Approximate Path-Tracking Control of Snake Robot Joints with Switching Constraints

Motoyasu Tanaka, *Member, IEEE,* Kazuo Tanaka, *Fellow, IEEE,* and Fumitoshi Matsuno, *Member, IEEE*

***Abstract*—This paper presents an approximate path-tracking control method for all joints of a snake robot, along with the verification of this method by simulations and experiments. We consider a wheeled snake robot that has passive wheels and active joints. The robot can switch the wheels that touch the ground by lifting the required parts of its body. The model of the robot becomes a kinematically redundant system if certain wheels are lifted. Using this kinematic redundancy, and selecting the appropriate lifted parts, we design a controller for approximate path-tracking. Simulations and experimental results show that the proposed controller effectively reduces the path-tracking error for all joints of the snake robot.**

***Index Terms*—Snake robot, Path-tracking, Switching constraints, Redundancy, Kinematics.**

I. Introduction

SNAKE robots can realize various types of movement and, because of their slim body shapes, are useful for inspecting pipelines and disaster sites. Hirose has produced several snake robots, and models the snake using a wheeled link mechanism with no sideslip (where the snake robot body slides out of its track) [1]. In the wheeled link model, passive wheels are attached to the side of the snake robot body to achieve the friction mechanism of a real snake. This paper deals with a snake robot that moves via body friction and its own bending motion, without generating propulsion using active wheels or active crawlers. The robot locomotes without energy loss by reducing the sideslip of the wheels, because the propulsion of the robot is generated by its bending motion and the constraint force of the wheels. Also, by assuming that the passive wheel does not slide sideways, controllers can be designed to ensure the convergence of the tracking error of the robot's head [2]–[6]. Thus, it is beneficial to limit sideslip from the viewpoint of control theory based on an ideal model. However, if the wheel does not slide in the sideways direction, differences between the paths followed by each link are generated, similar to the difference between the paths of the front and rear wheels in four-wheeled vehicles. Given that snake robots are expected to be useful for searching in confined spaces, this phenomenon would be a drawback. If the trajectories of the robot's body parts (follower links) depart from the path of the robot's head in a narrow space, there is a possibility that the robot will become stuck on obstacles and holes on the ground after the head has steered around them (Fig. 1). Therefore, the robot's follower links should track the path of the head. This paper proposes a control method that accomplishes whole-body path-tracking in the snake robot, i.e., where the follower links of the snake robot track the path of the head. Snake robots can produce thrust by pushing against obstacles and studies have been performed on obstacle-aided locomotion [7]–[9]. However, the robot may encounter a slim cable that causes the robot to stick, breakable walls or high-temperature substances with which the robot should not be in contact, and gaps that make the robot unstable. Thus, both obstacle-aided locomotion and obstacle avoidance are required, and this paper focuses on whole-body path-tracking processes related to obstacle avoidance.

Conkur proposed a path-tracking method for a whole highly redundant manipulator along B-spline curves [10]. The redundant manipulator has a fixed end, and all joints can rotate freely. However, the method of [10] cannot be applied directly to a snake robot, because the snake robot structure is unlike that of the redundant manipulator, in that the snake robot does not have a fixed end, and the rotation of its joints is limited by the wheel.

Path-tracking control of an articulated vehicle with active wheels was proposed in [11], [12], and a snake-like robot with a screw-drive unit was reported in [13]. In contrast, a snake robot without active wheels generates its propulsion force by actuating its joints to produce a bending motion, and the principle of locomotion of the snake robot described in this paper is different to that of robots with active units that generate their propulsion force without any bending motion of the body [11]–[13]. For the path-tracking control of snake robots, Liljebäck et al. [14] proposed a path-tracking controller for the robot's center of gravity, but no previous research has focused on the path-tracking of all links.

Methods that considered motion along a reference path by treating the snake robot as a continuous curve were proposed in [15]–[19]. However, these approaches require discretization, which introduces discretization errors when the method is implemented in the robot. Additionally, these methods do not ensure the convergence of the tracking error of the robot's head, and the trajectories of the head and the follower links may depart from the reference path because of the accumulation of such errors.

Path-tracking methods for articulated trailers were proposed in [20], [21]. However, it is difficult to apply these methods to snake robots because of the differences between the mechanisms used and their degrees of freedom. Snake robots can dynamically switch the grounding and non-grounding of their wheels by lifting up some parts of their body, and have a

M. Tanaka and K. Tanaka are with the Department of Mechanical Engineering and Intelligent Systems, The University of Electro-Communications, Tokyo 182-8585, Japan. (e-mail: mtanaka@uec.ac.jp).

F. Matsuno is with Graduate School of Engineering, Kyoto University, Kyoto 615-8540, Japan. (e-mail:matsuno@me.kyoto-u.ac.jp).

Manuscript received Month Day, 20XX; revised Month Day, 20XX.

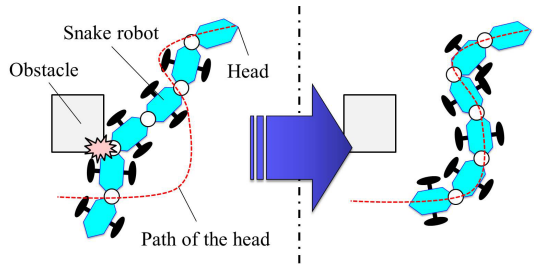


Fig. 1. Path-tracking of whole body of a snake robot.

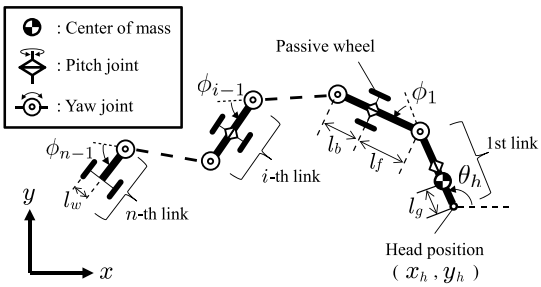


Fig. 2. An n-link snake-like robot [6].

greater degree of freedom than articulated trailers. We have proposed a controller that tracks the trajectory of the robot's head and avoids obstacles by switching the parts of its body that are grounded [6]. If the robot makes efficient use of this switching mechanism, then it is possible to reduce the path-tracking error of the follower links.

This paper aims to achieve whole-body path-tracking in a snake robot. Specifically, we present an approximate path-tracking controller for all joints of the robot using kinematic redundancy and switching of the grounded wheels, and report the results of simulations and experiments performed to determine the effectiveness of the proposed controller.

II. Model

We consider an n-link wheeled snake robot, as shown in Fig. 2. Only a brief description of the model is given here, because the mechanism and the mathematical model of the snake robot have been reported previously [6].

