A waypoint guidance strategy for underwater snake robots

E. Kelasidi, K. Y. Pettersen and J. T. Gravdahl

Abstract— **In this paper, a waypoint guidance strategy is proposed for an underwater snake robot. The robot is directed to follow a path which is derived using path planning techniques. A first version of the path is derived by the path planner by using the artificial potential field method for obstacle avoidance. Afterwards, by subsampling the derived path, a set of waypoints are chosen along the path. The path is then defined by interconnecting these waypoints by straight lines. Secondly, a straight line path following controller is proposed, to make the underwater snake robot follow the desired path. Simulation results are presented, illustrating the performance of the proposed guidance control strategy for both lateral undulation and eel-like motion.**

I. Introduction

For centuries, engineers and scientists have gained inspiration from the natural world, while searching for ideal solutions to technical problems. More recently, this process has been termed as biomimetics. Every biological organism living in an aquatic environment, swims by generating a propulsive force through the interaction between the body and the surrounding fluid that is created through a rhythmic body movement. Generally, studies of hyper-redundant mechanisms (HRMs), also known as snake robots, have largely restricted themselves to land-based studies, while several models for snake robots have been proposed [1]. Empirical and analytic studies of snake locomotion were reported by Gray in [2], while, among the first attempts to develop a snake prototype, the work of Hirose [3] is essential. Recently, HRMs are presented that are suited for aquatic propulsion as well [4], [5], [6].

Motion planning, i.e. a path covering a certain area or moving the robot towards a desired location-goal taking into account energy consumption [7], is a challenging task for underwater snake robot locomotion. One important feature of motion planning is obstacle avoidance. Algorithms for obstacle avoidance for robots appeared as early as mid-1980s [8], [9]. [10] proposes the use of vector field histograms to steer the robot towards the direction of low obstacle density areas. In [11], the idea for potentials creation from obstacles that repel the robot and potentials from the target that attracts it, is presented. [12] presents a solution for collision and obstacle free formation flight and reconfiguration of groups of autonomous helicopters. Artificial Potential Fields (APF) have since been utilised in various applications as they offer a fast and simple method for obstacle avoidance [13]. Also, this method is used by [14] in the case of a fish robot. In this paper we will use artificial potential fields as the first step in the motion planner, creating a first rough description of the desired path.

Given the desired path that the robot should follow, the next task is to develop a path following controller. Regarding control for underwater snake robots, several types of controllers have been proposed in the literature [4], [15], [16], [17]. However, it is worth mentioning that the emphasis so far has mainly been on achieving forward and turning locomotion [4]. The next step would be not only to achieve forward locomotion, but also to make the snake robot follow a desired path, i.e. solving the path following control problem, and this paper presents preliminary results towards this end. Research on robotic fish and underwater snake robots (also referred to as eel-like robots) are related since these mechanisms have important similarities. The works of [4], [18] and [19] synthesize gaits for translational and rotational motion of various fish-like mechanisms and propose controllers for tracking straight and curved trajectories. The work of [15] demonstrates the use of central pattern generators (CPGs) for crawling and swimming in a real robot. Eel-like motion is considered in [4] and [16], where controllers for tracking straight and curved trajectories are proposed. In this paper we are motivated by recent results for land-based snake robots. In particular, [20] proposes a waypoint guidance strategy for steering a land-based snake robot along a path, defined by straight lines of interconnected points. Waypoint guidance strategy, where the waypoints are defined a priori, is proposed for a Carangiform swimmer in [17]. Path following control for underwater snake robots, however, still remains an open problem.

In this paper we combine the use of an artificial potential fields-based path planner with a new waypoint guidance strategy for an underwater snake robot. In particular, the first contribution of this paper is to describe how an approach for straight line path following, that previously has been presented for a ground snake robot in [20], can be extended and used for path following control of underwater snake robots. The paper presents preliminary results towards solving the path following control problem for underwater snake robots, by proposing a waypoint guidance control strategy. Simulations are presented, and these are seen to support the hypothesis that the proposed control strategy can solve the path following control problem. The paper is thus a first step towards solving the control problem, and lays the foundation for a full systems analysis of the closed-loop system. A full

E. Kelasidi, and J. T. Gravdahl are with the Department of Engineering Cybernetics at NTNU, NO-7491 Trondheim, Norway. E-mail: {Eleni.Kelasidi,Tommy.Gravdahl}@itk.ntnu.no

K. Y. Pettersen is with the Centre for Autonomous Marine Operations and Systems, Dept. of Engineering Cybernetics at NTNU, NO-7491 Trondheim, Norway. E-mail: Kristin.Y.Pettersen@itk.ntnu.no

This work was partly supported by the Research Council of Norway through project no. 205622 and its Centres of Excellence funding scheme, project no. 223254

systems analysis is challenging, since the kinematics and dynamics of underwater snake robots are complex, and is a topic of future work. In addition, as opposed to [17] and [20] where the waypoints are chosen a priori, we propose in this waypoint guidance strategy to instead derive the waypoints using a path planner based on the artificial potential field method in order to also address the obstacle avoidance problems. Simulation results that illustrate the performance of the proposed guidance strategy both for lateral undulation and eel-like motions are presented.

The paper is organized as follows. Section II presents the dynamic model of an underwater snake robot, while the motion planning algorith is outlined in Section III. A straight line path following controller for underwater snake robot is presented in Section IV, followed by simulation results in Section V. Finally, conclusions and suggestions for further research are given in Section VI.

II. Dynamic Modelling of Underwater Snake Robot

This section briefly presents the kinematics and dynamics of an underwater snake robot moving in a virtual horizontal plane. An extensive presentation of the kinematics and dynamics of the robot can be found in [6].

A. Notations and defined symbols

The robot consists of n rigid links of equal length $2l$ interconnected by $n-1$ joints. The links are assumed to have the same mass m and moment of inertia $J=\frac{1}{3}ml^2$. The mass of each link is uniformly distributed so that the link CM (center of mass) is located at its center point (at length l from the joint at each side). The total mass of the snake robot is therefore nm. In the following sections, the kinematics and dynamics of the underwater snake robot will be described in terms of the mathematical symbols described in Table I and illustrated in Fig. 1. The following vectors and matrices are used in the subsequent sections:

$$\mathbf{A}=\begin{bmatrix} 1 & 1 & & \\ & \ddots & \ddots & \\ & & 1 & 1 \end{bmatrix}, \mathbf{D}=\begin{bmatrix} 1 & -1 & & \\ & \ddots & \ddots & \\ & & 1 & -1 \end{bmatrix},$$

where $\mathbf{A},\mathbf{D}\in\mathbb{R}^{(n-1)\times n}$. Furthermore,

$$\mathbf{e}=\begin{bmatrix} 1 & \dots & 1 \end{bmatrix}^T\in\mathbb{R}^n, \mathbf{E}=\begin{bmatrix} \mathbf{e} & \mathbf{0}_{n\times 1} \\ \mathbf{0}_{n\times 1} & \mathbf{e} \end{bmatrix}\in\mathbb{R}^{2n\times 2},$$

$$\sin\theta = \begin{bmatrix} \sin\theta_1 & \dots & \sin\theta_n \end{bmatrix}^T\in\mathbb{R}^n, \mathbf{S}_\theta=\mathrm{diag}(\sin\theta)\in\mathbb{R}^{n\times n},$$

$$\cos\theta = \begin{bmatrix} \cos\theta_1 & \dots & \cos\theta_n \end{bmatrix}^T\in\mathbb{R}^n, \mathbf{C}_\theta=\mathrm{diag}(\cos\theta)\in\mathbb{R}^{n\times n}$$

$$\mathrm{sgn}\theta = \begin{bmatrix} \mathrm{sgn}\theta_1 & \dots & \mathrm{sgn}\theta_n \end{bmatrix}^T\in\mathbb{R}^n$$

