



Head trajectory tracking control of an extendable snake-like robot

Koki Harada¹ · Ryo Ariizumi¹ · Motoyasu Tanaka² · Toru Asai¹ · Shun-ichi Azuma¹

Received: 30 August 2021 / Accepted: 20 January 2022 / Published online: 3 March 2022
© International Society of Artificial Life and Robotics (ISAROB) 2022

Abstract

In recent years, extendable snake-like robots are attracting researchers' interest. A large number of degrees of freedom make it possible for the robots to perform new gaits. It is expected that the telescopic motion can also give the robots higher adaptability for the environment. In this paper, we propose a controller of an extendable snake-like robot for head trajectory tracking and show the effectiveness of the extendable snake-like robot by comparison to an ordinary snake-like robot.

Keywords Trajectory tracking control · Snake-like robot · Extendable snake-like robot · Prismatic joint

1 Introduction

Living snakes are highly adaptive to the environment. They can climb trees, swim in the water, and move on sand even though they have no limbs. The mysterious locomotion has stimulated researchers' interests. Hirose's development of snake-like robots is one of the notable studies. He performed experiments with living snakes and built the world's first snake-like robot [1].

Most snake-like robots consist of serial rigid links coupled with revolutionary joints that generate a large number of degrees of freedom. This enables a snake-like robot to perform various movements allowing it to perform interesting locomotions like climbing up pipes [2]. Moreover, the large number of degrees of freedom also gives robustness to the robot for joint malfunctions [3]. Because of the mobility and the robustness mentioned above, snake-like robots are expected to work as rescue robots or maintenance robots [4].

Research on extendable snake-like robots with prismatic joints is also progressing. The additional degrees of freedom by the prismatic joints make it possible for the snake-like robots to perform new gaits. One of these new gaits enables extendable snake-like robots to move into such a narrow path that normal snake-like robots cannot move forward [5]. Another study shows that the extension of the prismatic joints helps the robot climb a step [6]. Besides, general snake-like robots can fall into a singular posture, i.e., the posture in which there exists a direction that the robots cannot move [7]. It is expected that the prismatic joints enable the robot to avoid falling into the singular posture.

However, most previous studies on the extendable snake-like robots use prebuilt motion patterns to actuate the joints. In reality, the robot works in an environment unknown to the developer. It is difficult to develop a set of prebuilt motion patterns covering all environment cases. In addition, the robot is often required to move its head along a trajectory given by the operator. Usually, a snake-like robot has its sensors on its head, and moving the head to the desired position is a fundamental task, for example in rescue and maintenance missions. Although prebuilt motions with adjustable parameters may achieve the head tracking task, the appropriate choice of parameters is far from obvious. A head trajectory tracking controller can become a good solution for generating the joints' movement based on the operator's command. Moreover, such a controller can develop into a controller that enhances the robot's adaptivity to the environment.

In the current study, we propose a controller using the partial feedback linearization technique for head trajectory

This work was presented in part at the joint symposium with the 15th International Symposium on Distributed Autonomous Robotic Systems 2021 and the 4th International Symposium on Swarm Behavior and Bio-Inspired Robotics 2021(Online, June 1-4, 2021).

✉ Koki Harada
harada.koki.s8@s.mail.nagoya-u.ac.jp

¹ Nagoya University, Huro-cho, Chikusa-ku, Nagoya, Aichi 464-8603, Japan

² The University of Electro-Communications, 1-5-1, Chofugaoka, Chofu, Tokyo 182-8585, Japan

tracking. The controller can utilize the additional degrees of freedom introduced by the prismatic joints appropriately to achieve tasks that an unextendable snake-like robot cannot complete. The validity and efficacy of the proposed controller are shown by simulations. In what follows, Sect. 2 introduces the model of the robot briefly and illustrates the derivation of the controller. Then Sect. 3 shows the results of the simulations.

2 Head tracking control

As mentioned in the introduction, moving the head along the desired trajectory is an important task for snake-like robots. Thus, we set our control objective as head trajectory tracking of an extendable snake-like robot.

In this section, we introduce the model and the controller. The equation of motion of the extendable snake-like robot is shown in Subsection 2.1. The controller for head-tracking control is derived in Subsection 2.2.

2.1 The model of a snake-like robot with prismatic joints

Figure 1 shows the structure of the extendable snake-like robot. It consists of $2n$ rigid links serially connected by $n - 1$ yaw joints and n prismatic joints. All the links have the same length L , and viscous friction acts on each link. We assumed that the viscous coefficient of the longitudinal direction of the link is smaller than that of the lateral direction. The previous study shows that the anisotropy of frictional force plays a major role in generating thrust force [8]. Note that we use “simple model” to indicate the simple mathematical model (1), which represents the system depicted in Fig. 1.

