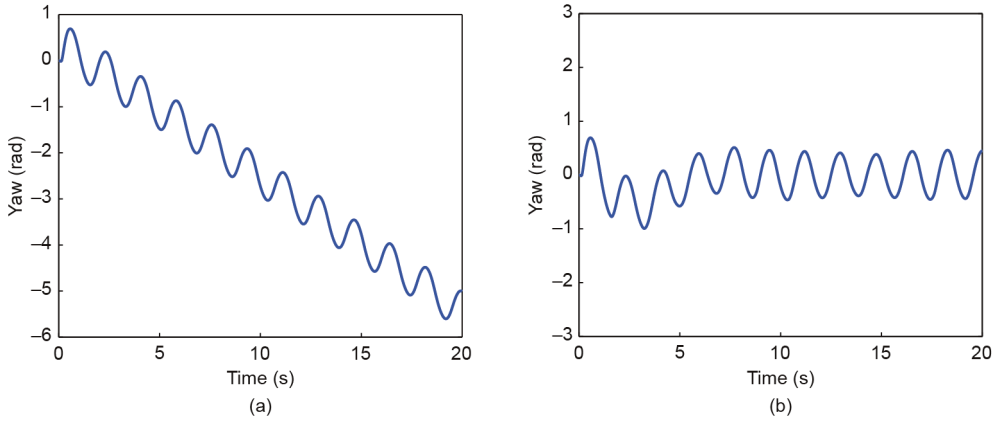


Fig. 5. Simulation of the system output when one of the tail joints is stuck. (a) System without control; (b) system with feedback controller.

${}^{i}H_0 = {}^{i}H_{i-1}\,{}^{i-1}H_{i-2}\ldots{}^{1}H_0$ and ${}^{i}H_{i-1}$ denote the transformation matrix, M_i denotes the inertial matrix of the ith joint, $f_{w,i}$ denotes the Coriolis force and hydrodynamic force on the ith link, and f_r is the remaining part of the head force.

Since the driving force of the robotic fish swimming in the water is mainly generated by the hydrodynamic force, the influence of f_r can be ignored in calculations. In this study, the robotic fish swims in a two-dimensional plane, so it is feasible to reduce the yaw effect caused by the fault by controlling the torque of the head. The sixth component of $\boldsymbol{\Gamma}$ represents the torque of the head. Therefore, the sixth line of $\boldsymbol{\Gamma}$ is extracted as follows:

$$\boldsymbol{\Gamma}_6 = \sum_{i=0}^{n} ({}^{i}H_0^{\mathrm{T}})_{6}\, f_{w,i} \tag{4}$$

where $(\cdot)_{k:}$ represents the kth line of matrix $\cdot$ and $\boldsymbol{\Gamma}_6$ denotes the sixth component of $\boldsymbol{\Gamma}$.

For $({}^{i}H_0^{\mathrm{T}})_{6:}$, it can be calculated from the following:

$$\begin{cases} {}^{i}R_{i-1} = \begin{pmatrix} \cos\theta_i & \sin\theta_i & 0 \\ -\sin\theta_i & \cos\theta_i & 0 \\ 0 & 0 & 1 \end{pmatrix} \\ P_{i-1} = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & -l_{i-1} \\ 0 & l_{i-1} & 0 \end{pmatrix} \\ {}^{i}H_{i-1} = \begin{pmatrix} {}^{i}R_{i-1} & -{}^{i}R_{i-1}\hat{P}_{i-1} \\ 0_{3\times3} & {}^{i}R_{i-1} \end{pmatrix} \end{cases} \tag{5}$$

where θ_i represents the angle of the ith joint and l_i represents the length of the ith link.

In a more explicit form, the following equations are introduced:

$$\begin{cases} ({}^{i}H_0^{\mathrm{T}})_{6:} = \left(\sum_{j=0}^{i-1} l_j\, {}^{i}S_j \quad \sum_{j=0}^{i-1} l_j\, {}^{i}C_j \quad 0 \quad 0 \quad 0 \quad 1 \right) \\ {}^{i}S_j = \sin\left(\sum_{k=j+1}^{i} \theta_k \right) \\ {}^{i}C_j = \cos\left(\sum_{k=j+1}^{i} \theta_k \right) \end{cases} \tag{6}$$

$f_{w,i}$ can be derived according to the following equation:

$$f_{w,i} = \sigma_i - f_{d,i} \tag{7}$$

where σ_i represents the total Coriolis force received by the ith link, and $f_{d,i}$ denotes the resistance of the ith link.

