

Backstepping Control-based Trajectory Tracking for Tail-actuated Robotic Fish

Maria L. Castaño, and Xiaobo Tan

Abstract—In this work, we propose a backstepping-based trajectory tracking control approach for a tail-actuated robotic fish, which has highly nonlinear and under-actuated dynamics. A modified heading error is introduced that augments the heading error with a term dependent on lateral tracking error, which enables the exploitation of dynamics coupling to address the under-actuation challenge and stabilize both the heading and lateral tracking errors. Furthermore, the input constraints imposed by physical limitations are accounted for with the incorporation of an auxiliary system. Simulation results support the efficacy of the proposed control approach and its advantages are shown via comparison with a PI controller.

I. INTRODUCTION

Due to their high maneuverability and simple mechanical design, robotic fish are gaining interest in a number of underwater sensing applications. While extensive work has been reported on motion control of robotic fish, it has mainly been focused on the generation of coordinated movements of the actuation components to produce some fish-like swimming gaits [1]–[9]. Some works have addressed trajectory tracking and stabilization problems [10], [11], but mostly with an emphasis on heading or depth control. On the other hand, some limited work has been done on model-based closed-loop motion control to achieve maneuvering, speed and orientation control, path following or point-to-point tracking [12]–[18]. Our recent work addressed the path-following control problem for a tail-actuated robotic fish [19], where a nonlinear model predictive control (NMPC) scheme was proposed. However, the computational complexity of NMPC poses great challenges in implementing such controllers on resource-constrained robotic fish.

Backstepping-based control design presents a practical, promising, and systematic approach for trajectory tracking with stability guarantees. In particular, it is computationally inexpensive, especially when compared to methods such as NMPC. Some limited work has been reported on backstepping-based control of robotic fish [20], [21]. In [20], the authors proposed a target-tracking hybrid controller that consists of an open-loop turning controller and a closed-loop backstepping controller, to drive a robotic fish to a specified target location. In [21], the authors proposed a backstepping controller for tracking control of a nonholonomic fish robot with dynamics That can be expressed in a chained form. The proposed approach guaranteed asymptotic convergence

to a desired trajectory generated by a reference fish robot, but limitations were placed on the desired trajectory since it had to be generated by dynamics in the same chained form. Furthermore, input constraints were not accommodated.

In this work, a backstepping-based control scheme for a tail-actuated robotic fish to track a desired trajectory is proposed. The scheme accommodates the under-actuation nature of the dynamics by exploiting input coupling, and it incorporates input constraints. The design is based on an experimentally validated high-fidelity average dynamic model for robotic fish. In particular, inspired by the work [22], a new error coordinate is introduced that augments the heading error with a term dependent on the lateral tracking error. Then by using the angular velocity as a virtual input to regulate the aforementioned modified heading error, the controller is able to handle the tracking of desired heading and lateral displacement trajectories, aside from that of the longitudinal displacement trajectory. Furthermore, to accommodate the input constraints, an auxiliary system is designed such that the error due to the difference between the feasible and “desired” inputs is compensated for. Finally, Lyapunov-based design is carried out to ensure closed-loop stability of the system. Simulation results demonstrate the effectiveness of the controller and show its advantage over a PI controller.

II. DYNAMIC MODEL OF ROBOTIC FISH

The tail-actuated robotic fish is modeled as a rigid body with a rigid tail that is actuated at its base, and it is assumed that the robot operates in an inviscid, irrotational, and incompressible fluid within an infinite domain [23].

Let $[X, Y, Z]^T$ and $[x, y, z]^T$ be defined as the inertial coordinate system and the body-fixed coordinate system, respectively, as illustrated in Fig. 1. In this work, only the planar motion is considered and it is assumed that the body is symmetric with respect to the xz -plane and that the tail moves in the xy -plane. Thus the system only has three degrees of freedom, namely surge (V_{c_x}), sway (V_{c_y}), and yaw (ω_z). Furthermore, let ψ denote the heading angle, formed by the x -axis relative to the X -axis and let α denote the tail deflection angle with respect to the negative x -axis.

Consider the following periodic pattern for the tail deflection angle:

$$\alpha(t) = \alpha_0 + \alpha_a \sin(\omega_\alpha t) \quad (1)$$

where α_0 , α_a , and ω_α represent the bias, amplitude, and frequency of the tail beat, respectively. By using classical averaging methods an averaged model can be obtained as proposed in [23]. In particular, define the states $x_1 = V_{c_x}$,

*This work was supported by the National Science Foundation (DGE1424871, ECCS 1446793, IIS 1715714).

¹Maria Castaño and Xiaobo Tan are with the Department of Electrical and Computer Engineering, Michigan State University, East Lansing, MI 48824, USA. Email: castanom@msu.edu (M.C.), xbtan@egr.msu.edu (X.T.)

$x_2 = V_{cy}$ and $x_3 = \omega_z$ so that the averaged dynamics takes the following form [23]

$$\dot{x}_1 = f_1(x_1, x_2, x_3) + K_f c_7 \bar{f}_4(\alpha_0, \alpha_a, \omega_\alpha) \quad (2)$$

$$\dot{x}_2 = f_2(x_1, x_2, x_3) + K_f c_8 \bar{f}_5(\alpha_0, \alpha_a, \omega_\alpha) \quad (3)$$

$$\dot{x}_3 = f_3(x_1, x_2, x_3) + K_m c_9 \bar{f}_6(\alpha_0, \alpha_a, \omega_\alpha) \quad (4)$$

with

$$f_1(x_1, x_2, x_3) = \frac{m_2}{m_1} x_2 x_3 - \frac{c_1}{m_1} x_1 \sqrt{x_1^2 + x_2^2} + \frac{c_2}{m_1} x_2 \sqrt{x_1^2 + x_2^2} \arctan\left(\frac{x_2}{x_1}\right) \quad (5)$$