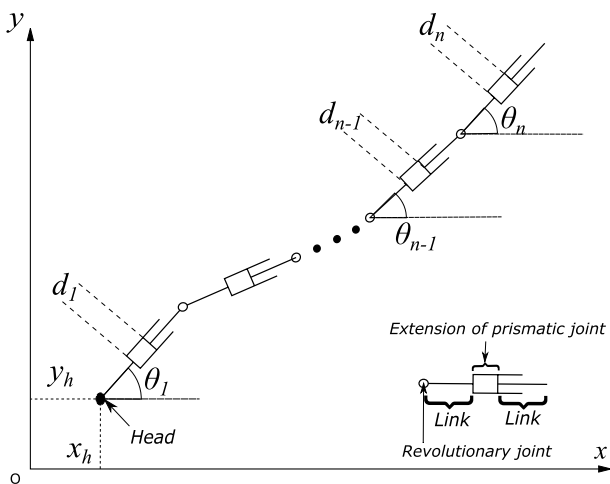


Fig. 1 The structure of the extendable snake-like robot

Let $r_h(t) = [x_h(t), y_h(t)]^T$ and θ_1 be the position and the posture of the head respectively. The angle of the i -th yaw joint is represented as $\phi_i(t)$ and the extension of the i -th prismatic joint is represented as $d_i(t)$. Let the torque of yaw joints and the force of prismatic joints as the inputs of the robot. The torque and the force corresponding $\phi_i(t)$ and $d_i(t)$ are denoted as $T_i(t)$ and $F_i(t)$ respectively. We use the vectorized notations for these variables defined as follows:

$$\begin{aligned} \phi &= [\phi_1, \dots, \phi_{n-1}]^T, \mathbf{d} = [d_1, \dots, d_n]^T, \\ \mathbf{T} &= [T_1, \dots, T_{n-1}]^T, \mathbf{F} = [F_1, \dots, F_n]^T, \\ \mathbf{u} &= [\mathbf{T}^T, \mathbf{F}^T]^T, \mathbf{q} = [\phi^T, \mathbf{d}^T]^T. \end{aligned}$$

Application of Newton’s second law and Euler’s equation gives the following equations of motion [9]:

$$\begin{bmatrix} E_2 & M_{12} & M_{13} \\ \mathbf{0}_{1 \times 2} & M_{22} & M_{23} \\ \mathbf{0}_{(2n-1) \times 2} & M_{32} & M_{33} \end{bmatrix} \begin{bmatrix} \ddot{r}_h \\ \ddot{\theta}_1 \\ \ddot{\mathbf{q}} \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ 0 \\ \mathbf{B}_q \mathbf{u} \end{bmatrix} + \begin{bmatrix} \mathbf{b}_h \\ b_\theta \\ \mathbf{b}_q \end{bmatrix}, \quad (1)$$

$$\begin{aligned} M_{12} &\in \mathbf{R}^{2 \times 1}, M_{13} \in \mathbf{R}^{2 \times (2n-1)}, \\ M_{22} &\in \mathbf{R}, M_{23} \in \mathbf{R}^{1 \times (2n-1)}, \\ M_{32} &\in \mathbf{R}^{(2n-1) \times 1}, M_{33} \in \mathbf{R}^{(2n-1) \times (2n-1)}, \\ \mathbf{b}_h &\in \mathbf{R}^2, b_\theta \in \mathbf{R}, \mathbf{b}_q \in \mathbf{R}^{2n-1}, \end{aligned}$$

where E_m is the $m \times m$ identity matrix and \mathbf{B}_q is a regular diagonal matrix. The matrices M_{ij} and the scalar value M_{22} are determined by θ_1 and \mathbf{q} . The vectors \mathbf{b}_h, b_θ and \mathbf{b}_q consist of the friction force, the centrifugal force and the Coriolis force. They depend on $\theta_1, \mathbf{q}, \dot{\theta}_1, \dot{\mathbf{q}}$ and \dot{r}_h . Note that the inertia matrix in the above equations of motion is not symmetric. Although we can transform it into a symmetric one, we leave it as it is, because symmetry is not required in the following discussions.

In the equation above, we make the following assumption.

Assumption 1 $M_{22} \neq 0$ and matrices $\mathcal{M}, \bar{\mathcal{M}}_{12}$ defined below are full rank.

$$\begin{aligned} \mathcal{M} &= M_{33} - M_{22}^{-1} M_{32} M_{23} \\ \bar{\mathcal{M}}_{12} &= M_{13} - M_{22}^{-1} M_{12} M_{23} \end{aligned}$$

This assumption means that the robot does not take a singular posture, in which there exists a direction where the robot cannot accelerate its head. We have confirmed that the assumption fails to be true if $d_i = -2L$ and $\phi_i = 0$ for any i . This situation means that all revolutionary joints gather at the same point. However, it is still unclear whether it is always true when $d_i > 0$.

