Set-based path following and obstacle avoidance for underwater snake robots

A. M. Kohl, S. Moe, E. Kelasidi, K. Y. Pettersen, and J. T. Gravdahl

Abstract— **This article presents a set-based guidance strategy for path following with obstacle avoidance for underwater snake robots conducting planar sinusoidal motion. The guidance scheme is designed such that the robot follows a straight path and avoids obstacles on the way by following a circle around them. In order to enable a switching between these two tasks, we generalize a strategy that was introduced for surface vessels, by making the switching condition independent of the dynamics and thus applicable for a larger class of systems. The guidance system is shown to fulfil the control objectives at a kinematic level. We present new test results that validate the set-based guidance scheme for the first time experimentally.**

I. Introduction

Increasing the autonomy of the technology has a large potential for making subsea operations safer, more efficient, cost-effective, and environmentally friendly. Already today, remotely operated vehicles (ROVs) [1] are replacing divers in subsea inspection, maintenance, and repair (IMR) operations. Employing fully autonomous underwater vehicles (AUVs) in addition or instead, has the potential to further improve the safety and cost-effectiveness of such operations. Underwater snake robots, which mimic the motion of eels, can be considered as self-propelled, hyperredundant manipulator arms. These robots thus provide a promising novel solution for subsea IMR operations. In particular, they provide a high transport range similar to survey AUVS, excellent access capabilities, similar to observation ROVs, and light intervention capabilities, like standard ROVs. In addition, the hyperredundant design provides high dexterity for all operations. In order to reach an offshore installation to perform IMR, efficient path following control laws are required.

Path following controllers for underwater snake robots and eel-like robots have previously been proposed in [2–4]. All of these approaches have in common that they consider only straight line path following in an obstacle-free environment. The control system in [5,6], on the other hand, is able to handle general curved paths. It is motivated by a maneuvering control system for terrestrial snake robots, which is shown to practically stabilize the tracking errors in [5], and is experimentally validated in [6]. Non of the approaches in [2–4,6], however, consider obstacle avoidance.

This work was funded by the Research Council of Norway through its Centres of Excellence funding scheme, project no. 223254-NTNU AMOS, and by VISTA - a basic research program in collaboration between The Norwegian Academy of Science and Letters, and Statoil. A. M. Kohl, E. Kelasidi, and K. Y. Pettersen are with the Centre for Autonomous Marine Operations and Systems (NTNU-AMOS), Dep. of Engineering Cybernetics, NTNU*. anna.kohl@itk.ntnu.no. S. Moe and J. T. Gravdahl are with the Dep. of Engineering Cybernetics, NTNU*, *Norwegian University of Science and Technology, NO-7491 Trondheim, Norway.

When following a path through an underwater installation, the snake robot has to be able to avoid obstacles that might interfere with its planned path. Examples of obstacles are other vehicles or stationary installed constructions that block the shortest path between the robot and its goal. Obstacle avoidance strategies for underwater snake robots, or the closely related fish robots, have been presented in [7–9]. The work in [7] presents experimental results for obstacle recognition and avoidance for a fish-like robot using neural networks. The focus of the work in [7] lies mostly on obstacle recognition and avoidance and less on following a predefined path. Similarly, another study in [8] uses an electric exteroceptive feedback loop in a fish-like robot to detect and avoid obstacles. Again, also [8] focuses only on the obstacle avoidance rather than path following. Another approach is presented in [9] for underwater snake robots, where an artificial potential field is used to plan a safe path towards a target within a cluttered environment. While [7,8] consider mainly local obstacle detection, the study in [9] requires a global map of the obstacle locations a priori. For terrestrial snake robots, obstacle avoidance strategies have been considered in [10–12]. In [10], a control strategy for a terrestrial snake robot with passive wheels is presented, where obstacles in the way of the robot are avoided by altering the sinusoidal shape that the robot adapts to. This approach considers local avoidance of obstacles, but no path following strategy is proposed. A similar strategy for moving obstacles is proposed and experimentally validated in [11], where a robot is remotely controlled to move forward and avoids obstacles on the way by lifting links that are about to collide sideways. In [12], a method for both trajectory planning and obstacle avoidance is proposed. However, the method in [12] is for snake robots with active wheels, and thus not applicable for underwater snake robots. There is a lot of literature on obstacle avoidance that is not specific for snake robots, see for instance the reviews on mobile robots [13], unmanned aerial vehicles [14], or autonomous ships [15]. A guidance strategy for autonomous ships was recently proposed in [16]. The strategy is based on the results for set-based control in [17], which facilitate a theoretical stability analysis. It is shown in [16] that the strategy guarantees obstacle avoidance, which is illustrated with simulations, but has not been tested experimentally yet. Compared to the strategy in [9], the approach for surface vessels in [16] has the advantage that it can easily be extended for an unknown environment, if an obstacle detection strategy is added.

The set-based path following and obstacle avoidance guidance that we propose in this paper is therefore motivated by

the set-based approach in [16], which enables a switching between one mode for path following and one mode for obstacle avoidance. The main idea is to make the robot converge to and follow a straight path and only leave it in order to circumvent an obstacle that is in the way. In that case, the guidance switches into obstacle avoidance mode, and the robot follows a circle around the obstacle. As soon as the obstacle is passed, the robot converges to the original straight path again. In order to apply the strategy to a snake robot that propels itself forward using body undulation, the switching conditions are reformulated in a more general, purely kinematic manner. This makes the switching strategy independent of the dynamic model, and thus applicable to a more general class of systems, including snake robots. For a snake robot, considering the dynamics would introduce oscillations to the guidance due to the oscillating nature of snake robot locomotion, and thus make the set-based system vulnerable to chattering. Furthermore, the new switching strategy is combined with a modified guidance law for snake robots from [5,6]. This guidance controller is suitable for generic paths, and can thus be applied both for the straight line path following mode, and the obstacle avoidance mode, which requires a circular reference path. In [5], the guidance law is shown to practically stabilize the robot to the path. In this article, the guidance law is modified in order to allow for circular path following in both directions, thus enabling the robot to choose the shortest way around the obstacles blocking its path. We show that the modifications to the guidance law still preserve the stability properties. Finally, the obstacle avoidance strategy is experimentally tested with a swimming snake robot for different stationary obstacles. The experimental results validate the proposed set-based path following and obstacle avoidance guidance scheme, and are the first test results of the set-based control strategy that have been obtained with a floating-base robot.