According to Ref. [30], the following equation can be obtained:

$$\begin{cases} \sigma_i = \begin{pmatrix} \Omega_i & 0_{3\times3} \\ 0_{3\times3} & \Omega_i \end{pmatrix} M_i \boldsymbol{V}_i \\ \Omega_i = \begin{pmatrix} 0 & -\omega_i & 0 \\ \omega_i & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} \\ f_{d,i} = \begin{pmatrix} c_{1,i}|V_{x,i}|V_{x,i} \\ c_{2,i}\int_0^{l_i} |V_{y,i} + \omega r|(V_{y,i} + \omega_i r)\mathrm{d}r \\ 0_{3\times1} \\ c_{2,i}\int_0^{l_i} r|V_{y,i} + \omega r|(V_{y,i} + \omega_i r)\mathrm{d}r \end{pmatrix} \end{cases} \tag{8}$$

where $\boldsymbol{V}_i$ is a (6×1) vector representing the velocity of the ith joint. Specifically, the first three dimensions of $\boldsymbol{V}_i$ form the linear velocity vectors, the last three dimensions form the angular velocity vectors. $\omega_i = (V_i)_{6:}$, $V_{x,i} = (V_i)_{1:}$, and $V_{y,i} = (V_i)_{2:}$; $c_{1,i}$ and $c_{2,i}$ are constants; M_i denotes the mass matrix of the ith link; and r is a variable.

By simplifying M_i to a diagonal matrix, we obtain the following:

$$f_{w,i} = -\begin{pmatrix} c_{1,i}|V_{x,i}|V_{x,i} + M_{i,2}\omega_i V_{y,i} \\ c_{2,i}\int_0^{l_i} |V_{y,i} + \omega r|(V_{y,i} + \omega_i r)\mathrm{d}r - M_{i,1}\omega_i V_{x,i} \\ 0_{3\times1} \\ c_{2,i}\int_0^{l_i} r|V_{y,i} + \omega r|(V_{y,i} + \omega_i r)\mathrm{d}r \end{pmatrix}^{\mathrm{T}} \tag{9}$$

where $M_{i,1}$ and $M_{i,2}$ are constants.

In order to obtain a concise form, we introduce the following denotations:

$$\begin{cases} F_{x,i} = -c_{1,i}|V_{x,i}|V_{x,i} - M_{i,2}\omega_i V_{y,i} \\ F_{y,i} = -c_{2,i}\int_0^{l_i} |V_{y,i} + \omega r|(V_{y,i} + \omega_i r)\mathrm{d}r + M_{i,1}\omega_i V_{x,i} \\ F_{\omega,i} = -c_{2,i}\int_0^{l_i} r|V_{y,i} + \omega r|(V_{y,i} + \omega_i r)\mathrm{d}r \end{cases} \tag{10}$$

It then follows that

$$\boldsymbol{\Gamma}_6 = \sum_{i=0}^{n}\left(\sum_{j=0}^{i-1} l_j\, {}^{i}S_j \right) F_{x,i} + \sum_{i=0}^{n}\left(\sum_{j=0}^{i-1} l_j\, {}^{i}C_j \right) F_{y,i} + F_{\omega,i} \tag{11}$$

$\boldsymbol{V}_i$ can be calculated by $\boldsymbol{V}_0$ from $\boldsymbol{V}_i = {}^{i}H_{i-1}\boldsymbol{V}_i + \dot{\theta}_i[0_{5\times1}, 1]^{\mathrm{T}}$. The value of $\boldsymbol{\Gamma}_6$ at each moment can be obtained iteratively by setting the initial value of V_0 to $0_{6\times1}$.