2.2 Controller

To achieve the control objective $\|r_h - r_{hr}\| \rightarrow 0$ ($t \rightarrow \infty$), where $r_{hr}(t)$ is the reference trajectory of the head, we used the partial feedback linearization technique. From (1), we have

$$\ddot{r}_h = -\bar{M}_{12}^{-1} \mathcal{M}^{-1} (\mathbf{B}_q \mathbf{u} + \mathbf{b}_q - M_{22}^{-1} \mathbf{M}_{32} b_\theta) - M_{22}^{-1} \mathbf{M}_{12} b_\theta + \mathbf{b}_h. \tag{2}$$

Define the weighted pseudo-inverse matrix \bar{M}_{12}^+ of \bar{M}_{12} as follows:

$$\bar{M}_{12}^+ = \mathbf{W}^{-1} \bar{M}_{12}^T (\bar{M}_{12} \mathbf{W}^{-1} \bar{M}_{12}^T)^{-1},$$

where \mathbf{W} is a $(2n-1) \times (2n-1)$ diagonal matrix with positive diagonal entries. It is easy to see that $\bar{M}_{12} \bar{M}_{12}^+ = \mathbf{E}_2$ holds. The weighted pseudo-inverse matrix regulates the movement of the joints. The details will be described later.

Using \bar{M}_{12}^+ , let the input \mathbf{u} be calculated as

$$\mathbf{u} = \mathbf{B}_q^{-1} (\mathcal{M} \mathbf{v} - \mathbf{b}_q + M_{22}^{-1} \mathbf{M}_{32} b_\theta), \tag{3}$$

$$\mathbf{v} = -\bar{M}_{12}^+ (\bar{\mathbf{v}} + M_{22}^{-1} \mathbf{M}_{12} b_\theta - \mathbf{b}_h) + (\mathbf{E}_{2n-1} - \bar{M}_{12}^+ \bar{M}_{12}) \mathbf{k}, \tag{4}$$

$$\bar{\mathbf{v}} = \ddot{r}_{hr} - \mathbf{K}_d (\dot{r}_h - \dot{r}_{hr}) - \mathbf{K}_p (r_h - r_{hr}), \tag{5}$$

where $\mathbf{k} \in \mathbf{R}^{2n-1}$ is an arbitrary vector, $\mathbf{K}_p, \mathbf{K}_d \in \mathbf{R}^{2 \times 2}$ are positive definite matrices.

Substituting the above expression for \mathbf{u} in the (2), the following equation is derived:

$$\ddot{r}_h - \ddot{r}_{hr} + \mathbf{K}_d (\dot{r}_h - \dot{r}_{hr}) + \mathbf{K}_p (r_h - r_{hr}) = \mathbf{0}, \tag{6}$$

which implies $\|r_h - r_{hr}\| \rightarrow 0$ as $t \rightarrow \infty$.

The weight \mathbf{W} determines which joint mainly contributes to the acceleration of the head. The (i, i) entry ($1 \leq i \leq n-1$) and the $(n+j-1, n+j-1)$ entry ($1 \leq j \leq n$) of \mathbf{W} correspond to the weight of the i -th revolutionary joint and the weight of the j -th prismatic joint respectively. For example, a relatively large $(1, 1)$ element to other elements restrains the movement of the first revolutionary joint ϕ_1 .

If the reference of joint values \mathbf{q}_r is given, \mathbf{k} can be used to reduce $\|\mathbf{q} - \mathbf{q}_r\|$ at every t as described in [10]. From (1) and the property of \bar{M}_{12}^+ , the closed loop system is expressed as follows:

$$\ddot{\mathbf{q}} = -\bar{M}_{12}^+ (\bar{\mathbf{v}} + M_{22}^{-1} \mathbf{M}_{12} b_\theta - \mathbf{b}_h) + (\mathbf{E}_{2n-1} - \bar{M}_{12}^+ \bar{M}_{12}) \mathbf{k}, \tag{7}$$

$$(\mathbf{E}_{2n-1} - \bar{M}_{12}^+ \bar{M}_{12})^2 = (\mathbf{E}_{2n-1} - \bar{M}_{12}^+ \bar{M}_{12}), \tag{8}$$

$$(\mathbf{E}_{2n-1} - \bar{M}_{12}^+ \bar{M}_{12}) \bar{M}_{12}^+ = \mathbf{0}. \tag{9}$$

Multiplying (7) by $(\mathbf{E}_{2n-1} - \bar{M}_{12}^+ \bar{M}_{12})$ from the left and applying (8) and (9) gives the following relation:

$$(\mathbf{E}_{2n-1} - \bar{M}_{12}^+ \bar{M}_{12}) \ddot{\mathbf{q}} = (\mathbf{E}_{2n-1} - \bar{M}_{12}^+ \bar{M}_{12}) \mathbf{k}. \tag{10}$$

Therefore, if \mathbf{k} is defined by

$$\mathbf{k} = \ddot{\mathbf{q}}_r - \mathbf{K}_{qd} (\dot{\mathbf{q}} - \dot{\mathbf{q}}_r) - \mathbf{K}_{qp} (\mathbf{q} - \mathbf{q}_r), \tag{11}$$

with positive definite matrices \mathbf{K}_{qp} and \mathbf{K}_{qd} , \mathbf{q} will get closer to \mathbf{q}_r .

The technique using \mathbf{k} to achieve a sub-objective is mentioned in [11]. The authors calculate \mathbf{k} as the gradient of a cost function to avoid the singular posture. However, the method requires a carefully designed cost function, therefore we employ the method in [10] with a modification for an extendable snake-like robot.

3 Simulation

We carried out four simulations as shown in Table 1.