This article is organized as follows. In Section II, the kinematic conventions of an underwater snake robot conducting planar motion are presented. In addition, the gait of the robot is introduced, as well as some basic assumptions. The control objectives are formulated in Section III. In Section IV, the set-based path following and obstacle avoidance guidance strategy for underwater snake robots is proposed. The analysis of the guidance strategy is presented in Section V. An experimental study, which validates the approach, is presented in Section VI. Concluding remarks are given in Section VII.

II. Underwater snake robots and preliminaries

This section gives an introduction to snake robots as a basis for the following sections. In particular, the kinematics of the robot is briefly presented, and the propulsion method is introduced along with some basic assumptions.

A. Kinematics

In this article we consider neutrally buoyant underwater snake robots, which move in a plane, i. e. swimming snake robots that have no joint rotation in pitch. Such robots

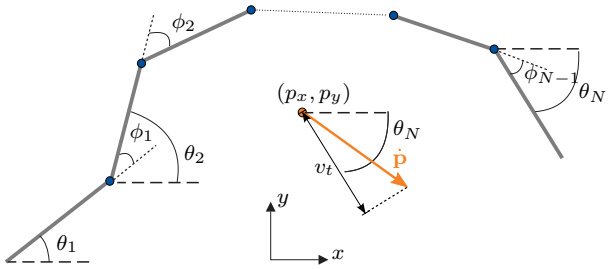


Fig. 1. Model conventions of an underwater snake robot.

can be considered a kinematic chain of N links that are interconnected with $N-1$ rotational joints. Each link has the same mass and length. In this article we define the position $\mathbf{p} = [p_x, p_y]^T$ of the robot as the position of its center of mass (CM), and the heading of the robot as the orientation of its head link, θ_N. Choosing the orientation of the head link as the controlled heading state instead of the mean orientation [18] or the orientation of a virtual chassis [19,20] has a significant advantage. It enables a combination of the control strategy with robot vision in the future, since cameras and other sensors would typically be placed in the head link. The angles of the individual links w. r. t. the global frame are given by $\theta_i, i = 1, \ldots, N$, and the angles between two adjacent links, the joint angles, are given by $\phi_i, i = 1, \ldots, N-1$. Fig. 1 shows such a robot with the corresponding angular conventions. Furthermore, the forward velocity v_t of the robot is defined as the velocity component that is aligned with the heading $v_t = \dot{p}_x \cos(\theta_N) + \dot{p}_y \sin(\theta_N)$.

Remark 1: A dynamic model of such robots, which takes into account the hydrodynamic effects acting on the robot, can for instance be found in [21].

B. Forward propulsion

Forward propulsion of terrestrial snake robots is often achieved by lateral undulation, which is the most common gait of biological snakes. It is achieved by propagating a sine wave along the body from head to tail. The underlying principle is anisotropic skin friction, where the scales of the snake skin create a larger friction sideways than along the body, which enables the snake to push itself forwards through body undulation [18]. A snake robot conducts lateral undulation by making each joint track the reference signal

$$\phi_i = \alpha \sin(\omega t + (i-1)\delta) + \phi_0, \quad i = 1, \ldots, N-1, \quad (1)$$

where α is the amplitude and ω the frequency of the sine wave, δ is the phase shift between adjacent joints, and ϕ_0 is an offset. For underwater snake robots, lateral undulation is a special case of a more general class of sinusoidal gaits mimicking the motion of eels [22], and is also a widely used way of propulsion. In this case, the anisotropy of the external forces is induced by a higher drag parameter in the sideways direction than along the body. The velocity of the robot is determined by α, ω, and δ, whereas the offset ϕ_0 induces turning motion. In this article, we will choose some positive constants α, ω, and δ to achieve sufficient forward velocity, and consider the offset ϕ_0 as an input for

the heading feedback controller. More details on the general sinusoidal gait for underwater snake robots are given in [22].

C. Basic assumptions

Assumption 1: The underwater snake robot is neutrally buoyant and moves in a plane according to (1), with a forward velocity $v_t > 0$.

Remark 2: It has been shown that lateral undulation according to (1) results in a forward velocity [18,23].

Assumption 2: The robot is not exposed to ocean currents.

Assumption 3: The robot moves within an environment containing k stationary obstacles. The single obstacles are located sufficiently far away from each other so that the robot can travel safely between them.

Remark 3: Assumption 3 implies that only one obstacle at a time has to be accounted for by the guidance system.

III. Control Objectives

The problem of following a path whilst avoiding obstacles along the way can be considered as a multiple-task control problem, where the path following objective defines one task, and the obstacle avoidance objective defines another, higher prioritized task. The snake robot is supposed to follow a straight path while avoiding k stationary obstacles on the way. This is achieved by ensuring that the robot always stays outside a circle with obstacle j located in the circle center, and a safe radius $r_{\mathrm{s},j}$. The tasks of path following and simultaneous obstacle avoidance can be in conflict with each other if an obstacle is on or close to the path. In this case, we will prioritize the obstacle avoidance task over the path following task in order to ensure a safe operation. Thus, we can formalize the above considerations in the following prioritized control objectives:

Objective 1: The distance between the CM of the snake robot (p_x, p_y) and the center of every obstacle j, $(p_{\mathrm{o},jx}, p_{\mathrm{o},jy})$, should always be greater than or equal to some safe radius $r_{\mathrm{s},j}$: $\sqrt{(p_x - p_{\mathrm{o},jx})^2 + (p_y - p_{\mathrm{o},jy})^2} \geq r_{\mathrm{s},j}$, $j = 1, \ldots, k$.

Remark 4: As opposed to conventionally propelled marine vehicles, an underwater snake robot changes its configuration continuously. It can therefore not be guaranteed that keeping the CM outside of the obstacle will at all times prevent collisions. We therefore propose to define the safe radius r_{s} as the maximal extent of the obstacle r_{o} plus half the snake length $r_{\mathrm{s}} = r_{\mathrm{o}} + \frac{Nl}{2}$.

Objective 2: The robot should, without loss of generality, converge to and follow the global y-axis: $\lim_{t\to\infty} \|p_y(t)\| = 0$. The difference between the tasks is that the path following task constitutes an equality task, whereas the obstacle avoidance task is set-based, which can be seen from the inequality in Objective 1.

