Trajectory Planning and Tracking of Robotic Fish Using Ergodic Exploration

Maria L. Castaño, Anastasia Mavrommati, Todd Murphey and Xiaobo Tan

***Abstract*— In recent years, underwater robots that propel and maneuver themselves like real fish, often called robotic fish, have emerged as promising mobile sensing platforms for freshwater and marine environments. For these active monitoring applications, efficient exploration along with economical locomotion is highly important, in order to optimize sensing coverage and guarantee long field operation time. As a result, optimization of the sensing trajectory and energy-saving tracking of the planned trajectory are of interest. In this paper we adopt an ergodic exploration method to calculate an optimal sensing trajectory for a tail-actuated robotic fish, and propose a nonlinear model predictive control (NMPC) approach for tracking the generated trajectory. A high-fidelity, averaged nonlinear dynamic model is used for trajectory planning and control. In particular, the bias and amplitude of the tail-beat pattern are treated as the control inputs, and their physical bounds and the constraints on their changing rates are properly accounted for in the optimization process. Finally, simulation results are presented to illustrate the effectiveness of the proposed approach.**

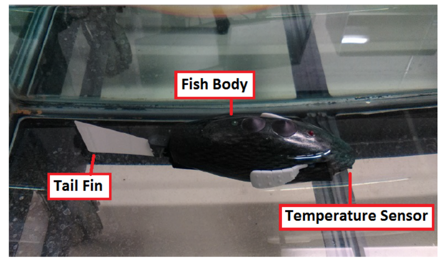

Fig. 1. A robotic fish developed by the Smart Microsystems Lab at Michigan State University.

I. Introduction

Monitoring and understanding aquatic environments has become key for securing sustainable water resources and for ensuring the longevity of aquatic ecosystems. Robotic fish, a class of underwater robots that mimic the movement of fish (Fig.1), have received increasing attention in recent years as they hold strong promise in underwater sensing. These robots have various actuation mechanisms from oscillating caudal or pectoral fins to undulation of the entire body, and like real fish, they are able to attain high maneuverability [1]. However, given that they are often battery-powered and require constant actuation to swim and maneuver, it is crucial for them to be highly efficient in order to attain long field operation time. The latter makes optimal exploration planning and control an important problem for robotic fish. There has been extensive work on using nonlinear control theory to achieve trajectory tracking for robotic fish [2], [3], [4], [5], [6], [7], [8]; however, little work has been reported on trajectory planning for exploration tasks that is combined with systematic control approaches incorporating performance objectives for such robots.

Ergodic exploration presents a promising framework for dealing with trajectory planning for active exploration [9]. Using an ergodic exploration approach for robotic fish is promising because it builds an exploration strategy that takes into account the complex nonlinear dynamics of the robotic fish in a continuous manner without requiring spatial grid decomposition. Such a planned trajectory is also amenable to tracking control. In particular, nonlinear model predictive control (NMPC) holds promise in dealing with uncertainties and constraints for trajectory tracking. Generating an optimal trajectory with the use of ergodic exploration assures efficient sampling of a domain, while using NMPC for tracking the resulting trajectory ensures the incorporation of performance and control effort objectives.

The application of both ergodic trajectory optimization and NMPC trajectory tracking for robotic fish has not been reported in the literature. In this paper we employ an ergodic exploration scheme [9], [10] for planning an exploration trajectory, by tailoring the optimization algorithm to robotic fish dynamics, and develop an NMPC scheme for tracking the generated trajectory. Both are implemented for a tail-actuated robotic fish, based on an experimentally validated, high-fidelity averaged dynamic model [11]. We generate optimal ergodic trajectories over two-dimensional probability density functions (PDFs) that provide the spatial probability that a particular feature (for example, target of interest) is located at each point in the domain, and demonstrate the effectiveness of the trajectory planning and tracking control approach through simulations.

The rest of the paper is organized as follows. In Section II, we review the original and averaged dynamic models for tail-actuated robotic fish. In Section III, we present the trajectory optimization approach for robotic fish using ergodic exploration. The NMPC trajectory tracking scheme is

*This research was supported in part by the National Science Foundation (DGE 1424871, IIS 1319602, CCF 1331852, ECCS 1446793).

[1]Maria Castaño is with the Department of Electrical Engineering and Computer Engineering, Michigan State University, East Lansing, MI, 48824, USA castanom@msu.edu,

[2]Anastasia Mavrommati is with the Department of Mechanical Engineering, Northwestern University, Evanston, IL 60208, USA anastasiamavrommati2012@u.northwestern.edu

[3]Todd Murphey is faculty at the Department of Mechanical Engineering, Northwestern University, Evanston, IL 60208, USA t-murphey@northwestern.edu

[4]Xiaobo Tan is faculty at the Department of Electrical and Computer Engineering, Michigan State University, East Lansing, MI, 48824, USA xbtan@egr.msu.edu

described in Section IV. The simulation results are presented in Section V. And finally, we provide concluding remarks and discuss future work in Section VI.