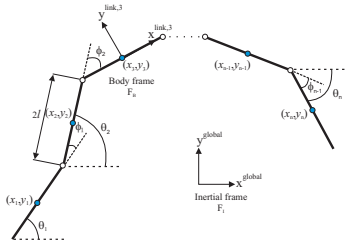
$$\dot\theta^2 = \begin{bmatrix} \dot\theta_1{}^2 & \dots & \dot\theta_n{}^2 \end{bmatrix}^T\in\mathbb{R}^n, \mathbf{J}=J\mathbf{I}_n, \mathbf{L}=l\mathbf{I}_n, \mathbf{M}=m\mathbf{I}_n$$

$$\mathbf{K}=\mathbf{A}^T\left(\mathbf{D}\mathbf{D}^T\right)^{-1}\mathbf{D}, \mathbf{V}=\mathbf{A}^T\left(\mathbf{D}\mathbf{D}^T\right)^{-1}\mathbf{A}$$

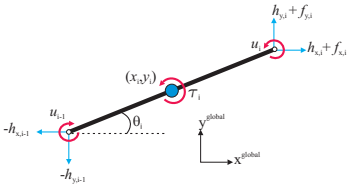
B. Kinematics of the underwater snake robot

The snake robot is assumed to move in a virtual horizontal plane, fully immersed in water, and has n+2 degrees of freedom (n links angles and the x-y position of the robot). The *link angle* of each link $i\in 1,\dots,n$ of the snake robot is denoted by $\theta_i\in\mathbb{R}$, while the *joint angle* of joint $i\in 1,\dots,n-1$ is given by $\phi_i=\theta_i-\theta_{i-1}$. The *heading* (or *orientation*) $\bar\theta\in\mathbb{R}$ of the snake is defined as the average of the links angles similar to land-based snake robots in [1]

$$\bar\theta=\frac{1}{n}\sum_{i=1}^{n}\theta_i. \tag{1}$$



(a) Kinematic parameters



(b) Forces and torques acting on each link

Fig. 1: Underwater snake robot

TABLE I: Definition of mathematical terms

Symbol	Description	Vector
n	The number of links	
l	The half length of a link	
m	Mass of each link	
J	Moment of inertia of each link	
θ_i	Angle between link i and the global x axis	$\theta\in\mathbb{R}^n$
ϕ_i	Angle of joint i	$\phi\in\mathbb{R}^{n-1}$
(x_i,y_i)	Global coordinates of the CM of link i	$\mathbf{X},\mathbf{Y}\in\mathbb{R}^n$
(p_x,p_y)	Global coordinates of the CM of the robot	$\mathbf{p}_{CM}\in\mathbb{R}^2$
u_i	Actuator torque of joint between link i and link $i+1$	$\mathbf{u}\in\mathbb{R}^{n-1}$
u_{i-1}	Actuator torque of joint between link i and link $i-1$	$\mathbf{u}\in\mathbb{R}^{n-1}$
$(f_{x,i},f_{y,i})$	Fluid force on link i	$\mathbf{f}_x,\mathbf{f}_y\in\mathbb{R}^n$
τ_i	Fluid torque on link i	$\tau\in\mathbb{R}^n$
$(h_{x,i},h_{y,i})$	Joint constraint force on link i from link $i+1$	$\mathbf{h}_x,\mathbf{h}_y\in\mathbb{R}^{n-1}$
$-(h_{x,i-1},h_{y,i-1})$	Joint constraint force on link i from link $i-1$	$\mathbf{h}_x,\mathbf{h}_y\in\mathbb{R}^{n-1}$

The global frame position $\mathbf{p}_{\mathrm{CM}}\in\mathbb{R}^2$ of the CM (center of mass) of the robot is given by

$$\mathbf{p}_{\mathrm{CM}}=\begin{bmatrix} p_x \\ p_y \end{bmatrix}=\begin{bmatrix} \frac{1}{nm}\sum_{i=1}^n mx_i \\ \frac{1}{nm}\sum_{i=1}^n my_i \end{bmatrix}=\frac{1}{n}\begin{bmatrix} \mathbf{e}^T\mathbf{X} \\ \mathbf{e}^T\mathbf{Y} \end{bmatrix}, \tag{2}$$

where (x_i,y_i) are the global frame coordinates of the CM of link i, $\mathbf{X}=[x_1,\dots,x_n]^T\in\mathbb{R}^n$ and $\mathbf{Y}=[y_1,\dots,y_n]^T\in\mathbb{R}^n$. The links are constrained by the joints according to

$$\mathbf{D}\mathbf{X}+l\mathbf{A}\cos\theta=\mathbf{0}, \quad \mathbf{D}\mathbf{Y}+l\mathbf{A}\sin\theta \quad =\mathbf{0}. \tag{3}$$

It is shown in [6] that the position of the individual links as a function of the CM position and the link angles of the robot is given by

$$\mathbf{X}=-l\mathbf{K}^T\cos\theta+\mathbf{e}p_x, \qquad \mathbf{Y}=-l\mathbf{K}^T\sin\theta+\mathbf{e}p_y, \tag{4}$$

where $\mathbf{K}=\mathbf{A}^T\left(\mathbf{D}\mathbf{D}^T\right)^{-1}\mathbf{D}\in\mathbb{R}^{n\times n}$, and where $\mathbf{D}\mathbf{D}^T$ is nonsingular and thereby invertible [1]. The linear velocities of the links are given by

$$\dot{\mathbf{X}}=l\mathbf{K}^T\mathbf{S}_\theta\dot\theta+\mathbf{e}\dot p_x, \quad \dot{\mathbf{Y}}=-l\mathbf{K}^T\mathbf{C}_\theta\dot\theta+\mathbf{e}\dot p_y. \tag{5}$$

The linear accelerations of the links are found by differentiating the velocity of the individual links (5) with respect to time, which gives

$$\ddot{\mathbf{X}}=l\mathbf{K}^T\left(\mathbf{C}_\theta\dot\theta^2+\mathbf{S}_\theta\ddot\theta\right)+\mathbf{e}\ddot p_x, \quad \ddot{\mathbf{Y}}=l\mathbf{K}^T\left(\mathbf{S}_\theta\dot\theta^2-\mathbf{C}_\theta\ddot\theta\right)+\mathbf{e}\ddot p_y. \tag{6}$$

Note that in this paper (6) has been adjusted compared to the corresponding expression presented in [6] in order to express the acceleration of the links in a more proper way, by also taking into account the acceleration of the CM.

C. Hydrodynamic modelling

The hydrodynamic modeling approach, that considered in this paper, takes into account both the linear and the nonlinear drag forces (resistive fluid forces), the added mass effect (reactive fluid forces), the fluid moments and current

effect (see, e.g. [6]). In particular, in [6], it is shown that the fluid forces on all links in vector form can be expressed as

$$\mathbf{f} = \begin{bmatrix} \mathbf{f_x} \\ \mathbf{f_y} \end{bmatrix} = \begin{bmatrix} \mathbf{f_{A_x}} \\ \mathbf{f_{A_y}} \end{bmatrix} + \begin{bmatrix} \mathbf{f^I_{D_x}} \\ \mathbf{f^I_{D_y}} \end{bmatrix} + \begin{bmatrix} \mathbf{f^{II}_{D_x}} \\ \mathbf{f^{II}_{D_y}} \end{bmatrix}. \tag{7}$$