In Table 1, ‘‘Sine curve*’’ represents a sine curve trajectory containing discontinuous reference velocity. The mathematical definitions of these trajectories are given in Subsection 3.1.

Subsection 3.2 explains the parameters of the extendable snake-like robots. In Subsection 3.3, the simple model of the unextendable snake-like robot is introduced briefly. The simulation results of Sim1 and Sim2 are shown in Subsection 3.4, and the result of Sim3 and Sim4 are shown in Subsection 3.5.

We used COPPELIA ROBOTICS V-REP version 3.6.2 and set the time step 10 ms. The physics engine was Open Dynamics Engine [12] and ‘‘Dynamics settings’’ was ‘‘Accurate’’. On the setting, the output of our controller was updated every two calculations of the movement of the robot.

Table 1 Simulated robots and trajectory

	Robot	Trajectory
Sim1	Extendable	Sine curve
Sim2	Extendable	Sine curve*
Sim3	Unextendable	Sine curve
Sim4	Unextendable	Sine curve*

‘‘Sine curve*’’ represents a sine curve trajectory containing discontinuous reference velocity

3.1 The reference trajectories

In Sim1 and Sim3, we used a sine curve as the reference trajectory to examine the trackability on the path where the robot’s head swings from side to side. The trajectory was defined as follows:

$$\begin{aligned} \mathbf{r}_{hr} &= [-v_x t, -0.2 \sin(-0.75t)]^T, \\ v_x &= 0.15. \end{aligned} \tag{12}$$

In Sim2 and Sim4, we examined the trackability when the reference trajectory contained discontinuous velocity change. We changed v_x in (12) as follows:

$$v_x = \begin{cases} 0.15 & (0 \leq t \leq 5, 6 < t), \\ 0.45 & (5 < t \leq 6). \end{cases}$$

The discontinuous velocity reference is one of the difficult references for a normal snake-like robot because it will lead the robot into a singular posture, e.g., a line shape posture and an arc shape posture. It will be discussed later how the prismatic joints ease the difficulty of the discontinuous velocity reference.

In all the simulations, the robot stood still at the initial time.

3.2 The parameters of the extendable snake-like robot

Figure 2 shows the 3D model of the extendable snake-like robot. The grey rectangle parts represent the prismatic joints and the red cylindrical parts represent the rotational joints. We use “simulation model” to indicate the 3D model in the physical simulator. Note that the physical simulator can simulate three-dimensional movement, however, we let the robot move slow and kept the robot on the floor. In addition, the simulator handles the friction as Coulomb friction, and to represent the anisotropy of friction in the physical simulator, we equipped passive wheels on the simulation model.

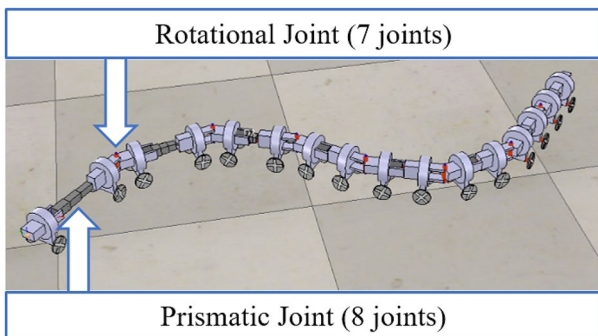


Fig. 2 3D model of the robot (16 links)

The number of links was 16, the link length was 8 cm, the weight of the link was 0.25 kg, and the maximum extension of the prismatic joint was 10 cm. We set $\mathbf{K}_p, \mathbf{K}_d, \mathbf{q}_r, \mathbf{K}_{qp}, \mathbf{K}_{qd}$ and \mathbf{W} as follows:

$$\begin{aligned} \mathbf{K}_p &= 5000\mathbf{E}_2, \\ \mathbf{K}_d &= 100\sqrt{2}, \\ \mathbf{q}_r &= [\phi_r^T \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0]^T, \\ \phi_r &= [\text{Ser}(t, 1), \dots, \text{Ser}(t, 7)]^T, \\ \text{Ser}(t, i) &= \frac{\pi^2}{14} \sin\left(\frac{\pi}{4}t - \frac{2\pi i}{7}\right), \\ \mathbf{K}_{qp} &= \text{block diag}(200\mathbf{E}_7, 10\mathbf{E}_8), \\ \mathbf{K}_{qd} &= \text{block diag}(20\sqrt{2}\mathbf{E}_7, 2\sqrt{10}\mathbf{E}_8), \\ \mathbf{W} &= \text{block diag}(\mathbf{E}_7, 1000\mathbf{E}_8). \end{aligned}$$

If the reference angle (ϕ_r) is not given, i.e., if we set $\mathbf{k} = \mathbf{0}$, the robot tends to solely use extensions to follow the reference path of the head. That causes the robot to converge to a singular posture: the fully stretched robot can be regarded as an ordinary snake-like robot without prismatic joints and the straight line is known to be a singular posture for the ordinary snake-like robot. To avoid such a problem, we set ϕ_r as above. In addition, when the robot does not take the singular posture, the head tracking is achieved without using ϕ_r . Therefore, the difference of the frequencies between ϕ_r and \mathbf{r}_{hr} does not cause problems.