A link comprises a pitch rotational joint and a pair of passive wheels mounted coaxially, and is connected to another link via a yaw rotational joint. All wheels are passive and all joints are active. We assume that a passive wheel does not slide in the sideways direction and that the environment is flat. Although the wheels do actually slide sideways and the model error is caused, it was verified previously that the feedback controller enabled the controlled variables to converge to the desired value [5], [6] . Also, the risk of slipping sideways can be reduced by applying a controller that considers the constraint force of the wheels [22], [23] . l_f is the distance from the anterior end of the link to the wheel axis, l_b is the distance from the posterior end of the link to the wheel axis, l_w is the distance from the center line of a link to the center of a wheel, and l_g is the distance from the anterior end of the link to the center of gravity of the link. Let $\boldsymbol{\psi} = [\psi_1, \cdots, \psi_{n-1}]^T$ be

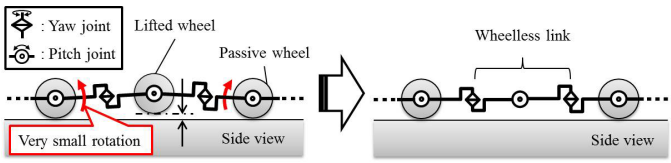


Fig. 3. Lifting links and wheelless links.

the pitch joint angles of each link, and $\boldsymbol{\phi} = [\phi_1, \cdots, \phi_{n-1}]^T$ be the yaw joint angles of the links. The lifting links of the snake robot are regarded as wheelless links of the two-dimensional snake robot, and the grounded links are regarded as the wheeled links (Fig. 3). Additionally, the pitch joint angles are only used for switching of the grounded links and we assume that $\boldsymbol{\psi} \simeq \mathbf{0}$ as shown in Fig. 3, i.e., that the motion of the pitch joints does not affect the position and attitude of the robot's head and links, because we set the desired pitch angle values to be small and the dynamic influence of the motion of the pitch joints to be negligible. We therefore model the lifting motion of the snake in three-dimensional space by switching the wheeled link to the wheelless link in the two-dimensional motion. Note that the pitch joint angles are not explicitly included in the model.

Let $\boldsymbol{w} = [x_h, y_h, \theta_h]^T$ give the position and attitude of the snake's head, and $\boldsymbol{q} = [\boldsymbol{w}^T, \boldsymbol{\phi}^T]^T$ be generalized coordinates. The snake robot in this paper is represented as a hybrid system that switches discretely because the grounded links switch, i.e., the velocity constraints are caused by the three-dimensional motion of switching of the grounded/non-grounded wheels. We assume that the time required to lift and ground the wheels is infinitesimal. Let ΔT be the switching time period. The system of the snake robot with switching constraints is then expressed as

$$\begin{aligned} \tilde{A}_\sigma(\boldsymbol{q})\dot{\boldsymbol{w}} &= \tilde{B}_\sigma(\boldsymbol{q})\boldsymbol{u} \\ \sigma(t) &= \sigma_k, \quad \forall t \in [t_k, t_{k+1}), \end{aligned} \tag{1}$$

where $\boldsymbol{u} = \dot{\boldsymbol{\phi}}$ is the joint input, $\sigma \in M$ is the discrete mode number, $M = \{1, 2, \cdots, N_m\}$, and $t_k = k\Delta T (k = 0, 1, 2, \cdots)$ is the switching time. The mode σ switches instantaneously at $t = t_k$, and is retained while $t_k \leq t < t_{k+1}$. In mode σ, the passive wheel of the head link is removed to control the position and attitude of the snake head at the same time, and the number of non-grounded links m_σ satisfies $(n - m_\sigma) \geq 3$ to ensure the uniqueness of the solution of the system [3].

By considering the first link to be wheelless and taking $n - m_\sigma \geq 3$, N_m is obtained as

$$\begin{aligned} N_m &= {}_{n-1}\mathrm{C}_{n-1} + {}_{n-1}\mathrm{C}_{n-2} + \cdots + {}_{n-1}\mathrm{C}_3 \\ &= \sum_{j=3}^{n-1} {}_{n-1}\mathrm{C}_j \end{aligned} \tag{2}$$

where ${}_i\mathrm{C}_j$ is the number of combinations from i to j and ${}_i\mathrm{C}_j = \frac{i!}{j!(i-j)!}$. The first link is wheelless and the robot can select each of the other links to be wheeled or wheelless. (2) is the number of combinations for selection of wheeled or wheelless links.

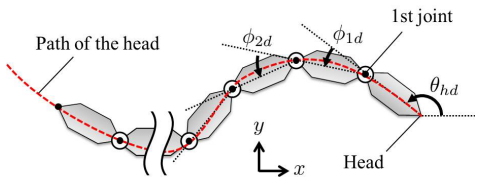


Fig. 4. The desired attitude of the head and the desired angles of all joints.

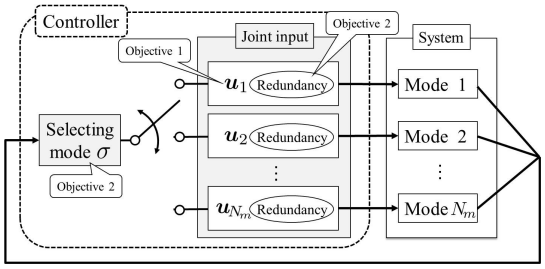


Fig. 5. Control strategy.

III. Controller design

We design a controller to track the path of the whole body of the snake robot, as shown in Fig. 1. Let $\boldsymbol{w}_d = [x_{hd}, y_{hd}, \theta_{hd}]^T$ be the desired value of $\boldsymbol{w}$. We assume that the desired attitude of the head $\theta_{hd}(t)$ and the desired angles of all joints $\boldsymbol{\phi}_d(t)$ to accomplish path-tracking at time t are given as shown in Fig. 4, i.e. where each joint is passing through the desired trajectory. If $\boldsymbol{w} \to \boldsymbol{w}_d$ and $\boldsymbol{\phi} \to \boldsymbol{\phi}_d$ are satisfied, then all joints follow the same path as the head. Therefore, we set the control objectives as follows:

1. The position and attitude of the head $\boldsymbol{w} \to \boldsymbol{w}_d$.
2. The yaw joint angles $\boldsymbol{\phi} \to \boldsymbol{\phi}_d$.

Using the controller proposed in [3], [6], $\boldsymbol{w}$ will converge to the desired trajectory. In contrast, the control of $\boldsymbol{\phi}$ has not been considered in the previous research.

In the case of the snake-like robot with the screw-drive mechanism, all the robot's joints can track the path of the freely moving head because the robot can move omnidirectionally and has a sufficient degree of freedom [13]. However, the joint angles of the snake robot treated in this paper cannot all converge completely to the reference path of the head, because the robot has non-holonomic constraints caused by the passive wheels and an insufficient number of inputs (i.e. yaw joints) that can be used for locomotion. By lifting some of the wheels, the robot generates kinematic redundancy and increases the number of controllable joints, but the relationship between the number of controlled variables and the number of wheels means that more than three links must be touching the ground [3]. Therefore, the robot cannot ensure that all joint angles completely and simultaneously converge to the reference.

In [6], we proposed a controller that uses "kinematic redundancy" and "mode selection" by lifting up some wheels dynamically, thus accomplishing both trajectory tracking of the head and moving obstacle avoidance. This paper applies the controller of [6] to the path-tracking of all robot joints. Specifically, we approximately accomplish control objective 2 using kinematic redundancy and mode selection so that all joint angles realize the desired motion as closely as possible. The control strategy is shown in Fig. 5. Objective 1 and 2 in Fig. 5 mean the control objective 1 and 2, respectively. The controller consists of the joint input and the mode selection. The joint input is the angular velocity of the yaw angles of the robot, and the mode selection relates to a constraint condition on the wheels. The joint input contains the kinematic redundancy, and is like a parallel distributed controller for each mode. The differences from the controller in [6] are the joint input and the method of mode selection, and they occur because of the difference in the control objective.