The vectors $\mathbf{f_{A_x}}$ and $\mathbf{f_{A_y}}$ represent the effects from added mass forces and are expressed as

$$\begin{aligned} \begin{bmatrix} \mathbf{f_{A_x}} \\ \mathbf{f_{A_y}} \end{bmatrix} = &- \begin{bmatrix} \mu_n(\mathbf{S}_\theta)^2 & -\mu_n \mathbf{S}_\theta \mathbf{C}_\theta \\ -\mu_n \mathbf{S}_\theta \mathbf{C}_\theta & \mu_n(\mathbf{C}_\theta)^2 \end{bmatrix} \begin{bmatrix} \ddot{\mathbf{X}} \\ \ddot{\mathbf{Y}} \end{bmatrix} \\ &- \begin{bmatrix} -\mu_n \mathbf{S}_\theta \mathbf{C}_\theta & -\mu_n(\mathbf{S}_\theta)^2 \\ \mu_n(\mathbf{C}_\theta)^2 & \mu_n \mathbf{S}_\theta \mathbf{C}_\theta \end{bmatrix} \begin{bmatrix} \mathbf{V}_x^a \\ \mathbf{V}_y^a \end{bmatrix} \dot{\theta}, \end{aligned} \tag{8}$$

where $\mathbf{V}_x^a = \operatorname{diag}(V_{x,1}, \ldots, V_{x,n}) \in \mathbb{R}^{n\times n}$, $\mathbf{V}_y^a = \operatorname{diag}(V_{y,1}, \ldots, V_{y,n}) \in \mathbb{R}^{n\times n}$ and $[V_{x,i}, V_{y,i}]^T$ is the current velocity expressed in inertial frame coordinates. The vectors $\mathbf{f^I_{D_x}}$, $\mathbf{f^I_{D_y}}$ and $\mathbf{f^{II}_{D_x}}$, $\mathbf{f^{II}_{D_y}}$ present the effects from the linear (9) and nonlinear drag forces (10), respectively, where the relative velocities are given from the Eq. (11).

$$\begin{bmatrix} \mathbf{f^I_{D_x}} \\ \mathbf{f^I_{D_y}} \end{bmatrix} = - \begin{bmatrix} c_t(\mathbf{C}_\theta)^2 + c_n(\mathbf{S}_\theta)^2 & (c_t - c_n)\mathbf{S}_\theta \mathbf{C}_\theta \\ (c_t - c_n)\mathbf{S}_\theta \mathbf{C}_\theta & c_t(\mathbf{S}_\theta)^2 + c_n(\mathbf{C}_\theta)^2 \end{bmatrix} \begin{bmatrix} \dot{\mathbf{X}} - \mathbf{V}_x \\ \dot{\mathbf{Y}} - \mathbf{V}_y \end{bmatrix} \tag{9}$$

$$\begin{bmatrix} \mathbf{f^{II}_{D_x}} \\ \mathbf{f^{II}_{D_y}} \end{bmatrix} = - \begin{bmatrix} c_t \mathbf{C}_\theta & -c_n \mathbf{S}_\theta \\ c_t \mathbf{S}_\theta & c_n \mathbf{C}_\theta \end{bmatrix} \operatorname{sgn}\left(\begin{bmatrix} \mathbf{V_{r_x}} \\ \mathbf{V_{r_y}} \end{bmatrix} \right) \begin{bmatrix} \mathbf{V_{r_x}}^2 \\ \mathbf{V_{r_y}}^2 \end{bmatrix} \tag{10}$$

$$\begin{bmatrix} \mathbf{V_{r_x}} \\ \mathbf{V_{r_y}} \end{bmatrix} = \begin{bmatrix} \mathbf{C}_\theta & \mathbf{S}_\theta \\ -\mathbf{S}_\theta & \mathbf{C}_\theta \end{bmatrix} \begin{bmatrix} \dot{\mathbf{X}} - \mathbf{V}_x \\ \dot{\mathbf{Y}} - \mathbf{V}_y \end{bmatrix} \tag{11}$$

In addition, the fluid torques on all links in matrix form are

$$\tau = -\Lambda_1 \ddot{\theta} - \Lambda_2 \dot{\theta} - \Lambda_3 \dot{\theta} |\dot{\theta}|, \tag{12}$$

where $\Lambda_1 = \lambda_1 \mathbf{I}_n$, $\Lambda_2 = \lambda_2 \mathbf{I}_n$ and $\Lambda_3 = \lambda_3 \mathbf{I}_n$ [6]. The coefficients c_t, c_n, λ_2, λ_3 represent the drag forces parameters due to the pressure difference between the two sides of the body, and the parameters μ_n, λ_1 stand for the added mass of fluid carried by the moving body.

D. Equations of motion

This subsection presents the equations of motion for the robot, in terms of the acceleration of the links angles, $\ddot{\theta}$, and the acceleration of the CM of the robot, $\mathbf{p}_{\mathrm{CM}}$. These coordinates describe all $n+2$ DOFs of the robot. In [6], it is shown that the force balance equations for all links may be expressed in matrix form as

$$m\ddot{\mathbf{X}} = \mathbf{D}^T \mathbf{h}_x + \mathbf{f_x}, \quad m\ddot{\mathbf{Y}} = \mathbf{D}^T \mathbf{h}_y + \mathbf{f_y}. \tag{13}$$

Note that the link accelerations may also be expressed by differentiating (3) twice with respect to time. This gives

$$\mathbf{D}\ddot{\mathbf{X}} = l\mathbf{A}\left(\mathbf{C}_\theta \dot{\theta}^2 + \mathbf{S}_\theta \ddot{\theta}\right), \quad \mathbf{D}\ddot{\mathbf{Y}} = l\mathbf{A}\left(\mathbf{S}_\theta \dot{\theta}^2 - \mathbf{C}_\theta \ddot{\theta}\right). \tag{14}$$

We obtain the acceleration of the CM by differentiating (2) twice with respect to time, inserting (13), and noting that the constraint forces $\mathbf{h}_x$ and $\mathbf{h}_y$, are cancelled out when the link accelerations are summed. This gives

$$\begin{bmatrix} \ddot{p}_x \\ \ddot{p}_y \end{bmatrix} = \frac{1}{n} \begin{bmatrix} \mathbf{e}^T \ddot{\mathbf{X}} \\ \mathbf{e}^T \ddot{\mathbf{Y}} \end{bmatrix} = \frac{1}{nm} \begin{bmatrix} \mathbf{e}^T & \mathbf{0}_{1\times n} \\ \mathbf{0}_{1\times n} & \mathbf{e}^T \end{bmatrix} \mathbf{f} \tag{15}$$

By inserting (6), (7) and (8) into (15) the acceleration of the CM may be expressed as

$$\begin{aligned} \begin{bmatrix} \ddot{p}_x \\ \ddot{p}_y \end{bmatrix} = &-\mathbf{M}_p \begin{bmatrix} \mathbf{e}^T \mu_n \mathbf{S}_\theta^2 & -\mathbf{e}^T \mu_n \mathbf{S}_\theta \mathbf{C}_\theta \\ -\mathbf{e}^T \mu_n \mathbf{S}_\theta \mathbf{C}_\theta & \mathbf{e}^T \mu_n \mathbf{C}_\theta^2 \end{bmatrix} \begin{bmatrix} l\mathbf{K}^T(\mathbf{C}_\theta \dot{\theta}^2 + \mathbf{S}_\theta \ddot{\theta}) \\ l\mathbf{K}^T(\mathbf{S}_\theta \dot{\theta}^2 - \mathbf{C}_\theta \ddot{\theta}) \end{bmatrix} \\ &- \mathbf{M}_p \begin{bmatrix} -\mathbf{e}^T \mu_n \mathbf{S}_\theta \mathbf{C}_\theta & -\mathbf{e}^T \mu_n \mathbf{S}_\theta^2 \\ \mathbf{e}^T \mu_n \mathbf{C}_\theta^2 & \mathbf{e}^T \mu_n \mathbf{S}_\theta \mathbf{C}_\theta \end{bmatrix} \begin{bmatrix} \mathbf{V}_x^a \\ \mathbf{V}_y^a \end{bmatrix} \dot{\theta} + \mathbf{M}_p \begin{bmatrix} \mathbf{e}^T \mathbf{f_{Dx}} \\ \mathbf{e}^T \mathbf{f_{Dy}} \end{bmatrix} \end{aligned} \tag{16}$$

where

$$\mathbf{M}_p = \begin{bmatrix} m_{11} & m_{12} \\ m_{21} & m_{22} \end{bmatrix} = \begin{bmatrix} nm + \mathbf{e}^T \mu_n \mathbf{S}_\theta^2 \mathbf{e} & -\mathbf{e}^T \mu_n \mathbf{S}_\theta \mathbf{C}_\theta \mathbf{e} \\ -\mathbf{e}^T \mu_n \mathbf{S}_\theta \mathbf{C}_\theta \mathbf{e} & nm + \mathbf{e}^T \mu_n \mathbf{C}_\theta^2 \mathbf{e} \end{bmatrix}^{-1}. \tag{17}$$

and $\mathbf{f_{Dx}} = \mathbf{f^I_{D_x}} + \mathbf{f^{II}_{D_x}}$ and $\mathbf{f_{Dy}} = \mathbf{f^I_{D_y}} + \mathbf{f^{II}_{D_y}}$ are the drag forces in x and y directions. Additionally, it is easily verifiable that the determinant $n^2m^2 + nm\mu_n + \mu_n^2 \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} (\sin(\theta_i - \theta_j))^2$ is nonzero for $n \neq 0$ and $m \neq 0$.

