# Head Trajectory Tracking Control of a Snake Robot and its Robustness Under Actuator Failure

Ryo Ariizumi, *Member, IEEE,* Ryota Takahashi, Motoyasu Tanaka, *Member, IEEE,* and Toru Asai, *Member, IEEE*

***Abstract*—This paper considers the problem of trajectory tracking of a planar snake robot without a lateral constraint. The reference trajectory of the head position and the orientation of link 1 are given, and torque control is determined to reduce tracking errors. The performance of the controller was tested in a number of simulations. The robustness during actuator failure was also studied. We assumed that one of the actuators was broken and the corresponding joint became passive. Furthermore, as a more realistic situation, we considered an instance when some of the states were not readily accessible from the sensor readings and needed to be estimated by an observer. The extended Kalman filter was employed for this purpose, and the performance of the closed-loop system with the observer was also tested in simulations.**

***Index Terms*—Snake robot, Trajectory tracking, Actuator malfunction**

## I. Introduction

Snake robots can negotiate environments such as uneven terrain [1], [2], [3], stairs [4], and the inside or outside of pipes [5], [6], [7] using their large number of degrees of freedom (DOFs). Therefore, snake robots are expected to be useful for a wide range of applications such as disaster response and pipe inspection. However, controlling such complex systems with so many DOFs, as well as propelling them by interacting with environments in a complicated way, is difficult and has received significant focus of recent research.

The locomotion of biological snakes was studied by Gray [8] in the 1940s, and he suggested that an anisotropy in friction might play an important role. In his pioneering work [9], Hirose observed biological snakes and suggested the use of a no side-slip (lateral) constraint on the body to model the snake locomotion mathematically. Hirose also proposed for the body shape of snake robots the serpenoid curve, which has a curvature determined by a sinusoidal function. Snake-like locomotion of such snake robots, with serpenoid curves and passive wheels, were successfully realized.

After Hirose's work, there has been a number of studies on the control of snake robots that assume a lateral constraint. Prautsch et al. [10] studied the control of the head position based on a dynamic model incorporating the lateral constraint. However, they observed that the robot often converged to a singular posture in which all joint angles were equal. Once the singular posture is reached, the robot is difficult to control. Although the movement of the snake robot was successfully approximated in those studies [11], [12], [13], lateral motion is actually very common in snake robots, even if they are equipped with passive wheels.

Another branch of snake robot research does not rely on the lateral constraint, but models the robot with explicit force from friction [14], [15], [16]. In [17], [18], a novel model that converts the effect of a rotational joint into link displacement in the lateral direction was proposed, and a straight path tracking control of the center of mass (CM) of a snake robot was developed. These studies relied on the serpenoid curve to make the robotic shape, and the controlled point in these studies is usually the CM; the control of the head is not considered. However, in some instances the control of the head position and the orientation of link 1 are more important than the control of the CM. For example, a snake robot is often equipped with a camera on its head, and the operator must rely on its image for operation. In [19], [20], a path-following control method based on sliding-mode control that achieved asymptotic convergence of the head orientation and path-following of the CM was proposed. However, the path-following problem is slightly different from the trajectory-tracking problem, that is considered for snake robots with lateral constraints [12].

The large number of DOFs in snake robots suggests that it can continue its task even if some of the actuators fail to move correctly. However, there are only a few studies [21] on a robot's movement under actuator malfunction. If the lateral constraint can be assumed, the control method proposed in [11], [12] would be applicable. However, this control method would fail if there were lateral motion. Therefore, it is necessary to take lateral motion into consideration to develop a control method when actuator malfunction is present.

In this paper, we develop a head-trajectory tracking control method for snake robots that can be used even if a joint becomes free accidentally. The resistance force from the environment is modeled as anisotropic viscous friction as in many studies [15], [14], [16], [22] to take the lateral motion into account. We first construct the control strategy without assuming actuator failure and then show how it can be applied in instances of a malfunctioning joint, which is assumed to be free. In other types of failure, we can cut the energy supply for the actuator and make it free, as long as the structure of the robot allows this.

In instances when lateral constraints are assumed, the trajectory-tracking problem was addressed by the partial feedback linearization (PFL) technique [12]. However, it is not easy to apply in our problem settings. This is because attaining trajectory tracking of the head does not imply boundedness of the joint variables. Motivated by the difficulty of PFL, in this study, the control purpose is relaxed to make it readily achievable. A controller is developed to solve the relaxed version of the problem. Note that in [16], [23], [24], they successfully used PFL for control of a snake robot without lateral constraints, with the purpose of controlling its shape. This can be achieved because the boundedness of joint variables implies the boundedness of the head position. In [25], in which the control of the body shape and head angle is considered, the boundedness of joint variables is proved using finite gain $\mathcal{L}$-stability analysis. In [23], maneuvering control problems were considered, i.e., the speed was also modulated. Our method gives also a simpler alternative to this.

In summary, the main contributions of the paper are the following: (i) we propose a novel controller that can achieve head-trajectory tracking in a weaker sense, and (ii) we show the robustness of the controller in instances of a malfunctioning joint.

The paper is organized as follows. In Section II, the model of a snake robot is briefly explained. Based on the equations of the motion,

This work was supported by the ImPACT Program of the Council for Science, Technology and Innovation (Cabinet Office, Government of Japan).
*(Corresponding author: Ryo Ariizumi)*

R. Ariizumi, R. Takahashi, and T. Asai are with the Department of Mechanical System Engineering, Graduate School of Engineering, Nagoya University, Nagoya 464-8603, Japan email: ryo.ariizumi@mae.nagoya-u.ac.jp, riota.takahashi@gmail.com, asai@nuem.nagoya-u.ac.jp.

M. Tanaka is with the Department of Mechanical and Intelligent Systems Engineering, Graduate School of Information Science and Engineering, The University of Electro-Communications, Tokyo 182-8585, Japan email: mtanaka@uec.ac.jp.

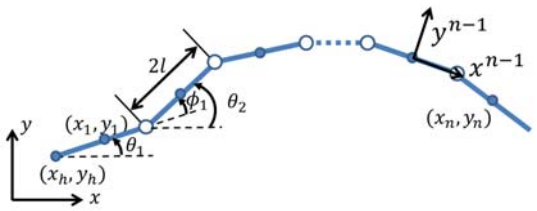



Fig. 1. Schematic of a snake robot with $n$ links and $n-1$ joints

the state-space representation is also given. Our proposed controller is explained in Section III. In Section IV, we introduce a malfunctioning joint in the system and explain the required modification to our controller. The controller is tested by a number of numerical examples in Section V, and Section VII concludes the paper.

## II. Model of a Planar Snake Robot

The dynamic model of a snake robot is given in [15] and in this section, some fundamentals of the model of a snake robot are reviewed. For the derivation of the equations of motion, we refer the reader to [15], [23], for example.