Note that, due to a constraint of the simulator, it was difficult to make the joints output specified torques and forces. Therefore, in the physical simulator, we controlled the joints by a position controller. To make it applicable, we converted our controller’s output to the reference position and the reference velocity. Let the reference value of \mathbf{q} and $\dot{\mathbf{q}}$ to be calculated as follows:

$$\begin{aligned} \ddot{\mathbf{q}} &= \mathcal{M}^{-1}(\mathbf{B}_q \mathbf{u} + \mathbf{b}_q - \mathbf{M}_{22}^{-1} \mathbf{M}_{32} \mathbf{b}_\theta), \\ \dot{\mathbf{q}}_{\text{sim}} &= \dot{\mathbf{q}} + \ddot{\mathbf{q}} \Delta t, \\ \mathbf{q}_{\text{sim}} &= \mathbf{q} + \dot{\mathbf{q}} \Delta t + \frac{1}{2} \ddot{\mathbf{q}} \Delta t^2, \end{aligned}$$

where \mathbf{q}_{sim} is the reference value of the joints, $\dot{\mathbf{q}}_{\text{sim}}$ is the reference velocity and Δt is the step time of the simulator (10 ms). A similar calculation is also used in Sim3 and Sim4. Position servo motors are often used as the joints of real snake-like robots. Using the method above, our controller can be applied to such real snake-like robots.

3.3 The parameters of the ordinary snake-like robot

The simulation model of the ordinary snake-like robot is shown in Fig. 3. Before mentioning its parameters, we

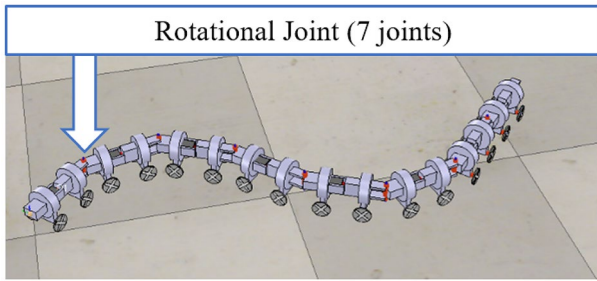


Fig. 3 Ordinary snake-like robot used in Sim3 and Sim4

introduce the equation of motion of the ordinary snake-like robot briefly.

$$\begin{bmatrix} E_2 & \tilde{M}_{12} & \tilde{M}_{13} \\ \mathbf{0}_{1 \times 2} & \tilde{M}_{22} & \tilde{M}_{23} \\ \mathbf{0}_{(n-1) \times 2} & \tilde{M}_{32} & \tilde{M}_{33} \end{bmatrix} \begin{bmatrix} \ddot{r}_h \\ \ddot{\theta}_1 \\ \ddot{\phi} \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ 0 \\ -u \end{bmatrix} + \begin{bmatrix} \tilde{b}_h \\ \tilde{b}_\theta \\ \tilde{b}_\phi \end{bmatrix},$$

$$\begin{aligned} \tilde{M}_{12} &\in \mathbb{R}^{2 \times 1}, \tilde{M}_{13} \in \mathbb{R}^{2 \times (n-1)}, \\ \tilde{M}_{22} &\in \mathbb{R}, \tilde{M}_{23} \in \mathbb{R}^{1 \times (n-1)}, \\ \tilde{M}_{32} &\in \mathbb{R}^{(n-1) \times 1}, \tilde{M}_{33} \in \mathbb{R}^{(n-1) \times (n-1)}, \\ \tilde{b}_h &\in \mathbb{R}^2, \tilde{b}_\theta \in \mathbb{R}, \tilde{b}_\phi \in \mathbb{R}^{n-1}. \end{aligned}$$

This equation is almost the same as (1). Because of the absence of prismatic joints, the variable q in (1) turned into ϕ , and the input u consists only of the torque.

We designed the controller for this robot in the same way as Section 2 using the partial feedback linearization technique.

$$\begin{aligned} u &= -(\tilde{\mathcal{M}}v - \tilde{b}_\phi + \tilde{M}_{22}^{-1}\tilde{M}_{32}\tilde{b}_\theta), \\ v &= -\tilde{M}_{12}^+(\tilde{v} + \tilde{M}_{22}^{-1}\tilde{M}_{12}\tilde{b}_\theta - \tilde{b}_h) \\ &\quad + (E_{2n-1} - \tilde{M}_{12}^+\tilde{M}_{12})k, \\ \tilde{v} &= \ddot{r}_{hr} - \tilde{K}_d(\dot{r}_h - \dot{r}_{hr}) - \tilde{K}_p(r_h - r_{hr}), \end{aligned}$$

where

$$\begin{aligned} \tilde{\mathcal{M}} &= \tilde{M}_{33} - \tilde{M}_{22}^{-1}\tilde{M}_{32}\tilde{M}_{23}, \\ \tilde{M}_{12} &= \tilde{M}_{13} - \tilde{M}_{22}^{-1}\tilde{M}_{12}\tilde{M}_{23}, \\ \tilde{M}_{12}^+ &= \tilde{M}_{12}^T(\tilde{M}_{12}\tilde{M}_{12}^T)^{-1}, \\ k &= \ddot{\phi}_r - \tilde{K}_{\phi d}(\dot{\phi} - \dot{\phi}_r) - \tilde{K}_{\phi p}(\phi - \phi_r), \end{aligned}$$

and $\tilde{K}_p, \tilde{K}_d, \tilde{K}_{\phi p}, \tilde{K}_{\phi d}$ are arbitrary positive definite matrices.