The torque balance equations for all links is expressed in matrix form as

$$\mathbf{J}\ddot{\theta} = \mathbf{D}^T \mathbf{u} - l\mathbf{S}_\theta \mathbf{A}^T \mathbf{h}_x + l\mathbf{C}_\theta \mathbf{A}^T \mathbf{h}_y + \tau, \tag{18}$$

where τ is given from (12) [6]. What now remains is to remove the constraint forces from (18). By premultiplying (13) by $\mathbf{D}$ and solving for $\mathbf{h}_x$ and $\mathbf{h}_y$, we can write the expression for the joint constraint forces as

$$\begin{aligned} \mathbf{h}_x = &(\mathbf{D}\mathbf{D}^T)^{-1} \mathbf{D}(m\ddot{\mathbf{X}} + \mu_n(\mathbf{S}_\theta)^2 \ddot{\mathbf{X}} - \mu_n \mathbf{S}_\theta \mathbf{C}_\theta \ddot{\mathbf{Y}} \\ &- \mu_n \mathbf{S}_\theta \mathbf{C}_\theta \mathbf{V}_x^a \dot{\theta} - \mu_n(\mathbf{S}_\theta)^2 \mathbf{V}_y^a \dot{\theta} - \mathbf{f^I_{D_x}} - \mathbf{f^{II}_{D_x}}) \\ \mathbf{h}_y = &(\mathbf{D}\mathbf{D}^T)^{-1} \mathbf{D}(m\ddot{\mathbf{Y}} - \mu_n \mathbf{S}_\theta \mathbf{C}_\theta \ddot{\mathbf{X}} + \mu_n(\mathbf{C}_\theta)^2 \ddot{\mathbf{Y}} \\ &+ \mu_n(\mathbf{C}_\theta)^2 \mathbf{V}_x^a \dot{\theta} + \mu_n \mathbf{S}_\theta \mathbf{C}_\theta \mathbf{V}_y^a \dot{\theta} - \mathbf{f^I_{D_y}} - \mathbf{f^{II}_{D_y}}). \end{aligned} \tag{19}$$

Inserting in (18) the joint constraints forces (19) and also replacing $\mathbf{D}\ddot{\mathbf{X}}$, $\mathbf{D}\ddot{\mathbf{Y}}$ with (14), $\ddot{\mathbf{X}}$,$\ddot{\mathbf{Y}}$ with (6) and $\ddot{p}_x$, $\ddot{p}_y$ with (16), and solving for $\ddot{\theta}$, we can finally express the model of an underwater snake robot as

$$\mathbf{M}_\theta \ddot{\theta} + \mathbf{W}_\theta \dot{\theta}^2 + \mathbf{V}_\theta \dot{\theta} + \Lambda_3 |\dot{\theta}| \dot{\theta} + \mathbf{K_{Dx}} \mathbf{f_{Dx}} + \mathbf{K_{Dy}} \mathbf{f_{Dy}} = \mathbf{D}^T \mathbf{u}, \tag{20}$$

where $\mathbf{M}_\theta$, $\mathbf{W}_\theta$, $\mathbf{V}_\theta$, $\mathbf{K_{Dx}}$ and $\mathbf{K_{Dy}}$ are defined as

$$\mathbf{M}_\theta = \mathbf{J} + ml^2 \mathbf{S}_\theta \mathbf{V} \mathbf{S}_\theta + ml^2 \mathbf{C}_\theta \mathbf{V} \mathbf{C}_\theta + \Lambda_1 + l^2 \mu_n \mathbf{K}_1 \mathbf{K}^T \mathbf{S}_\theta + l^2 \mu_n \mathbf{K}_2 \mathbf{K}^T \mathbf{C}_\theta \tag{21}$$

$$\mathbf{W}_\theta = ml^2 \mathbf{S}_\theta \mathbf{V} \mathbf{C}_\theta - ml^2 \mathbf{C}_\theta \mathbf{V} \mathbf{S}_\theta + l^2 \mu_n \mathbf{K}_1 \mathbf{K}^T \mathbf{C}_\theta - l^2 \mu_n \mathbf{K}_2 \mathbf{K}^T \mathbf{S}_\theta \tag{22}$$

$$\mathbf{V}_\theta = \Lambda_2 - l\mu_n \mathbf{K}_2 \mathbf{V}_x^a - l\mu_n \mathbf{K}_1 \mathbf{V}_y^a \tag{23}$$

$$\mathbf{K_{Dx}} = l\mu_n m_{11} A_1 \mathbf{e}\mathbf{e}^T - l\mu_n m_{21} A_2 \mathbf{e}\mathbf{e}^T - l\mathbf{S}_\theta \mathbf{K} \tag{24}$$

$$\mathbf{K_{Dy}} = l\mu_n m_{12} A_1 \mathbf{e}\mathbf{e}^T - l\mu_n m_{22} A_2 \mathbf{e}\mathbf{e}^T + l\mathbf{C}_\theta \mathbf{K} \tag{25}$$

where $\mathbf{K}_1 = \mathbf{A}_1 + \mu_n \mathbf{A}_1 \mathbf{e}\mathbf{e}^T (m_{12}\mathbf{S}_\theta \mathbf{C}_\theta - m_{11}\mathbf{S}_\theta^2) - \mu_n \mathbf{A}_2 \mathbf{e}\mathbf{e}^T (m_{22}\mathbf{S}_\theta \mathbf{C}_\theta - m_{21}\mathbf{S}_\theta^2)$, $\mathbf{K}_2 = \mathbf{A}_2 - \mu_n \mathbf{A}_1 \mathbf{e}\mathbf{e}^T (m_{11}\mathbf{S}_\theta \mathbf{C}_\theta - m_{12}\mathbf{C}_\theta^2) + \mu_n \mathbf{A}_2 \mathbf{e}\mathbf{e}^T (m_{21}\mathbf{S}_\theta \mathbf{C}_\theta - m_{22}\mathbf{C}_\theta^2)$, $\mathbf{A}_1 = \mathbf{S}_\theta \mathbf{K} \mathbf{S}_\theta^2 + \mathbf{C}_\theta \mathbf{K} \mathbf{S}_\theta \mathbf{C}_\theta$, $\mathbf{A}_2 = \mathbf{S}_\theta \mathbf{K} \mathbf{S}_\theta \mathbf{C}_\theta + \mathbf{C}_\theta \mathbf{K} \mathbf{C}_\theta^2$.

Remark 1: The model (16,20) has been adjusted compared to the model in [6] by redefining the expression of the link accelerations as in (6) in order to avoid a singularity issue of the model presented in [6].

In summary, the equations of motion for the underwater snake robot are given by (16) and (20). By introducing the state variable $\mathbf{x} = \left[\theta^T, \mathbf{p}_{CM}^T, \dot{\theta}^T, \dot{\mathbf{p}}_{CM}^T\right]^T \in \mathbb{R}^{2n+4}$, we can rewrite the model of the robot compactly in state space form as

$$\dot{\mathbf{x}} = \left[\dot{\theta}^T, \dot{\mathbf{p}}_{CM}^T, \ddot{\theta}^T, \ddot{\mathbf{p}}_{CM}^T\right]^T = \mathbf{F}(\mathbf{x}, \mathbf{u}) \tag{26}$$

where the elements of $\mathbf{F}(\mathbf{x}, \mathbf{u})$ are found by solving (16) and (20) for $\ddot{\mathbf{p}}_{CM}$ and $\ddot{\theta}$, respectively.