A. Design of joint input $\boldsymbol{u}$

In system (1), we set the input $\boldsymbol{u}$ as

$$\begin{aligned}\boldsymbol{u}(t) &= \boldsymbol{u}_{\sigma(t)} \\ &= \tilde{B}_\sigma^\dagger \tilde{A}_\sigma \{\dot{\boldsymbol{w}}_d - K(\boldsymbol{w} - \boldsymbol{w}_d)\} + (I - \tilde{B}_\sigma^\dagger \tilde{B}_\sigma)\dot{\boldsymbol{\phi}}_r, \quad (3)\end{aligned}$$

where $\boldsymbol{u}_\sigma$ is the input for the σ-th mode, $\tilde{B}_\sigma^\dagger$ is a pseudo-inverse matrix of $\tilde{B}_\sigma$, $K > 0$ is the feedback gain, and $\dot{\boldsymbol{\phi}}_r$ is the velocity input for kinematic redundancy.

By substituting (3) into (1), the closed-loop system is expressed as

$$\tilde{A}_\sigma \{\dot{\boldsymbol{e}} + K\boldsymbol{e}\} = \boldsymbol{0}, \quad (4)$$

where $\boldsymbol{e} = \boldsymbol{w} - \boldsymbol{w}_d$. If the matrix $\tilde{A}_\sigma$ is of full column rank (i.e., the robot does not have a singular configuration), then the solution is guaranteed to be unique. The solution of (4) is given as

$$\dot{\boldsymbol{e}} + K\boldsymbol{e} = \boldsymbol{0}, \quad (5)$$

and $\boldsymbol{w}$ converges to the desired trajectory $\boldsymbol{w}_d$. Note that (5) is independent of the mode σ. The state variable $\boldsymbol{w}$ to be controlled converges on its desired vector $\boldsymbol{w}_d$ exponentially, and does not depend on the switching constraints unless the robot attains a singular configuration.

The second term on the right-hand side of (3) expresses the kinematic redundancy, and approximates $\boldsymbol{\phi}_d$ as closely as possible by designing $\dot{\boldsymbol{\phi}}_r$ as follows [24]:

$$\dot{\boldsymbol{\phi}}_r = \dot{\boldsymbol{\phi}}_d - K_r(\boldsymbol{\phi} - \boldsymbol{\phi}_d), \quad (6)$$

where $K_r > 0$ is a gain for redundancy.

B. Mode selection by designing σ

We consider the reduction of the path-tracking errors of all joints by selecting an appropriate mode σ. We use three conditions in [6] for static stable locomotion under the effects of the switching modes. At $t = t_k$, let $c(t_k)$ be the set of grounded links, $C_G(t_k)$ be the center of gravity of the whole body of the snake robot, and $P(\boldsymbol{q}(t_k); c(t_k))$ be the supporting polygon area constructed by the passive wheels of the grounded links. We introduce the conditions

$$C_G(t_k) \in P(\boldsymbol{q}(t_k); c(t_k) \cap c(t_{k-1})) \quad (7)$$

$$\hat{C}_G(t) \in P(\hat{\boldsymbol{q}}(t); c(t_k)), \quad t_k \le t < t_{k+1} \quad (8)$$

$$\hat{k}(t) > c_s, \quad t_k \le t < t_{k+1}, \quad (9)$$

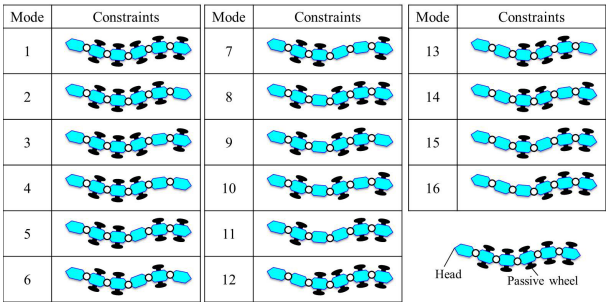


Fig. 6. Various constraint situations and modes ($n = 6$).

where $k(t) = \det(\tilde{A}^T_{\sigma(t)}\tilde{A}_{\sigma(t)})$, c_s is a minimal positive value, and $\hat{C}_G$, $\hat{k}$, and $\hat{\boldsymbol{q}}$ are the estimated values of C_G, k, and $\boldsymbol{q}$, respectively. The estimated value $\hat{\boldsymbol{q}}$ is calculated according to (3) and the solution of (5), and $\hat{C}_G$ and $\hat{k}$ are calculated using $\hat{\boldsymbol{q}}$. The condition (7) means that the center of gravity of the robot is contained in the supporting polygon constructed by the common grounded links $(c(t_k) \cap c(t_{k-1}))$ before and after switching. Condition (8) means that the snake robot is statically stable while $t_k \leq t < t_{k+1}$. The static stability of the snake robot's gait is ensured, and impractical switching (e.g., where all the grounded links are switched to lifted links or vice versa) is avoided by introducing conditions (7) and (8). Condition (9) means that $\tilde{A}_\sigma$ has full column rank, i.e., the snake robot does not attain a singular configuration while $t_k \leq t < t_{k+1}$.

Next, we consider a cost function $V(t)$ for selection of the correct mode. We set the position of the i-th joint and its desired vector to $\boldsymbol{p}_i$ and $\boldsymbol{p}_{id} \in \mathbf{R}^2$, respectively, at time t. $\boldsymbol{p}_i$ is geometrically calculated using $\boldsymbol{w}$ and $\boldsymbol{\phi}$, and $\boldsymbol{p}_{id}$ is computed from $\boldsymbol{w}_d$ and $\boldsymbol{\phi}_d$. We set the norm of the positional error of joint i to $f_i = ||\boldsymbol{p}_i - \boldsymbol{p}_{id}|| \in \mathbf{R}^1$ and $\boldsymbol{f} = [f_1, f_2, \cdots, f_n]^T \in \mathbf{R}^n$. Then, the cost function V is calculated as

$$V(t) = ||\boldsymbol{f}(t)||. \tag{10}$$

We set $V_\sigma(t)$ to be the value of $V(t)$ in mode σ, and formulate the decrease in V as a finite-time optimal mode selection problem at $t = t_k$:

$$\min_{\sigma} \quad \hat{V}_\sigma(t_{k+1}) \tag{11}$$
$$\text{subject to} \quad \text{Eqs.}(7), (8) \text{ and } (9)$$

where $\hat{V}_\sigma(t_{k+1})$ is the estimated value of V_σ at $t = t_{k+1} = t_k + \Delta T$, and σ_k should be selected as the optimal mode that attains the minimum value of (11). The optimal control problem (11) is solved at $t = t_k$, and σ_k is selected as $\sigma(t)$ at $t < t_{k+1}$. σ_k is calculated based on a full search of all combinations of the switching mode. Specifically, we set $\sigma = \sigma_k^* \in M$ at $t = t_k$, calculate the estimated value $\hat{V}_{\sigma_k^*}(t_{k+1})$, and select the optimal σ_k^* as σ_k. We can expect V to decrease by selecting the mode according to (11).