A sinusoidal wave approximation can be used for the signal produced by the CPG model. For a situation in which one of the joints is stuck, the wave approximation can be expressed as follows:

$$\begin{cases} \theta_i = r\sin(wt + \varphi_i) + b_0, \quad i \neq f \\ \theta_f = \beta \end{cases} \tag{12}$$

where b_0 is the compensation bias, $f \in \{1, \ldots, n\}$ marks the location of the stuck joint, and β denotes the angle of the faulty joint.

By substituting Eq. (12) into Eq. (11), $\boldsymbol{\Gamma}_6$ can be expressed as $\boldsymbol{\Gamma}_6(t,\ \beta,\ b_0,\ f)$.

As previously analyzed, in order to minimize the impact of the yaw caused by the fault, a feedforward compensator was designed as follows:

$$b_c = \arg\min_{b_0}\left\{ \left| \int_{t=0}^{T}\int_{u=0}^{t} \boldsymbol{\Gamma}_6(u,\ \beta,\ b_0,\ f)\mathrm{d}u\mathrm{d}t \right| \right\} \tag{13}$$

where $T = 2\pi/w$ is the period and u is variable. Once the range of b is limited, the value of b_c can be obtained through optimization.

The complete controller is proposed as follows:

$$b_i(t) = b_c + K_P^i e(t) + K_I^i \int_0^t e(u)\mathrm{d}u \tag{14}$$

A block diagram of the fault-tolerant control system is provided in Fig. 6, and Fig. 7(a) shows the simulated output of the system with a feedforward compensator. The feedforward compensator can adequately correct the effect of the fault. Fig. 7(b) shows the simulated output of a system with a feedforward-feedback controller, which permits eventual convergence to the target yaw with a small overshot and short convergence time when the fault occurs. The system with a feedforward-feedback controller has a shorter convergence time than the system with a feedback controller.

4. Experiments and results

To verify the effectiveness of the fault-tolerant control of the robotic fish, a motion measurement system with a global vision camera was employed [34]. The global vision camera was set 190 cm above the center of the experimental setup, which was 500 cm long, 400 cm wide, and 120 cm deep. Based on the image data returned by the camera, the host computer of the recording system provides the current position and velocity by identifying the colored marks on the head of the robotic fish.

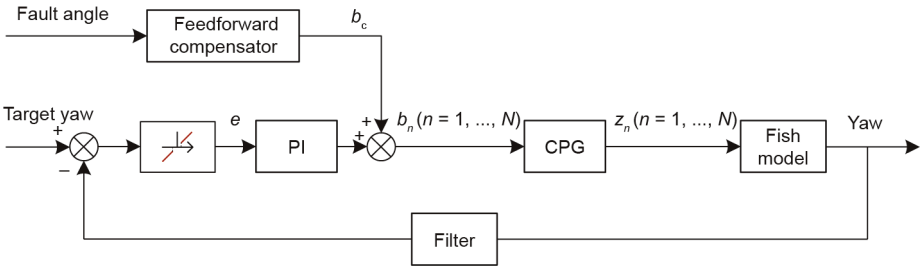


Fig. 6. Block diagram of the feedforward-feedback fault-tolerant control system.

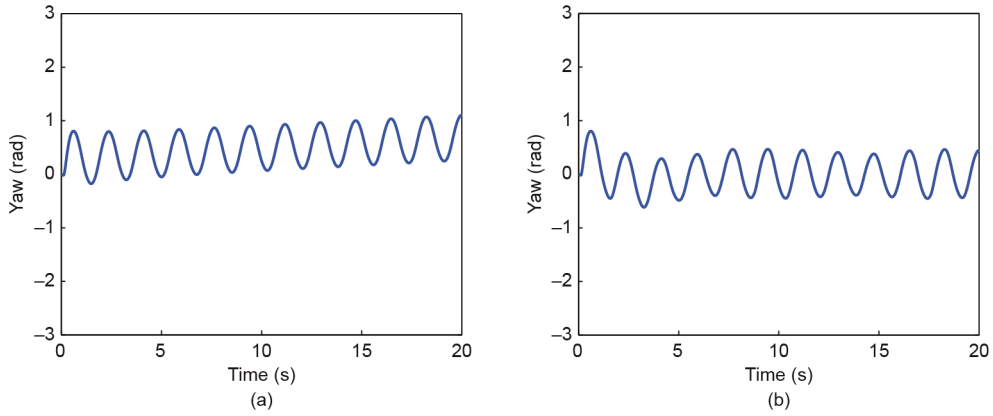


Fig. 7. Simulated output of the system with a feedforward compensator in a situation with a stuck tail joint. (a) System with feedforward compensator; (b) system with feedforward-feedback controller.