To reduce the difference of inertial parameters of the two simulation models, the simulation model of the ordinary snake-like robot is made from the extendable snake-like robot’s simulation model by fixing the prismatic joints’ extension to 0. We can regard a pair of two links of the extendable snake-like robot connected by a prismatic joint as

one link of the ordinary snake-like robot. Thus the number of links was 8, the link length was 16 cm, and the weight of the link was 0.5 kg. The gain and ϕ_r are defined as follows:

$$\begin{aligned} \tilde{K}_p &= 10000E_2, \\ \tilde{K}_d &= 200, \\ \phi_r &= [\text{Ser}(t, 1), \dots, \text{Ser}(t, 7)]^T, \\ \tilde{K}_{\phi p} &= 9.5E_7, \\ \tilde{K}_{\phi d} &= 6.1644E_7. \end{aligned}$$

Note that different gains were used from the ones for the extendable snake-like robot. In the present study, we chose the gain which gave almost the same error value in Sim1 and Sim3.

3.4 The results of Sim1 and Sim2

Figure 4 shows the position and the posture of the extendable snake-like robot at $t = 0, 5, 6, 10$ s in Sim1. Figure 5 shows the position and the posture in Sim2. The red ball shows the reference position at the time. The robot seemed to track the reference well in both simulations.

Figure 6 shows the discrepancy between the reference and the actual head positions $\|r_h - r_{hr}\|$ in Sim1. It shows that the error value stayed around 1 cm. The reason why