$$f_2(x_1, x_2, x_3) = -\frac{m_1}{m_2} x_1 x_3 - \frac{c_1}{m_2} x_2 \sqrt{x_1^2 + x_2^2} - \frac{c_2}{m_2} x_1 \sqrt{x_1^2 + x_2^2} \arctan\left(\frac{x_2}{x_1}\right) \quad (6)$$

$$f_3(x_1, x_2, x_3) = (m_1 - m_2) x_1 x_2 - c_4 \omega_z^2 \text{sgn}(\omega_z) \quad (7)$$

$$\bar{f}_4(\alpha_0, \alpha_a, \omega_\alpha) = \omega_\alpha^2 \alpha_a \left(3 - \frac{3}{2} \alpha_0^2 - \frac{3}{8} \alpha_a^2\right) \quad (8)$$

$$\bar{f}_5(\alpha_0, \alpha_a, \omega_\alpha) = \omega_\alpha^2 \alpha_a^2 \alpha_0 \quad (9)$$

$$\bar{f}_6(\alpha_0, \alpha_a, \omega_\alpha) = \omega_\alpha^2 \alpha_a^2 \alpha_0 \quad (10)$$

where $m_1 = m_b - m_{ax}$, $m_2 = m_b - m_{ay}$, $J_3 = J_{bz} - J_{az}$, $c_1 = \frac{1}{2} \rho S C_D$, $c_2 = \frac{1}{2} \rho S C_L$, $c_4 = \frac{K_D}{J_3}$, $c_7 = \frac{m L^2}{12 m_1}$, $c_8 = \frac{m L^2}{4 m_2}$, and $c_9 = -\frac{c m L^2}{4 J_3}$. Here S denotes the reference surface area for the robot body, C_D, C_L and K_D represent the drag force coefficient, lift coefficient, and drag moment coefficient, respectively, ρ is the density of water, L is the tail length, c is the distance from the body center to the pivot point of the actuated tail and m represents the mass of water displaced by the tail per unit length and is approximated by $\frac{\pi}{4} \rho d^2$ with d denoting the tail depth. K_f is a scaling constant, and K_m is a scaling function affine in α_0 . Finally, the kinematic equations for the robotic fish are given by

$$\dot{X} = V_{cx} \cos \psi - V_{cy} \sin \psi \quad (11)$$

$$\dot{Y} = V_{cx} \sin \psi + V_{cy} \cos \psi \quad (12)$$

$$\dot{\psi} = \omega_z \quad (13)$$

To further facilitate control design, in this work K_m is considered as a constant by taking the average of K_m for a given range of α_0 . The resulting model is referred to as the simplified averaged model.

III. TRAJECTORY TRACKING CONTROL ALGORITHM

A. Trajectory Tracking Error Coordinates

The trajectory tracking problem involves controlling the robotic fish such that it tracks a reference trajectory that is parametrized in time t . In essence, this means that the robot is required to be at a desired position with a desired orientation at any given time t . Fig.1 illustrates the idea.

Let the vectors $\bar{\mathbf{C}}(t)$ and $\bar{\mathbf{P}}(t)$ denote the position of the center of the robotic fish (point C) and the desired position with respect to the inertial frame $\{I\}$ at a given time t , respectively. Let $\bar{\mathbf{C}}$ and $\bar{\mathbf{P}}$ be defined as

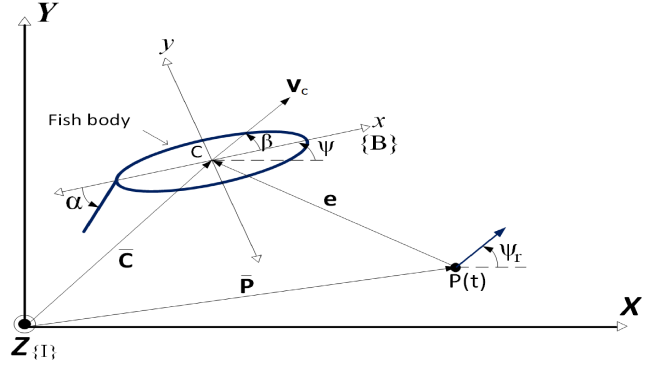


Fig. 1: Top view of the tail-actuated robotic fish undergoing planar motion.

$$\bar{\mathbf{C}} = \begin{bmatrix} X \\ Y \\ \psi \end{bmatrix}, \quad \bar{\mathbf{P}} = \begin{bmatrix} X_r \\ Y_r \\ \psi_r \end{bmatrix} \quad (14)$$

Let ${}^I \mathbf{R}_B$ denote the rotation matrix from the inertial frame $\{I\}$ to the body-fixed frame $\{B\}$ and ${}^B \mathbf{R}_I$ denote the rotation matrix from $\{B\}$ to $\{I\}$, with

$${}^B \mathbf{R}_I = \begin{bmatrix} \cos \psi & -\sin \psi & 0 \\ \sin \psi & \cos \psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (15)$$

Furthermore, let $\mathbf{e} = [X_e; Y_e; \psi_e]$ denote the tracking error vector expressed in the body-fixed frame such that

$$\mathbf{e} = {}^I \mathbf{R}_B (\bar{\mathbf{C}} - \bar{\mathbf{P}}) \quad (16)$$

The derivative of $\mathbf{e}$ expressed in $\{B\}$ is given by

$$\frac{d\mathbf{e}}{dt} = -\mathbf{S}(\omega_z) {}^I \mathbf{R}_B (\bar{\mathbf{C}} - \bar{\mathbf{P}}) + {}^I \mathbf{R}_B \left(\frac{d\bar{\mathbf{C}}}{dt} - \frac{d\bar{\mathbf{P}}}{dt} \right) \quad (17)$$

where

$$\left(\frac{d\mathbf{e}}{dt} \right)_{\{B\}} = \begin{bmatrix} \dot{X}_e \\ \dot{Y}_e \\ \dot{\psi}_e \end{bmatrix} \quad (18)$$

$$\frac{d\bar{\mathbf{P}}}{dt} = \begin{bmatrix} \dot{X}_r \\ \dot{Y}_r \\ \dot{\psi}_r \end{bmatrix} = \begin{bmatrix} \cos \psi_r & -\sin \psi_r & 0 \\ \sin \psi_r & \cos \psi_r & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} u_r \\ v_r \\ \omega_r \end{bmatrix} \quad (19)$$

$$\mathbf{S}(\omega_z) = \begin{bmatrix} 0 & -\omega_z & 0 \\ \omega_z & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad (20)$$

where u_r , v_r and ω_r are desired surge, sway and angular velocities, respectively.