### A. Dynamic Model of Planar Snake Robot

The snake robot considered in this research consists of $n$ rigid links that are connected serially by $n-1$ active yaw joints (Fig. 1). All links are assumed to have the same length and uniform mass distribution. Some notation used in the analysis is defined as follows:

$l$ : Half of the length of each link
$\theta_i$ : Orientation of link $i$
$\phi_i$ : Yaw angle of joint $i$. $\phi_i = \theta_{i+1} - \theta_i$
$(x_h, y_h)$ : Position of the head
$\boldsymbol{w}$ : $\boldsymbol{w} = [x_h \quad y_h \quad \theta_1]^T$
$\boldsymbol{\theta}, \boldsymbol{\phi}$ : $\boldsymbol{\theta} = [\theta_1 \quad \cdots \quad \theta_n]^T$, $\boldsymbol{\phi} = [\phi_1 \quad \cdots \quad \phi_{n-1}]^T$
$\boldsymbol{q}$ : General coordinate. $\boldsymbol{q} = [x_h \quad y_h \quad \theta_1 \quad \boldsymbol{\phi}^T]^T$
$\boldsymbol{\tau}$ : the torque input vector

For convenience, the equations of motion of a snake robot are listed below:

$$H\ddot{\boldsymbol{q}} + W\mathrm{diag}(J_{q\bar{q}}\dot{\boldsymbol{q}})J_{q\bar{q}}\dot{\boldsymbol{q}} + C\dot{\boldsymbol{q}} = E\boldsymbol{\tau}, \tag{1}$$

where $H$ is the positive-definite inertia matrix, $J_{q\bar{q}}$ a transformation matrix that satisfies $J_{q\bar{q}}\boldsymbol{q} = [x_h \quad x_y \quad \boldsymbol{\theta}^T]^T$, $W\mathrm{diag}(J_{q\bar{q}}\dot{\boldsymbol{q}})J_{q\bar{q}}\dot{\boldsymbol{q}}$ describe the Coriolis and centrifugal force term, and $C\dot{\boldsymbol{q}}$ is the viscous friction term.

### B. State-Space Representation

The above equations of motion can now be written more concisely in the form,

$$H(\boldsymbol{q})\ddot{\boldsymbol{q}} + \boldsymbol{h}(\boldsymbol{q}, \dot{\boldsymbol{q}}) = E\boldsymbol{\tau}, \tag{2}$$

where $\boldsymbol{h}(\boldsymbol{q}, \dot{\boldsymbol{q}}) = W\mathrm{diag}(J_{q\bar{q}}\dot{\boldsymbol{q}})J_{q\bar{q}}\dot{\boldsymbol{q}} + C\dot{\boldsymbol{q}}$. Being positive-definite, $H$ is therefore always invertible [26].

By multiplying (2) by $H^{-1}$ from the left, we have

$$\ddot{\boldsymbol{q}} = -H^{-1}(\boldsymbol{q})\boldsymbol{h}(\boldsymbol{q}, \dot{\boldsymbol{q}}) + H^{-1}(\boldsymbol{q})E\boldsymbol{\tau}. \tag{3}$$

Let $\boldsymbol{\eta}$ be the state vector defined as $\boldsymbol{\eta} = [\boldsymbol{q}^T \quad \dot{\boldsymbol{q}}]^T$, then the state equations can be written as follows,

$$\dot{\boldsymbol{\eta}} = \boldsymbol{f}(\boldsymbol{\eta}) + \mathcal{B}(\boldsymbol{\eta})\boldsymbol{\tau}, \tag{4}$$

where

$$\boldsymbol{f}(\boldsymbol{\eta}) = \begin{bmatrix} \dot{\boldsymbol{q}} \\ -H^{-1}(\boldsymbol{q})\boldsymbol{h}(\boldsymbol{q}, \dot{\boldsymbol{q}}) \end{bmatrix}, \quad \mathcal{B}(\boldsymbol{\eta}) = \begin{bmatrix} O_{(n+2)\times(n-1)} \\ H^{-1}(\boldsymbol{q})E \end{bmatrix}. \tag{5}$$

## III. Controller Design

We next define the tracking problem and propose a novel controller. A straightforward design of PFL by setting head position as the output may fail in avoiding a singular posture, such as a straight line, and the control input diverges. This is because the tracking of the head does not imply the stability of the joint variables nor singular posture avoidance. For the definition of the singular posture, see [26] for details. Essentially, the singular posture is the posture at which manipulability vanishes; here manipulability is the metric indicating how easy it is to move the head in all direction. Clearly, a straight-line shape is singular because moving the head forward is impossible. To overcome the drawback in using the PFL-based controller, we loosen the control requirement and propose a controller that achieves the original requirement with some inevitable tracking errors.

### A. Trajectory Tracking Problem

We consider the problem of tracking the trajectory of the robot head and the orientation of the first link. More precisely, the problem can be stated as follows:

**Problem 1.** *Given the reference trajectory of* $\boldsymbol{w}^r(t) = [x_h^r(t) \quad y_h^r(t) \quad \theta_h^r(t)]^T$, *find the control input* $\boldsymbol{\tau}(t)$ *that attains the convergence of the tracking error* $e(t) = \|\boldsymbol{w}(t) - \boldsymbol{w}^r(t)\|^2/2 \to 0$ *as* $t \to \infty$, *where* $\boldsymbol{w} = [x_h \quad y_h \quad \theta_h]^T$ *and concurrently keep* $\|\boldsymbol{\phi}(t)\|$ *bounded for any* $t \geq 0$.

With no constraint on lateral motion, this problem has been addressed by PFL techniques [12]. PFL was successful in such instances because, under the lateral constraint, the internal motion of $\boldsymbol{\phi}$ is completely determined by $\boldsymbol{w}$. However, in our problem settings, there is no guarantee that $\boldsymbol{\phi}$ is bounded even if $\boldsymbol{w}$ and its derivatives are bounded. Therefore, we need to ensure all the states are bounded by a controller. Unfortunately, a snake robot is an under-actuated system and it is very difficult to assure the boundedness of all states if some of the DOFs are used to render the exact head tracking. Therefore, we have loosened the requirement for tracking and propose a controller for the modified problem.

### B. Problem Modification and the Proposed Controller

As mentioned above, a straightforward design of a PFL-based controller for head tracking suffers from convergence to a singular posture or instability. Although this implies that exact tracking is difficult to achieve, asymptotic convergence is not required if we only want the robot to follow the trajectory on average. In other words, we want the robot to move more or less, but not exactly, on the predefined trajectory. This is not true if, for example, the intended usage is on a factory line. However, the arranged environment of a factory would not be an ideal location for the use of snake robots as other robots that are designed especially for this purpose can be used. Unarranged environments would be ideal locations, and this is why we suppose the looser version of the requirement would be sufficient.

The looser version of the control purpose is stated as follows:

**Problem 2.** *Given the reference trajectory of* $\boldsymbol{w}^r(t) = [x_h^r(t) \quad y_h^r(t) \quad \theta_h^r(t)]^T$, *find the control input* $\boldsymbol{\tau}(t)$ *that makes the tracking error* $e(t) = \|\boldsymbol{w}(t) - \boldsymbol{w}^r(t)\|^2/2$ *as small as possible for any* $t \geq 0$ *and at the same time keep* $\|\boldsymbol{\phi}\|$ *bounded.*

To achieve the objective, we consider a control method that tries to make a cost function small as time grows. Let $V(\boldsymbol{\eta})$ be a cost function that appropriately encodes the purpose. The time derivative of $V$ is

$$\dot{V}(\boldsymbol{\eta}) = \frac{\partial V}{\partial \boldsymbol{\eta}}\dot{\boldsymbol{\eta}} = \frac{\partial V}{\partial \boldsymbol{\eta}}\{\boldsymbol{f}(\boldsymbol{\eta}) + \mathcal{B}(\boldsymbol{\eta})\boldsymbol{\tau}\}. \tag{6}$$

Therefore, if we define $\boldsymbol{\tau}$ as

$$\boldsymbol{\tau} = -\mathcal{B}^{\dagger}(\boldsymbol{\eta}) \left(\frac{\partial V}{\partial \boldsymbol{\eta}}\right)^{T} - \mathcal{B}^{\dagger}(\boldsymbol{\eta}) \boldsymbol{f}(\boldsymbol{\eta}), \tag{7}$$

then we have

$$\dot{V}(\boldsymbol{\eta}) = -\frac{\partial V}{\partial \boldsymbol{\eta}} \mathcal{B}\mathcal{B}^{\dagger}(\boldsymbol{\eta}) \left(\frac{\partial V}{\partial \boldsymbol{\eta}}\right)^{T} + \frac{\partial V}{\partial \boldsymbol{\eta}} \{I - \mathcal{B}\mathcal{B}^{\dagger}\} \boldsymbol{f}(\boldsymbol{\eta}). \tag{8}$$