III. Motion planning algorithm

A. Artificial Potential Field

In this subsection, the artificial potential field (APF) method is applied to the obstacle avoidance problem of an underwater snake robot. The design of the path is an important step for the control of the robot. The objective is to determine the optimal path that should be followed in order to reach the goal of the mission while avoiding obstacles. Generally, path planners are divided into two categories; local path planners and global path planners [21]. In this study the artificial potential field method, which is introduced by [11], is used to derive the path for the underwater snake robot. In APF the robot moves in an area with a potential created by the obstacles and the target [11], [21]. The obstacles repel the robot, while the target attracts it. The potential functions are chosen so that the target position

is a global minimum while the obstacles are maxima of the function. Denoting the repulsive potential from the $i-$th obstacle by U_r^i, $i = 1,...,o$, where o is the number of the obstacles, and the attractive potential from the goal position by U_a, the potential of the area can be expressed as [21]

$$U = U_a + \sum_{i=1}^{o} U_r^i. \tag{27}$$

It should be noted that, generally, in APF theory the obstacles are considered as points, without specific volume. However, it is obvious that the volume of the obstacles and the volume of the robot should be considered in order to derive the function for the potential. In this paper, we follow a similar approach as described in [21], where the volume of the obstacles and the robot is set as the volume of a circle around the obstacles. Hence, the repulsive potential function is defined as [11]

$$U_r^i = K_{rep} \frac{1}{d_o^{iN} - a_r^N}, \tag{28}$$

where K_{rep} is a positive gain constant which determines the intensity of the attractive potential, the parameter d_o^i denotes the distance to the $i-$th obstacle, and a_r is the radius of the circle around the obstacle taking into account also the dimensions of the robot. The radius of the circle that takes into account the dimension of the robot should be at least twice the amplitude of the sinusoidal motion of the robot. The parameter N is chosen equal to 2 and K_{rep} is chosen close to 1 in order to take into account only the local influence of the repulsive potential. The attractive potential function is defined as [11]

$$U_a = -K_{att} \frac{1}{d^N}. \tag{29}$$

where K_{att} is a positive gain constant which determines the intensity of the attractive potential, the parameter d denotes the distance to the goal and N is chosen equal to 2 in this study. It is worth to mention that increasing the parameter N will lead to a steeper curve of the attractive potential function. A gradient-based optimization method is used to derive the path [21]. In addition, in this paper, we decide to increase the radius of the circle around the robot with a small value, in order to take into account the joint constraints of the robot, cf. [11]. When following the resulting path, the robot is moving towards the global minimum of the potential function (27).

A simulation result of the described motion planning method is presented below for a space ($8\text{m} \times 8\text{m}$) with obstacles (obstacle positions : $\{(1,1)\,(3,-2)\,(1,4)\,(4,2)\,(4,5)\}$) and a selected target (target position: $(7,5)$). The obstacles are rectangular shapes with dimensions (0.3×0.21). The simulation results are performed with the parameter $K_{rep} = 1$ giving the radius of the circumscribed circle around the obstacles and $K_{att} = 200$. Fig. 2a shows the described scenario, while in Fig. 2b the resulting potential is shown. Fig. 2c shows the resulting path that is created based on the obstacles and target scenario presented in Fig. 2a.

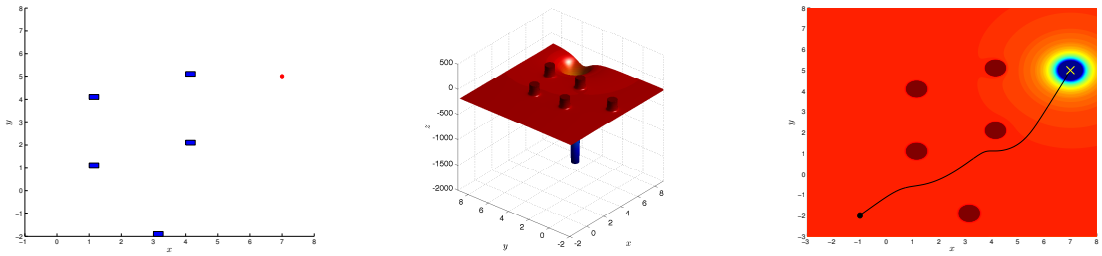
Remark 2: It should be noted that the disadvantage of this method is that there is a risk that the robot in some cases can be stuck at a local minimum. This problem can be solved using harmonic potential field as presented in [21], but in this study the typical non-harmonic potential method is chosen to avoid adding more complexity in our system.

B. Waypoint guidance

Future applications of snake robots will generally involve motion in challenging and unstructured environments [1]. It is worth to note that compared to traditional snake robots the underwater snake robots have the advantage of adaptability to aquatic environments. The most recent fields of interest include the integration of snake robots into underwater exploration, monitoring, and surveillance [6]. This brings the need for steering the robot to a specific target location(s) avoiding obstacles in the plane of motion. In these situations, it is generally less important to follow an exact curved path as long as the robot reaches the target(s) within a reasonable amount of time, avoiding obstacles in the environment. Waypoint Guidance (WPG) is the most widely used scheme in the field of autonomous underwater vehicles (AUVs) (see e.g. [7]), but has, to the authors' best knowledge, not been considered for motion control of underwater snake robots. It was shown in [20] how waypoint guidance can be used for ground snake robots, and based on these results we here propose a waypoint guidance strategy for underwater snake robots. The first version of the desired path is derived by the AFP as described in Section III.A.

Waypoint guidance can then be achieved between the start and end points of this path, $[x_d(t_o), y_d(t_o)]$ and $[x_d(t_f), y_d(t_f)]$, by splitting the path between the points into a number of waypoints $[x_d(k), y_d(k)]$ for $k = 1,2,...,N_w$, with N_w being the number of the waypoints. The waypoint guidance system switches from one waypoint to the next when the underwater snake robot reaches the vicinity of the current waypoint, i.e. when it comes within a circle of acceptance with radius r_{accept} of the current waypoint $[x_d(k), y_d(k)]$ (see e.g. [7]). The circle of acceptance is typically chosen as two times the length of the vehicle [7]. Note that the distance between the waypoints on the path derived by the path planner using APF should be sufficiently small in order to keep the straight line path approximation as close to the original path that the collision avoidance properties are kept. A disadvantage of the waypoint guidance method is that undesirable control energy consumption due to overshoot can occur during the change from one straight line path to the next. Therefore, selection of the reference path to follow is important to reduce the overshoot width of path and thus to decrease the control energy consumption. [22] employs turning simulations to determine modified waypoints to avoid overshoot. Waypoint guidance, and specifically Line-of-Sight (LOS) guidance is a key feature in the majority of guidance systems for marine vehicles [7], and we believe that this is a promising approach also for swimming snake robots.

In this paper, an acceptance region is used, instead of the common approach of acceptance circle (see e.g. [20]). By this definition, it is guaranteed that the robot will reach the acceptance region of the current waypoint no matter how the waypoints are defined. The reason is that using an acceptance circle instead gives the risk that the robot misses a waypoint which is placed too close to the previous waypoint.



(a) Space with obstacles and target (b) The potential of the defined space (c) Potential of the space and the derived path

Fig. 2: Derived path for obstacle avoidance purposes using artificial potential field

IV. Path following control along straight lines

In this section a path following controller is proposed for lateral undulation and eel-like motion. The path following control objective that is considered for the underwater snake robot is to make the robot converge to the desired straight line path and subsequently progress along the path at some nonzero forward velocity. The authors consider it as less important to accurately control the forward velocity of the robot. The path following controller of the robot, which is proposed in this paper, consists of two main components [23]. The first component is the gait pattern controller, which propels the robot forward according to the gait pattern lateral undulation or eel-like motion. The second component is the heading controller, which steers the snake robot towards and subsequently along the desired path. An inner-loop PD controller is used to compensate the effects of the snake's dynamics, while an outer loop controller is used for the formation of the reference joint angles by tracking the desired heading (Fig. 3). The two components of the path following controller are now presented.