Let $\mathbf{V}_r = \sqrt{\dot{X}_r^2 + \dot{Y}_r^2}$, $\bar{\psi}_r = \arctan(\frac{\dot{Y}_r}{\dot{X}_r})$ and $\bar{\psi}_e = \psi - \bar{\psi}_r$. By solving for $\frac{d\mathbf{e}}{dt}$ from Eq. (17) one can obtain the error state model, which is then augmented with the simplified averaged scaled dynamics as seen in Section II, particularly Eqs. (2)-(4), to obtain

$$\begin{bmatrix} \dot{X}_e \\ \dot{Y}_e \\ \dot{\psi}_e \\ \dot{V}_{cx} \\ \dot{V}_{cy} \\ \dot{\omega}_z \end{bmatrix} = \begin{bmatrix} V_{cx} - \mathbf{V}_r \cos(\bar{\psi}_e) + \omega_z Y_e \\ V_{cy} + \mathbf{V}_r \sin(\bar{\psi}_e) - \omega_z X_e \\ \omega_z - \bar{\omega}_r \\ f_1(V_{cx}, V_{cy}, \omega_z) + K_f c_7 \bar{f}_4(\alpha_0, \alpha_a, \omega_\alpha) \\ f_2(V_{cx}, V_{cy}, \omega_z) + K_f c_8 \bar{f}_5(\alpha_0, \alpha_a, \omega_\alpha) \\ f_3(V_{cx}, V_{cy}, \omega_z) + K_m c_9 \bar{f}_6(\alpha_0, \alpha_a, \omega_\alpha) \end{bmatrix} \quad (21)$$

where $\bar{\omega}_r = \frac{d}{dt}(\arctan \frac{\dot{Y}_r}{\dot{X}_r})$.

B. Backstepping Control Design

By formulating the tracking problem in terms of the error dynamics, the control objective has become a stabilization problem. In particular, the trajectory-tracking problem is to find a control law such that, for an arbitrary initial error, the state $(X_e, Y_e, \bar{\psi}_e)$ of system (21) can be held near the origin $(0,0,0)$. To design the controller, the robot's simplified averaged model is considered. Here the control inputs are functions of the actual physical variables, namely the tail-beat pattern parameters α_0 , α_a , and ω_α . To simplify discussion, a fixed tail-beat frequency ω_α is assumed. The control inputs are then chosen as

$$u_1 = \alpha_a(3 - \frac{3}{2}\alpha_0^2 - \frac{3}{8}\alpha_a^2) \quad (22)$$

$$u_2 = \alpha_a^2 \alpha_0 \quad (23)$$

which are present in functions $f_4(\alpha_0, \alpha_a, \omega_\alpha)$ to $f_6(\alpha_0, \alpha_a, \omega_\alpha)$ in Eqs. (2)-(4). Note that the control inputs defined this way appear linearly in the dynamic equations.

1) Choice of Lyapunov Function: Given the under-actuated nature of the system and the coupling effects between the inputs specifically in Eqs. (3)-(4), the choice of Lyapunov function is not trivial. From Eq. (21) one can see that the error system has a triangular structure with two stages. In other words, one could attempt to design the virtual velocity control V_{c_x} , V_{c_y} and ω_z to stabilize the subsystem X_e , Y_e , and $\bar{\psi}_e$ at the origin, and then design the controls u_1 and u_2 based on the backstepping technique. In this case, the natural choice for a Lyapunov function would be $V = \frac{1}{2}(X_e^2 + Y_e^2 + \bar{\psi}_e^2)$. However, given the nature of the under-actuated dynamics (only two inputs u_1 and u_2 are at our disposal), this would lead to an over-constrained problem where the two inputs need to be chosen to stabilize three states. Similarly, from Eq. (21), one could attempt to stabilize X_e and Y_e at the origin by using virtual inputs V_{c_x} and V_{c_y} , by the choice $V = \frac{1}{2}(X_e^2 + Y_e^2)$. Although this could allow the tracking of the position, the orientation error is not guaranteed to be bounded, and given the input coupling between $\dot{V}_{c_y}$ and $\dot{\omega}_z$, this leads to unstable turning and whirling of the robotic fish. Finally, if one chooses to stabilize X_e and $\bar{\psi}_e$, which entails choosing $V = \frac{1}{2}(X_e^2 + \bar{\psi}_e^2)$, then the lateral displacement error will not be controlled, which does not guarantee actual convergence to the trajectory.