Because $\mathcal{B}$ is a column of full-rank, the pseudo-inverse $\mathcal{B}$ is expressed as $\mathcal{B}^{\dagger} = (\mathcal{B}^T\mathcal{B})^{-1}\mathcal{B}^T$, and therefore, $\mathcal{B}\mathcal{B}^{\dagger} = \mathcal{B}(\mathcal{B}^T\mathcal{B})^{-1}\mathcal{B}^T$ is symmetric positive semidefinite. This suggests that the first term of (8) has the effect of making $V$ smaller. Although the effect of the drift term $\boldsymbol{f}(\boldsymbol{\eta})$ cannot be completely canceled, the second term of (7) tries to minimize it. As can be understood from (8), this controller cannot ensure a monotonic decrease in $V$ because of the effect of $\boldsymbol{f}$. However, if $\dot{\boldsymbol{q}} = [O_{n+2} \;\; I_{n+2}]\boldsymbol{\eta}$ is small enough, $\boldsymbol{f}$ becomes small and the second term of (8) can be neglected.

The cost function $V$ is defined as follows:

$$V(\boldsymbol{\eta}) = \|(\dot{\boldsymbol{w}}^r - \dot{\boldsymbol{w}}) + K_{w1}(\boldsymbol{w}^r - \boldsymbol{w})\|^2_{K_{w2}} + \|(\dot{\boldsymbol{\phi}}^r - \dot{\boldsymbol{\phi}})\|^2_{K_\phi} \tag{9}$$

where $\|\boldsymbol{x}\|^2_G = \boldsymbol{x}^T G \boldsymbol{x}$ for any vector $\boldsymbol{x}$ and a positive semidefinite matrix $G$. The reference $\boldsymbol{w}^r$ is set through a user command, and $\boldsymbol{\phi}^r$ is defined to avoid a singularity.

Even though the purpose of the control is the trajectory tracking of the head and the control of the joint variables is not important, the term $\|(\dot{\boldsymbol{\phi}}^r - \dot{\boldsymbol{\phi}})\|^2_{K_\phi}$ will be required to have the robot continue moving. In Theorem 3 of [16], it is suggested that the controller must be time-variant to make our system stable. As a direct consequence, we need at least one of the following to hold: (i) $K_{w2} \neq O$ and $\dot{\boldsymbol{w}}^r$ is time variant, (ii) $K_{w1} \neq 0$, $K_{w2} \neq O$, and $\dot{\boldsymbol{w}}^r$ is time variant, or (iii) $K_\phi \neq O$ and $\dot{\boldsymbol{\phi}}^r$ is time variant.

Regarding the gains for $\boldsymbol{w}$, i.e., $K_{w1}$, $K_{w2}$, and for $\boldsymbol{\phi}$, i.e., $K_\phi$, it is desirable to use larger gain for $\boldsymbol{w}$ and smaller gain for $\boldsymbol{\phi}$, as our goal is to make $\boldsymbol{w}$ follow the reference trajectory. Also, empirically, making the gain $K_{w1}$ smaller than $K_{w2}$ is best[1].

Note that in the literature a similar idea has been used for different purposes. In [12], a similar approach was applied to avoid the singular posture of a snake robot with lateral constraints by increasing the manipulability.

## IV. Introduction of a Malfunctioning Actuator

Because of the large number of DOFs, snake robots are expected to be robust to actuator faults. We shall explain the required modification to our controller, which is relatively simple, in instances of a malfunctioning actuator.

### A. Problem description

Among many conceivable actuator malfunctions, we consider one where the energy supply to the actuator is lost. Let joint $i$ be the corresponding joint, then this fault can be formalized as the constraint of $\tau_i \equiv 0$, i.e., joint $i$ is passive. In many other faults, it is possible to cut the energy supply on purpose to make the joint passive. Therefore, this case is expected to be rather general and of practical importance. With $\tau_i$ set to 0, the state equation (4) reads

$$\dot{\boldsymbol{\eta}} = \boldsymbol{f}(\boldsymbol{\eta}) + \tilde{\mathcal{B}}(\boldsymbol{\eta})\tilde{\boldsymbol{\tau}}, \tag{10}$$

where

$$\tilde{\boldsymbol{\tau}} = \begin{bmatrix} \tau_1 & \cdots & \tau_{i-1} & \tau_{i+1} & \cdots & \tau_{n-1} \end{bmatrix}^T \in \mathbb{R}^{n-2}, \tag{11}$$

[1]A PDF file with some more detailed discussion, along with additional numerical simulation results (figures and mat files) can be found at: https://nuss.nagoya-u.ac.jp/index.php/s/LJpedA1NSALQdoX.

$$\tilde{\mathcal{B}} = \begin{bmatrix} \mathcal{B}_1 & \cdots & \mathcal{B}_{i-1} & \mathcal{B}_{i+1} & \cdots & \mathcal{B}_{n-1} \end{bmatrix} \in \mathbb{R}^{(2n+4)\times(n-2)}, \tag{12}$$

and $\mathcal{B}_j$ is the $j$-th column of the matrix $\mathcal{B}$. In addition, we assume that joint angle information is also lost because of the malfunction.

In this study, the detection and classification of actuator malfunctions are not discussed. Although they are very interesting and important problems, they are beyond the scope of our present paper. Note that some actuators are capable of sending data that tell their status including errors and this may be used to detect malfunctions.

### B. Modifications of the Controller

From (10), the controller is given by:

$$\tilde{\boldsymbol{\tau}} = -\tilde{\mathcal{B}}^{\dagger}(\boldsymbol{\eta}) \left(\frac{\partial V}{\partial \boldsymbol{\eta}}\right)^{T} - \tilde{\mathcal{B}}^{\dagger}(\boldsymbol{\eta}) \boldsymbol{f}(\boldsymbol{\eta}). \tag{13}$$

The only difference with (7) is the use of $\tilde{\mathcal{B}}$ instead of $\mathcal{B}$. This controller requires the angle and angular velocity of the malfunctioning joint. The actuator malfunction may result in the loss of the corresponding angle data if the sensors and the actuator are packaged together, as they are in many servo motors. For such instances, we need to use an observer to estimate the angle.

As is imaginable, the loss of a joint affects the system negatively and significantly. This effect was demonstrated in [26] as a significant decrease in dynamic manipulability. This decrease may cause a loss in robustness to external perturbations and modeling errors. However, a detailed examination of this effect is outside the scope of the present paper.

## V. Numerical validation

Our controller was tested by a number of simulations. The parameters of the snake robot are shown in Table I. For the reference angular velocity of the joint, we used the following:

$$\dot{\phi}_i = \frac{2\pi T}{n} \alpha v \cos\left(vt - \frac{2\pi T}{n} i\right), \quad i = 1, \; \cdots, \; n-1, \tag{14}$$

where $v$ is the angular frequency of bending and the spatial frequency $T$ indicates how many periods are formed within the robot. This is the joint angular velocity for the serpenoid curve [9]. The weights in the cost function (9) were defined as follows:

$$\begin{aligned} &K_{w1} = \mathrm{diag}(2, 2, 3), K_{w2} = \mathrm{diag}(400, 400, 45), \\ &K_\phi = \mathrm{diag}(10, 20, ..., 20). \end{aligned} \tag{15}$$

As it is not necessary for $\dot{\boldsymbol{\phi}}$ to follow the reference exactly, the gain $K_\phi$ was set to be smaller than the gains for the head trajectory. The serpenoid parameters used to avoid singularity are $\alpha = \pi/6$, $v = 2$, and $T = 1.5$. Note that these weights were used for all the following simulations, regardless of reference or the presence or absence of the free joint.