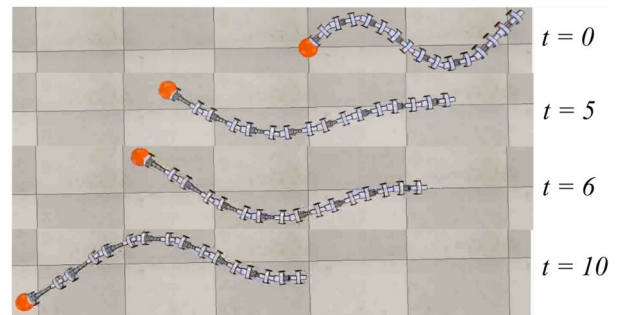


Fig. 4 The motion of the robot ($t = 0 - 10$ s): Sim1

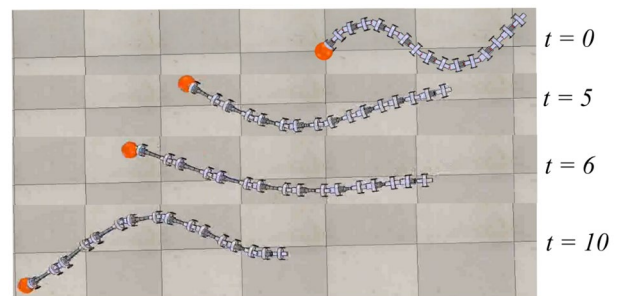


Fig. 5 The motion of the robot ($t = 0 - 10$ s): Sim2

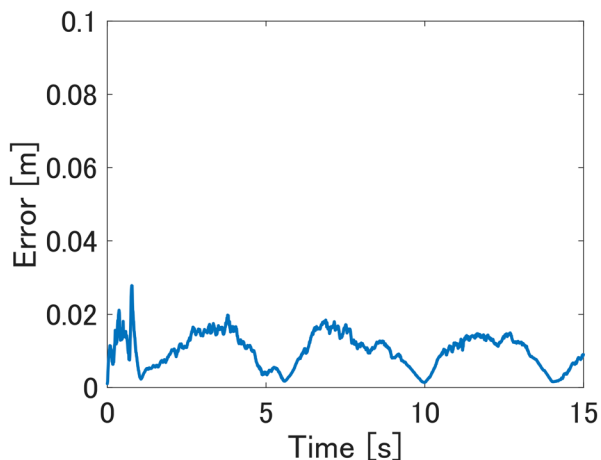


Fig. 6 The tracking error $\|r_h - r_{hr}\|$: Sim1

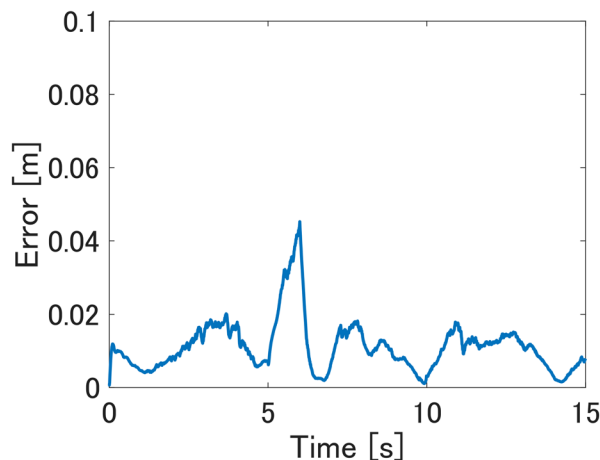


Fig. 7 The tracking error $\|r_h - r_{hr}\|$: Sim2

$\|r_h - r_{hr}\| \rightarrow 0 (t \rightarrow \infty)$ was not achieved will be the difference in the friction between the simple model and the simulator. In the simple model, we assumed that anisotropic friction force was distributed over the link. Because of a constraint on the physical simulator, we used passive wheels to represent anisotropy. This makes the frictional torque acting on links close to zero because the wheel lifts the link in the air. Besides, the friction is treated as Coulomb friction in the physical simulator. Thus, using the wheels and the property of the simulator led to a modeling error. However, even the largest error value (about 2.8 cm) was only 2.2 % of the length of the robot. Thus, we conclude that the controller worked well even with a modeling error. Moreover, Fig. 8 shows that the prismatic joints gradually extended. This is because the robot tried catching up with the reference position from the initial head velocity (0 m/s) using both the movement of the entire body and extension of the prismatic joints. That behavior can seem reasonable and effective to catch up with the reference. Therefore, we conclude that the controller successfully generated the prismatic joints' motion based on the given task.

Figure 7 shows the error in Sim2. As in Sim1, the error value stayed around 1 cm most of the time. The difference is that the error value between 5 s and 6 s was relatively large. Furthermore, Fig. 9 shows that sharp extension of the prismatic joints occurred around $t \approx 6$ s, which was not observed in Sim1. This implies that the controller generated the extension to track the sudden velocity change. Taking a closer look, around $t \approx 6$ s in Fig. 9, d_1 and d_2 reached the maximum extension (10 cm) and could not extend further. Thus, the large error around $t \approx 6$ s was caused by both the discontinuous change of the reference velocity and the limit of the extension. However, the largest error (about 4.5 cm) was also relatively small to the length of the robot (3.5 %).

Therefore, we conclude that the proposed controller worked well in the situation.

3.5 The results of Sim3 and Sim4

Figures 10 and 11 show the position and the posture of the ordinary unextendable snake-like robot at $t = 0, 5, 6, 10$ s in Sim3 and Sim4 respectively.

Figure 12 shows the error value during Sim3. Unlike Sim1, the error between $t = 5$ s and $t = 10$ s was relatively large. As can be seen in Fig. 10, the posture of the robot is close to an arc at $t = 5$ s and $t = 6$ s, which is known to be a singular posture. This suggests that it was difficult for the robot to track the reference trajectory without the aid