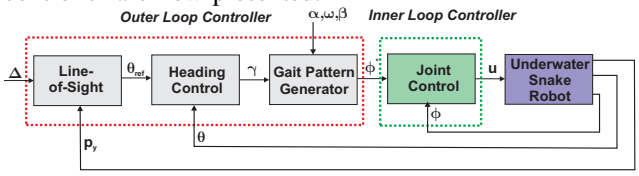


Fig. 3: Controller Structure

A. Lateral undulation

The gait pattern lateral undulation is the fastest and most common form of snake locomotion [1]. In order to achieve lateral undulation, the snake is commanded to follow the serpenoid curve [3]. The proposed lateral undulation is realized by controlling each joint of the snake robot according to the sinusoidal reference

$$\phi_i^* = \alpha \sin(\omega t + (i-1)\beta) + \gamma, \quad i = 1, \ldots, n-1, \tag{30}$$

where the parameters α and ω correspond to the amplitude and angular frequency of the sinusoidal joint motion, β determines the phase shift between the sequential joints, and γ is the joint offset.

B. Eel-like motion

Eel-like motion is achieved by propagating lateral axial undulations with increasing amplitude from nose to tail [6]. A simple equation is derived for the eel-like motion by controlling each joint of the snake robot according to the reference signal

$$\phi_i^* = \alpha \left(\frac{n-i}{n+1} \right) \sin(\omega t + (i-1)\beta) + \gamma, \quad i = 1, \ldots, n-1, \tag{31}$$

where the parameter $\alpha(n-i)/(n+1)$ corresponds to the increasing amplitude, from nose to tail [6].

C. Outer-Loop Controller

It is well-known that the parameters α and β are typically fixed and the parameters ω, γ are used to control the speed and the direction of the snake robot [23]. In this paper, the same idea will be used in order to steer the underwater snake robot to a desired orientation. In particular, the outer-loop controller will be responsible for generating the reference joints angles in order to ensure that the desired orientation is achieved. The orientation $\bar{\theta}$ of the robot is given by Eq. (1). Motivated by [20], [24] we propose to define the reference orientation using the following LOS guidance law

$$\bar{\theta}_{\text{ref}} = -\arctan\left(\frac{p_y}{\Delta} \right), \tag{32}$$

where p_y is the cross-track error (i.e., the position of the robot along the global y axis), and Δ is a design parameter referred to as the *look-ahead distance*. Note that LOS guidance law is much used in practice for path following control of marine surface vessels [7], [25].

Remark 3: The *look-ahead distance* Δ is an important design parameter that directly influences the transient motion of the underwater snake robot. Choosing Δ large results in a well-damped transient motion, but the rate of convergence to the path will be slow. On the other hand, choosing Δ too small results in poor performance or even instability. A rule of thumb is to choose Δ larger than twice the length of the robot (see e.g. [7]).

As we already mentioned, using the parameter γ, it is possible to control the direction of the locomotion of the snake robot. To steer the heading $\bar{\theta}$ according to the LOS angle in (32), we set the joint angle offset according to

$$\gamma = k_\theta \left(\bar{\theta} - \bar{\theta}_{\text{ref}} \right), \tag{33}$$

where $k_\theta > 0$ is a control gain [1].

Remark 4: It should be noted that this traditional LOS guidance is not designed to handle the current effect or any other environmental disturbances such as wind or waves. If the current effect has a component acting in the direction perpendicular to the path, the underwater snake robot, can not stay identically on the path with zero heading. This is a drawback of traditional LOS guidance laws that results in deviation problems when using LOS control in the presence of current. One approach that has given good results for marine vehicles, is to allow the vehicle to side-slip such that a component of the forward velocity of the robot can counteract the effect of the current effect [7].

D. Inner-Loop Controller

In order to make the joint angle ϕ_i follow its reference signal ϕ_i^*, a PD controller is used:

$$u_i = \ddot{\phi}_i^* + k_d(\dot{\phi}_i^* - \dot{\phi}_i) + k_p(\phi_i^* - \phi_i), \quad i = 1,\ldots,n-1, \tag{34}$$

where $k_p > 0$ and $k_d > 0$ are the gains of the controller. This controller exponentially stabilizes the origin of the joint error dynamics.

V. Simulation Study

This section presents simulation results in order to investigate the performance of the proposed waypoint guidance strategy described in Section III-IV. The dynamics was calculated using the ode23tb solver in Matlab R2011b with a relative and absolute error tolerance of 10^{-4}.

A. Implementation of guidance strategy

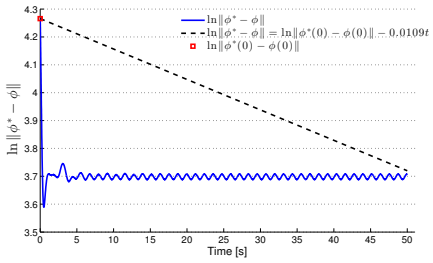
The efficacy of the proposed scheme has been examined through three different simulation studies; case 1 – the straight line path following controller, case 2 – the waypoint guidance strategy for a priori defined points and case 3 – the waypoint guidance strategy with obstacle avoidance. An underwater snake robot was considered with $n = 10$ links, each one having length $2l = 0.14$ m and mass $m = 0.6597$ kg. The initial values of all states of the robot were set to zero for case 2, while the value for the initial position of the CM is selected as $p_{CM}(0) = [0,1]$ in case 1 and $p_{CM}(0) = [-1,-2]$ in case 3.

The hydrodynamic related parameters are set to $c_t = 0.2639$, $c_n = 8.4$, $\mu_n = 0.3958$, $\lambda_1 = 2.298810^{-7}$, $\lambda_2 = 4.310310^{-4}$ and $\lambda_3 = 2.262910^{-5}$. An extensive discussion about the values of the fluid parameters can be found in [6]. We do not take into account currents in the simulation, as handling currents will be a topic of future work. In addition, the radius of the acceptance circle (i.e. acceptance circle being a subset of the acceptance region) enclosing each waypoint is $r_{accept} = 0.8$. The joint PD controller (34) is used for each joint with parameters $k_p = 20$, $k_d = 5$, and lateral undulation or eel-like motion are achieved by moving the joints according to (30) or (31), respectively, with gait parameters $a = 30^o$, $\beta = 40^o$ and $\omega = 150^o/s$. Furthermore, the parameter $k_\theta = 0.3$ and the parameter $\Delta = 1.4$ m.

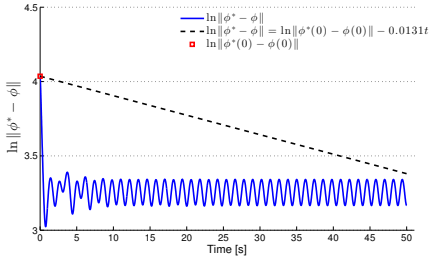
B. Simulation results

1) **Case 1***: The straight line path following controller:* First, we tested the straight line path following controller proposed in Section IV. In Fig. 4 we see that the joint angles converge exponentially to their reference values. Furthermore, we see in Fig. 5a and 6a that (33) together with (30) and (31), respectively, makes the heading angle converge to the desired heading angle given by (32). Moreover, Fig. 5b and 6b show that the hypothesis that the LOS guidance law (32) will make the cross track error oscillate around zero is verified. An *xy*-plot illustrating the behavior of the underwater snake robot for lateral undulation and eel-like motion is shown in Fig. 5c and 6c, respectively.