To handle the under-actuated nature of the robot, we define a new Lyapunov function that will allow us to stabilize the subsystem X_e , Y_e , and $\bar{\psi}_e$ with only two virtual velocities. Choosing the sway velocity V_{c_y} as a virtual input is impractical in reality and leads to uncontrolled twirling of the robot; therefore, the only viable choices are V_{c_x} and ω_z . We are thus motivated to define a modified error Z_e that incorporates both the heading error $\bar{\psi}_e$ and the lateral error Y_e :

$$Z_e = \bar{\psi}_e + \phi \quad (24)$$

where the “correction” angle

$$\phi = k_{ze1} \arctan(k_{ze2} u_r Y_e) \quad (25)$$

and k_{ze1} and k_{ze2} are some positive design constants to be chosen later. In particular, k_{ze1} is used to adjust on the correction angle within the total error Z_e , while k_{ze2} tunes the magnitude of the angle. Intuitively, by choosing Z_e in this manner the robotic fish heading angle can be used to steer the robot towards a desired trajectory. In particular, the heading angle error $\bar{\psi}_e$ is “corrected” by an angle ϕ that is dependent on the error Y_e , such that it captures the rotation needed for the robot to point towards the desired trajectory. Furthermore, by utilizing the arctan function, ϕ is guaranteed to be bounded and by choosing Z_e in this manner, tracking of $\bar{\psi}_r$ is implied when Y_e is small enough. Finally, Eq. (24) is well defined and the convergence of Y_e and Z_e implies that of $\bar{\psi}_e$.

2) Trajectory Tracking Control Synthesis: First, to stabilize the (X_e, Z_e) subsystem, the following candidate Lyapunov function is chosen

$$V_1 = \frac{1}{2}X_e^2 + \frac{1}{2}Z_e^2 \quad (26)$$

Let α_1 and α_2 represent the virtual inputs, which will be chosen shortly. Furthermore, let α_{d1} and α_{d2} be the desired virtual inputs, and let the virtual errors be given as

$$Z_1 = \alpha_1 - \alpha_{d1} \quad (27)$$

$$Z_2 = \alpha_2 - \alpha_{d2} \quad (28)$$

The time derivative of Eq. (26) is given by

$$\begin{aligned} \dot{V}_1 &= X_e \dot{X}_e + Z_e \dot{Z}_e \\ &= X_e(V_{c_x} - V_r \cos(\bar{\psi}_e) + \omega_z Y_e) + \\ &\quad Z_e(\mu \omega_z - \bar{\omega}_r + \tau) \end{aligned} \quad (29)$$

where

$$\tau = k_{ze1} k_{ze2} \frac{u_r(V_r \sin(\bar{\psi}_e) + V_{c_y}) + \dot{u}_r Y_e}{(k_{ze2} u_r Y_e)^2 + 1} \quad (30)$$

$$\mu = 1 - \frac{k_{ze1} k_{ze2} u_r X_e}{(k_{ze2} u_r Y_e)^2 + 1} \quad (31)$$

Let $\alpha_1 = V_{c_x}$ and $\alpha_2 = \mu \omega_z$. With Eq. (27), Eq. (29) can be rewritten as

$$\begin{aligned} \dot{V}_1 &= X_e(Z_1 + \alpha_{d1} - V_r \cos(\bar{\psi}_e) + \omega_z Y_e) \\ &\quad + Z_e(Z_2 + \alpha_{d2} - \bar{\omega}_r + \tau) \end{aligned} \quad (32)$$

Let the desired virtual inputs be defined as

$$\alpha_{d1} = V_r \cos(\bar{\psi}_e) - \omega_z Y_e - K_{X_e} X_e \quad (33)$$

$$\alpha_{d2} = \bar{\omega}_r - \tau - K_{Z_e} Z_e \quad (34)$$

so that

$$\dot{V}_1 = X_e(Z_1 - K_{X_e} X_e) + Z_e(Z_2 - K_{Z_e} Z_e) \quad (35)$$

We then define a new Lyapunov function

$$V_2 = V_1 + \frac{1}{2}Z_1^2 + \frac{1}{2}Z_2^2 \quad (36)$$

The time derivative of Eq. (36) is given by

$$\dot{V}_2 = \dot{V}_1 + \dot{Z}_1 Z_1 + \dot{Z}_2 Z_2 \quad (37)$$

With Eqs. (2)-(4) along with the input definition Eq. (22)-(23), one can expand the $\dot{V}_{c_x}$, $\dot{V}_{c_y}$ and $\dot{\omega}_z$ terms that appear

in Eq. (36). After simplification, u_1 and u_2 can be chosen using

$$K_f c_7 u_1 + K_m c_9 Y_e u_2 = -f_1 + \dot{\mathbf{V}}_r \cos(\bar{\psi}_e) - \mathbf{V}_r \sin(\bar{\psi}_e) \dot{\bar{\psi}}_e - f_3 Y_e - \omega_z \dot{Y}_e - K_{X_e} \dot{X}_e - K_{z_1} Z_1 \quad (38)$$

$$\eta u_2 = -f_3 \mu - \dot{\mu} \omega_z + \dot{\omega}_r - \delta - K_{Z_e} \dot{Z}_e - K_{z_2} Z_2 \quad (39)$$

where

$$\delta = k_{ze1} k_{ze2} \frac{F}{(k_{ze2} u_r Y_e)^2 + 1} - k_{ze1} k_{ze2} \frac{G}{((k_{ze2} u_r Y_e)^2 + 1)^2} \quad (40)$$

$$F = \ddot{u}_r Y_e + \dot{u}_r \dot{Y}_e + \dot{u}_r (\mathbf{V}_r \sin(\bar{\psi}_e) + V_{cy}) + u_r (\dot{\mathbf{V}}_r \sin(\bar{\psi}_e) + \mathbf{V}_r \cos(\bar{\psi}_e) \dot{\bar{\psi}}_e + f_2) \quad (41)$$

$$G = 2k_{ze2} Y_e u_r (u_r \dot{Y}_e + Y_e \dot{u}_r) (u_r (\mathbf{V}_r \sin(\bar{\psi}_e) + V_{cy}) + Y_e \dot{u}_r) \quad (42)$$

$$\eta = k_{ze1} k_{ze2} \frac{u_r K_f c_8}{(k_{ze2} u_r Y_e)^2 + 1} + K_m c_9 \mu \quad (43)$$

such that

$$\dot{V}_2 = -K_{X_e} X_e^2 - K_{Z_e} Z_e^2 + X_e Z_1 + Z_e Z_2 - K_1 Z_1^2 - K_2 Z_2^2 \quad (44)$$