The following two trajectories were used:

1) Connection of two straight lines with a right turn:

$$\boldsymbol{w}^r = \begin{cases} [-0.1t, \; 0, \; 0]^T, \; (t < 50 \text{ s}) \\ [-5, \; 0.1(t-50), \; -\pi/2]^T \; (t \geq 50 \text{ s}) \end{cases} \tag{16}$$

2) Sine curve:

$$\boldsymbol{w}^r = [-v_x t, \; 0.6\sin(2v_x t), \; \arctan(-0.6\cos(2v_x t))]^T \tag{17}$$

If not specified explicitly, $v_x = 0.05$ is used. Because we simulated situations where the trajectory is specified by velocity commands from the operator, we assumed that the initial position of the head is exactly on the trajectory, unless otherwise stated explicitly. However,

TABLE I
PARAMETERS OF THE ROBOT

| | | | |
|---|---|---|---|
| $m$ | 0.182 kg | $c_x$ | 0.1 Ns/m |
| $J$ | 0.22 kgm$^2$ | $c_y$ | 10 Ns/m |
| $2l$ | 0.12 m | $c_\theta$ | 0.0014 Nms |
| $n$ | 10 | | |

TABLE II
FRICTION PARAMETERS FOR SIMULATION MODEL IN CASE 3

| | |
|---|---|
| Viscous coeff. in lateral direction | 8.0 Ns/m |
| Viscous coeff. in longitudinal direction | 0.1 Ns/m |
| Coulomb coeff. in lateral direction | 0.15 |
| Coulomb coeff. in longitudinal direction | 0.0 |

for the head angle $\theta_h$, we did not let its initial value coincide with the reference. Instead, $\theta_h(0) = \alpha$ is used.

For each reference, we performed three types of simulations with different assumptions for the friction and free joint:

1) **Case 1**: There is no discrepancy in the friction model between the simulation model and the controller model. There is no free joint.
2) **Case 2**: The friction setting is the same as Case 1. Joint 5 is free (malfunctioning).
3) **Case 3**: In the simulation model, we add Coulomb friction in the lateral direction along with viscous friction. Joint 5 is free (malfunctioning).

In all cases, we assumed that only the position-level state values, i.e., $\boldsymbol{q}$, were available directly from observation and velocity-level state values, i.e., $\dot{\boldsymbol{q}}$, were not. Therefore, the state is estimated by the extended Kalman filter (EKF). For details on EKF, the reader is referred to standard textbooks such as [27]. Furthermore, to make the problem more realistic, measurements of available states were assumed to collapse with independent Gaussian noise. The standard deviations of noise were for torque inputs $1.0 \times 10^{-3}$ Nm and for observations $1.0 \times 10^{-2}$ m or rad according to the corresponding element.

In Cases 2 and 3, we assumed that the angular data of the malfunctioning actuator were lost. Therefore, in these instances, the state estimation using EKF is performed without using angle information of the malfunctioning joint. Although the observability of the system has not been formally proven, we confirmed that the linearized systems around the moving points were observable in our simulations, including the case where there is a joint the angle of which cannot be observed directly.

In Case 3, a slightly different model from the controller design was assumed for simulations. Specifically, the viscosity in the lateral direction was slightly smaller than the controller model (8.0 Ns/m instead of 10.0 Ns/m) and Coulomb friction was added, with a friction coefficient of 0.15. The friction settings are listed in Table II. For details of the Coulomb friction model that we used, see [14]. For a derivation of the equation of motion by the Lagrange method, see [23], for example.

In the remainder of this section, solid lines in figures refer to observed data and dashed lines refer to the reference. In all cases, we show the $t - x_h$ plot, $t - y_h$ plot, $t - \theta$ plot, head path, $t - V$ plot, and $t - \|\boldsymbol{\tau}\|$ plot.

### A. Case 1

*1) Right turn:* Results are shown in Fig. 2. The maximum observed tracking error in the simulation was 0.270 m (22.5 % of total robot length) for $x_h$, 0.190 m (15.8 % of total robot length) for $y_h$, and 1.63 rad in $\theta_h$. Note that the maximum tracking error of $\theta_h$ occurred at $t = 50$, where the reference changes discretely and a large tracking error is inevitable. The mean of the tracking error of $\theta_h$ was 0.0602 rad. Although our controller does not guarantee the cost function $V$ decreases with time, nevertheless it does so on average. From the plot of $\|\boldsymbol{\tau}\|$, it was confirmed that the singular posture avoidance mechanism works well and there was no occasion where the norm of the torque exceeded 3.

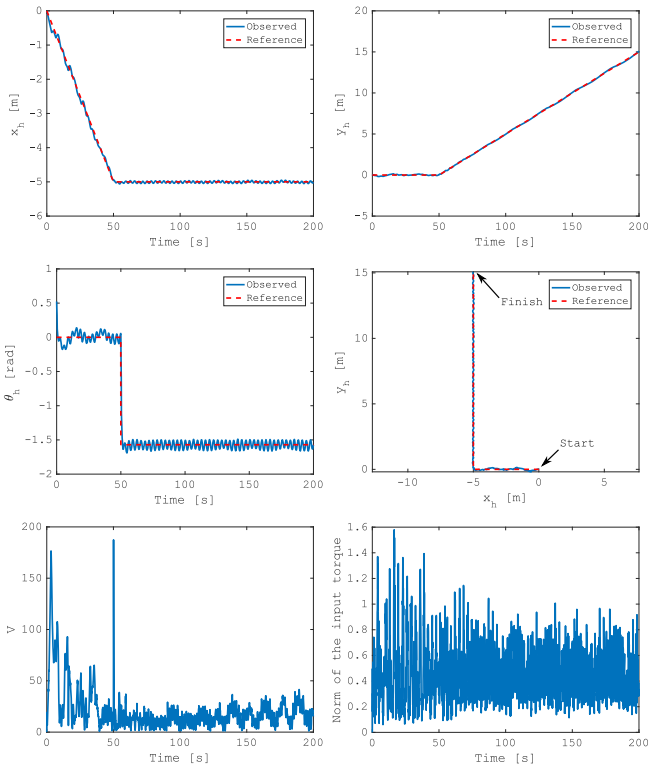
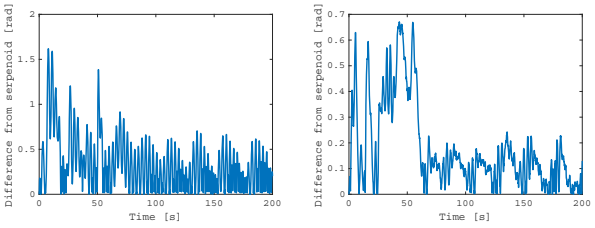



Fig. 2. Result of the simulation without discrepancy or a free joint in the model (Case 1). The right turn reference is used.




(a) Joint 1 (b) Joint 5

Fig. 3. Differences between the angle of joints and the serpenoid references for the joint 1 and 5.

In Fig. 3, the differences between the angles and the serpenoid references for the joints 1 and 5 are shown. As expected from our settings on the weight of the cost function $V$, the tracking of the joint angle to that of the serpenoid is not realized because the controller can render the head to track the reference.