During this experiment, the third tail joint of the robotic fish was fixed at $\pi/9$. Fig. 8 shows the straight swimming performance of the robotic fish without fault-tolerant control. When one of the tail joints of the robotic fish was stuck, the swimming direction of the robotic fish was extremely affected; it was difficult for the robotic fish to swim normally.

Fig. 9 shows the swimming performance of the robotic fish with the feedback controller. As shown in Fig. 9(a), the feedback controller gave the robotic fish a satisfactory motion performance in the faulty condition. The filtered yaw error during the motion is shown in Fig. 9(b). It can be seen that the system was able to stably converge to the target yaw angle. However, the system still had the disadvantage of a long convergence time. The robotic fish took nearly 7 s to adjust the yaw to the threshold. A lateral displacement of 53.4 cm occurred during the posture adjustment. The main reason for this displacement was that the filtering window for the yaw angle must be set in order to obtain the correct motion direction, by at least one period of CPG, which results in a large delay to the system and which extends the control period.

The motion performance of the robotic fish with the feedforward-feedback controller is shown in Fig. 10. The parameters of the feedforward compensator were set according to Ref. [19]. As shown in Fig. 10(a), the feedforward compensator gave the robotic fish a smoother motion curve than the simulation with the single feedback controller. In this case, the lateral displacement of the robotic fish was limited to 24.7 cm. Fig. 10(b) shows the filtered yaw error during the motion of the robotic fish. Due to the influence of the feedforward compensator, the robotic fish was able to quickly adapt to the fault when it occurred, and the convergence time of the system was reduced to approximately 1 s. Compared with the system with the feedback controller alone, the system with the feedforward compensator had better fault tolerance. Fig. 11 provides a sequence of screenshots from a video of the robotic fish swimming in the faulty condition.

This experiment demonstrates that the swimming performance of the robotic fish with fault-tolerant control is satisfactory when the third joint is damaged. To further validate this fault-tolerant control method for a situation in which a different

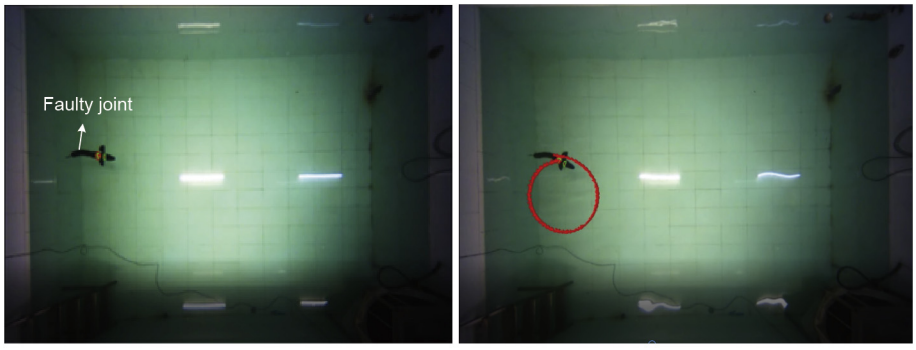


Fig. 8. Motion of the faulty robotic fish without fault-tolerant control. (a) Initial posture; (b) swimming path.