IV. Simulations

Simulations were carried out to demonstrate the effectiveness of the proposed controller. We consider a six-link snake robot where $l_f = l_b = l_g = 0.088$ m and $l_w = 0.043$ m, and we set the relationship between the various constraint situations and the mode number as per Fig. 6. We use a *serpenoid curve* [1] as the desired path. The curvature κ of a serpenoid curve can be represented as

$$\kappa(s) = -2\pi\alpha T \sin(2\pi T s), \tag{12}$$

where s is the arc length of the curve, T is the period of the curve, and the winding angle α is defined as the maximum angle between the body shape and the direction of movement [25]. By integrating (12) with respect to s, the x, y coordinates of the serpenoid curve are obtained as follows:

$$\nu(s) = \nu(0) + \int_0^s \kappa(s)ds, \tag{13}$$
$$x(s) = x(0) + \int_0^s \cos\nu(s)ds, \tag{14}$$
$$y(s) = y(0) + \int_0^s \sin\nu(s)ds. \tag{15}$$

We then set the parameters of the serpenoid curve to $\alpha = 1$ rad, $T = 1.5$, $x(0) = y(0) = 0$, $\nu(0) = \alpha$, and $s = 0.0075t$, and calculate the desired position (x_{hd}, y_{hd}) according to (12)–(15). We set the initial position of each joint to lie on the serpenoid curve, as in Fig. 7.

Obtaining $\theta_{hd}(t)$ and $\phi_d(t)$ when all joints are on the serpenoid curve requires the solution of complex nonlinear equations, and the computational load makes it difficult to obtain an accurate solution in real time. Accordingly, we assume that the head of the robot tracks the desired position (x_{hd}, y_{hd}) completely. $\theta_{hd}(t)$ and $\phi_d(t)$can be obtained by using a

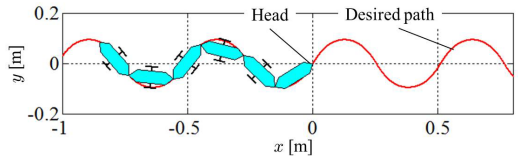
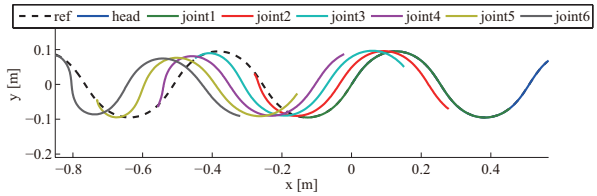
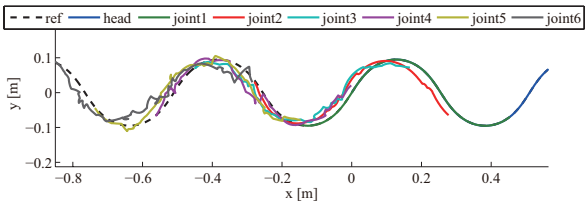


Fig. 7. The desired path and the initial posture.

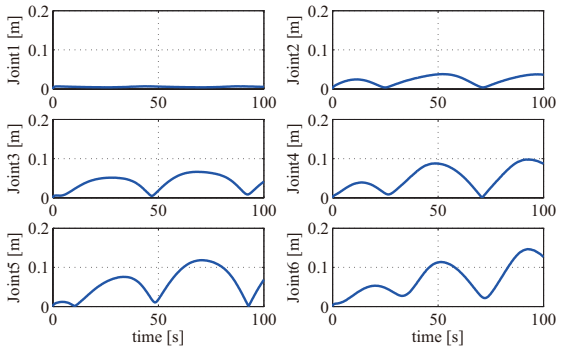


(a) Case 1

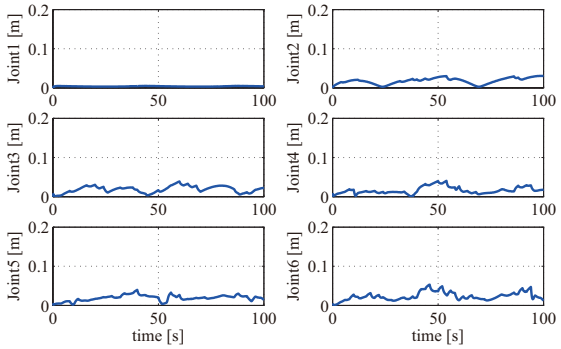


(b) Case 2

Fig. 8. The paths of the joints in the simulation.

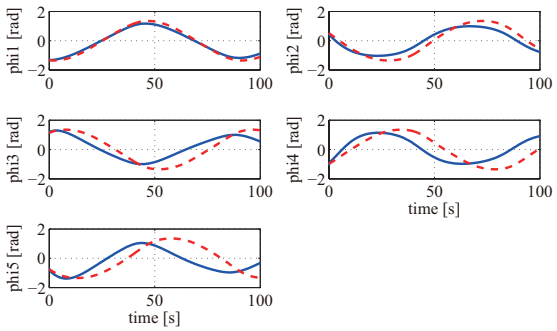


(a) Case 1

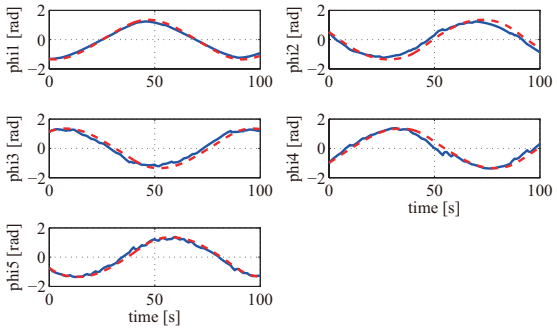


(b) Case 2

Fig. 9. The norm of the positional error of joint f_i in the simulation.



(a) Case 1



(b) Case 2

Fig. 10. Time responses of the joint angles in the simulation.

bisection search method [26]. In this paper, we obtain $\theta_{hd}(t)$ and $\boldsymbol{\phi}_d(t)$ using a bisection search in off-line calculations based on the desired trajectory of the head $(x_{hd}(t), y_{hd}(t))$, and prepare a look-up table of these values. The table has θ_{hd} and $\boldsymbol{\phi}_d$ for each sampling time of simulations and experiments. θ_{hd} and $\boldsymbol{\phi}_d$ can be obtained by using the table at each sampling time. Additionally, we define $K = 2$, $K_r = 1$, and $\Delta T = 2$ s as the controller parameters for the simulations.

Case 1 uses a fixed mode ($\sigma = 1$) without switching, and Case 2 uses the proposed controller. Fig. 8 shows all paths of the joints, Fig. 9 shows the norm of the positional error of each joint f_i, and Fig. 10 shows the time responses of the joint angles, where the broken line indicates the reference path.