II. Dynamic Model of Robotic Fish

We model the tail-actuated robotic fish as was done in [11]. We consider the robot to be a rigid body with a rigid tail that is actuated at its base, and we assume that the robot operates in an inviscid, irrotational, and incompressible fluid within an infinite domain.

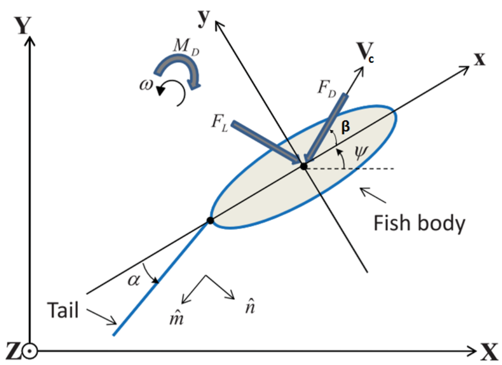

Fig. 2. Top view of the tail-actuated robotic fish undergoing planar motion [11].

As illustrated in Fig. 2, we define $[X,Y,Z]^T$ and $[x, y, z]^T$ as the inertial coordinate system and the body-fixed coordinate system, respectively. The velocity of the center of mass in the body-fixed coordinates is expressed as $V_c = [V_{c_x}, V_{c_y}, V_{c_z}]$, where V_{c_x} indicates surge, V_{c_y} indicates sway, and V_{c_z} indicates heave. The angular velocity expressed in the body-fixed coordinate system is given by $\omega = [\omega_x, \omega_y, \omega_z]$, which is composed of roll (ω_x), pitch (ω_y), and yaw (ω_z). We let β denote the angle of attack, formed by the direction of V_c with respect to the x-axis, and ψ denote the heading angle, formed by the x-axis relative to the X-axis. Finally, we let α denote the tail deflection angle with respect to the negative x-axis.

We only consider the planar motion, and further assume that the body is symmetric with respect to the xz-plane and that the tail moves on the xy-plane. As a result, the system only has three degrees of freedom, surge (V_{c_x}), sway (V_{c_y}), and yaw (ω_z). We further assume that we can neglect the inertial coupling between yaw, sway and surge motions, and arrive at the following equations of planar motion:

$$(m_b - m_{a_x})\dot{V}_{c_x} = (m_b - m_{a_y})V_{c_y}\omega_z + f_x \tag{1}$$

$$(m_b - m_{a_y})\dot{V}_{c_y} = -(m_b - m_{a_x})V_{c_x}\omega_z + f_y \tag{2}$$

$$(J_{bz} - J_{a_z})\dot{\omega}_z = (m_{a_y} - m_{a_x})V_{c_x}V_{c_y} + M_z \tag{3}$$

where m_b is the mass of the body, J_{bz} is the inertia of the body about the z-axis. m_{a_x},and m_{a_y} are the hydrodynamic derivatives that represent the added masses of the robotic fish along the x and y directions, respectively, and J_{a_z} represents the added inertia effect of the body about the z direction. The hydrodynamic forces and moment due to tail fin actuation and the interaction of the body itself with the fluid are captured by f_x, f_y, and M_z. To evaluate the hydrodynamics forces exerted by the tail, Lighthill's large amplitude elongated body theory is used, as shown in [11]. The kinematic equations for the robotic fish are given by

$$\dot{X} = V_{c_x}\cos\psi - V_{c_y}\sin\psi \tag{4}$$

$$\dot{Y} = V_{c_x}\sin\psi + V_{c_y}\cos\psi \tag{5}$$

$$\dot{\psi} = \omega \tag{6}$$

With the rhythmic nature of the robotic fish movement, averaging has proven to be a useful approach in studying the effect of the input parameters on the dynamics of the robotic fish and fin movement [11]. In practical applications, it is more natural to control the parameters for periodic fin beats than directly controlling the fin position at every moment. For this reason, an averaged model is best suited for trajectory planning and tracking control. In the following we review the scaled averaging model proposed in [11]. We consider the following periodic pattern for the tail deflection angles

$$\alpha(t) = \alpha_0 + \alpha_a \sin(\omega_\alpha t) \tag{7}$$

where α_0, α_a, and ω_α represent the bias, amplitude, and frequency of the tail beat, respectively. The original hydrodynamic force and moment terms in Eqs. (1)-(3) are scaled by some scaling functions dependent on the tail beat parameters, α_0, α_a, and ω_α, and classical averaging is then conducted over these scaled dynamics. In particular, we define the states $x_1 = V_{c_x}$, $x_2 = V_{c_y}$ and $x_3 = \omega$, and then the averaged dynamics takes the following form