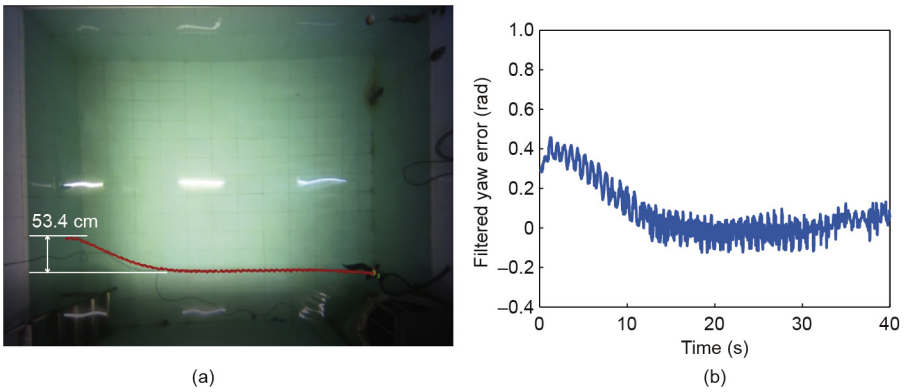


Fig. 9. Motion performance of the faulty robotic fish with the feedback controller. (a) Swimming path; (b) filtered yaw error.

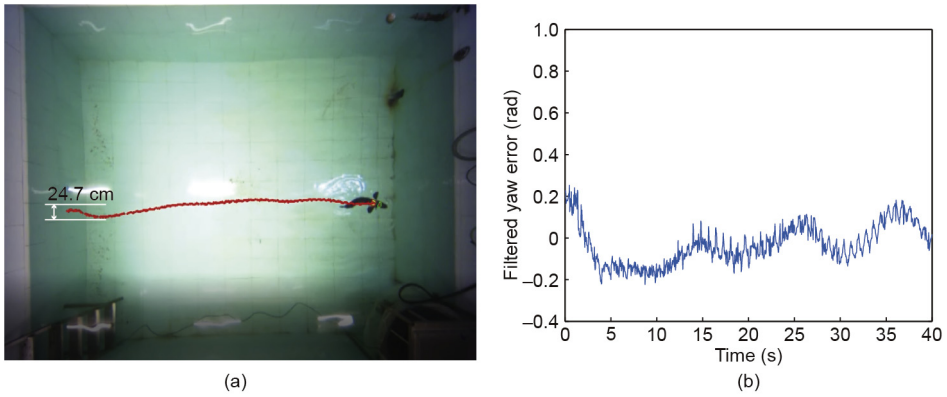


Fig. 10. Motion performance of the faulty robotic fish with the feedforward-feedback controller. (a) Swimming path; (b) filtered yaw error.

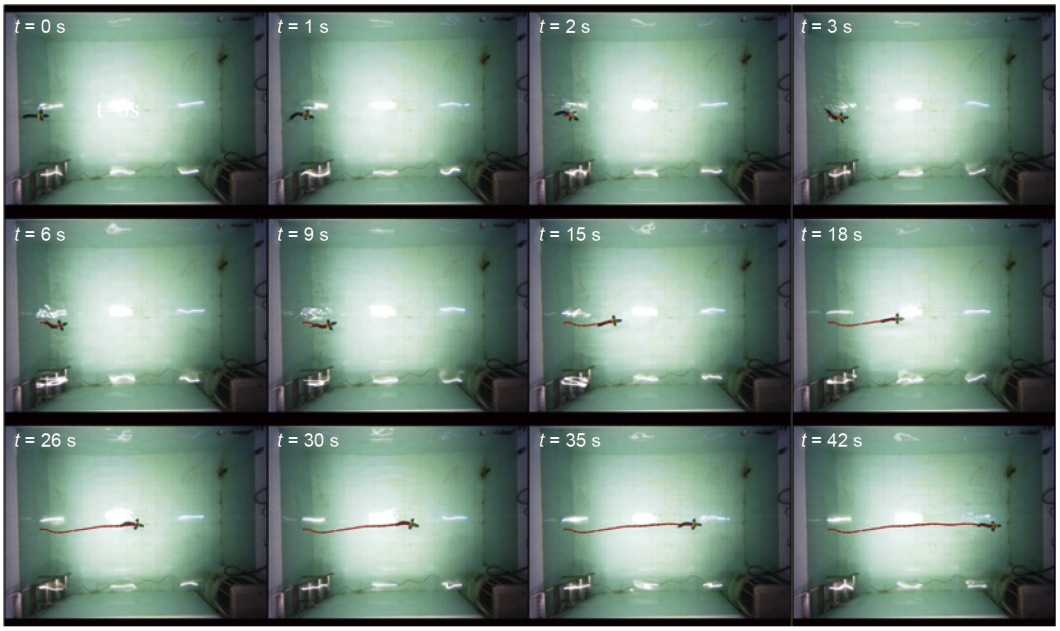
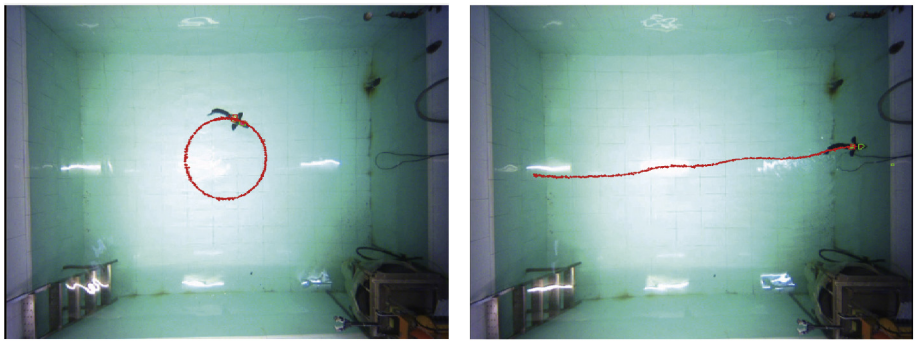