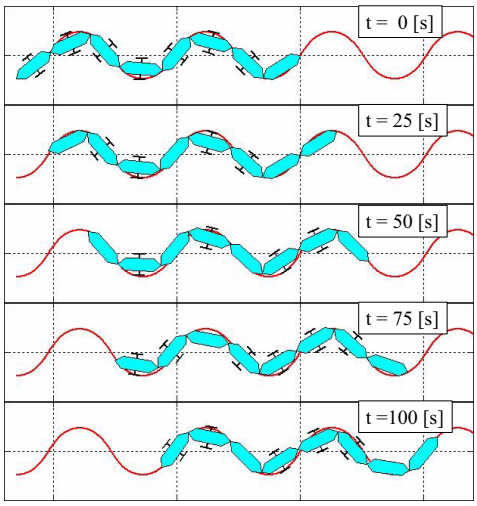
In Case 1, the path of the follower joints departs from the desired path, as shown in Fig. 8 (a), and the f_i value of the rear links was larger than that of the forward links, as shown in Fig. 9 (a). In contrast, in Case 2, f_i was lower when compared with Case 1, as shown in Figs. 8 (b) and 9 (b). From Fig. 10, we see that the joint angle in Case 2 remains closer to the reference path than that in Case 1.

Additionally, we consider the case where the number of links of the robot increases. If we apply the controller to the actual robot, the number of links that the pitch joint can lift has a limit because the actuator has a torque limit. Therefore, inappropriate modes should be excluded by considering the limit to the number of links that the pitch joint can lift. We consider the case where $n = 8$ and exclude the inappropriate modes where more than three consecutive wheels are lifted. Thus, the number of modes becomes 67. Fig. 11 shows simulation results for $n = 8$. It was found that the robot could achieve approximate path-tracking of all joints by selecting the appropriate mode. Although the inappropriate modes are excluded, the path-tracking error of all joints is low because the number of modes is sufficiently large.

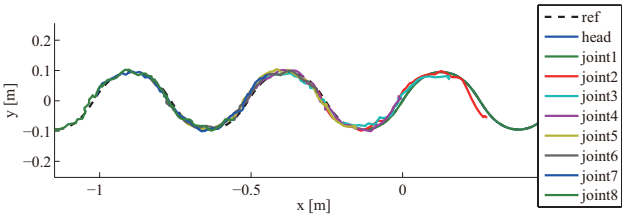
As a result, we found that the robot could achieve a reduction in the path-tracking error of all joints using the proposed controller with optimal switching.

V. Experiments

Experiments were carried out to determine the effectiveness of the proposed control strategy. The experimental system is shown in Fig. 12. The snake robot has links on which the passive wheels and the pitch joints are mounted, and the links are connected by yaw joints, where l_f, l_b, l_g, and l_w are as they were in the simulations. The pitch and yaw joints use Dynamixel MX-64R (ROBOTIS, Irvine, CA, USA) actuators. The yaw joints are used for propulsion in the x, y plane and correspond to ϕ_i in Fig. 2, and the pitch joints are used to switch modes. The real-time position and attitude of the robot's head $\boldsymbol{w}$ were measured by OptiTrack, an optical motion-capture system with tracking software (NaturalPoint, Inc.), and the real-time joint angles $\boldsymbol{\phi}$ were measured by the absolute encoder in the Dynamixel MX-64R. All joint positions were geometrically calculated using the measured values of $\boldsymbol{w}$ and $\boldsymbol{\phi}$. A PC was used to calculate the input and control the actuators of the robot according to the proposed



(a) The motion of the robot.



(b) Paths of the joints.

Fig. 11. Simulation results in the case where $n = 8$.

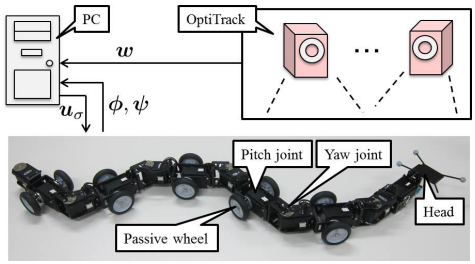


Fig. 12. Experimental system.

controller, and the PC and actuators were connected with a daisy chain via an RS485 interface.

A. Implementation of switching constraints [6]

Switching constraints are generated by the motion of the pitch joints of each link, as discussed in [6]. Fig. 13 shows a model of the lifting motion of the snake robot. Let ψ_i be the angle of the pitch joint of the i-th link and ψ_{id} be the desired value of ψ_i. The wheel of the i-th link can be lifted using the pitch joints of the $i, i-1, i+1$-th links. We set ψ_{id} according to Table I to switch modes depending on the constraint conditions (grounding/lifting) of the $i, i-1, i+1$-th links, where ϵ is a small value that satisfies the assumption $\boldsymbol{\psi} \simeq \boldsymbol{0}$ mentioned in section II. Using a proportional-integral controller, we set the angular velocity of the pitch joint $\dot{\psi}_i$ as:

$$\dot{\psi}_i = -K_P(\psi_i - \psi_{id}) - K_I \int (\psi_i - \psi_{id})dt, \qquad (16)$$

TABLE I
CONSTRAINT CONDITIONS AND ψ_{id}.

i-th link	$(i-1)$-th link	$(i+1)$-th link	ψ_{id}
grounded	lifting	lifting	2ϵ
	lifting	grounded	ϵ
	grounded	lifting	ϵ
	grounded	grounded	0
lifting	lifting	lifting	0
	lifting	grounded	$-\epsilon$
	grounded	lifting	$-\epsilon$
	grounded	grounded	-2ϵ

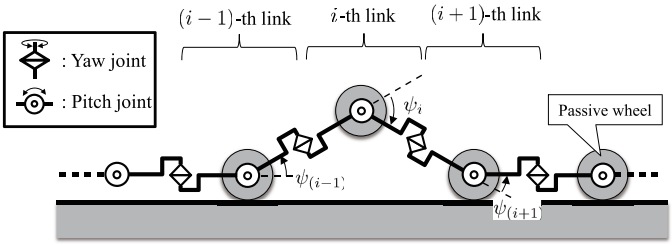


Fig. 13. Model of lifting motion using pitch joints [6].

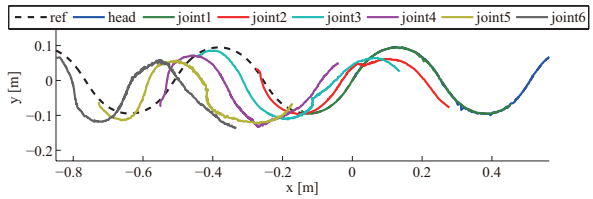
where $K_P, K_I > 0$ are the feedback gains.

VI. EXPERIMENTAL RESULTS AND DISCUSSION

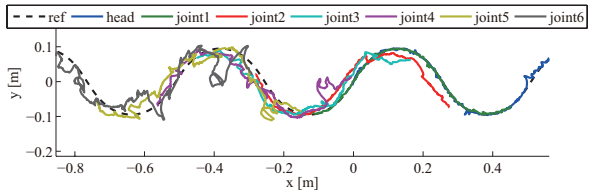
We set $\boldsymbol{\phi}(0)$, $\boldsymbol{w}_d$, $\boldsymbol{\phi}_d$, ΔT, and the gains to be equal to their respective values in the simulations, and fixed $K_P = 4$, $K_I = 0.1$, and $\epsilon = \pi/200$. Case 1 again considers a fixed mode ($\sigma = 1$) without switching, and Case 2 uses the proposed controller.