$$\dot{x}_1 = f_1(x_1, x_2, x_3) + K_{force} \cdot \bar{f}_4(\alpha_0, \alpha_a, \omega_\alpha) \tag{8}$$

$$\dot{x}_2 = f_2(x_1, x_2, x_3) + K_{force} \cdot \bar{f}_5(\alpha_0, \alpha_a, \omega_\alpha) \tag{9}$$

$$\dot{x}_3 = f_3(x_1, x_2, x_3) + K_{moment} \cdot \bar{f}_6(\alpha_0, \alpha_a, \omega_\alpha) \tag{10}$$

with

$$\begin{aligned} f_1(x_1, x_2, x_3) =& \frac{m_2}{m_1}x_2x_3 - \frac{c_1}{m_1}x_1\sqrt{x_1^2 + x_2^2} + \\ & \frac{c_2}{m_1}x_2\sqrt{x_1^2 + x_2^2}\arctan(\frac{x_2}{x_1}) \end{aligned} \tag{11}$$

$$\begin{aligned} f_2(x_1, x_2, x_3) =& -\frac{m_1}{m_2}x_1x_3 - \frac{c_1}{m_2}x_2\sqrt{x_1^2 + x_2^2} \\ & -\frac{c_2}{m_2}x_1\sqrt{x_1^2 + x_2^2}\arctan(\frac{x_2}{x_1}) \end{aligned} \tag{12}$$

$$f_3(x_1, x_2, x_3) = (m_1 - m_2)x_1x_2 - c_4\omega^2 sgn(\omega) \tag{13}$$

$$\bar{f}_4(\alpha_0, \alpha_a, \omega_\alpha) = \frac{m}{12m_1L^2}\omega_\alpha^2\alpha_a(3 - \frac{3}{2}\alpha_0^2 - \frac{3}{8}\alpha_a^2) \tag{14}$$

$$\bar{f}_5(\alpha_0, \alpha_a, \omega_\alpha) = \frac{m}{4m_2L^2}\omega_\alpha^2\alpha_a^2\alpha_0 \tag{15}$$

$$\bar{f}_6(\alpha_0, \alpha_a, \omega_\alpha) = -\frac{m}{4J_3L^2}c\omega_\alpha^2\alpha_a^2\alpha_0 \tag{16}$$

where $m_1 = m_b - m_{a_x}$, $m_2 = m_b - m_{a_y}$, $J_3 = J_{bz} - J_{a_z}$, $c_1 = \frac{1}{2}\rho SC_D$, $c_2 = \frac{1}{2}\rho SC_L$, $c_3 = \frac{1}{2}mL^2$, $c_4 =$

$\frac{1}{(J_3)}K_D,\ c_5 = \frac{1}{(2J_3)}L^2 mc,\ c_6 = \frac{1}{(3J_3)}L^3 m.S$ 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 joint 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_{force}, and K_{moment} are the scaling functions. For a tail-beat pattern with $\alpha_0 \neq 0$, the robotic fish trajectory converges to a circle. To determine the scaling functions K_{force}, K_{moment}, a blanket-search of these parameters can be conducted for each set of (α_0, α_a, ω_α) such that the averaged model matches the original dynamics in turning radius and period. It was found in [11] that K_{force} is constant, while K_{moment} is affine in the tail beat bias; in particular, for the robotic fish prototype used in [11],

$$K_{moment} = -0.0074 + 0.008432\alpha_0 \tag{17}$$

where α_0 is given in degrees. To further facilitate control design,in this paper we consider K_{moment} as a constant during the NMPC design. This term is found by taking the average of K_{moment} for a given range of α_0. We call the resulting model the simplified averaged model.The performance of the controller, however, will be checked against the non-simplified average model during simulation.

III. Ergodic Exploration

Active sensing in an efficient manner is highly important for a robotic fish. When exploring, there is a trade-off between energy consumption and area coverage. It is thus desirable to spend more time exploring areas that provide more information or that have a higher likelihood of containing features of interest. The idea of using ergodicity allows the comparison between the time-averaged behavior of a system to a map of expected information density (EID). Thus using it as a metric guarantees the generation of optimal trajectories for which the amount of time spent on a particular space is proportional to how much information is contained there [9], [10].