IV. Set-based guidance for path following and obstacle avoidance

In [17], a singularity-robust multiple task-priority inverse kinematics framework was proposed, which can handle both equality and set-based tasks. The framework from [17] was later used for collision avoidance of underactuated surface vessels in [16]. In this section, we will generalize the framework from [16], by making the conditions that switch between the single tasks independent of the dynamic model and thus applicable for other systems as well. In order to make it suitable for snake robots, the new set-based switching strategy is combined with a general path following guidance for snake robots. The guidance method provides both the option to follow a straight reference path, which is well-suited to for the path following task, and a circular reference path, which can be used for the obstacle avoidance task. In the following, we will derive the strategy for one obstacle for simplicity. Note that it is straightforward to extend it to k obstacles by repeating the strategy k times, once for each obstacle, as long as the obstacles are not overlapping.

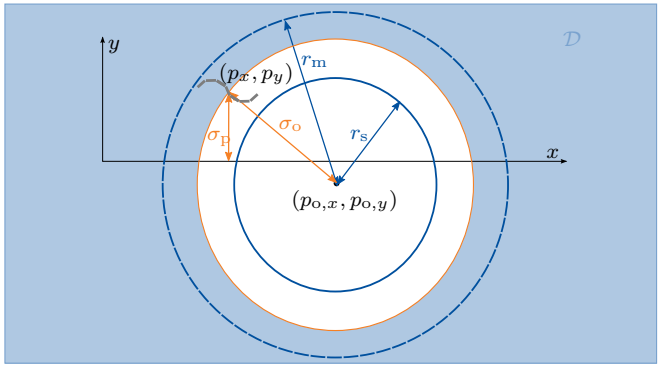


Fig. 2. Definitions of the set-based framework.

A. Definitions

As a preparation for the set-based guidance scheme that will be introduced in Sections IV-B and IV-C, we make the following definitions:

Definition 1: The path following mode is defined as Mode 1, and the obstacle avoidance mode as Mode 2.

Definition 2: The set-based obstacle avoidance task σ_{o} is defined as the distance between the robot and the obstacle center $\sigma_{\mathrm{o}} = \sqrt{(p_x - p_{\mathrm{o},x})^2 + (p_y - p_{\mathrm{o},y})^2}$, and the path following equality task σ_{p} as the distance between the robot and the path $\sigma_{\mathrm{p}} = p_y$.

The key idea of the framework [17], that we will base the guidance strategy on, is to include set-based tasks into the multiple task control framework by freezing the set-based task, thus turning it into an equality task, if it is about to leave its valid set $\mathcal{D}$.

Definition 3: The valid set for σ_{o} is $\mathcal{D} = [\sigma_{\min}, \infty)$ with $\sigma_{\min} = \min(r_{\mathrm{m}}, \max(\sigma_{\mathrm{o}}, r_{\mathrm{s}}))$, and the tangent cone to $\mathcal{D}$ is

$$T_{\mathcal{D}}(\sigma_{\mathrm{o}}) = \begin{cases} \mathbb{R}_o^+, & \sigma_{\mathrm{o}} = \sigma_{\min} \\ \mathbb{R}, & \sigma_{\mathrm{o}} > \sigma_{\min} \end{cases}. \tag{2}$$

The tangent cone $T_{\mathcal{D}}(\sigma_{\mathrm{o}})$ is used as an indicator of whether the task is about to leave the set $\mathcal{D}$: this is the case if the task derivative $\dot{\sigma}_{\mathrm{o}} \notin T_{\mathcal{D}}(\sigma_{\mathrm{o}})$.

It was pointed out in [16] that the framework from [17] has to be adapted when used with underactuated dynamic systems, which can be achieved by introducing a mode change radius $r_{\mathrm{m}} > r_{\mathrm{s}}$. Within the circle with radius r_{m},

switching from path following to obstacle avoidance mode can be activated if certain conditions are satisfied. These conditions will be discussed in Section IV-C.

Definition 4: The radius within which obstacle avoidance can be active is called the mode change radius r_{m}.

As soon as these conditions are no longer valid, the control system switches back to the default path following mode. The above definitions are illustrated in Fig. 2. In order to ensure that both control objectives are fulfilled, we make an additional assumption:

Assumption 4: The mode change radius r_{m} is chosen sufficiently large for the robot to converge to steady state in obstacle avoidance mode without overshooting.

B. The guidance law

The general formulation of a reference velocity vector $\boldsymbol{\mu} = [\mu_x, \mu_y]^T$ for the robot was proposed for terrestrial snake robots in [5] as

$$\boldsymbol{\mu}(\mathbf{p}) = -\frac{dh_p^T}{\|dh_p\|^2}\big(k_{\mathrm{tran}} h(\mathbf{p})\big) + \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} dh_p^T \frac{v}{\|dh_p\|}, \quad (3)$$

where $h(\mathbf{p})$ is an error function implicitly defining the desired path, $dh_p^T = \nabla h(\mathbf{p})$ is a vector that is normal to the level sets of h, $k_{\mathrm{tran}} > 0$ is a transversal gain, and $v > 0$ is the desired along-path velocity. It was shown in [6] that (3) can also be used for underwater snake robots, as long as there are no currents, which is in accordance with Assumption 2.

In this article, the guidance law (3) is modified in order to enable a choice of direction:

$$\boldsymbol{\mu}(\mathbf{p}) = -\frac{dh_p^T}{\|dh_p\|^2}\big(k_{\mathrm{tran}} h(\mathbf{p})\big) + \nu \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} dh_p^T \frac{v}{\|dh_p\|},$$
$$\nu = \begin{cases} -1, & \mathrm{Mode} = 2 \ \wedge \ p_{\mathrm{o},y} > 0 \\ +1, & \mathrm{else.} \end{cases} \quad (4)$$

If $p_{\mathrm{o},y} > 0$, the sign of the second term is changed from positive to negative in obstacle avoidance mode, which makes the robot follow the circle counter-clockwise instead of clockwise, thus always choosing the shortest way around the obstacle. The reference heading θ_{ref} for the robot is then obtained from (4) by the relation

$$\theta_{\mathrm{ref}} = \arctan(\tfrac{\mu_y}{\mu_x}). \quad (5)$$

For Mode 1, the error function $h(\mathbf{p})$ in (4) is simply given by $h_1(\mathbf{p}) = p_y$. In the second case, Mode 2, $h(\mathbf{p})$ is defined as $h_2(\mathbf{p}) = (p_x - p_{\mathrm{o},x})^2 + (p_y - p_{\mathrm{o},y})^2 - r_{\mathrm{s}}^2$, describing a circle with radius r_{s} around the obstacle.