Fig. 14 shows all paths of the joints, Fig. 15 shows the motion of the robot, Fig. 16 shows the norm of the positional error of each joint f_i, Fig. 17 shows the time responses of the joint angles, Fig. 18 shows the time responses of $\boldsymbol{w}$ and σ in Case 2, and Fig. 19 shows the time response of $\boldsymbol{e} = \boldsymbol{w} - \boldsymbol{w}_d$ in Case 2, where the broken line is the reference path.

From Figs. 14 (a) and 16 (a), we can see that the path of the rear joints departed from the desired path in Case 1, and that f_i of the rear links was larger than that of the forward links, particularly in the sixth joint, which had the maximum value of 0.19 m. In contrast, in Case 2, f_i was reduced when compared with Case 1, and the maximum value was 0.12 m in the sixth joint, as shown in Figs. 14 (b) and 16 (b). From Fig. 17, we can see that the joint angle in Case 2 was closer to the reference than in Case 1. As shown in Fig. 18, $\boldsymbol{w}$ tracked the reference path. However, from Fig. 19, it can be seen that the errors in x_h and y_h became larger for $t \in [86, 88]$ s. We believe that the errors in x_h and y_h are modeling errors caused by the wheels sliding to the side in spite of the assumption that the wheel does not slide in the sideways direction. Moreover, f_4 and f_6 were larger for $t \in [86, 95]$ s in Case 2. If the errors of x_h and y_h increase, the first term of the input (3) increases to make the errors converging to zero. In contrast to the second term of (3), the first term sometimes increases f_i. We believe that the cause of this increase in f_i is the first term which makes the errors in x_h and y_h converging to zero. If the robot reduces its head error by reducing the wheel-slide, then

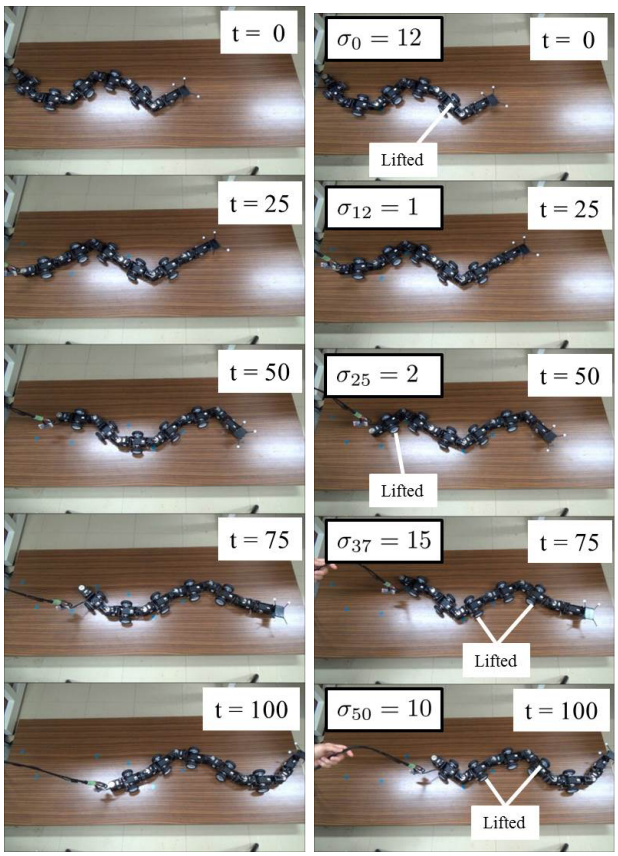


(a) Case 1



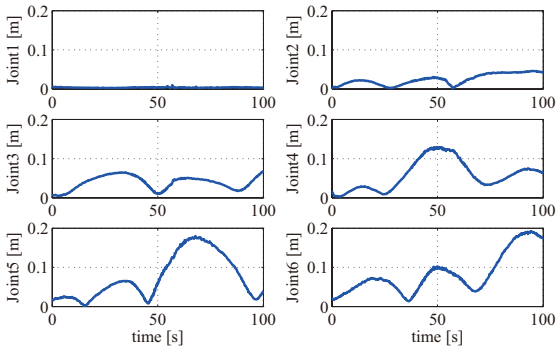
(b) Case 2

Fig. 14. Paths of the joints in the experiment.

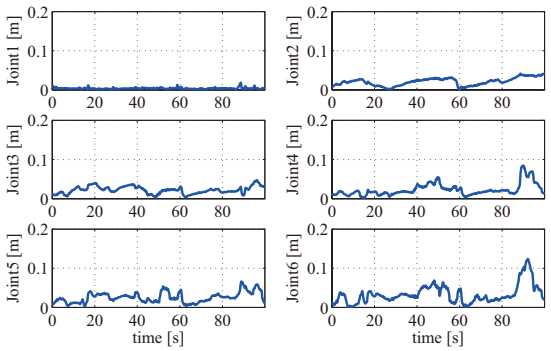


(a) Case 1 (b) Case 2

Fig. 15. Motion of the robot in the experiment.



(a) Case 1



(b) Case 2

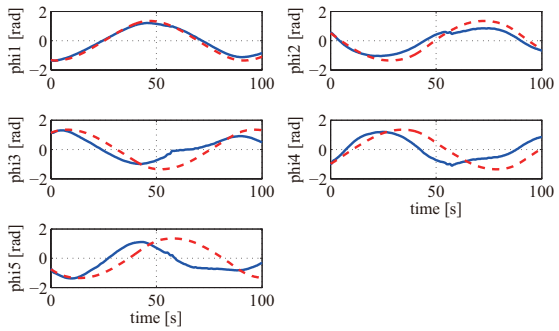
Fig. 16. Norms of the positional errors of the joints f_i in the experiment.

we could expect to obtain the same results as we obtained in the simulations.

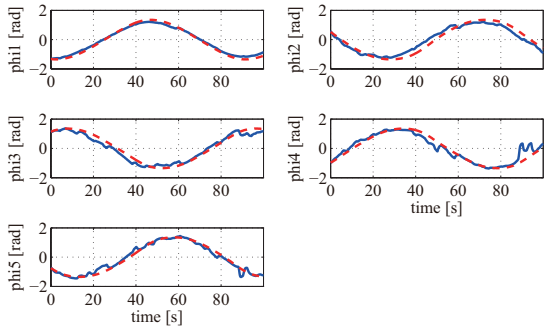
The value of σ switched dynamically, as shown in Fig. 18. Mode 16 was never selected. We believe that the reason for this is that the wheels of only the rear three links touch the ground in mode 16 and the robot cannot maintain static stability during motion based on the serpenoid curve. From Fig. 18, it was found that impractical switching could be avoided at each switching time because of conditions (7) and (8). For example, in the case of switching from $\sigma_{11} = 11$ to $\sigma_{12} = 1$ at $t = 24$, the common grounded links before and after switching are the second, fifth, and sixth links, and we find that the robot selected the mode when the supporting polygon P became larger to satisfy (7). Fig. 20 shows the time responses of d and $\hat{k}$, where d is the minimum distance between the center of gravity of the robot and the supporting polygon. We can see that d and $\hat{k}$ jumped at each switching time but did not become zero, although the minimum value of d is 0.002 m. This means that the robot was statically stable and did not become a singular configuration. Thus, it was confirmed that conditions (8) and (9) worked well.