*2) Sine Curve:* Results are shown in Fig. 4. A similar performance as for the right turn simulation was confirmed. The maximum observed tracking error in the simulation was 0.314 m for $x_h$, 0.234 m for $y_h$, and 1.06 rad in $\theta_h$. Note that the maximum tracking error for

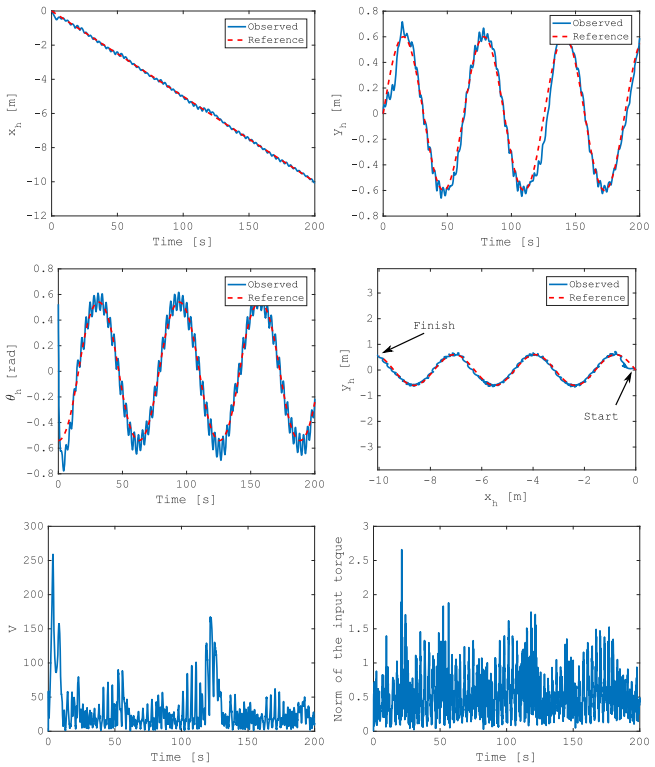



Fig. 4. Result of the simulation without discrepancy or a free joint in the model (Case 1). The sine curve reference is used.

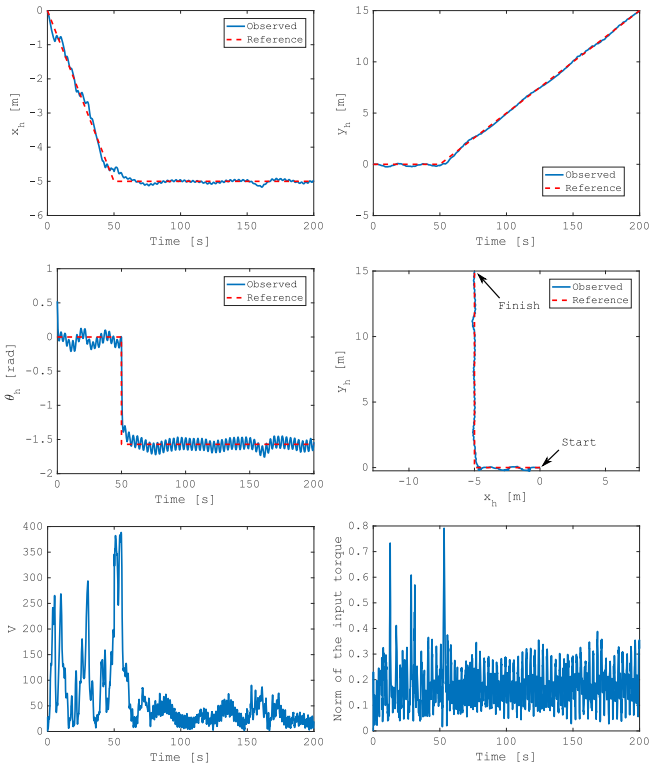



Fig. 5. Result of the simulation without discrepancy in the model. Joint 5 was free and the right turn reference was used (Case 2).

$\theta_h$ is observed at $t = 0$ s and the error gets much smaller within a couple of seconds. The maximum error of $\theta_h$ in $t \geq 10$ s is 0.163 rad and the mean error of $\theta_h$ was 0.0615 rad.

### B. Case 2

*1) Right turn:* The results are shown in Fig. 5. Although, in this case, one of the joints cannot produce torque, the controller works reasonably well. The maximum tracking error was 0.414 m for $x_h$ and 0.395 m for $y_h$, which are 34.5 % and 32.9 % of the total length of the snake robot, respectively. The maximum error in $\theta_h$ was almost the same as the previous case with a value of 1.56 rad at $t = 50$ s. The mean tracking error in $\theta_h$ was 0.0788 rad.

*2) Sine Curve:* The results are shown in Fig. 6. The maximum errors were 0.398 m for $x_h$, 0.338 m for $y_h$, and 1.06 rad for $\theta_h$. Also, in this case, the large tracking error in $\theta_h$ is because of the initial error. The maximum error of $\theta_h$ in $t \geq 10$ s is 0.269 rad and the mean error of $\theta_h$ is 0.0810 rad. Although the errors and $V$ seem bounded, no decreasing trend in $V$ is seen unlike Case 1. This results from the difficulties caused by a free joint.

### C. Case 3

*1) Right turn:* The results are shown in Fig 7. Despite modeling error, the tracking control worked reasonably well, which shows that the proposed controller was robust. The maximum tracking errors are 0.291 m for $x_h$ and 0.401 m for $y_h$, which are 24.3 % and 33.4 % of the total length of the robot. For $\theta_h$, the maximum tracking error was 1.57 rad at $t = 50$ s as in the previous cases, and the mean was 0.0762 rad.

The cost function $V$ had several large peaks, which shows that it is difficult to make the residual term $(\partial V/\partial \boldsymbol{\eta})(I - \mathcal{B}\mathcal{B}^{\dagger})\boldsymbol{f}(\boldsymbol{\eta})$ in (8) small enough given the modeling error considered. However, there was a decreasing trend in $V$.

*2) Sine Curve:* The results are shown in Fig 8. Large errors are seen to occur for $x_h$ and $y_h$ compared with other cases. However, as this is a very difficult case for the controller and the EKF, this performance is acceptable. The maximum tracking errors were 1.04 m for $x_h$, 0.553 m for $y_h$, and 1.06 rad for $\theta_h$. Although it seems that the error of $\theta_h$ becomes large at around $t = 120$, the maximum is taken at $t = 0$ s as in the previous cases. The maximum error of $\theta_h$ in $t \geq 10$ s is 0.389 rad and the mean error is 0.134 rad.

### D. Case 1 with Initial Tracking Error

Here, we assume that both of the head position and the head orientation at the initial time are not on the reference trajectory, but there is the initial tracking error of

$$\begin{aligned} \boldsymbol{w}(0) - \boldsymbol{w}^r(0) &= \begin{bmatrix} 2l & 12l & \pi/3 + \alpha - \theta_h^r(0) \end{bmatrix} \\ &= \begin{bmatrix} 0.12 & 0.72 & \pi/3 + \alpha - \theta_h^r(0) \end{bmatrix}. \end{aligned} \tag{18}$$

*1) Right Turn:* The results are shown in Fig 9. The tracking performance is not affected significantly compared with the case without initial tracking error. The maximum tracking errors were 0.258 m for $x_h$ and 0.720 for $y_h$. Note that the maximum error of $y_h$ is observed at $t = 0$ s and the mean error was 0.0658 m. The maximum error of $\theta_h$ was 1.63 rad at $t = 50$ s, at which the large error is inevitable. The mean tracking error of $\theta_h$ was 0.0724 rad.