2) **Case 2***: The waypoint guidance strategy for a priori defined waypoints:* The proposed guidance strategy is then tested when having chosen a set of waypoints a priori. The chosen waypoints in global frame coordinates are (0 0), (3 0), (6 3), (6 8) and (0 8), respectively. The heading of the robot is presented in the Fig. 7a, Fig. 8a and Fig. 7b, Fig. 8b show



(a) Lateral Undulation



(b) Eel-like motion

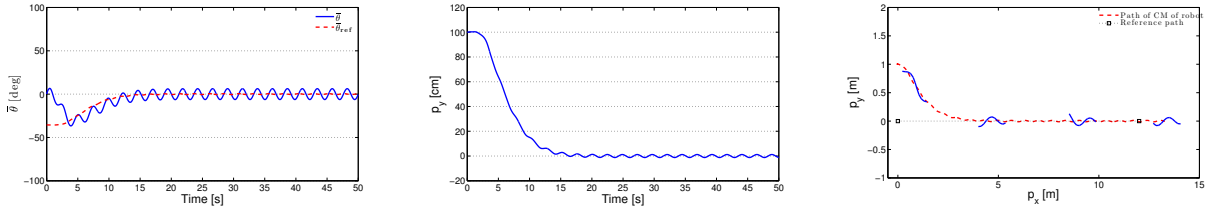
Fig. 4: Convergence of joint angles

the cross-track error. From Fig. 7c and Fig. 8c we see that the robot achieves a smooth path towards each waypoint, and from Fig. 7a and Fig. 8a we can see that the robot is able to follow the direction of the desired path. Fig. 7b and Fig. 8b show the transient behavior discussed in Section IV.C, and show that the cross-track error oscillates around zero after each waypoint switching both for lateral undulation and eel-like motion, respectively. Fig. 7c and Fig. 8c show the motion of the CM of the underwater snake robot.

3) **Case 3***: The waypoint guidance strategy for obstacle avoidance using APF:* Finally, the proposed control strategy is tested in the case of an obstacle avoidance scenario described in Section III.A. The heading of the robot is presented in the Fig. 9a, 10a, 11a and 12a. It may be noted that the cross-track error does not converge to zero (Figs 9b, 10b, 11b and 12b) in all cases, because the waypoints are really close to each other, so there is not enough time to achieve zero convergence. In Fig. 9c, Fig. 10c, Fig. 11c and Fig. 12c we can see that the robot manages to follow the desired path, derived using artificial potential field for obstacle avoidance purposes. As we can see from Fig. 9b, 10b and Fig. 11b, 12b, with larger subsampling step, we achieve better convergence results in cross-track error. This illustrates, as discussed in Section III.B., that a larger step in the subsampling process can give better results in cross-track error, but as the step is chosen larger the deviation between the original path and the straight line path becomes larger, and thus the risk for the robot to collide with an object increases. The important point is that, even if the cross-track error does not converge exactly to zero, the heading of the robot oscillates around zero (Fig. 9a, 10a, 11a and 12a) and the robot manages to follow, with almost zero error, the desired path without colliding with any obstacle.

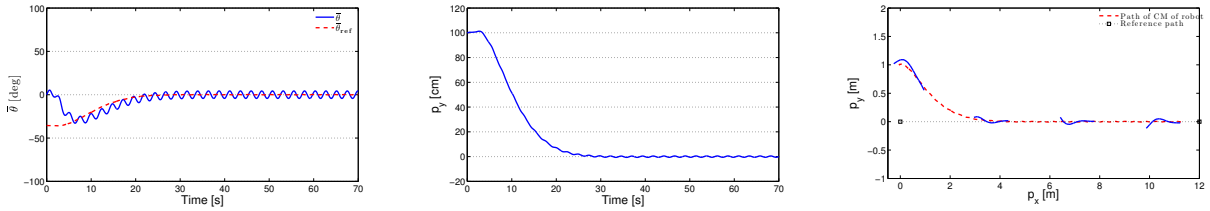
VI. Conclusions

In this paper preliminary results for path following control of underwater snake robots were presented. In particular, a waypoint guidance control strategy was proposed: A motion



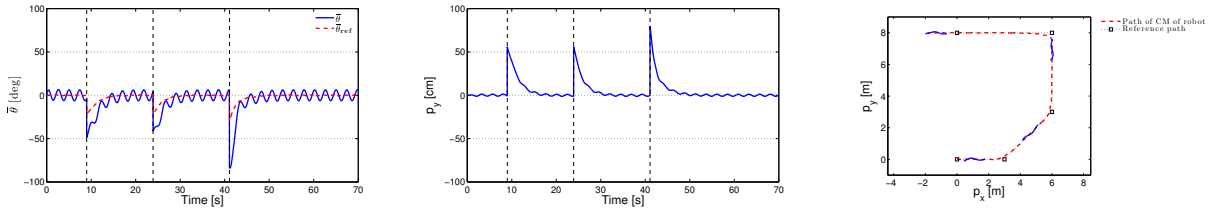
(a) Heading angle (b) Cross track error (c) Path of the robot

Fig. 5: Straight line path following controller for lateral undulation



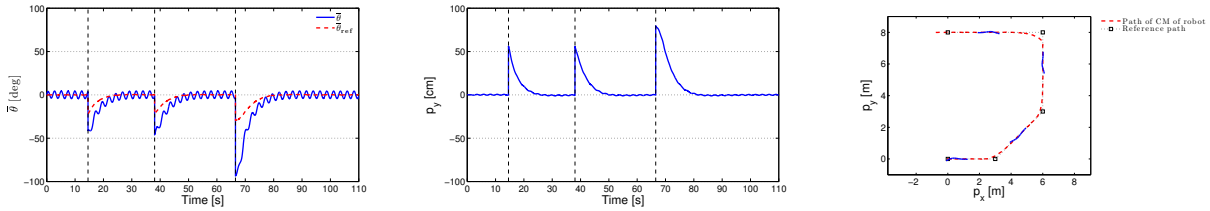
(a) Heading angle (b) Cross track error (c) Path of the robot

Fig. 6: Straight line path following controller for eel-like motion



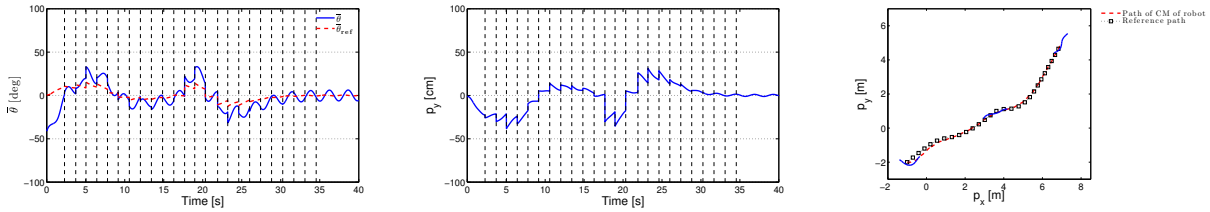
(a) Heading angle (b) Cross track error (c) Path of the robot

Fig. 7: Path following of underwater snake robot during waypoint guidance for lateral undulation



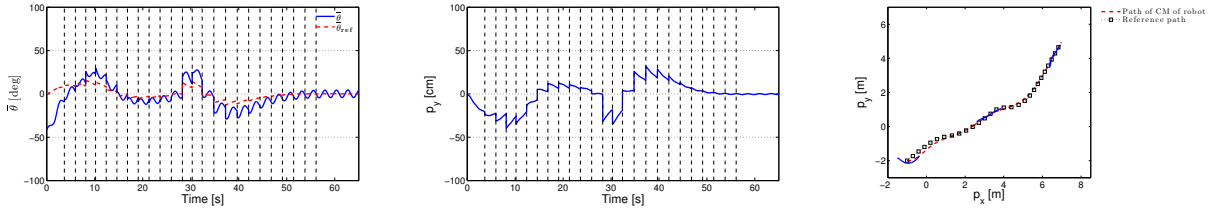
(a) Heading angle (b) Cross track error (c) Path of the robot

Fig. 8: Path following of underwater snake robot during waypoint guidance for eel-like motion



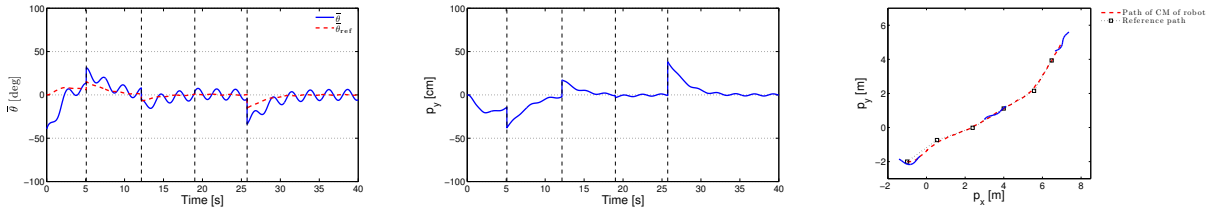
(a) Heading angle (b) Cross track error (c) Path of the robot