As a result, although the effect was smaller than that in the simulations, it was found that the robot could reduce the path-tracking errors of all joints using the proposed control method. The proposed method is both effective and significant because, unlike traditional controllers, it can approximately track the paths of all of the snake robot's joints.

In the case of locomotion in a real environment, e.g., where the snake robot is operating in a narrow space, the pre-calculated ϕ_d used in this paper is not applicable, because the trajectory of the robot's head changes as necessary. The real-time calculation of ϕ_d, which is obtained by solving nonlinear equations for the path of the head, is difficult because of the computational load. This problem can be solved by applying a discrete approximation method for a continuous curve [17] to the calculation of ϕ_d. However, when this approximate method



(a) Case 1



(b) Case 2

Fig. 17. Time responses of the joint angles in the experiment.

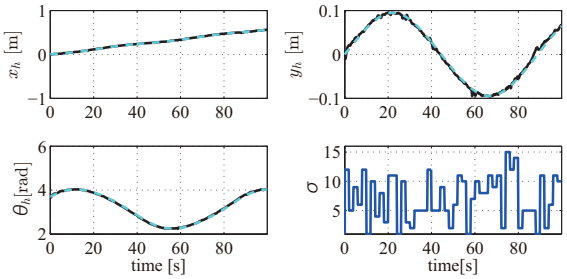


Fig. 18. Time responses of $\boldsymbol{w}$, σ in the experiment (Case 2).

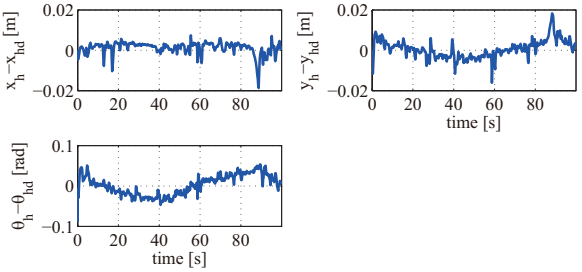


Fig. 19. Time response of $\boldsymbol{w} - \boldsymbol{w}_d$ in the experiment (Case 2).

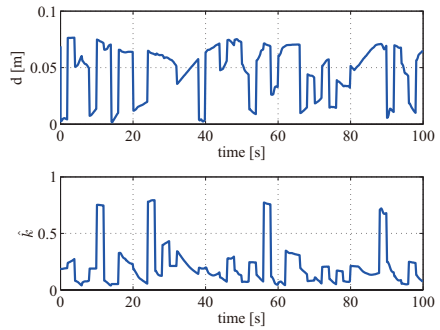


Fig. 20. Time responses of d and $\hat{k}$ in the experiment (Case 2).

is used to calculate $\boldsymbol{\phi}_d$, it becomes difficult to separate the path-tracking error caused by the proposed controller from that caused by the approximate method. Therefore, we carried out simulations and experiments using a look-up table of accurate values of $\boldsymbol{\phi}_d$ to demonstrate the path-tracking error that is due solely to the proposed controller.

Any increase in the number of links leads to a large increase in the computational cost of mode selection, making it difficult to apply the proposed controller to a robot that has more than seven links, as discussed in [6]. In future work, we will aim to reduce the calculation costs of this system.

VII. Conclusion

This paper has presented an approximate path-tracking controller for all joints of a snake robot. We designed the controller to reduce the path-tracking errors of all joints by tracking the trajectory of the robot's head using kinematic redundancy and mode-switching. Simulations and experimental results demonstrated the effectiveness of the proposed controller.

Further studies are required to reduce the calculation time for the mode selection process, to apply a real-time method to the calculation of the desired angles $\boldsymbol{\phi}_d$, and to expand the control method to three-dimensional environments.

Acknowledgment

This work was supported by JSPS KAKENHI Grant Number 26870198.

References

[1] S. Hirose, Biologically Inspired Robots (Snake-like Locomotor and Manipulator), Oxford University Press, 1993.
[2] P. Prautsch, T. Mita, and T. Iwasaki, "Analysis and Control of a Gait of Snake Robot," *Trans. of IEEJ*, vol.120-D, pp.372–381, 2000.
[3] F. Matsuno and K. Mogi, "Redundancy Controllable System and Control of Snake Robot with Redundancy based on Kinematic Model," *Proc. IEEE Conf. on Decision and Control*, pp. 4791–4796, 2000.
[4] S. Ma, Y. Ohmameuda, K. Inoue, and B. Li, "Control of a 3-Dimensional Snake-like Robot," *Proc. IEEE Int. Conf. on Robotics and Automation*, pp.2067–2072, 2003.
[5] M. Tanaka and F. Matsuno, "Modeling and Control of Head Raising Snake Robots by Using Kinematic Redundancy," *Journal of Intelligent and Robotic Systems*, vol.75, no.1, pp.53–69, 2014.
[6] 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.
[7] H. Date and Y. Takita, "Adaptive Locomotion of a Snake Like Robot Based on Curvature Derivatives," *Proc. IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, pp.3554–3559, 2007.
[8] A. Transeth, R. Leine, C. Glocker, K. Pettersen, and P. Liljebäck, "Snake Robot Obstacle-Aided Locomotion: Modeling, Simulations, and Experiments," *IEEE Trans. on Robotics*, vol.24, no.1, pp.88–104, 2008.