Proposition 1: If the heading reference (5) obtained from the guidance law (4) is tracked by an underwater snake robot in a sufficiently accurate manner, the robot is practically stabilized to the desired path. In particular, for any $\epsilon > 0$, there exists a k_{tran} such that the set $\{h(\mathbf{p}) \leq \epsilon\}$ is asymptotically stable, i. e. the path following error function $h(\mathbf{p})$ stays close to zero.

Proof: The result follows from the proof of Prop. 17 in [5]. It is straightforward to extend the Lyapunov analysis in [5] for the adapted guidance law in (4), because the second part on the RHS of (3) leads to an indefinite term that can be cancelled out by a sufficiently large k_{tran}. The analysis therefore does not depend on the sign of ν. ■

Remark 5: In this article we focus only on the heading control of the robot, which means that we do not actively control the velocity. This is in accordance with Assumption 1, which states that the sinusoidal gait (1) ensures some forward velocity v_t. The desired velocity v in (4) will therefore be treated as a positive control constant.

C. Switching conditions

In the following we propose a strategy for switching between the two modes of the system that were presented in the previous paragraphs. In contrast to the condition in [16], the switching condition in this article is independent of the vehicle dynamics. This is necessary because snake robots moving according to (1) display an oscillating behaviour. Using the dynamical model of such a robot in the switching conditions would therefore result in a control system that is susceptible to chattering.

The general idea of the strategy is that it is safe to do path following as long as the robot is outside of the mode change radius of the obstacle, $\sigma_o \geq r_{\mathrm{m}}$, and thus the default mode of the system is chosen to be Mode 1. In this case, the equality based task is active, and the set-based task is implicitly satisfied and thus inactive. As soon as the vehicle is inside the mode change radius, i. e. $\sigma_{\mathrm{o}} < r_{\mathrm{m}}$, Mode 1 might no longer be safe. Whether Mode 1 is safe or not can be determined by checking if the reference velocity according to Mode 1 will drive the robot closer towards the obstacle. In this case, the system needs to switch into Mode 2, which means that the set-based task is activated by changing the control objective to $\sigma_o = r_{\mathrm{s}}$ and thus making the robot follow a circle with the safe radius r_{s} around the obstacle, and hence achieving obstacle avoidance. As soon as the guidance according to Mode 1 will take the robot further away from the obstacle, it is safe to switch back to Mode 1, move back to the set-based objective $\sigma_o \geq r_{\mathrm{s}}$, and thus fulfil both control objectives. This strategy can be formalized using the task derivative $\dot{\sigma}_{\mathrm{o}}$ and the set $\mathcal{D}$ with its tangent cone $T_{\mathcal{D}}(\sigma_{\mathrm{o}})$ from Def. 3. The set $\mathcal{D}$ corresponds to the area that is safe for the robot to move towards. If $\dot{\sigma}_{\mathrm{o}} \in T_{\mathcal{D}}(\sigma_{\mathrm{o}})$, the time evolution of the task σ_{o} remains in the set $\mathcal{D}$. If $\dot{\sigma}_{\mathrm{o}} \notin T_{\mathcal{D}}(\sigma_{\mathrm{o}})$, the time evolution of the task σ_{o} moves out of the set $\mathcal{D}$, i. e. the vehicle comes closer to the obstacle, and the guidance switches into Mode 2.

Whether the task derivative $\dot{\sigma}_{\mathrm{o}}$ is in the tangent cone $T_{\mathcal{D}}(\sigma_{\mathrm{o}})$ can be checked by looking at the orientation of the reference velocity vector $\boldsymbol{\mu}$. This relies on the assumption that the controllers will ensure tracking of the heading reference orientation θ_{ref}. How to check if the task derivative is in the tangent cone $\dot{\sigma}_{\mathrm{o}} \in T_{\mathcal{D}}(\sigma_{\mathrm{o}})$ by analysing $\boldsymbol{\mu}$ is visualized in Fig. 3. It is done by comparing the reference angle θ_{ref} to the angle θ_{o}, which is defined as

$$\theta_{\mathrm{o}} = \arctan(\tfrac{p_y - p_{\mathrm{o},y}}{p_x - p_{\mathrm{o},x}}). \quad (6)$$

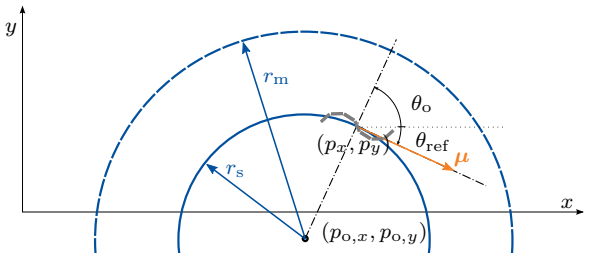


Fig. 3. Checking if the conditions for switching back to Mode 1 are met: if $|\theta_{\mathrm{ref}} - \theta_{\mathrm{o}}| \leq \frac{\pi}{2}$, the vector $\boldsymbol{\mu}$ is safe to track.

The angle θ_{o} is the angle between the global x-axis and the virtual line between the robot and the obstacle. If their absolute values add up to less than $90°$, the vector $\boldsymbol{\mu}$ points away from the obstacle and is safe to track.

Proposition 2: Given the guidance system (4), (5). If the condition $|\theta_{\mathrm{ref}} - \theta_{\mathrm{o}}| \leq \frac{\pi}{2}$ holds for the reference heading θ_{ref}, Mode 1 of the guidance provides a reference that fulfils both Objectives 1 and 2.

Proof: A reference velocity vector with an orientation that fulfils the above condition will increase the distance between the robot and the obstacle, and thus not violate Objective 1. It will furthermore, according to Proposition 1, make the robot converge to the set $\{p_y \leq \epsilon\}$, which can be made arbitrarily small, and thus fulfil also Objective 2. ■

If the reference of the guidance law is tracked exactly by the low level controllers, the switching condition is equivalent to the one in [16].