By adding and subtracting $\frac{Z_1^2}{4K_{X_e}}$ and $\frac{Z_2^2}{4K_{Z_e}}$ and completing the square, one can arrive at

$$\dot{V}_2 = -K_{X_e} (X_e - \frac{1}{2K_{X_e}} Z_1)^2 - K_{Z_e} (Z_e - \frac{1}{2K_{Z_e}} Z_2)^2 - Z_1^2 (K_1 - \frac{1}{4K_{X_e}}) - Z_2^2 (K_2 - \frac{1}{4K_{Z_e}}) \quad (45)$$

If $K_{X_e} > 0$, $K_{Z_e} > 0$, $K_1 > \frac{1}{4K_{X_e}}$ and $K_2 > \frac{1}{4K_{Z_e}}$, then $\dot{V}_2 < 0$ unless $X_e = Z_e = Z_1 = Z_2 = 0$. From Lasalle's Invariance Principle [24], one can conclude the convergence of (X_e, Z_e, Z_1, Z_2) to zero.

Finally, u_1 and u_2 can be obtained via

$$\begin{bmatrix} u_1 \\ u_2 \end{bmatrix} = \begin{bmatrix} K_f c_7 & K_m c_9 Y_e \\ 0 & \eta \end{bmatrix}^{-1} \begin{bmatrix} \Gamma_1 \\ \Gamma_2 \end{bmatrix} \quad (46)$$

where Γ_1 and Γ_2 represent the right hand side of Eqs. (38)-(39), respectively.

3) Control Synthesis Incorporating Input Constraints:

Given that robot's actuators have physical limitations, the backstepping-based controller design should accommodate such constraints so that the control scheme can be successfully implemented. In order to address magnitude constraints on the control inputs the following scheme inspired by [25] and [26] is proposed.

Let v_1 and v_2 represent the nominal backstepping control inputs, and let u_1 and u_2 be the inputs that can be practically implemented. To obtain the value for u_1 and u_2 , first the values given by v_1 and v_2 are used to solve for the tail-beat parameters α_0, α_a using Eqs. (22)-(23), and then the tail-beat parameter values are saturated such that they lie within the range $[\alpha_{0min}, \alpha_{0max}]$, and $[\alpha_{amin}, \alpha_{amax}]$. Finally u_1 and u_2 are obtained using Eqs. (22)-(23) with the saturated values.

To analyze the influence of the input constraints, the

following auxiliary system is chosen,

$$\begin{cases} \dot{\lambda}_1 = -\zeta_1 \lambda_1 + \lambda_2 \\ \dot{\lambda}_2 = -\zeta_2 \lambda_2 + K_f c_7 (u_1 - v_1) + Y_e K_m c_9 (u_2 - v_2) \\ \dot{\lambda}_3 = -\zeta_3 \lambda_3 + \lambda_4 \\ \dot{\lambda}_4 = -\zeta_4 \lambda_4 - \eta (u_2 - v_2) \end{cases} \quad (47)$$

The variables λ_1 - λ_4 defined above represent the filtered effect of the non-achievable portion of the virtual and control inputs. In particular, the additional tracking error that arises because of mismatch between the nominal and implementable inputs is represent by λ_1 and λ_3 , while λ_2 and λ_4 represent the error propagated to the virtual inputs. As a result, the modified tracking errors are defined as follows:

$$\bar{X}_e = X_e - \lambda_1 \quad (48)$$

$$\bar{Z}_e = Z_e - \lambda_3 \quad (49)$$

Furthermore, let the modified virtual errors be given as

$$\bar{Z}_1 = \alpha_1 - \alpha_{d1} - \lambda_2 \quad (50)$$

$$\bar{Z}_2 = \alpha_2 - \alpha_{d2} - \lambda_4 \quad (51)$$

To stabilize the $(\bar{X}_e, \bar{Z}_e)$ subsystem the following Lyapunov function is chosen

$$\bar{V}_1 = \frac{1}{2} \bar{X}_e^2 + \frac{1}{2} \bar{Z}_e^2 \quad (52)$$

With these new definitions, similar stability analysis as done in Eq. (29)-(35) is carried out. Let the desired virtual inputs be defined as

$$\alpha_{d1} = \mathbf{V}_r \cos(\bar{\psi}_e) - \omega_z Y_e - K_{X_e} \bar{X}_e - \zeta_1 \lambda_1 \quad (53)$$

$$\alpha_{d2} = \bar{\omega}_r - \tau - K_{Z_e} \bar{Z}_e - \zeta_3 \lambda_3 \quad (54)$$

so that

$$\dot{\bar{V}}_1 = \bar{X}_e (\bar{Z}_1 - K_{X_e} \bar{X}_e) + \bar{Z}_e (\bar{Z}_2 - K_{Z_e} \bar{Z}_e) \quad (55)$$

Then a new Lyapunov function is defined as

$$\bar{V}_2 = \bar{V}_1 + \frac{1}{2} \bar{Z}_1^2 + \frac{1}{2} \bar{Z}_2^2 \quad (56)$$

The time derivative of Eq. (56) is given by

$$\dot{\bar{V}}_2 = \dot{\bar{V}}_1 + \dot{\bar{Z}}_1 \bar{Z}_1 + \dot{\bar{Z}}_2 \bar{Z}_2 \quad (57)$$