[9] T. Kano, T. Sato, R. Kobayashi, and A. Ishiguro, "Local reflexive mechanisms essential for snakes' scaffold-based locomotion," *Bioinspiration& Biomimetics*, vol. 7, 046008, pp.1–11, 2012.
[10] E. S. Conkur, "Path following algorithm for highly redundant manipulators," *Robotics and Autonomous Systems*, vol.45, no.1, pp.1–22, 2003.
[11] T. Kamegawa, T. Yamasaki, H. Igarashi, and F. Matsuno, "Development of The Snake-like Rescue Robot'KOHGA'," *IEEE Int. Conf. on Robotics and Automation*, pp. 5081–5086, 2004.
[12] B. Murugendran, A. A. Transeth, and S. A. Fjerdingen, "Modeling and Path-following for a Snake Robot with Active Wheels," *Proc. IEEE/RSJ Int. Conf. on Intell. Robots and Syst.*, pp.3643–3650, 2009.
[13] H. Fukushima, S. Satomura, T. Kawai, M. Tanaka, and F. Matsuno, "Modeling and Control of a Snake-like Robot Using the Screw Drive Mechanism," *IEEE Trans. on Robotics*, vol.28, no.3, pp.541–554, 2012.
[14] P. Liljebäck, I. U. Haugstuen, and K. Y. Pettersen, "Path Following Control of Planar Snake Robots Using a Cascaded Approach," *IEEE Trans. on Cont. Sys. Tech*, vol.20, no.1, pp.111–126, 2012.
[15] G. S. Chirikjian and W. Burdick, "The Kinematics of Hyper-Redundant Robot Locomotion," *IEEE Trans. on Robotics and Automation*, vol.11, no.6, pp.781–793, 1995.
[16] H. Date and Y. Takita, "Control of 3D snake-like locomotive mechanism based on continuum modeling," *Proc. ASME2005 Int. Design Engineering Technical Conf.*, no. DETC2005-85130, 2005.
[17] H. Yamada and S. Hirose, "Study on the 3D Shape of Active Cord Mechanism," *IEEE Int. Conf. on Robotics and Automation*, pp.2890–2895, 2006.
[18] R. L. Hatton and H. Choset, "Generating gaits for snake robots: annealed chain fitting and keyframe wave extraction," *Autonomous Robots*, vol.28, pp.271–281, 2010.
[19] T. Kamegawa, T. Baba, and A. Gofuku, "V-shift control for snake robot moving the inside of a pipe with helical rolling motion," *Proc. IEEE Int. Symp. on Safety, Security and Rescue Robotics*, pp.1–6, 2011.
[20] M. Sampei, T. Tamura, T. Kobayashi, and N. Shibui, "Arbitrary Path Tracking Control of Articulated Vehicles Using Nonlinear Control Theory," *IEEE Trans. Cont. Sys. Tech*, vol.3, pp.125–131, 1995.
[21] C. Altafini, "Path Following With Reduced Off-Tracking for Multibody Wheeled Vehicles," *IEEE Trans. Cont. Sys. Tech*, vol.11, pp.598–605, 2003.
[22] H. Date, Y. Hoshi, M. Sampei, and S. Nakaura, "Locomotion Control of a Snake Robot with Constraint Force Attenuation," *Proc. American Control Conference*, pp.113–118, 2001.
[23] G. Watanabe, M. Iwase, S. Hatakeyama, and T. Maruyama, "Control Strategy for a Snake-Like Robot Based on Constraint Force and Verification by Experiment," *Advanced Robotics*, vol.23, pp.907–937, 2009.
[24] T. Yoshikawa, Foundations of robotics: Analysis and control, MIT Press, Cambridge, pp.247–248, 1990.
[25] S. Toyoshima, M. Tanaka, and F. Matsuno, "A Study on Sinus-Lifting Motion of a Snake Robot With Sequential Optimization of a Hybrid System," *IEEE Trans. on Automation Science and Engineering*, vol.11, no.1, pp.139-143, 2014.
[26] S. B. Andersson, "Discretization of a Continuous Curve," *IEEE Trans. on Robotics*, vol.24, no.2, pp.456-,461, 2008.

Motoyasu Tanaka (S'05 - M'12) received his B.E., M.S., and Ph.D. degrees in Engineering from the Department of Mechanical Engineering and Intelligent Systems at the University of Electro-Communications, Japan in 2005, 2007, and 2009, respectively. From 2009 to 2012, he worked at Canon, Inc., Tokyo, Japan.

He is currently an Assistant Professor in the Department of Mechanical Engineering and Intelligent Systems at the University of Electro-Communications. His research interests include control of biologically inspired robots, flying robots, and nonlinear systems.

Dr. Tanaka received the Hatakeyama Memorial Prize and the Miura Memorial Prize from The Japan Society of Mechanical Engineers in 2004 and 2006, respectively, and the IEEE Robotics and Automation Society Japan Chapter Young Award from the IEEE Robotics and Automation Society Japan Chapter in 2006. He is a member of the IEEE, SICE, and RSJ.

Kazuo Tanaka (S'87 - M'91 - SM'09 - F'14) received the B.S. and M.S. degrees in Electrical Engineering from Hosei University, Tokyo, Japan, in 1985 and 1987, and Ph.D. degree, in Systems Science from Tokyo Institute of Technology, in 1990, respectively.

He is currently a Professor in Department of Mechanical Engineering and Intelligent Systems at The University of Electro-Communications. He was a Visiting Scientist in Computer Science at the University of North Carolina at Chapel Hill in 1992 and 1993. He received the Best Young Researchers Award from the Japan Society for Fuzzy Theory and Systems in 1990, the Outstanding Papers Award at the 1990 Annual NAFIPS Meeting in Toronto, Canada, in 1990, the Outstanding Papers Award at the Joint Hungarian-Japanese Symposium on Fuzzy Systems and Applications in Budapest, Hungary, in 1991, the Best Young Researchers Award from the Japan Society for Mechanical Engineers in 1994, the Outstanding Book Awards from the Japan Society for Fuzzy Theory and Systems in 1995, 1999 IFAC World Congress Best Poster Paper Prize in 1999, 2000 IEEE Transactions on Fuzzy Systems Outstanding Paper Award in 2000, the Best Paper Selection at 2005 American Control Conference in Portland, USA, in 2005, the Best Paper Award at 2013 IEEE International Conference on Control System, Computing and Engineering in Penang, Malaysia, in 2013, the Best Paper Finalist at 2013 International Conference on Fuzzy Theory and Its Applications, Taipei, Taiwan in 2013. His research interests include intelligent systems and control, nonlinear systems control, robotics, brain-machine interface and their applications.

He is currently serving as an Associate Editor for Automatica and for the IEEE Transactions on Fuzzy Systems, and is on the IEEE Control Systems Society Conference Editorial Board. He is the author of two books and a co-author of 17 books. Recently, he co-authored (with Hua O. Wang) the book Fuzzy Control Systems Design and Analysis: A Linear Matrix Inequality Approach (Wiley-Interscience, 2001). He is a fellow of IEEE.

Fumitoshi Matsuno (M'94) received a Ph.D. (Dr. Eng.) degree from Osaka University in 1986. In 1986, he joined the Department of Control Engineering at Osaka University. He became a Lecturer in 1991 and an Associate Professor in 1992 in the Department of Systems Engineering at Kobe University. In 1996, he joined the Department of Computational Intelligence and Systems Science in the Interdisciplinary Graduate School of Science and Engineering at Tokyo Institute of Technology, as an Associate Professor. In 2003 he became a Professor in the Department of Mechanical Engineering and Intelligent Systems at The University of Electro- Communications. Since 2009, he has been a Professor in the Department of Mechanical Engineering and Science at Kyoto University. He also holds a post as the Vice-President of the NPO International Rescue System Institute (IRS). His current research interests lie in robotics, control of distributed parameter systems and nonlinear systems, fire and disaster rescue support systems, and geographic information systems.

Dr. Matsuno has received many awards, including Outstanding Paper Awards in 2001 and 2006, the Takeda Memorial Prize in 2001 from the Society of Instrument and Control Engineers (SICE), and the Outstanding Paper Award in 2013 from the Information Processing Society of Japan. He is a Fellow of the Japan Society of Mechanical Engineers and SICE, and a member of the IEEE, RSJ, and ISCIE, among other organizations.

He is a co-chair of the IEEE RAS Technical Committee on Safety, Security, and Rescue Robotics, a chair of the Steering Committee for the SICE Annual Conference, an editor of the Journal of Intelligent and Robotic Systems, an associate editor of Advanced Robotics and the International Journal of Control, Automation, and Systems, and is on the Conf. Editorial Board of IEEE CSS.