Remark 6: The task derivative $\dot{\sigma}_{\mathrm{o}}$ depends on the dynamical model of the robot. In this article, we disregard the dynamics of the system in the design of the switching conditions, and base these solely on the kinematics. This choice was made because of the oscillating nature of snake robot locomotion according to (1), which inevitably implies oscillations of the center of mass. Therefore, basing the switching condition on the actual velocity of the robot predicted by a dynamic model would result in chattering between the two modes, as the robot oscillates. The reference velocity, on the other hand, is only based on the kinematics, and is therefore much less exposed to oscillations, and thus a better fit for designing the switching conditions. With the geometric considerations illustrated in Fig. 3, it is possible to determine if $\dot{\sigma}_{\mathrm{o}} \in T_{\mathcal{D}}(\sigma)$, provided that the control system tracks the reference, without explicitly computing $\dot{\sigma}_{\mathrm{o}}$. Note that this approach makes the guidance strategy more general in the sense that it can be used for any type of vehicle, because it is independent of the dynamic model.

D. The switching algorithm

The implementation of the switching strategy from the previous paragraphs is summarized in Algorithm 1. As an extension to k obstacles, the algorithm is run in a for-loop for $i = 1, \ldots, k$, with each iteration checking one obstacle.

Remark 7: The proposed approach is general in the sense that it is independent of the dynamics of the robot. This implies that the proposed guidance for path following with obstacle avoidance can also be applied to terrestrial snake robots without any modifications.

Initialize:
last_mode = path_following;
while *True* **do**
  if $\sigma_o \geq r_{\mathrm{m}}$ **then**
    $h(\mathbf{p}) = h_1(\mathbf{p})$;
    mode = path_following;
  else
    if $|\theta_{\mathrm{ref}} - \theta_{\mathrm{o}}| \leq \frac{\pi}{2}$ **then**
      $h(\mathbf{p}) = h_1(\mathbf{p})$;
      mode = path_following;
    else
      $h(\mathbf{p}) = h_2(\mathbf{p})$;
      mode = obstacle_avoidance;
    end
  end
  last_mode = mode
end

Algorithm 1. The set-based path following and obstacle avoidance guidance scheme for underwater snake robots.

V. Main Result

Theorem 1: Consider a neutrally buoyant underwater snake robot conducting planar motion according to (1). Provided Assumptions 1 to 4 hold and the reference heading given by the set-based guidance algorithm in Algorithm 1 is tracked, Objective 1 is fulfilled. Furthermore, as long as the system is in Mode 1, Objective 2 is also satisfied in the sense that $\lim_{t\to\infty} \|p_y(t)\|$ can be made arbitrarily small.

Proof: According to Proposition 1, the guidance law (4) makes sure that the robot converges to the set $\mathcal{H} = \{h_2(\mathbf{p}) \leq \epsilon\}$ in Mode 2. Because the reference $\boldsymbol{\mu}$ is aligned with the path tangential when the robot is on the path, the robot will converge to a circle with a radius larger than r_{s}, the offset can be made small by making ϵ small, i. e. choosing a large k_{tran}. Given Assumption 4, we can use the result from [17] regarding satisfaction of set-based tasks with the valid set $\mathcal{D}$ from Def. 3 to show that the robot converges to $\mathcal{H}$ without violating Objective 1. It follows directly from Proposition 1, that Objective 2 is satisfied in the sense that $\lim_{t\to\infty} \|p_y(t)\|$ can be made arbitrarily small if the system is in Mode 1. ■

VI. Experimental Study

In this section we present an experimental study that validates the proposed set-based guidance scheme. First, the set-up of the experimental tests is described, before the experimental results are demonstrated.

A. Experimental set-up

The set-based path following and obstacle avoidance guidance control scheme in Algorithm 1 was tested in the Marine Cybernetics lab (MC-lab) at the Norwegian University of Science and Technology (NTNU). The MC-lab comprises a 1.5 m deep basin of 40×6.45 m^2 and is equipped with six cameras of the underwater motion capture system Qualisys, that are used to measure the position of reflective markers inside the basin. The camera set-up is shown in Fig. 4.

The snake robot Mamba served as the test platform for the guidance strategy. Mamba is a modular snake robot that was developed at NTNU [24]. In our case, the robot consisted of 18 joints, nine mounted horizontally and nine vertically.

Fig. 4. The MC-lab is equipped with six motion capture cameras, three mounted on each side of the basin.

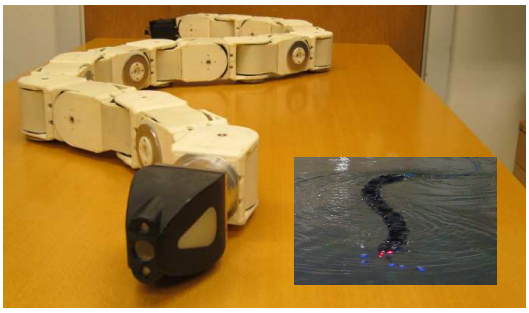


Fig. 5. The snake robot Mamba. In the basin, the robot is additionally equipped with reflective markers and a synthetic skin.

It has a thin, positively buoyant cable for power supply and communication. For the two-dimensional approach that was tested, the vertical joints were stiffened by setting their reference to zero in order to enable purely planar motion. A picture of the robot is displayed in Fig. 5. Despite the waterproof design of the single joints, the robot is additionally equipped with a synthetic, waterproof skin for swimming tests. The amount of air inside the skin can be varied with a pneumatic valve, which influences the buoyancy of the robot. In our tests, enough air was left inside the skin to provide a slightly positive buoyancy in order to keep the robot close to the surface, since depth control was not considered in the control system. During the tests, a construction with five reflective markers was attached to the head of the robot in order to provide a reference for the camera positioning system. The geometry of the attachment makes sure that its three-dimensional pose can be determined by the positioning system as soon as it is within the range of at least two cameras. The position of the CM of the robot according to the definition in Section II-A was computed from the position of the markers by using the kinematic relations that are sketched in Fig. 1. As discussed in Section II-A, the heading of the robot was defined by the orientation of the head link, and was therefore obtained directly from the motion capture system. Each joint module of Mamba comprises a servo motor, a microcontroller card, and various sensors like a temperature and a water leakage sensor. For more details on the joint modules, the reader is referred to [24].