A. Trajectory Optimization

A trajectory $x(t)$ is ergodic with respect to some distribution $\phi(x)$ when the percentage of time spent over any subset of the domain is equal to the measure of that subset. Here we assume that there is a PDF $\phi(x)$ representing the information density over an n-dimensional domain $X \subset \mathbb{R}^n$ defined as $[0, L_1] \times [0, L_2] ... \times [0, L_n]$. The ergodic metric will be used to determine how far a trajectory is from being ergodic with respect to this distribution and it is defined as

$$\varepsilon = \sum_{k=0}^{K} \Lambda_k |c_k - \phi_k|^2 \tag{18}$$

where $\phi_k, k = 0, \cdots, K$, are the Fourier coefficients of the spatial distribution $\phi(x)$, and $c_k, k = 0, \cdots, K$, are the Fourier coefficients of the basis functions along a trajectory $x(t)$ averaged over time, the calculation of which will be detailed shortly. K is the number of basis functions used in the exploration domain. Λ_k is defined as

$$\Lambda_k = \frac{1}{(1 + ||k||^2)^s} \tag{19}$$

with $s = \frac{n+1}{2}$.

The ergodic objective function $J(\cdot)$ is defined as the sum of the ergodic metric and the control effort, and it takes as an argument any (feasible or unfeasible) curve $\xi(t) = (\alpha(t), \mu(t))$, where $\alpha(t)$ and $\mu(t)$ represent a path trajectory and a control trajectory, respectively,

$$\begin{aligned} J(\xi(t)) :=& q \sum_{k=0}^{K} \Lambda_k \left(\frac{1}{T} \int_0^T F_k(\alpha(\tau)) d\tau - \phi_k \right)^2 \\ &+ \int_0^T \frac{1}{2} \mu(\tau) R(\tau) \mu(\tau) d\tau \end{aligned} \tag{20}$$

where $q \in \mathbb{R}$ and $R(\tau) \in \mathbb{R}^{(m \times m)}$ are weighting parameters used to defined the relative importance between ergodicity and control effort, and T represents the planning horizon. Here $F_k(\alpha(\tau)), k = 0, \cdots, K$, are the Fourier basis functions used to approximate a distribution defined over n dimensions and they are evaluated as

$$F_k(\alpha(\tau)) = \frac{1}{h_k} \prod_{i=1}^{n} \cos(\frac{k\pi}{L_i}\alpha_i(\tau)), \quad \text{for} \quad k = 0, 1, 2 \cdots, K \tag{21}$$

where α_i is the ith component of α and h_k is a normalizing factor. Note that in (20) the time-averaged $F_k(\alpha)$ is exactly c_k in (18). To solve for feasible continuous time trajectories, the following function is minimized

$$\arg\min_{\xi(t)} = J(\xi(t)) \tag{22}$$

This optimization problem is subject to nonlinear constraints and is not written in the form of a Bolza problem [9]. However, with the use of a projection operator, the calculation at each iteration of the optimization occurs in the infinite dimensional tangent space of the constraint. This removes the nonlinear constraint imposed by the dynamics during the descent direction search at each iteration. As a result, the optimization problem is reformulated as an equivalent, unconstrained optimization problem. As discussed in [9], the problem is reduced to solving a linear quadratic problem which can be solved using a Riccati differential equation.

B. Trajectory Planning for Robotic Fish

We utilize the ergodicity-based trajectory optimization method discussed above to solve for a continuous exploratory trajectory for sampling a region using a robotic fish. We implement the algorithm with the robot's simplified averaged model, in which the controls represent functions of the actual control variables, namely, the tail-beat pattern parameters α_0, α_a, and ω_α. To simplify discussion, we assume that the robotic fish has a fixed tail-beat frequency ω_α, and the actual control variables are α_0 and α_a. From the robotic fish model Eqs. (8)-(10), in particular in functions

$f_4(\alpha_0, \alpha_a, \omega_\alpha)$-$f_6(\alpha_0, \alpha_a, \omega_\alpha)$, the control inputs are defined as

$$\boldsymbol{u} = \begin{pmatrix} u_1 \\ u_2 \end{pmatrix} = \begin{pmatrix} \alpha_a^2(3 - \frac{3}{2}\alpha_0^2 - \frac{3}{8}\alpha_a^2) \\ \alpha_a^2\alpha_0 \end{pmatrix} \tag{23}$$

To avoid changing the tail beat patterns much faster than 1 Hz, we imposed an indirect constraint in the rate of change on the controls by adding the inputs u_1 and u_2 as states (denoted as x_7 and x_8 below) and using a barrier cost function to maintain them in the desired range. Consequently, the new inputs (v_1, v_2) are the time derivative of the original controls. Our robotic fish model thus takes the form

$$\dot{x}_1 = f_1(x_1, x_2, x_3) + K_{force}\frac{m}{12m_1L^2}\omega_\alpha^2 x_7 \tag{24}$$

$$\dot{x}_2 = f_2(x_1, x_2, x_3) + K_{force}\frac{m}{4m_2L^2}\omega_\alpha^2 x_8 \tag{25}$$

$$\dot{x}_3 = f_3(x_1, x_2, x_3) - K_{moment}\frac{m}{4J_3L^2}c\omega_\alpha^2 x_8 \tag{26}$$