Fig. 9: Path following of underwater snake robot during waypoint guidance for obstacle avoidance for lateral undulation: subsampling every 40 points –waypoints distance approximately 0.4 m



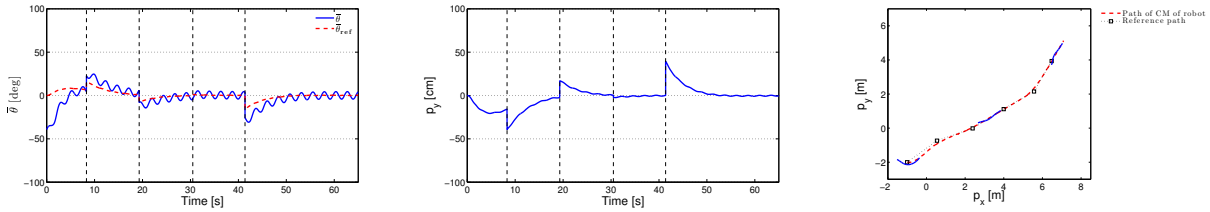
(a) Heading angle (b) Cross track error (c) Path of the robot

Fig. 10: Path following of underwater snake robot during waypoint guidance for obstacle avoidance for eel-like motion: subsampling every 40 points –waypoints distance approximately 0.4 m



(a) Heading angle (b) Cross track error (c) Path of the robot

Fig. 11: Path following of underwater snake robot during waypoint guidance for obstacle avoidance for lateral undulation: subsampling every 200 points–waypoints distance approximately 2 m



(a) Heading angle (b) Cross track error (c) Path of the robot

Fig. 12: Path following of underwater snake robot during waypoint guidance for obstacle avoidance for eel-like motion: subsampling every 200 points–waypoints distance approximately 2 m

planning algorithm was proposed for deriving reference paths based on APF techniques, in order to achieve collision free paths. The path was then split into linear path segments defined by waypoints. Finally, a LOS guidance law was proposed to make the underwater snake robot converge to the straight line paths. Simulation results were presented, illustrating the successful integration of the aforementioned strategy. It remains a topic of future work to investigate the stability properties of the proposed path following controller.

REFERENCES

[1] P. Liljebäck, K. Y. Pettersen, Ø. Stavdahl, and J. T. Gravdahl, *Snake Robots: Modelling, Mechatronics, and Control*. Springer-Verlag, Advances in Industrial Control, 2013.

[2] J. Gray, "Studies in animal locomotion," *Journal of Experimental Biology*, vol. 10, no. 1, pp. 88–104, 1933.

[3] S. Hirose, *Biologically Inspired Robots: Snake-Like Locomotors and Manipulators*. Oxford: Oxford University Press, 1993.

[4] K. McIsaac and J. Ostrowski, "Motion planning for anguilliform locomotion," *IEEE Transactions on Robotics and Automation*, vol. 19, no. 4, pp. 637–625, 2003.

[5] F. Boyer, M. Porez, and W. Khalil, "Macro-continuous computed torque algorithm for a three-dimensional eel-like robot," *IEEE Transactions on Robotics*, vol. 22, no. 4, pp. 763 –775, 2006.

[6] E. Kelasidi, K. Y. Pettersen, J. T. Gravdahl, and P. Liljebäck, "Modelling of underwater snake robots," in *Proc. IEEE International Conference on Robotics and Automation (ICRA)*, Hong Kong, China, May 31 - June 7 2014, accepted. [Online]. Available: https://www.dropbox.com/s/4v5qk9shui28jl7/ICRA140679.pdf

[7] T. I. Fossen, *Handbook of Marine Craft Hydrodynamics and Motion Control*. John Wiley & Sons, Ltd, 2011.

[8] S. LaValle, *Planning Algorithms*, Cambridge, Ed. UK: Cambridge University Press, 2006.

[9] H. Choset, K. M. Lynch, S. Hutchinson, G. Kantor, W. Burgard, L. E. Kavraki, and S. Thrun, *Principles of Robot Motion: Theory, Algorithms, and Implementations.* U.K. Cambridge, MIT Press, 2005.

[10] J. Borenstein and Y. Koren, "The vector field histogram-fast obstacle avoidance for mobile robots," *IEEE Transactions on Robotics and Automation*, vol. 7, no. 3, pp. 278–288, 1991.

[11] O. Khatib, "Real-time obstacle avoidance for manipulators and mobile robots," in *Proc. IEEE International Conference on Robotics and Automation (ICRA)*, vol. 2, 1985.

[12] T. Paul, T. Krogstad, and J. Gravdahl, "UAV formation flight using 3D potential field," in *Proc. 16th Mediterranean Conference on Control and Automation (MED)*, Ajaccio-Corsica, France, June 25-27 2008.

[13] H. Igarashi and M. Kakikura, "Path and posture planning for walking robots by artificial potential field method," in *Proc. IEEE International Conference on Robotics and Automation (ICRA)*, vol. 3, New Orleans, LA, USA, April 26-May 1 2004.

[14] M. Porez, V. Lebastard, A. Ijspeert, and F. Boyer, "Multi-physics model of an electric fish-like robot: Numerical aspects and application to obstacle avoidance," in *Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, San Francisco, California, Sept. 25-30 2011.

[15] A. Crespi and A. J. Ijspeert, "AmphiBot II: An Amphibious Snake Robot that Crawls and Swims using a Central Pattern Generator," in *Proc. of the 9th International Conference on Climbing and Walking Robots (CLAWAR 2006)*, Brussels, Belgium, Sept. 12-14 2006.

[16] K. A. McIsaac and J. P. Ostrowski, "A framework for steering dynamic robotic locomotion systems," *Int. J. Robot. Res.*, vol. 22, no. 2, pp. 83–97, February 2003.

[17] J. Guo, "A waypoint-tracking controller for a biomimetic autonomous underwater vehicle," *Ocean Engineering*, vol. 33, pp. 2369 – 2380, 2006.

[18] P. A. Vela, K. A. Morgansen, and J. W. Burdick, "Underwater locomotion from oscillatory shape deformations," in *Proc. IEEE Conf. Decision and Control*, vol. 2, Las Vegas, NV, USA, Dec. 10-13 2002.

[19] K. Morgansen, B. Triplett, and D. 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.

[20] P. Liljebäck and K. Y. Pettersen, "Waypoint guidance control of snake robots," in *Proc. IEEE International Conference on Robotics and Automation (ICRA)*, Shanghai, China, May 9-13 2011.

[21] I. Arvanitakis and A. Tzes, "Trajectory optimization satisfying the robot's kinodynamic constraints for obstacle avoidance," in *Proc. 20th Mediterranean Conference on Control Automation (MED)*, Barcelona, Spain, July 3-6 2012.

[22] D. Yeo, "Design of AUV tracking system using the sliding mode control and the optimal control theory," Ph.D. dissertation, MS Thesis, Department of Naval Architecture and Ocean Engineering, Seoul National University, Korea, 1999.

[23] E. Kelasidi and A. Tzes, "Serpentine motion control of snake robots for curvature and heading based trajectory - parameterization," in *Proc. 20th Mediterranean Conference on Control Automation (MED)*, Barcelona, Spain, July 3-6 2012.

[24] P. Liljebäck, I. U. Haugstuen, and K. Y. Pettersen, "Path following control of planar snake robots using a cascaded approach," *IEEE Transactions on Control Systems Technology*, vol. 20, no. 1, pp. 111–126, 2012.

[25] E. Fredriksen and K. Y. Pettersen, "Global κ-exponential way-point maneuvering of ships: Theory and experiments," *Automatica*, vol. 42, pp. 677 – 687, 2006.