*2) Sine Curve:* The results are shown in Fig 10. The tracking control is seen to work well, as for the right turn reference case. The maximum tracking errors were 0.181 m for $x_h$, 0.720 m for $y_h$, and 2.11 rad for $\theta_h$. As in previous cases with a sine curve reference, the maximum errors of $y_h$ and $\theta_h$ were observed at $t = 0$ and the mean errors are 0.0739 m and 0.0732 rad.

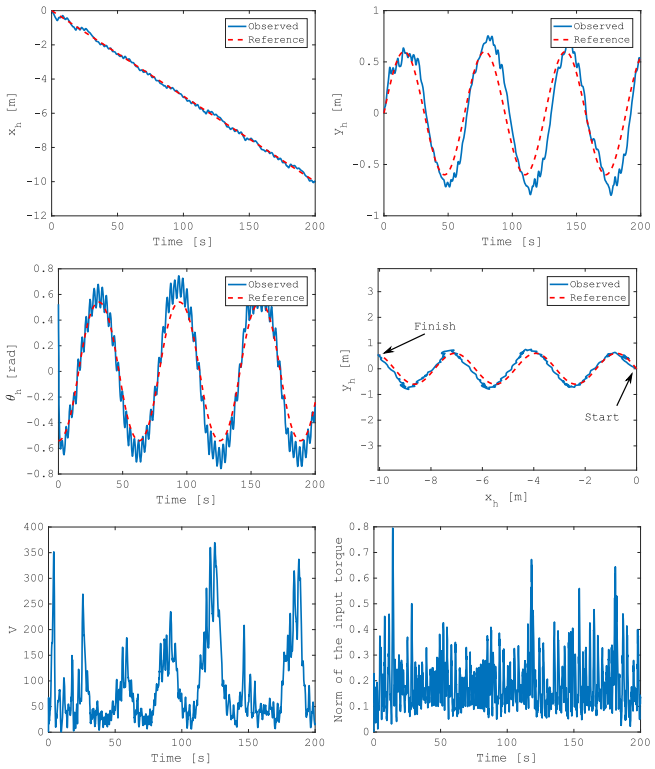



Fig. 6. Result of the simulation without discrepancy in the model. Joint 5 was free and the sine curve reference was used (Case 2).

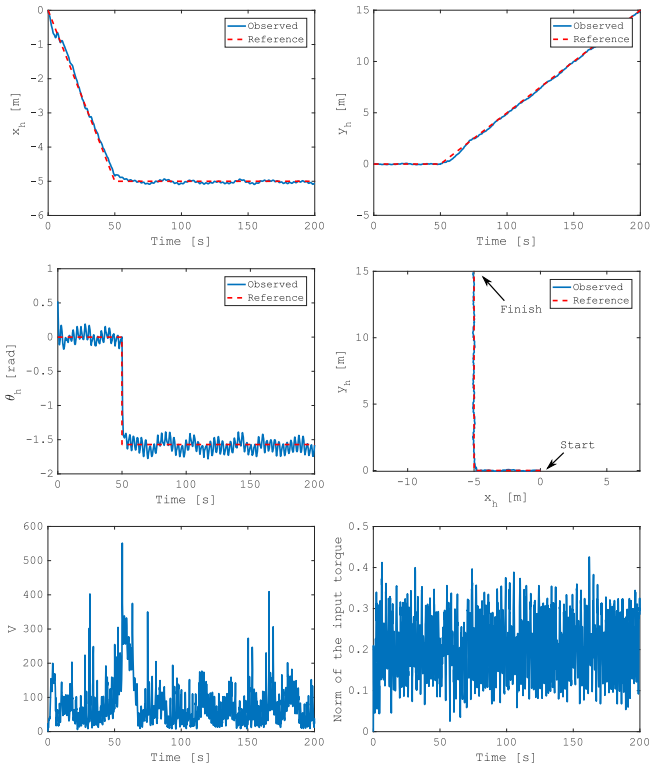



Fig. 7. Result of the simulation with discrepancy in the model. Joint 5 was free and the right turn reference was used (Case 3).

### *E. Case 1 with Various Sine References*

In this subsection, the tracking performance for the sine curve reference with various $v_x \in [0.01, 0.15]$ is investigated. Apart from $v_x$, all other settings were set the same. The performance is assessed by the following two metrics:

$$e_p = \frac{1}{L}\int_0^{t_f} \left\| \begin{bmatrix} x_h(t) \\ y_h(t) \end{bmatrix} - \begin{bmatrix} x_h^r(t) \\ y_h^r(t) \end{bmatrix} \right\| dt, \tag{19}$$

$$e_o = \frac{1}{L}\int_0^{t_f} |\theta_h(t) - \theta_h^r(t)| dt, \tag{20}$$

where $L = |x_h(t_f)|$. We divided by $L$ because the integral is expected to increase as the final position gets further from the start. In general, larger $e_p$ or $e_o$ implies worse tracking performance.

The results are shown in Fig. 11. Note that, from the result of $e_o$ (Fig. 11(b)), we exclude the result for $v_x = 0.145$ because it is reasonable to regard it as an outlier with $e_o = 7.03 \times 10^3$. From this figure, the tracking performance tends to become worse as $v_x$ increases. One of the possible factors of the phenomena is the settings of the serpenoid curve: $\alpha$, $v$, and $T$. As we do not change the serpenoid parameters that are used as the reference for joint angles, the motion of the robot may not be able to be much faster than the serpenoid motion.

## VI. Simulation Using Physical Engine

To make our discussion more reliable, we performed additional simulations using Vortex, running on V-REP. The model is shown in Fig 12. The robot was composed of $n = 8$ links with equal length of $2l = 0.176$ m, equal mass of 0.417 kg, and equal moment of inertia of $1.20 \times 10^{-3}$ kgm$^2$. The anisotropy in friction was achieved by equipping a pair of passive wheels on each link.

Friction coefficients for translational motion $(c_n, c_t)$ and gains that are used in the simulation were set the same as in the simulations in the previous section. The friction coefficient for a link rotation $c_\theta$ was set $4.00 \times 10^{-3}$ Nms. In the estimate of the time derivative of the general coordinate $\dot{\boldsymbol{q}}$, we used a pseudo-differential, the transfer function being $2\pi f_c s/(2\pi f_c\ s + 1)$, where the cut-off frequency $f_c$ was set to 10 Hz. Although the estimated velocity by the pseudo-differential will not be as accurate as that by EKF with the accurate model, it works more robustly to modeling errors. The time step was set to be 10 ms. Because the simulator does not accept the torque as input directly, we calculated the angular velocity of each joint that is assumed to be realized by the controller model using the state equation (4) and fed it as the reference joint angular velocity. The maximum feasible torque is set to 6.0 Nm. The maximum feasible torque for joint $i$ at time $t$ is set to $|\tau_i(t)|$. The center of mass of each link does not coincide with the geometric center in the Vortex model. However, in the controller model, we assumed that it does.

The reference trajectory is defined as follows:

$$\boldsymbol{w}^r = \left[ v_x t \quad -L\sin(bt) \quad \operatorname{atan}\left(-\tfrac{bL}{v_x}\cos(bt)\right) \right]^T, \tag{21}$$

where $v_x = 0.15$, $L = 2nl$, and $b = v_x/3$.

The results are shown in Fig. 13. The control is seen to work reasonably well also in the physical simulation, despite a large modeling error. Figure 12 shows the torque calculated by (7) and the actual torque applied to joints 4 and 7 for the first 60 s. There is a large difference especially in joint 4, which indicates a large modeling error.