Fig. 11. A sequence of screenshots from a video of the robotic fish swimming with a stuck tail joint.

joint fails, we carried out an experiment in which the second joint was set as the faulty joint. The corresponding comparative experiment is illustrated in Fig. 12. As can be observed, when the fault location changed, the fault-tolerant control method was still able to ensure that the robotic fish achieved good swimming performance. To be specific, the closer the fault location is to the caudal fin, the more severe the loss of the swimming speed of the robotic fish is. This is due to the fact that the wetted surface of the caudal fin is the main source of hydrodynamic force. Also, the closer the fault is to the head, the more serious the effect of the fault is on the swimming direction of the robotic fish.



(a) (b)

Fig. 12. A comparison experiment in which the second joint was set as the faulty joint. (a) Without control; (b) fault-tolerant control.

The experimental results show that the fault-tolerant control method proposed in this paper effectively improves the performance of the robotic fish in a faulty condition. The deflection in the direction of motion that is caused by the stuck tail joint can be effectively counteracted by the feedforward-feedback controller. With the feedback controller, the control system eventually converges to the target yaw with almost no static error. Through the comparative experiment, it was found that the feedforward compensator solves the delay problem of the system. The prior knowledge obtained through the dynamic analysis allows the other joints of the robotic fish to rapidly adapt to the impact of the stuck joint. As the convergence time is shortened, the lateral displacement produced by the posture adjustment is reduced. Considering that the head of the robotic fish swings on the plane of the yaw angle, the swimming performance in this state indicates that the proposed fault-tolerant control method has some stability and anti-interference ability.

5. Conclusion and future work

In this paper, we proposed a fault-tolerant control method for a self-propelled multi-jointed robotic fish with a stuck tail joint. Based on the CPG model, the feedback controller was designed by analyzing the influence of the fault. In order to obtain better performance, a feedforward compensator was designed based on dynamic analysis. More specifically, controlling the specific parameters of the CPG makes the robotic fish robust to faults. The performance of each part of the fault-tolerant control system was analyzed using the simulation. Finally, the validity of the fault-tolerant control algorithm in the real world was verified experimentally. The control method proposed in this paper is able to effectively control the motion of a faulty robotic fish. Hence, the fault-tolerant control of a robotic fish based on the CPG and on a mathematical model has been accomplished.

Our future work will focus on improving the anti-interference ability of these fault-tolerant control methods. In addition, the experiments showed that the speed of the faulty robotic fish was greatly affected; thus, it will be necessary to seek an alternative method to enhance the swimming speed tolerance under faulty conditions. We will also concentrate on adapting the fault-tolerant control method to other faulty conditions, such as joint power failure.