$$\dot{x}_4 = x_1 \cos x_6 - x_2 \sin x_6 \tag{27}$$

$$\dot{x}_5 = x_1 \sin x_6 + x_2 \cos x_6 \tag{28}$$

$$\dot{x}_6 = x_3 \tag{29}$$

$$\dot{x}_7 = v_1 \tag{30}$$

$$\dot{x}_8 = v_2 \tag{31}$$

IV. Design of Trajectory Tracking Control

To be suitable for aquatic monitoring applications, it is highly desirable for robotic fish to have energy-efficient locomotion in order to prolong battery life. Therefore, it is important to design a controller that is able to meet performance objectives such as minimizing the trajectory-tracking error while accommodating consideration of control effort. In this section, a trajectory tracking problem is posed and an NMPC scheme is developed in detail for the simplified averaged model for robotic fish. The resulting controller will be subject to the non-simplified average model in simulation. As mentioned earlier, in this work the control input consists of two of the tail-beat parameters, specifically the tail bias, and tail amplitude, while the angular frequency is kept constant.

A. Trajectory Tracking Problem Formulation

The trajectory tracking problem involves controlling a robot to track a given trajectory that is parameterized in time t. In essence, this means that the robot is required to track a specified position and orientation at any given time. Fig. 2 illustrates the idea. We let point C denote the center of the robotic fish, and the vectors $\bar{\boldsymbol{C}}$ denote the positions of C as well as the orientation of the robot, and $\bar{\boldsymbol{P}}$ a point P(t) on the trajectory at a given time, in the inertial frame. Let the reference trajectory P(t) be defined as

$$\bar{\boldsymbol{P}} = \begin{bmatrix} X_d \\ Y_d \\ \theta_d \end{bmatrix} \tag{32}$$

We thus let $\boldsymbol{e} = [X_e; Y_e; \theta_e]$ describe the difference between the trajectory position and orientation and that of the robotic fish at time t. The trajectory tracking problem is to find suitable control laws for the tail-beat bias α_0 and amplitude α_a, to drive the error of X_e, Y_e, and θ_e to zero.

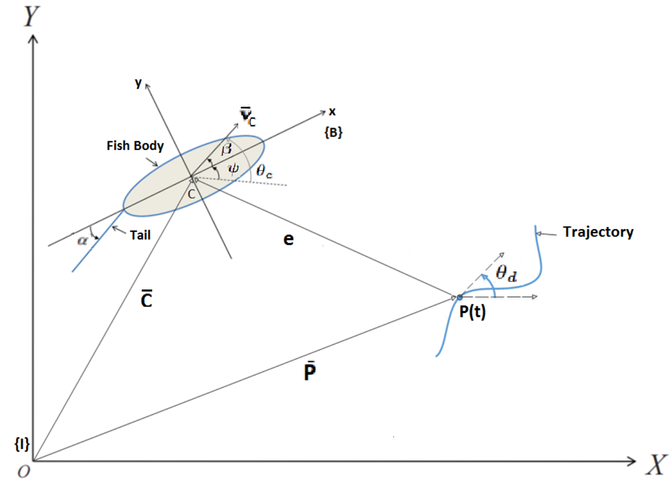

Fig. 3. Illustration of trajectory-tracking for robotic fish.

B. Nonlinear Model Predictive Control Design

Nonlinear model predictive control (NMPC) is an optimization-based method for feedback control of nonlinear systems where a finite-horizon optimization problem subject to state and input constraints is solved repeatedly. Fig. 4 depicts this principle. NMPC is an attractive controller choice because it allows explicit consideration of state and input constraints, is capable of handling nonlinear models, and can optimize control performance [12], [13].

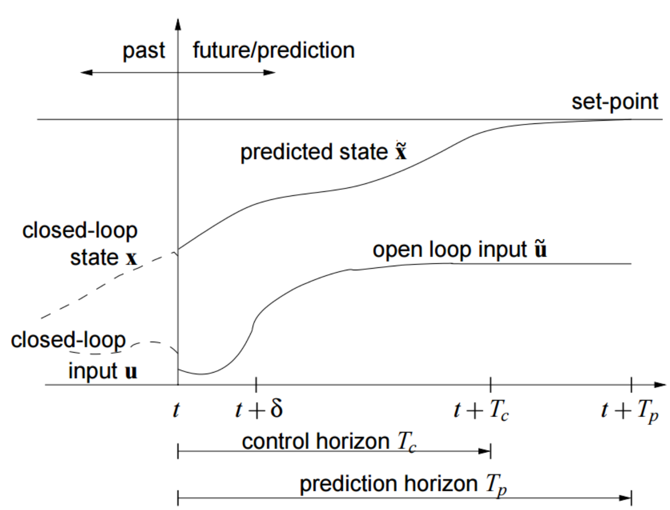

Fig. 4. Principle of nonlinear model predictive control [12].