The guidance algorithm, Algorithm 1, was implemented in LabVIEW 2013. The parameters in (4) were chosen as $k_{\text{tran}} = 0.02$ and $v = 0.05$ for Mode 1, and $k_{\text{tran}} = 0.012$ and $v = 0.05$ for Mode 2. The required position measurements were sent from the Qualisys Motion Tracker (QTM) software to LabVIEW at a frequency of 10 Hz in order to be available for the guidance. In order to make the switching feasible for the robot, the smoothing function $f_{\text{s}}(t, t_{\text{switch}})$ was used to interpolate between the heading reference of the active mode and the last heading reference before the previous switch:

$$f_{\text{s}}(t, t_{\text{switch}}) = \tfrac{1}{2}\tanh(0.8(t - t_{\text{switch}} - 2)) + \tfrac{1}{2}. \quad (7)$$

The time constant of the smoothing function was chosen such that the jump in the reference was smoothed within one oscillation of the snake robot. The resulting reference signal θ_{ref} was then sent to the heading controller. For the heading control input ϕ_0 in (1), a PD-controller was implemented:

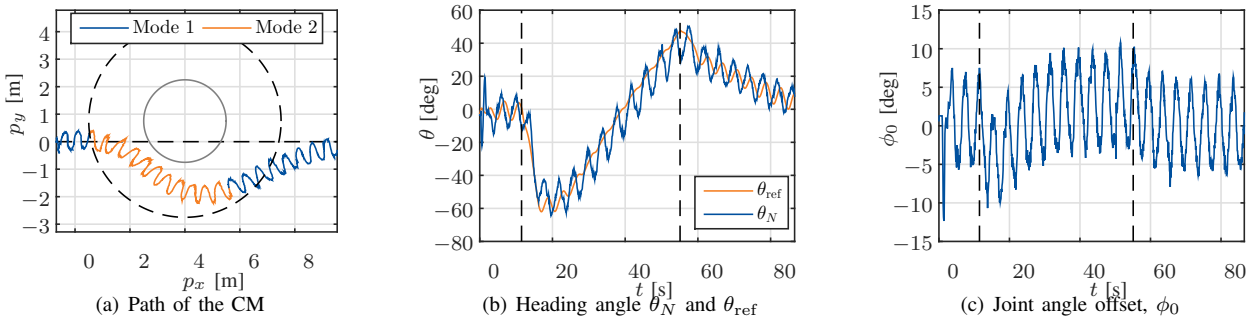
$$\phi_o = -k_p(\theta_N - \theta_{\text{ref}}) - k_d(\dot{\theta}_N - \dot{\theta}_{\text{ref}}). \quad (8)$$

The gains of the controller were tuned as $k_p = 0.56$ and $k_d = 0.03$. The time derivative $\dot{\theta}_N$ was obtained by numerically differentiating the angular measurement from the positioning system in LabVIEW and the time derivative of the reference heading, $\dot{\theta}_{\text{ref}}$, was obtained from (5) through a third order low-pass filtering reference model. The frequency of the reference model was chosen as $\frac{\pi}{2}$, and the damping was set to one. Details on the filtering reference model can be found in [18]. The resulting signal ϕ_0 was saturated at $\phi_{0,\max} = \pm 20^\circ$ to respect the joint limitations, and sent to the low level joint controllers that enforce the reference (1) on the single joints. These are P-controllers that are included in the servo motors inside each joint. The parameters in (1) were chosen as $\alpha = 30^\circ$, $\omega = 90^\circ$, and $\delta = 50^\circ$. In our tests the position of the obstacle was assumed to be known and directly included in the implementation. Combining the set-based guidance system with robot vision in order to detect the obstacles autonomously will be subject of future work.

B. Experimental results

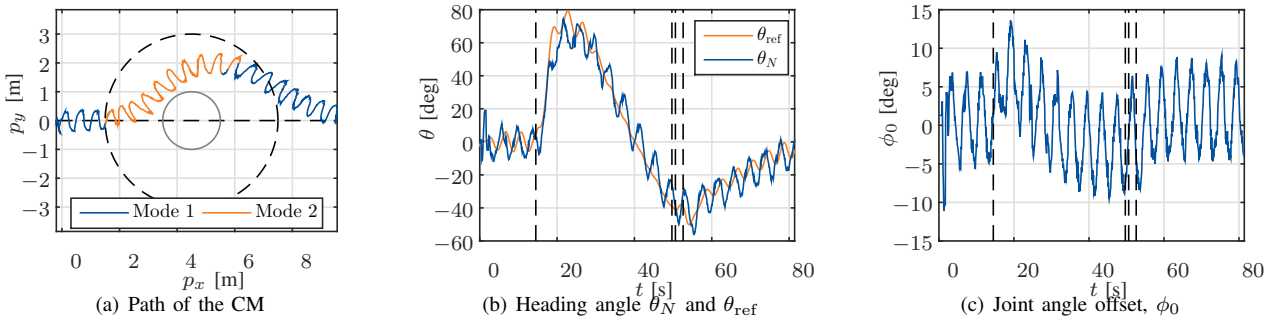
The set-based path following and obstacle avoidance guidance was tested in three different scenarios with obstacles of different sizes and location. Because the range of the motion capture system was limited to a length of ca. 10 m, only one obstacle at a time was considered. Results of one case of each scenario are illustrated in Figs. 6 to 8. Photos of the robot Mamba during a test run are presented in Fig. 9.

Because of short available range of the camera system, the initial conditions were chosen such that the robot started on and parallel to the path: $p_y(0) \approx 0, \theta_N(0) \approx 0$. The path of the robot is plotted in Figs. 6(a), 7(a) and 8(a). It can be seen that in every scenario, the robot left the path right after entering the mode change circle. The robot then followed a circular path without violating the safety radius. The offset of the robot to the grey circle was intentionally achieved by the choice of the transversal gain k_{tran} for Mode 2: in accordance with Proposition 1, the offset to the circle can be made small by a higher choice of k_{tran}. A better tracking of the circle would however lead to an intersection of some of the robot's oscillations with the safety radius, so in accordance with Objective 1, we chose a small gain k_{tran}. In all three scenarios, the robot managed to converge back to the path just before leaving the range of the camera system. The reference θ_{ref} that was provided by the guidance law is plotted in Figs. 6(b), 7(b) and 8(b). It can be seen