Acknowledgements

This work was supported by the National Natural Science Foundation of China (61725305, 61633020, 61633004, and 61633017), the Beijing Natural Science Foundation (4161002), and the Beijing Advanced Innovation Center for Intelligent Robots and Systems (2016IRS02).

Compliance with ethics guidelines

Yueqi Yang, Jian Wang, Zhengxing Wu, and Junzhi Yu declare that they have no conflict of interest or financial conflicts to disclose.

References

[1] Luo Z, Shang J, Zhang Z. Innovative design of six wheeled space exploration robot using module combination. In: Proceedings of the 19th International Conference on Mechatronics and Machine Vision in Practice; 2012 Nov 28–30; Auckland, New Zealand. New York: IEEE; 2012. p. 460–5.
[2] Chen K, Kamezaki M, Katano T. Fundamental development of a virtual reality simulator for four-arm disaster rescue robot OCTOPUS. In: Proceedings of 2016 IEEE International Conference on Advanced Intelligent Mechatronics; 2016 July 12–15; Banff, AB, Canada. Canada: IEEE; 2016. p. 721–6.
[3] Abbaspour R. Design and implementation of multi-sensor based autonomous minesweeping robot. In: Proceedings of 2010 International Congress on Ultra Modern Telecommunications and Control Systems and Workshops; 2010 Oct 18–20; Moscow, Russia. New York: IEEE; 2010. p. 443–9.
[4] Koh M, Norton M, Khoo S. Robust fault-tolerant leader-follower control of four-wheel steering mobile robots using terminal sliding mode. Austra J Electri Electro Eng 2012;9(3):247–53.
[5] Suzuki H, Asano M, Hamaya A, Onozawa T. Space demonstration of a fault tolerant computer system using commercial MPU. Space Technol 2004;24 (1):35–41.
[6] Wang J, Cao J, Jiang S. Fault-tolerant pattern formation by multiple robots: a learning approach. In: Proceedings of 2017 IEEE 36th Symposium on Reliable Distributed Systems; 2017 Sep 26–29; Hong Kong, China. New York: IEEE; 2017. p. 268–9.
[7] Mi Y, Xu F, Tan J, Wang X, Liang B. Fault-tolerant control of a 2-DOF robot manipulator using multi-sensor switching strategy. In: Proceedings of the 36th Chinese Control Conference; 2010 Oct 18–20; Dalian, China. New York: IEEE; 2017. p. 7307–14.
[8] Ben-Gharbia KM, Maciejewski AA, Roberts RG. A kinematic analysis and evaluation of planar robots designed from optimally fault-tolerant Jacobians. IEEE Trans Robot 2014;30(2):516–24.
[9] Kawata T, Kamiyama K, Kojima M, Horade M, Mae Y. Fault-tolerant adaptive gait generation for multi-limbed robot. In: Proceedings of 2016 IEEE/RSJ International Conference on Intelligent Robots and Systems; 2016 Oct 9–14; Daejeon, Korea. New York: IEEE; 2016. p. 3381–6.
[10] Mavrovouniotis M, Van M, Ge SS. An adaptive backstepping nonsingular fast terminal sliding mode control for robust fault tolerant control of robot manipulators. IEEE Trans Syst Man Cybern Syst 2018 Jan 17:1–11.
[11] Zhang Y, Zeng J, Li Y, Sun Y. Research on reconstructive fault-tolerant control of an X-rudder AUV. In: Proceedings of the OCEANS 2016 MTS/IEEE; 2016 Sep 19–23; Monterey, CA, USA. New York: IEEE; 2016. p. 1–5.
[12] Triantafyllou MS, Triantafyllou GS. An efficient swimming machine. Sci Am 1995;272(3):64–70.
[13] Yu J, Chen S, Wu Z, Wang W. On a miniature free-swimming robotic fish with multiple sensors. Int J Adv Robot Syst 2016;13:62.
[14] Yu J, Su Z, Wang M, Tan M, Zhang J. Control of yaw and pitch maneuvers of a multilink dolphin robot. IEEE Trans Robot 2012;28(2):318–29.
[15] Yu J, Tan M, Wang S, Chen E. Development of a biomimetic robotic fish and its control algorithm. IEEE Trans Syst Man Cybern B Cybern 2004;34(4): 1798–810.