We consider the general form of a nonlinear system expressed as:

$$\begin{gathered} \dot{x} = f(\boldsymbol{x}(t), \boldsymbol{u}(t)),\ x(0) = x_0, \\ \text{subject to } \boldsymbol{u}(t) \in U,\ \boldsymbol{x}(t) \in X,\ t \in [0, \infty] \end{gathered} \tag{33}$$

where $\boldsymbol{x}(t) \in R^N$, $\boldsymbol{u}(t) \in R^M$ are the state vector and input vector, respectively, and the set X and U represent the feasible states and inputs of the system.

The input that is applied to the system is given by the solution to the following finite horizon open-loop optimal control problem, which is solved at every sampling instant δ:

$$\min_{\tilde{u}(\cdot)} = J(\tilde{\boldsymbol{x}}(t), \tilde{u}(\cdot)) \tag{34}$$

subject to

$$\dot{\tilde{\boldsymbol{x}}}(t) = f(\tilde{\boldsymbol{x}}(t), \tilde{\boldsymbol{u}}(t)), \tilde{\boldsymbol{x}}(0) = x_0, \tag{35}$$

$$\tilde{\boldsymbol{u}}(\tau) \in U, \forall \tau \in [t, t + T_c], \tag{36}$$

$$\tilde{\boldsymbol{u}}(\tau) = \tilde{\boldsymbol{u}}(t + T_c), \forall \tau \in [t + T_c, t + T_p], \tag{37}$$

$$\tilde{\boldsymbol{x}}(\tau) \in X, \forall \tau \in [t, t + T_p] \tag{38}$$

where $J(\boldsymbol{x}(t), \tilde{u}(\cdot))$ denotes the objective function, and is given as

$$J(\boldsymbol{x}(t), \tilde{\boldsymbol{u}}(\cdot)) := \int_t^{t+T_p} F(\tilde{\boldsymbol{x}}(\tau), \tilde{\boldsymbol{u}}(\tau) d\tau + E(\tilde{x}(t + T_p)) \tag{39}$$

where F is the cost function specifying the desired performance objective, E is the terminal cost, T_p and T_c denote the prediction horizon and the control horizon, respectively ($T_c \leq T_p$). The terminal constraint is given by $\tilde{x}(t+T_p) \in \Omega$. $\tilde{x}(\cdot)$ is the solution to Eq. (22) driven by the input signal $\tilde{u}(\cdot) : [t, t + T_p] \to U$ under the initial condition $\boldsymbol{x_0}$; in other words, the tilde denotes predicted values or internal controller variables. The optimal solutions to this problem are denoted by $\tilde{u}^*(\cdot; x(t)) : [t, t+T_p] \to U$, and the input applied to the system is the sequence of these optimal solutions obtained at every sampling instant δ so that $\boldsymbol{u} = \tilde{\boldsymbol{u}}^*(\cdot; x(\delta))$. The nominal closed loop system is then given by :

$$\dot{\boldsymbol{x}}(t) = f(\boldsymbol{x}(t), \boldsymbol{u} = \tilde{\boldsymbol{u}}^*(\cdot; x(\delta))) \tag{40}$$

To steer the robotic fish to track the desired trajectory, we utilize the NMPC approach as described above, and choose the cost function as shown in (39) and define the following quadratic cost as the stage cost function

$$F(\boldsymbol{\Omega}_e, \boldsymbol{u}) = (\boldsymbol{\Omega}_e(\tau))^T Q(\boldsymbol{\Omega}_e(\tau)) + \boldsymbol{u}(\tau)^T R \boldsymbol{u}(\tau) \tag{41}$$

where $\boldsymbol{\Omega}_e(\tau)$ is defined as

$$\boldsymbol{\Omega}_e = \begin{bmatrix} X_e(\tau) \\ Y_e(\tau) \\ \theta_e(\tau) \end{bmatrix} \tag{42}$$

and Q and R are positive definite weighting matrices that penalize deviations from the desired values. In [14] it was shown that closed-loop stability can be achieved for relative;y long horizons by just tuning Q, R and T. Based on this, our cost function will not contain a terminal penalty. We use the robotic fish model Eqs. (8)-(10) to design the controller. In particular from the functions in $f_4(\alpha_0, \alpha_a, \omega_\alpha)$-$f_6(\alpha_0, \alpha_a, \omega_\alpha)$, we choose $\boldsymbol{u}$ as

$$\boldsymbol{u} = \begin{pmatrix} u_1 \\ u_2 \end{pmatrix} = \begin{pmatrix} \alpha_a^2(3 - \frac{3}{2}\alpha_0^2 - \frac{3}{8}\alpha_a^2) \\ \alpha_a^2\alpha_0 \end{pmatrix} \tag{43}$$

By solving the optimal control problem in (34) we obtain the optimal control sequence for α_0, and α_a.