## VII. Conclusion

We proposed a head-trajectory tracking control method for a snake robot. A looser version of the trajectory-tracking problem, which is still of practical use, was first stated. Then, a controller that is based on the minimization of a cost function is constructed. Although

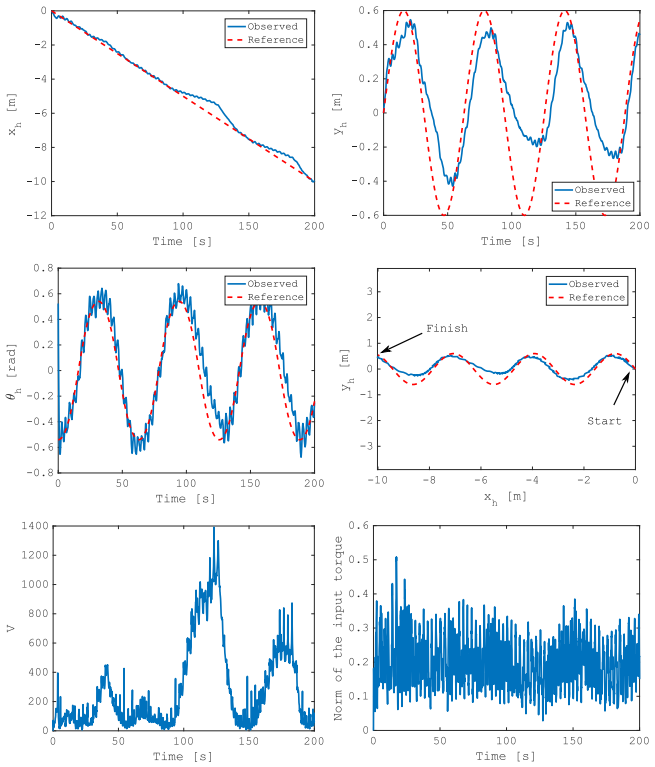



Fig. 8. Result of the simulation with discrepancy in the model. Joint 5 was free and the sin curve reference was used (Case 3).

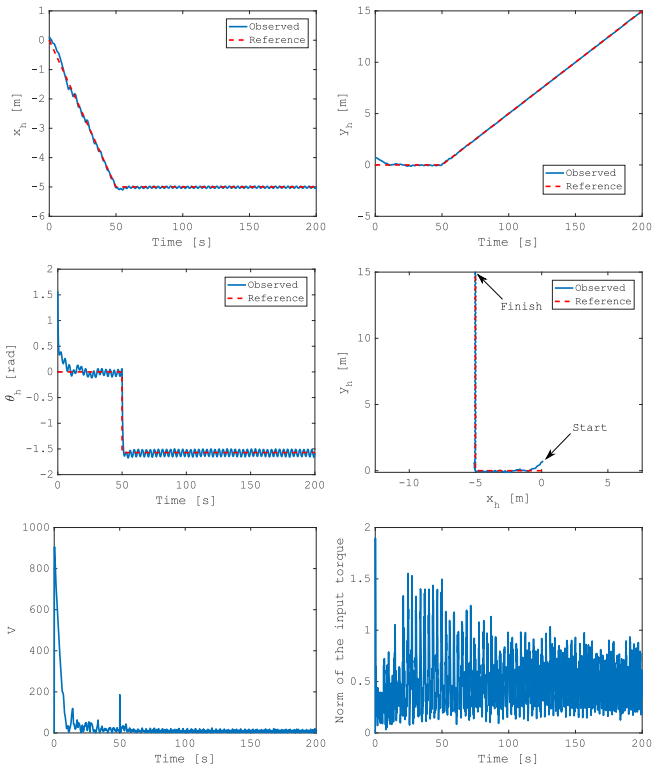



Fig. 9. Result of the simulation with initial tracking error. Simulation was without discrepancy in the model or a free joint (Case 1 with initial tracking error). The right turn reference is used.

the stability of the control method was not theoretically proven, simulation results showed that the tracking error was bounded. Moreover, the same control with the same weight settings was shown to be applicable when there was a free joint. By using the EKF, the control method can be adopted even if the angle and angular velocity of the free joint is not available. One of the merits of our method is that the performance was relatively insensitive to weight parameters, and the same parameter settings can be used in a variety of cases (e.g., with or without a free joint, with or without complete knowledge of the state, and for a variety of reference trajectories). This suggests the possibility that our proposed control method is valid in instances of a broken joint. We believe this is an important step toward making a control method that is robust to actuator failures.

Although the simulations showed that the tracking errors were bounded with our control method, it was not formally proven, and therefore proving the stability under certain conditions is to be the focus of future work. We are also interested in testing our method with a real snake robot.

## REFERENCES

[1] P. Liljebäck, K.Y. Pettersen, Ø. Stavdahl, and J.T. Gravdahl, "Hybrid Modeling and Control of Obstacle-Aided Snake Robot Locomotion," *IEEE Trans. Robot.*, vol. 26, no. 5, pp. 781-799, 2010.

[2] M. Travers, C. Gong, and H. Choset, "Shape-Constrained Whole-Body Adaptivity," in *Proc. IEEE International Symposium on Safety, Security, and Rescue Robotics*, pp.1-6, 2015.

[3] M. Travers, J. Whitman, P. Schiebel, D. Goldman, and H. Choset, "Shape-Based Compliance in Locomotion," in *Proc. Robotics: Science and Systems*, 2016.

[4] M. Tanaka, and K. Tanaka, "Control of a Snake Robot for Ascending and Descending Steps," *IEEE Transactions on Robotics*, vol.31, no.2, pp. 511-520, 2015

[5] T. Baba, Y. Kameyama, T. Kamegawa, and A. Gofuku, "A snake robot propelling inside of a pipe with helical rolling motion," *SICE Annual Conference*, pp. 2319-2325, 2010

[6] T. Takemori, M. Tanaka, and F. Matsuno, "Gait Design of a Snake Robot by Connecting Simple Shapes," *IEEE International Symposium on Safety, Security, and Rescue Robotics*, pp. 189-194, 2016

[7] D. Rollinson, and H. Choset, "Pipe Network Locomotion with A Snake Robot," *Journal of Field Robotics*, vol. 33, no. 3, pp. 322-336, 2016

[8] J. Gray, "The Mechanism of Locomotion in Snakes," *J. Exp. Biol.*, vol. 23, no. 2, pp. 101-123, 1946

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

[10] P. Prautsch, T. Mita, and T. Iwasaki, "Analysis and control of a gait of snake robot," *Trans. Electr. Electron. J.*, vol. 120-D, pp. 372-381, 2000.

[11] F. Matsuno, and K. Mogi, "Redundancy controllable system and control of snake robots based on kinematic model," *IEEE Conference on Decision and Control*, pp. 4791-4796, 2000.

[12] F. Matsuno, and H. Sato "Trajectory tracking control of snake robots based on dynamic model," *IEEE International Conference on Robotics and Automation*, pp. 3040-3045, 2005.

[13] R. Ariizumi, M. Tanaka, and F. Matsuno, "Analysis and heading control of continuum planar snake robot based on kinematics and a general solution thereof," *Advanced Robotics*, vol. 30, no. 5, pp. 301-314, 2016.

[14] M. Saito, M. Fukaya, and T. Iwasaki, "Serpentine Locomotion with Robotic Snakes," *IEEE Control Syst. Mag.*, vol. 22, no. 1, pp. 64-81, 2002.

[15] R. Ariizumi, and F. Matsuno, "Dynamic Analysis of Three Snake Robot Gaits," *IEEE Transactions on Robotics*, vol. 33, no. 5, pp. 1075-1087, 2017.

[16] P. Liljebäck, K. Pettersen, Ø. Stavdahl, and J. Gravdahl, "Controllability and Stability Analysis of Planar Snake Robot Locomotion," *IEEE Trans. Autom. Control*, vol. 56, no. 6, pp. 1365-1380, 2011.