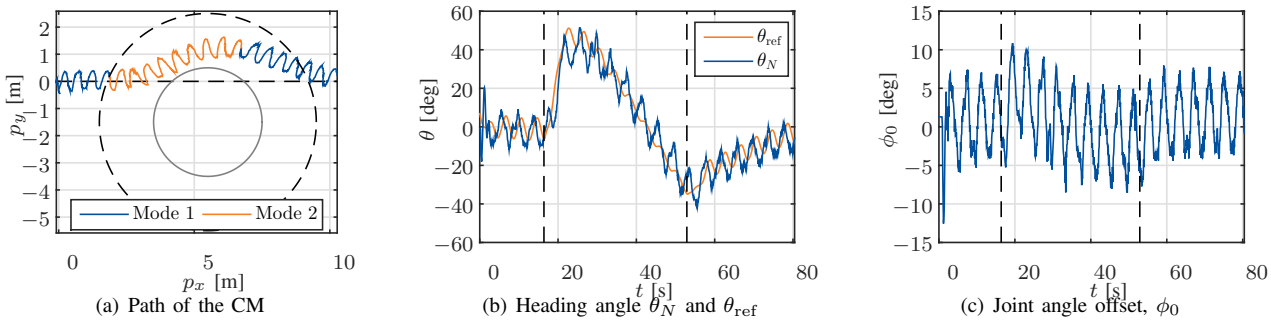
(a) Path of the CM (b) Heading angle θ_N and θ_{ref} (c) Joint angle offset, ϕ_0

Fig. 6. Experimental results of the first scenario: An obstacle with safety radius $r_{\text{s}} = 1.5$ m was placed 0.75 m to the left side of the path. In (a), the safe radius r_{s} is displayed as a grey, and the mode change radius r_{m} as a dashed circle. The dashed lines in (b) and (c) indicate the switching times.



(a) Path of the CM (b) Heading angle θ_N and θ_{ref} (c) Joint angle offset, ϕ_0

Fig. 7. Experimental results of the second scenario: An obstacle with safety radius $r_{\text{s}} = 1$ m was placed on the path. In (a), the safe radius r_{s} is displayed as a grey, and the mode change radius r_{m} as a dashed circle. The dashed lines in (b) and (c) indicate the switching times.



(a) Path of the CM (b) Heading angle θ_N and θ_{ref} (c) Joint angle offset, ϕ_0

Fig. 8. Experimental results of the third scenario: An obstacle with safety radius $r_{\text{s}} = 2$ m was placed 1.5 m to the right side of the path. In (a), the safe radius r_{s} is displayed as a grey, and the mode change radius r_{m} as a dashed circle. The dashed lines in (b) and (c) indicate the switching times.

in Figs. 6(b), 7(b) and 8(b) that the smoothing function (7) smooths the reference signal. The heading controller turned out to track the reference nicely, and the control input ϕ_0 remained within reasonable values, as displayed in Figs. 6(c), 7(c) and 8(c). The oscillations in the control input ϕ_0 are a result of the oscillations of the CM, that enter ϕ_0 via the heading reference θ_{ref}. Such oscillations are inherent to snake locomotion according to (1) and we do not attempt to suppress them in the control design. In some cases, the guidance system switched back and forth between the two modes when exiting obstacle avoidance mode. An example is provided in Fig. 7. The heading reference θ_{ref} in Fig. 7(b) and the control input ϕ_0 in Fig. 7(c) do, however, not demonstrate chattering behaviour. This is owed to the smoothing function (7), which prevents the chattering from entering the controllers.

VII. Conclusions

This article proposed a set-based guidance strategy for path following with obstacle avoidance for planar underwater snake robots. The guidance system utilizes a switching strategy originally developed for autonomous surface vessels, which we generalized in this article by introducing a new switching condition It is used in combination with a guidance scheme for underwater snake robots, which has been extended to enable a choice of direction. The new switching condition makes the set-based switching independent from the underlying dynamic model by considering only the kinematics, and thus generalizes the approach in [16] to a larger class of systems, including terrestrial and underwater snake robots. This approach was seen to be more suitable for snake robot locomotion, where the dynamic model will induce oscillations due to the sinusoidal snake locomotion, and thus increase the risk of chattering behaviour. The guidance system was shown to fulfil the control objectives on a kinematic level. The proposed guidance strategy was combined with a PD-heading controller and experimentally validated with a swimming snake robot for several stationary obstacles of different sizes and locations with respect to the path. These were the first tests of the set-based theory for

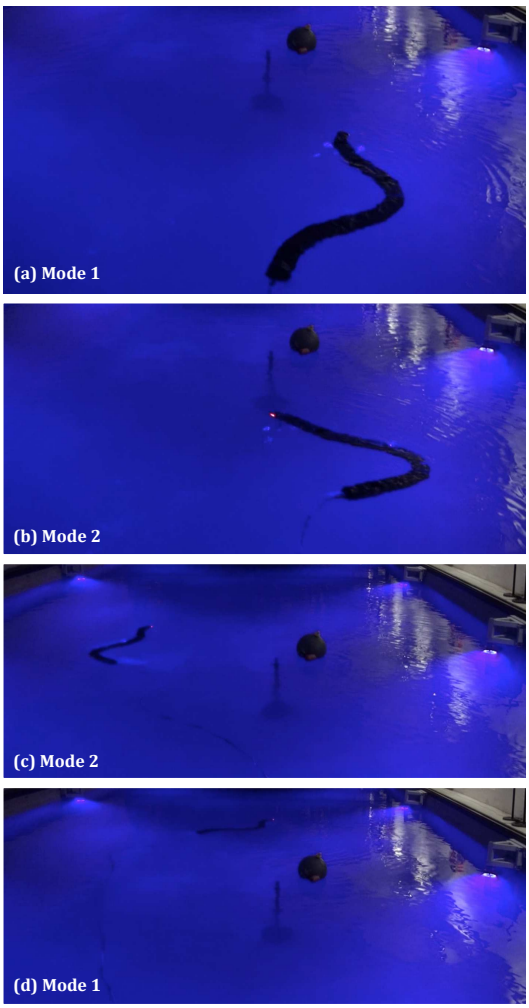


Fig. 9. The snake robot Mamba during a test run. (a) The robot is started on the path and towards the obstacle. (b) The set-based guidance strategy switches into obstacle avoidance mode and the robot starts turning. (c) The robot circumvents the obstacle at a safe distance. (d) After passing the obstacle, the robot safely converges back to the path. We have included a supplementary file which contains the full video.

a floating-base robot, and the results validate the proposed guidance scheme. Future work will include a combination of the approach with autonomous obstacle detection and an extension to the three-dimensional case. In addition, Assumption 2 will be relaxed by extending the guidance to handle unknown ocean currents.