V. SIMULATION

We present the optimal exploratory trajectory obtained in simulation, and demonstrate the trajectory tracking performance obtained with the NMPC. The parameters used (Tab. 1) for the dynamic averaged model were based on the robotic fish prototype in [11].

TABLE I
PARAMETERS OF THE ROBOTIC FISH.

PARAMETER	VALUE
m_b	0.311 kg
m_{ax}	-0.0621 kg
m_{ay}	-0.2299 kg
J_{bz}	5.0797×10^{-4} kg · m^2
J_{az}	-1.0413×10^{-4} kg · m^2
L	0.08 m
d	0.025 m
c	0.07 m
ρ	1000 kg/m^3
S	0.0108 m^2
C_D	0.386
C_L	4.50
K_D	7.82 $\times10^{-4}$ kg·m^2

The following were the tail-beat parameter constraints considered for the robotic fish

$\alpha_{a_{max}} = 30°$
$\alpha_{0_{max}} = 50°$
$\alpha_{a_{min}} = 0°$
$\alpha_{0_{min}} = -50°$

From practical considerations,α_0 and α_a should change slowly with respect to the tail beat frequency, which introduces rate constraints on the control inputs. In particular, we assume

$$\begin{bmatrix} -\frac{\pi}{2} & \leq \dot{\alpha}_a \leq \frac{\pi}{2} \\ -\frac{\pi}{2} & \leq \dot{\alpha}_0 \leq \frac{\pi}{2} \end{bmatrix} \tag{44}$$

Considering these constraints and the tail-beat parameters ranges we obtain the following constraints on (u_1, u_2) and their derivatives (v_1, v_2):

$$\begin{bmatrix} 0.0013 & \leq u_1 \leq 0.7038 \\ -0.0006 & \leq u_2 \quad \leq 0.0006 \end{bmatrix} \tag{45}$$

$$\begin{bmatrix} -3.0 & \leq v_1 \leq 3.4 \\ -1.1 & \leq v_2 \leq 1.2 \end{bmatrix} \tag{46}$$

For trajectory planning using ergodic exploration, we assume a unimodal Gaussian PDF for the distribution of information in the specified domain, as illustrated in Fig.

5. The distribution has a mean of (3,1) and covariance of [0.15 0; 0 0.09].

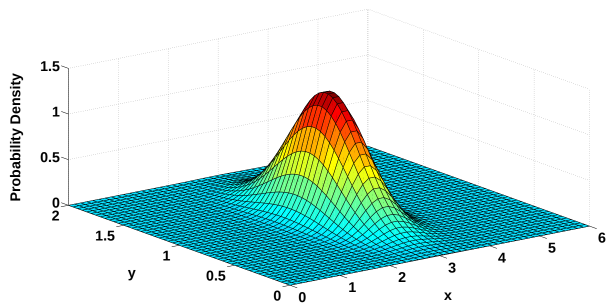

Fig. 5. 3D plot of the spatial PDF

The parameters used to solve the ergodic optimization problem were as follows:

Length of optimization horizon : $T = 5$
Sampling interval : $\delta = 0.1$
Ergodic cost weighting matrix : $q = 20\boldsymbol{I}_8$
Control effort weighting matrix : $R = 0.01\boldsymbol{I}_2$
Max number of harmonics : $K = 20$

Fig. 6 shows the optimized ergodic trajectory for the robotic fish with respect to the PDF. The optimized trajectory is plotted in red, while the gray-scale contours represent its corresponding spatial distribution.

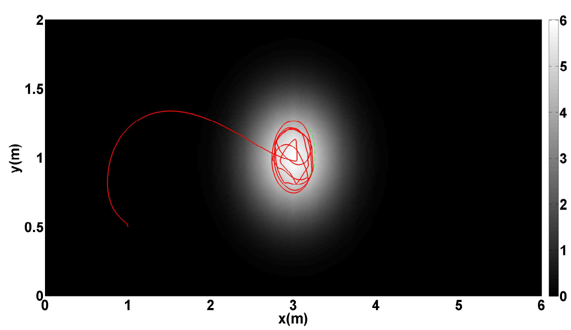

Fig. 6. Optimal trajectory for the robotic fish generated by ergodic exploration.

As seen from the simulation results the optimized trajectory spends more time exploring near the peaks of the PDF, which indicates that the planned trajectory is optimally ergodic.

In order to evaluate the effectiveness of the designed controller, simulations were carried out using the MATLAB interface of the ACADO Model Predictive Control Toolkit [15].