As previously done, the terms $\dot{V}_{cx}$, $\dot{V}_{cy}$ and $\dot{\omega}_z$ that appear in Eq. (57) can be further expanded. After simplifying the above, v_1 and v_2 can be chosen as

$$K_f c_7 v_1 + K_m c_9 Y_e v_2 = -f_1 + \dot{\mathbf{V}}_r \cos(\bar{\psi}_e) - \mathbf{V}_r \sin(\bar{\psi}_e) \dot{\bar{\psi}}_e - f_3 Y_e - \omega_z \dot{Y}_e - K_{X_e} \dot{X}_e + \zeta_1^2 \lambda_1 - (\zeta_1 + \zeta_2) \lambda_2 - K_{z_1} \bar{Z}_1 \quad (58)$$

$$\eta v_2 = -f_3 \mu - \dot{\mu} \omega_z + \dot{\omega}_r - \delta - K_{Z_e} \dot{Z}_e + \zeta_3^2 \lambda_3 - (\zeta_3 + \zeta_4) \lambda_4 - K_{z_2} \bar{Z}_2 \quad (59)$$

By following similar stability analysis as previously done for the constraints-free case, one can arrive at the following

$$\dot{\bar{V}}_2 = -K_{X_e} (\bar{X}_e - \frac{1}{2K_{X_e}} \bar{Z}_1)^2 - K_{Z_e} (\bar{Z}_e - \frac{1}{2K_{Z_e}} \bar{Z}_2)^2 - \bar{Z}_1^2 (K_1 - \frac{1}{4K_{X_e}}) - \bar{Z}_2^2 (K_2 - \frac{1}{4K_{Z_e}}) \quad (60)$$

If $K_{X_e} > 0$, $K_{Z_e} > 0$, $K_1 > \frac{1}{4K_{X_e}}$ and $K_2 > \frac{1}{4K_{Z_e}}$, then $\dot{\bar{V}}_2 < 0$ unless when $\bar{X}_e = \bar{Z}_e = \bar{Z}_1 = \bar{Z}_2 = 0$

implying the convergence of $(\bar{X}_e, \bar{Z}_e, \bar{Z}_1, \bar{Z}_2)$ to zero as time approaches infinity. Furthermore, since $0 \leq \bar{V}_2(t) \leq \bar{V}_2(0)$, one can conclude that $(\bar{X}_e, \bar{Z}_e, \bar{Z}_1, \bar{Z}_2)$ are each in $\mathcal{L}_2$. This shows that even when physical limitations do not allow the desired control signals to be implemented, the quantities $\bar{X}_e$ and $\bar{Z}_e$ do not diverge. In other words, this guarantees convergence for the compensated tracking errors $\bar{X}_e$ and $\bar{Z}_e$ but not the actual tracking errors X_e and Z_e . The latter may actually increase during periods when input limitations are in effect given that the desired control signal is not being implemented (i.e. $u_1 \neq v_1$ and/or $u_2 \neq v_2$). However, when the control signal limitations are not in effect, (i.e. $u_1 = v_1$ and $u_2 = v_2$), $\lambda_1 - \lambda_4$ approach zero, and $(\bar{X}_e, \bar{Z}_e)$ converges towards (X_e, Z_e) .

IV. SIMULATION RESULTS

To evaluate the effectiveness of the designed controller, simulations were carried out using MATLAB. Furthermore, a PI controller was implemented to provide performance comparison. The robotic fish parameters used for simulation are listed Tab. I.

TABLE I: PARAMETERS OF THE ROBOTIC FISH.

PARAMETER	VALUE	PARAMETER	VALUE
m_b	0.725 kg	m_{ax}	-0.217 kg
m_{ay}	-0.7888 kg	c	0.105 m
J_{az}	-7.93×10^{-4} kg · m ²	L	0.071 m
J_{bz}	2.66×10^{-3} kg · m ²	d	0.04 m
ρ	1000 kg/m ³	S	0.03 m ²
C_D	0.97	C_L	3.9047
K_D	4.5×10^{-3} kg·m ²	K_f	0.7
$K_m(\text{averaged})$	0.45		

The tunable backstepping and PI controller parameters in the simulation were chosen as follows:

$$\begin{array}{llll}
K_{X_e}=0.6 & K_{Z_e}=0.7 & K_{Z_1}=0.46 & K_{Z_2}=0.2 \\
\zeta_1=0.4 & \zeta_2=0.8 & \zeta_3=0.9 & \zeta_4=0.9 \\
K_{Z_{e1}}=5 & K_{Z_{e2}}=63.3 & K_{P1}=3.5 & K_{P2}=0.85 \\
K_{I1}=0.021 & K_{I2}=0.023 & \alpha_{0_{min}}=-50^\circ & \alpha_{0_{max}}=50^\circ \\
\alpha_{a_{min}}=0^\circ & \alpha_{a_{max}}=30^\circ & t_s=0.66 \text{ s} & \omega_\alpha=3\pi
\end{array}$$

where $\alpha_{0_{min}}, \alpha_{0_{max}}, \alpha_{a_{min}}$ and $\alpha_{a_{max}}$ are the physical limits on the tail-beat bias and amplitude respectively. Furthermore, K_{P1}, K_{P2}, K_{I1} , and K_{I2} are the PI controller tunable parameters. The variable t_s is the sampling interval which pertains to the amount of time between an update to the control inputs. In this design, we chose $t_s = 0.66$ seconds given that the tail-beat frequency is 1.5 Hz. The controller parameters were chosen such that under the right values the backstepping controller was able to regulate the error system to the origin. We found that K_{X_e} and K_{Z_e} controlled the balance between the convergence rate of the X_e and Z_e error, respectively, while varying $\zeta_1 - \zeta_4$ controlled the convergence rate of the errors $\lambda_1 - \lambda_4$, which are a consequence of the effect of the input constraints. For the PI controller, its gain parameters were tuned carefully. Note that although the

backstepping controller was designed using the simplified averaged model, the simulations were performed on the

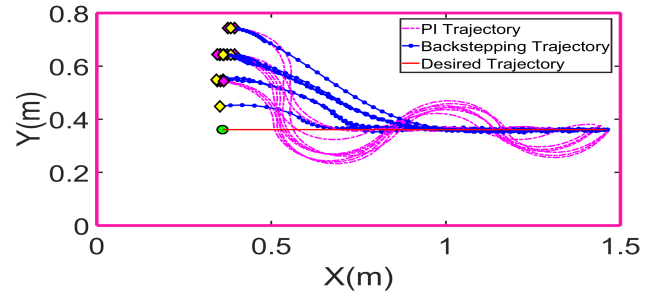


Fig. 2: Simulation: line-tracking trajectories for backstepping-based and PI control.