[17] P. Liljebäck, I.U. Haugstuen, and K.Y. Pettersen, "Path Following Control of Planar Snake Robots Using a Cascaded Approach," *IEEE Trans. Control Syst. Tech.*, vol. 20, no. 1, pp. 111–126, 2012.

[18] P. Liljebäck, K.Y. Pettersen, Ø. Stavdahl, and J.T. Gravdahl, "Lateral undulation of snake robots: a simplified model and fundamental properties," *Robotica*, Available on CJO 2013 doi:10.1017/S0263574713000295

[19] J. Mukherjee, S. Mukherjee, and I. Narayan Kar, "Sliding Mode Control of Planar Snake Robot with Uncertainty Using Virtual Holonomic

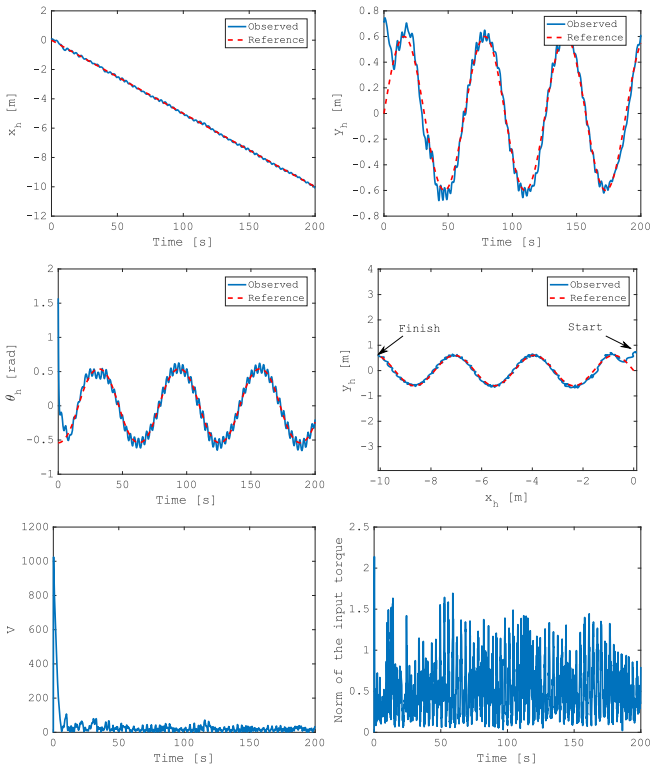
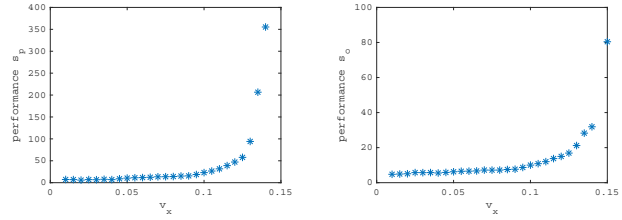



Fig. 10. Result of the simulation with initial tracking error. Simulation was without discrepancy in the model or a free joint (Case 1 with initial tracking error). The sine curve reference is used.




(a) $e_p$ (b) $e_o$

Fig. 11. Tracking performances against various $v_x$.

Fig. 12. Snake robot in V-REP

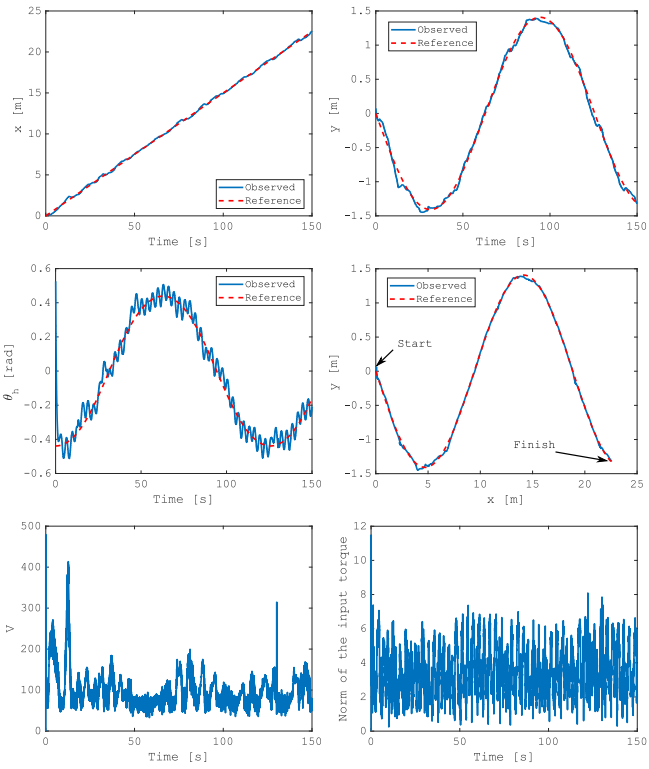
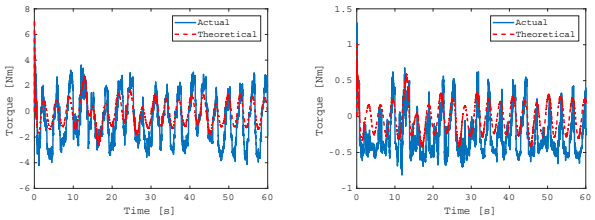



Fig. 13. Result of the simulation using Vortex.




(a) Joint 4 (b) Joint 7

Fig. 14. Torque input calculated by (7) and the actual torque applied to the joints 4 and 7.

Constraints," *IEEE Robotics and Automation Letters*, vol. 2, no. 2, pp. 1077-1084, 2017.

[20] E. Rezapour, K.Y. Pettersen, P. Liljebäck, and J.T. Gravdahl, "Differential Geometric Modelling and Robust Path Following Control of Snake Robots Using Sliding Mode Techniques," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2014, pp. 4532-4539.

[21] V. Mehta, S. Brennan, and F. Gandhi, "Experimentally Verified Optimal Serpentine Gait and Hyperredundancy of a Rigid-Link Snake Robot," *IEEE Transactions on Robotics*, vol. 24, no. 2, 2008.

[22] G. Hicks, and K. Ito, "A Method for Determination of Optimal Gaits With Application to a Snake-Like Serial-Link Structure," *IEEE Trans. Autom. Control,*, vol. 50, no. 9, pp. 1291-1306, 2005

[23] E. Rezapour, K.Y. Pettersen, P. Liljebäck, J.T. Gravdahl, and E. Kelasidi, "Path following control of planar snake robots using virtual holonomic constraints: theory and experiments", *Robotics and Biomimetics*, Vol. 1, No. 1, doi: 10.1186/s40638-014-0003-6, 2014.

[24] A. Mohammadi, E. Rezapour, M. Maggiore, and K.Y. Pettersen, "Maneuvering Control of Planar Snake Robots Using Virtual Holonomic Constraints," *IEEE Trans. Control Syst. Tech.*, vol. 24, no. 3, pp. 884-899, 2016.

[25] E. Rezapour, K.Y. Pettersen, P. Liljebäck, and J.T. Gravdahl, "Body shape and orientation control for locomotion of biologically-inspired snake robots." In *Proc. 5th IEEE RAS/EMBS Int. Conf. on Biomedical Robotics and Biomechatronics*, 2014.

[26] R. Ariizumi, and M. Tanaka, "Manipulability Analysis of a Snake Robot Without Lateral Constraint for Head Position Control," *Asian J. Control* (submitted).

[27] S. Thrun, W. Burgard, and D. Fox, *Probabilistic Robotics*, The MIT Press, 2005.