ACKNOWLEDGMENT

The authors thankfully acknowledge the support of Albert Sans-Muntadas in the lab, and of Glenn Angell and Daniel Bogen during the preparations of the experiments.

REFERENCES

[1] R. Christ and R. Wernli, *The ROV Manual: A User Guide for Remotely Operated Vehicles*. Butterworth-Heinemann, Oxford, 2014.

[2] K. A. McIsaac and J. P. Ostrowski, "Motion planning for anguilliform locomotion," *IEEE Trans. Robot. Autom.*, vol. 19, no. 4, pp. 637–652, 2003.

[3] E. Kelasidi, P. Liljebäck, K. Y. Pettersen, and J. T. Gravdahl, "Innovation in underwater robots: Biologically inspired swimming snake robots," *IEEE Robotics Automation Magazine*, vol. 23, no. 1, pp. 44–62, March 2016.

[4] A. M. Kohl, K. Y. Pettersen, E. Kelasidi, and J. T. Gravdahl, "Planar path following of underwater snake robots in the presence of ocean currents," *IEEE Robotics and Automation Letters*, vol. 1, no. 1, pp. 383–390, 2016.

[5] A. Mohammadi, E. Rezapour, M. Maggiore, and K. Y. Pettersen, "Maneuvering control of planar snake robots using virtual holonomic constraints," *IEEE Trans. Control Syst. Technol.*, vol. 24, no. 3, pp. 884 – 899, 2015.

[6] A. M. Kohl, E. Kelasidi, A. Mohammadi, M. Maggiore, and K. Pettersen, "Planar maneuvering control of underwater snake robots using virtual holonomic constraints," *Bioinspiration & Biomimetics*, vol. 11, no. 065005, 2016.

[7] S. Y. Na, D. Shin, J. Y. Kim, S.-J. Baek, and S. H. Min, *Obstacle Recognition and Collision Avoidance of a Fish Robot Based on Fuzzy Neural Networks*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2007, pp. 337–344.

[8] M. Porez, V. Lebastard, A. J. Ijspeert, and F. Boyer, "Multi-physics model of an electric fish-like robot: Numerical aspects and application to obstacle avoidance," in *Proc. IEEE/RSJ Int. Conf. Intelligent Robots and Systems*, San Francisco, CA, Sept 2011.

[9] E. Kelasidi, K. Y. Pettersen, and J. T. Gravdahl, "A waypoint guidance strategy for underwater snake robots," in *Proc. IEEE Mediterranean Conf. on Control & Automation*, Palermo, Italy, June 2014.

[10] Y. Hitaka, T. Yoshitake, and M. Yokomichi, "Obstacle avoidance of snake robot by switching control constraint," *Artificial Life and Robotics*, vol. 17, no. 2, pp. 180–185, 2012.

[11] M. Tanaka and F. Matsuno, "Control of snake robots with switching constraints: trajectory tracking with moving obstacle," *Advanced Robotics*, vol. 28, no. 6, pp. 415–429, 2014.

[12] M. S. Menon, V. C. Ravi, and A. Ghosal, "Trajectory Planning and Obstacle Avoidance for Hyper-Redundant Serial Robots," *Journal of Mechanisms and Robotics*, vol. 9, no. 041010-1, 2017.

[13] F. Kamil, S. H. Tang, W. Khaksar, N. Zulkifli, and A. S. A., "A review on motion planning and obstacle avoidance approaches in dynamic environments," *Advances in Robotics & Automation*, vol. 4, no. 2-1000134, 2015.

[14] B. M. Albaker and N. A. Rahim, "A survey of collision avoidance approaches for unmanned aerial vehicles," in *Int. Conf. Technical Postgraduates (TECHPOS)*, Kuala Lumpur, Malaysia, Dec. 2009.

[15] T. Statheros, G. Howells, and K. McDonald-Maier, "Autonomous ship collision avoidance navigation concepts, technologies and techniques," *The Journal of Navigation*, vol. 61, pp. 129–142, 2008.

[16] S. Moe and K. Y. Pettersen, "Set-Based Line-of-Sight (LOS) Path Following with Collision Avoidance for Underactuated Unmanned Surface Vessel," in *Proc. 24th Mediterranean Conf. on Control and Automation*, Athens, Greece, June 2016.

[17] S. Moe, G. Antonelli, A. R. Teel, K. Y. Pettersen, and J. Schrimpf, "Set-Based Tasks within the Singularity-Robust Multiple Task-Priority Inverse Kinematics Framework: General Formulation, Stability Analysis, and Experimental Results," *Frontiers in Robotics and AI*, vol. 3, no. 16, pp. 1–18, 2016.

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

[19] D. Rollinson, A. Buchan, and H. Choset, "Virtual Chassis for Snake Robots: Definition and Applications," *Advanced Robotics*, vol. 26, no. 17, 2012.

[20] R. L. Hatton, R. A. Knepper, H. Choset, D. Rollinson, C. Gong, and E. Galceran, "Snakes on a plan: Toward combining planning and control," in *Proc. 2013 IEEE Int. Conf. Robotics and Automation*, Karlsruhe, Germany, May 2013.

[21] T. I. Fossen, E. Kelasidi, and K. Y. Pettersen, *Encyclopedia of Robotics*. Springer, 2017, ch. Modeling of underwater vehicles, (submitted).

[22] E. Kelasidi, P. Liljebäck, K. Y. Pettersen, and J. T. Gravdahl, "Experimental investigation of efficient locomotion of underwater snake robots for lateral undulation and eel-like motion patterns," *Robotics and Biomimetics*, vol. 2, no. 8, pp. 1–27, 2015.

[23] A. M. Kohl, K. Y. Pettersen, E. Kelasidi, and J. T. Gravdahl, "Analysis of underwater snake robot locomotion based on a control-oriented model," in *Proc. IEEE Int. Conf. Robotics and Biomimetics*, Zhuhai, China, Dec. 2015.

[24] P. Liljebäck, Ø. Stavdahl, K. Pettersen, and J. Gravdahl, "Mamba - A waterproof snake robot with tactile sensing," in *Proc. IEEE/RSJ Int. Conf. Intelligent Robots and Systems*, Chicago, IL, Sep. 2014.