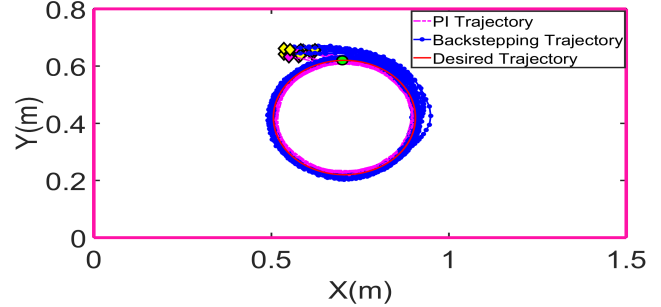


Fig. 3: Simulation: circle-tracking trajectories for backstepping-based and PI control.

original dynamic model. The following line and circular trajectories were considered

$$\begin{aligned}
\dot{X}_r &= u_r, \quad \dot{Y}_r = 0, \quad \dot{\psi}_r = \omega_r \\
u_r &= 0.02 \text{ m/s}, \quad v_r = 0 \text{ m/s}, \quad \omega_r = 0 \text{ rad/s} \\
\dot{X}_r &= R_1 \omega_r \cos(\omega_r t), \quad \dot{Y}_r = -R_1 \omega_r \sin(\omega_r t) \\
\dot{\psi}_r &= \omega_r, \quad R_1 = 0.2 \text{ m}, \quad \omega_r = 0.09 \text{ rad/s}
\end{aligned} \tag{61}$$

where $\dot{X}_r$ and $\dot{Y}_r$ represent the velocity of the trajectory in the $\{I\}$ frame. In Figs. 2-3 the desired and the closed-loop trajectories of the robotic fish are compared for the backstepping-based and PI controller in both the line and circular cases. Note that the diamonds represents the starting position of the robotic fish, while the green circle represents the starting point of the path. In particular, 10 simulations trials were run for each type of trajectory with different initial conditions. Figs. 4-5 illustrate the averaged magnitude of the position errors over time along with the corresponding standard deviations for line and circular tracking, respectively, for both the proposed backstepping scheme and the PI controller. From simulation results, in particular from Figs. 4-5, one can see that, with the proposed backstepping scheme, smaller position tracking error in the allotted time is obtained, as well as a faster convergence to the desired trajectory.

V. CONCLUSIONS

In this paper, we proposed a backstepping-based trajectory tracking backstepping-based scheme for a tail-actuated robotic fish. A high-fidelity averaged nonlinear dynamic model was used for controller design. Furthermore, a Lyapunov function was designed to achieve trajectory tracking and deal with the under-actuated system dynamics. The input

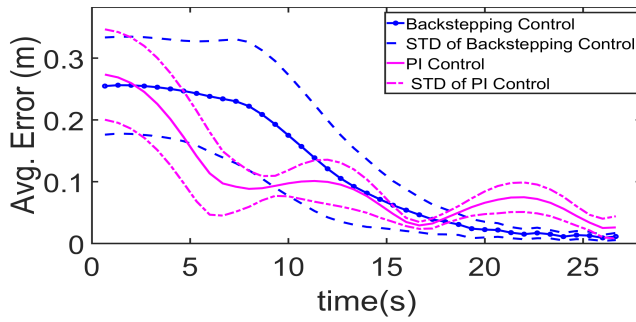


Fig. 4: Simulation: line-tracking position error for backstepping-based and PI control. “STD” curves represent the standard deviation envelopes for the average error under each controller.

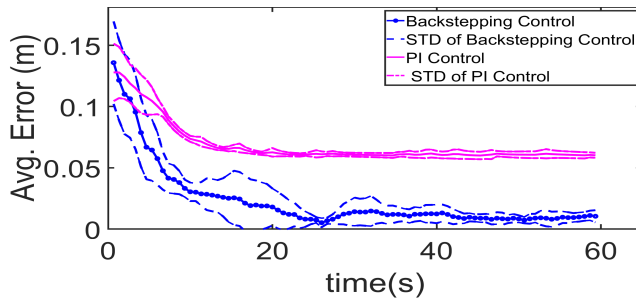


Fig. 5: Simulation: circle-tracking position error for backstepping-based and PI control. “STD” curves represent the standard deviation envelopes for the average error under each controller.

constraints were accommodated in the design via the use of an auxiliary system and the closed-loop stability was guaranteed with Lyapunov analysis. Finally, simulation results demonstrated the effectiveness of the proposed scheme, and showed its advantages over an alternative PI controller.

For future work, the proposed backstepping algorithm will be verified with experiments, and the closed-loop system stability will be analyzed further to demonstrated how the modified error coordinate guarantees the convergence of the heading and lateral error, which seems to be implied from the simulation results.