[16] Liang J, Wang T, Wen L. Development of a two-joint robotic fish for real-world exploration. J Field Robot 2011;28(1):70–9.
[17] Zhou C, Low KH. Design and locomotion control of a biomimetic underwater vehicle with fin propulsion. IEEE/ASME Trans Mechatron 2012;17(1):25–35.
[18] Wang M, Yu J, Tan M. CPG-based sensory feedback control for bio-inspired multimodal swimming. Int J Adv Robot Syst 2014;11(10):70.
[19] Chen S, Yu J, Li X, Yuan J. Design and implementation of a smart robotic shark with multi-sensors. In: Proceedings of the 18th International Conference on CLAWAR 2015; 2015 Sep 6–9; Hangzhou, China; 2015. p. 199–206.
[20] Zhong Y, Li Z, Du R. A novel robot fish with wire-driven active body and compliant tail. IEEE/ASME Trans Mechatron 2017;22(4):1633–43.
[21] Romero P, Sensale-Rodriguez B, Astessiano D, Canetti R. Fisho: a cost-effective intelligent autonomous robot fish. In: Proceedings of 2013 16th International Conference on Advanced Robotics; 2013 Nov 25–29; Montevideo, Uruguay. New York: IEEE; 2014. p. 1–6.
[22] Yu J, Wang L, Shao J, Tan M. Control and coordination of multiple biomimetic robotic fish. IEEE Trans Contr Syst Technol 2007;15(1):176–83.
[23] Zhou C. Research on modeling, control and cooperation of a marsupial biomimetic robotic fish [dissertation]. Beijing: Institute of Automation, Chinese Academy of Sciences; 2008. Chinese.
[24] Liang X, Zhang J, Li W. Sensor fault tolerant control for AUVs based on replace control. Sensors Transducers 2013;158(11):408–13.
[25] Ahmadzadeh SR, Leonetti M, Carrera A, Carreras M, Kormushev P, Caldwell DG. Online discovery of AUV control policies to overcome thruster failures. In: Proceedings of IEEE International Conference on Robotics and Automation; 2014 May 31–June 7; Hong Kong, China. New York: IEEE; 2014. p. 6522–8.
[26] Rauber JG, Santos CHFD, Chiella ACB, Motta LRH. A strategy for thruster fault-tolerant control applied to an AUV. In: Proceedings of 2012 17th International Conference on Methods and Models in Automation and Robotics; 2012 Aug 27–30; Miedzyzdrojie, Poland. New York: IEEE; 2012. p. 184–9.
[27] Su Z, Yu J, Tan M, Zhang J. Implementing flexible and fast turning maneuvers of a multijoint robotic fish. IEEE/ASME Trans Mechatron 2014;19(1):329–38.
[28] Yu J, Tan M, Chen J, Zhang J. A survey on CPG-inspired control models and system implementation. IEEE Trans Neural Netw Learn Syst 2014;25(3):441–56.
[29] Liu C, Wang D, Chen Q. Central pattern generator inspired control for adaptive walking of biped robots. IEEE Trans Syst Man Cybern Syst 2013;43(5):1206–15.
[30] Yu J, Yuan J, Wu Z, Tan M. Data-driven dynamic modeling for a swimming robotic fish. IEEE Trans Ind Electron 2016;63(9):5632–40.
[31] Wu Z, Yu J, Su Z, Tan M, Li Z. Towards an *Esox lucius* inspired multimodal robotic fish. Sci China Inf Sci 2015;58(5):052203.
[32] Wu Z. Three-dimensional maneuvering locomotion and gliding control for the robotic fish [dissertation]. Beijing: University of Chinese Academy of Sciences; 2015. Chinese.
[33] Yu J, Wu Z, Wang M, Tan M. CPG network optimization for a biomimetic robotic fish via PSO. IEEE Trans Neural Netw Learn Syst 2016;27(9):1962–8.
[34] Yuan J, Yu J, Wu Z, Tan M. Precise planar motion measurement of a swimming multi-joint robotic fish. Sci China Inf Sci 2016;59(9):092208.