The parameters used to solve the optimization problem and implement the NMPC were as follows

Length of optimization horizon : $T_c = T_p = 15$
Sampling interval : $\delta = 1$
Weighting matrix : $Q = 0.9\boldsymbol{I}_3$
Control weighting matrix : $R = 0.001\boldsymbol{I}_2$

In Fig. 7 we compare the optimized trajectory and the closed loop robotic fish trajectory obtained using the averaged model. Note that the dotted line represents the closed-loop trajectory of the robotic fish, while the red line is the path. Furthermore, the red dot represents the starting position of the robotic fish, while the green square represents the starting point of the trajectory. It can be seen that the actual robot trajectory converges to the planned trajectory as time proceeds, and the robotic fish spends much of its exploration time around the peak of the information density.

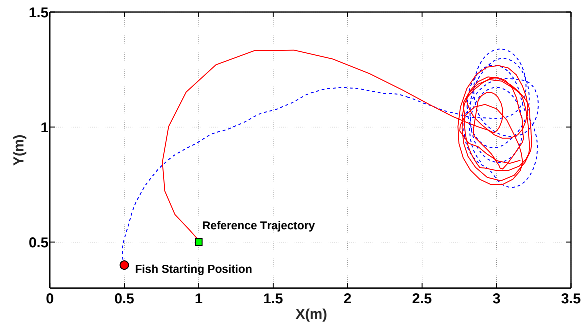

Fig. 7. Simulation results on trajectory tracking of robotic fish.

VI. Conclusion

In this paper we reported the first investigation on using ergodic exploration for trajectory planning of a robotic fish in active sensing, where a highly nonlinear dynamic model and specific physical constraints were accommodated. Furthermore, we proposed a nonlinear model predictive control for tracking the planned trajectory while minimizing control effort. This controller featured tunable parameters that provided trade-off between tracking error and control effort, and was designed based on varying only the amplitude and bias of the tail-beat. Overall we were able to track the given trajectory while expending low control effort.

For future work, we plan to carry out experiments to validate the proposed approach. In particular, a robotic fish with temperature-sensing capability (Fig. 1) will be used to explore a tank with a certain profile of temperature distribution (which can be adjusted using heaters at different locations), for example, to search for the location of highest temperature. In these experiments, the robot needs to estimate the PDF online based on its temperature measurement, which means trajectory planning will need to be implemented online.

References

[1] X. Tan, "Autonomous robotic fish as mobile sensor platforms: Challenges and potential solutions," 2007.

[2] S. Saimek and P. Y. Li, "Motion planning and control of a swimming machine," *Int. J. Robot. Res.*, vol. 23, no. 1, pp. 27–54, 2004.

[3] L. Lapierre and B. Jouvencel, "Robust nonlinear path-following control of an auv," *IEEE Journal of Oceanic Engineering*, vol. 33, no. 2, pp. 89–102, 2008.

[4] R. M. J. B. K. A. Morgansen, V. Duindam and R. Murray, "Nonlinear control methods for planar carangiform robot fish locomotion," (Seoul, Korea), pp. 427–434, 2001 IEEE International Conference on Robotics and Automation, 2001.

[5] S. W. J. Yu, M. Tan and E. Chen, "Development of a biomimetic robotic fish and its control algorithm," vol. 34, no. 4, pp. 1798–1810, 2004.
[6] N. Kato, "Control performance in the horizontal plane of a fish robot with mechanical pectoral fins," *IEEE Jourlnal of Oceanic Engineering*, vol. 25, no. 1, pp. 121–129, 2000.
[7] N. Kato, "Pectoral fin controllers," *Neurotechnology for Biomimetic Robots*, 2002.
[8] C.-y. H. Y.-y. Gao, Fu-dong Pan 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.
[9] M. M. L. Miller, Y. Silverman and T. Murphey, "Ergodic exploration of distributed information," *IEEE Transactions on Robotics*, vol. 32, no. 1, pp. 36–51, 2016.
[10] L. M. Miller and T. D. Murphey, "Trajectory optimization for continuous ergodic exploration on the motion group se (2)," in *52nd IEEE Conference on Decision and Control*, pp. 4517–4522, IEEE, 2013.
[11] X. T. J. Wang, "Averaging tail-actuated robotic fish dynamics through force and moment scaling," *IEEE Transactions on Robotics*, vol. 31, no. 4, pp. 906–917, 2015.
[12] D. M. P. C. E. Garcia and M. Morari, "Model predictive control: Theory and practice- a survey," *Automatica*, vol. 25, no. 3, pp. 335–348, 1989.
[13] R. F. F. Allgower and Z. K. Nagy, "Nonlinear model predictive control:from theory to application," *J. Chin. Inst. Chem. Engrs*, vol. 35, no. 3, pp. 299–315, 2004.
[14] L. UNE and J. Pannek, "Nonlinear model predictive control: Theory and algorithms, communications and control engineering," 2011.
[15] D. Ariens, B. Houska, H. Ferreau, and F. Logist, "Acado: Toolkit for automatic control and dynamic optimization."