REFERENCES

- [1] C. Zhou, Z. Cao, S. Wang, and M. Tan, “The posture control and 3D locomotion implementation of biomimetic robot fish,” in *Intelligent Robots and Systems, 2006 IEEE/RSJ International Conference on*. IEEE, 2006, pp. 5406–5411.
- [2] K. A. Morgansen, T. M. La Fond, and J. X. Zhang, “Agile maneuvering for fin-actuated underwater vehicles,” in *Proceedings of the 2006 Second International Symposium on Communications, Control and Signal Processing*.
- [3] J. Yu, L. Wang, and M. Tan, “A framework for biomimetic robot fish’s design and its realization,” in *American Control Conference, 2005. Proceedings of the 2005*. IEEE, 2005, pp. 1593–1598.
- [4] M. Makrodimitris, K. Nanos, and E. Papadopoulos, “A novel trajectory planning method for a robotic fish,” in *Control and Automation (MED), 2017 25th Mediterranean Conference on*. IEEE, 2017, pp. 1119–1124.
- [5] L. Wang, S. Wang, Z. Cao, M. Tan, C. Zhou, H. Sang, and Z. Shen, “Motion control of a robot fish based on CPG,” in *Industrial Technology, 2005. ICIT 2005. IEEE International Conference on*. IEEE, 2005, pp. 1263–1268.
- [6] W. Zhao, Y. Hu, L. Zhang, and L. Wang, “Design and CPG-based control of biomimetic robotic fish,” *IET Control Theory & Applications*, vol. 3, no. 3, pp. 281–293, 2009.
- [7] W. Wang, J. Guo, Z. Wang, and G. Xie, “Neural controller for swimming modes and gait transition on an ostraciiform fish robot,” in *Advanced Intelligent Mechatronics (AIM), 2013 IEEE/ASME International Conference on*. IEEE, 2013, pp. 1564–1569.
- [8] Y. Hu, W. Tian, J. Liang, and T. Wang, “Learning fish-like swimming with a CPG-based locomotion controller,” in *Intelligent Robots and Systems (IROS), 2011 IEEE/RSJ International Conference on*. IEEE, 2011, pp. 1863–1868.
- [9] Q. Ren, J. Xu, and X. Li, “A data-driven motion control approach for a robotic fish,” *Journal of Bionic Engineering*, vol. 12, no. 3, pp. 382–394, 2015.
- [10] K. A. Morgansen, B. I. Triplett, and D. J. Klein, “Geometric methods for modeling and control of free-swimming fin-actuated underwater vehicles,” *IEEE Transactions on Robotics*, vol. 23, no. 6, pp. 1184–1199, 2007.
- [11] K. A. Morgansen, P. A. Vela, and J. W. Burdick, “Trajectory stabilization for a planar carangiform robot fish,” in *Robotics and Automation, 2002. Proceedings. ICRA’02. IEEE International Conference on*, vol. 1. IEEE, 2002, pp. 756–762.
- [12] S. Saimek and P. Y. Li, “Motion planning and control of a swimming machine,” *The International Journal of Robotics Research*, vol. 23, no. 1, pp. 27–53, 2004.
- [13] N. Kato, “Control performance in the horizontal plane of a fish robot with mechanical pectoral fins,” *IEEE Journal of Oceanic Engineering*, vol. 25, no. 1, pp. 121–129, 2000.
- [14] P. Suebsaiprom and C.-L. Lin, “Maneuverability modeling and trajectory tracking for fish robot,” *Control Engineering Practice*, vol. 45, pp. 22–36, 2015.
- [15] K. Zou, C. Wang, G. Xie, T. Chu, L. Wang, and Y. Jia, “Cooperative control for trajectory tracking of robotic fish,” in *American Control Conference, 2009. ACC’09*. IEEE, 2009, pp. 5504–5509.
- [16] P. Suebsaiprom and C.-L. Lin, “Sliding mode path tracking control for fish-robot under ocean current perturbation,” in *Control and Automation (ICCA), 2016 12th IEEE International Conference on*. IEEE, 2016, pp. 972–977.
- [17] F. D. Gao, C. Pan, Y. Han, and X. Zhang, “Nonlinear trajectory tracking control of a new autonomous underwater vehicle in complex sea conditions,” *Journal of Central South University*, vol. 19, no. 7, pp. 1859–1868, 2012.
- [18] K. A. Morgansen, V. Duidam, R. J. Mason, J. W. Burdick, and R. M. Murray, “Nonlinear control methods for planar carangiform robot fish locomotion,” Seoul, Korea: In Proceedings of 2001 IEEE International Conference on Robotics and Automation, 2001, pp. 427–434.
- [19] M. L. Castano and X. Tan, “Model predictive control-based path following for tail-actuated robotic fish,” *Journal of Dynamic Systems, Measurement, and Control*, 2018 in press.
- [20] S. Chen, J. Wang, and X. Tan, “Backstepping-based hybrid target tracking control for a carangiform robotic fish,” in *ASME 2013 Dynamic Systems and Control Conference*. American Society of Mechanical Engineers, 2013, pp. V002T32A005–V002T32A005.
- [21] E. Yang, T. Ikeda, and T. Mita, “Nonlinear tracking control of a nonholonomic fish robot in chained form,” in *SICE 2003 Annual Conference (IEEE Cat. No. 03TH8734)*, vol. 2. IEEE, 2003, pp. 1260–1265.
- [22] K. D. Do and J. Pan, *Control of Ships and Underwater Vehicles: Design for Underactuated and Nonlinear Marine Systems*. Springer Science & Business Media, 2009.
- [23] J. Wang and X. Tan, “Averaging tail-actuated robotic fish dynamics through force and moment scaling,” *IEEE Transactions on Robotics*, vol. 31, no. 4, pp. 906–917, 2015.
- [24] H. K. Khalil, *Nonlinear Control*. Pearson New York, 2015.
- [25] J. Farrell, M. Polycarpou, and M. Sharma, “Adaptive backstepping with magnitude, rate, and bandwidth constraints: Aircraft longitude control,” in *American Control Conference, 2003. Proceedings of the 2003*, vol. 5. IEEE, 2003, pp. 3898–3904.
- [26] R. Mei and Q. Cui, “Backstepping control for a 3dof model helicopter with input and output constraints,” *International Journal of Advanced Robotic Systems*, vol. 14, no. 1, p. 1729881416687133, 2016.