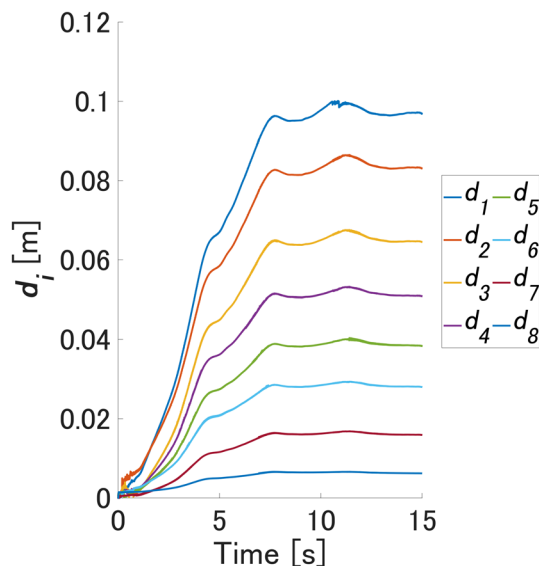


Fig. 8 The extension of the prismatic joints: Sim1

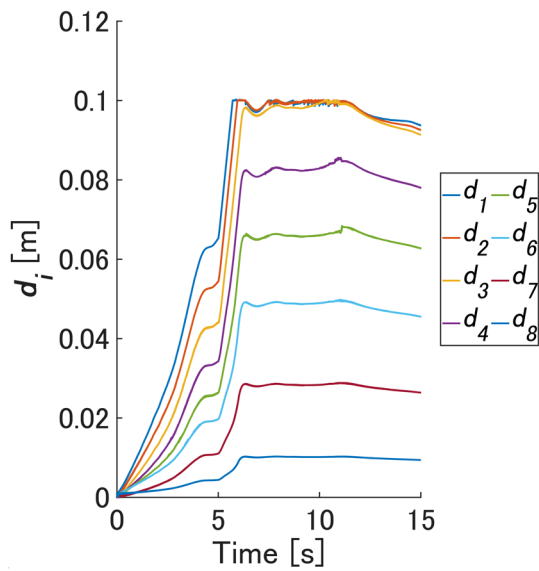


Fig. 9 The extension of the prismatic joints: Sim2

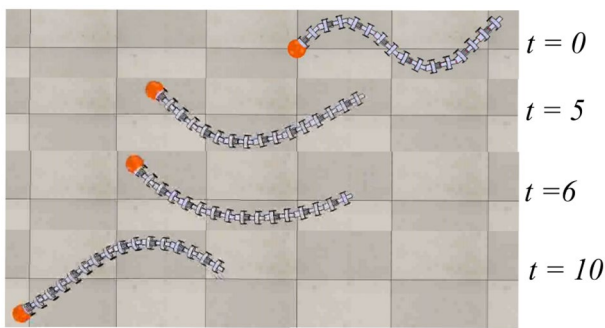


Fig. 10 The motion of the robot ($t = 0 - 10$ s): Sim3

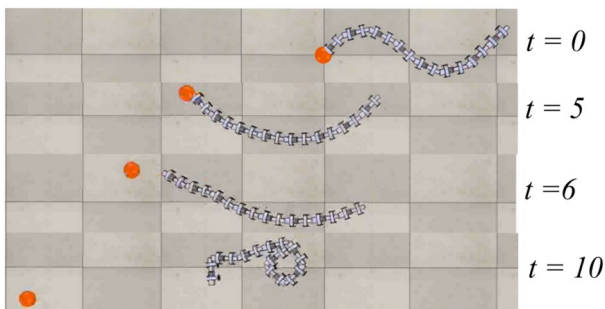


Fig. 11 The motion of the robot ($t = 0 - 10$ s): Sim4

of additional degrees of freedom at those time moments. Considering this result, in Sim1, the prismatic joints helped to keep the robot from falling into an arc shape by extending the body length. Except for the error between $t = 5$ s and $t = 6$ s, the error value stayed around 1 cm as the extendable

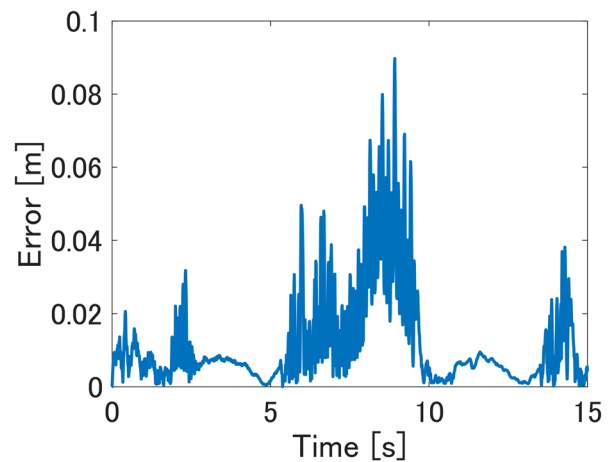


Fig. 12 The tracking error $\|r_n - r_{nr}\|$: Sim3

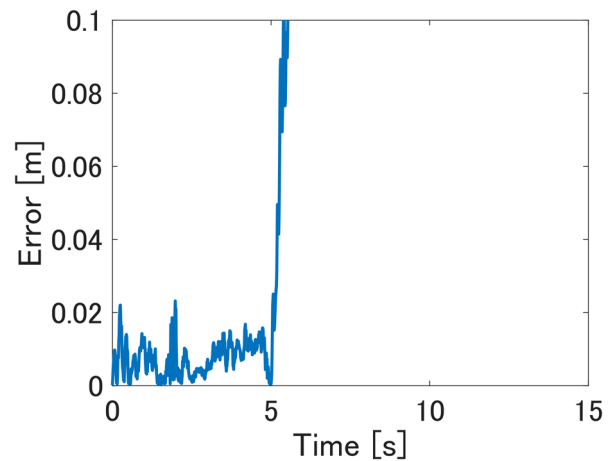


Fig. 13 The tracking error $\|r_n - r_{nr}\|$: Sim4

snake-like robot (Sim1). Thus, when (12) is given as a reference, both controllers exhibit almost the same performance.

Figure 13 shows the error value during Sim4. It indicates that the ordinary snake-like robot failed in tracking the reference trajectory when the discontinuous velocity input was given. Figure 11 also shows that the robot was left behind at $t = 6$ s and coiled itself up at $t = 10$ s far behind the reference position at the time. This result illustrates the effectiveness of equipping snake-like robots with prismatic joints.

4 Conclusion

In this paper, we developed the controller for the extendable snake-like robot and verified the effectiveness of the controller and the prismatic joint by the simulations. Our new controller successfully generates the motion of the joints based

on the given trajectory of the robot. That will enhance the use of extendable snake-like robots. Furthermore, the results also showed the superiority of extendable snake-like robots against ordinary unextendable snake-like robots.

However, the following problems remain. First, the sufficient condition for Assumption 1 to hold is unclear. Though we found that some negative extension of the prismatic joints makes Assumption 1 fail, however, it is not quite sure that the assumption always holds when all the joints' lengths take positive values. In addition, the sufficient condition under the existence of the movable limit of the joints also needs to be discussed. Our future work needs to consider the movable limit. Besides, the robustness of the controller against the environmental change has not been examined in this paper. We assumed the robot moved on a floor where the frictional property did not change. In the real world, the friction can vary from position to position, which can cause a source of an error. Future works include the application of robust control techniques to our controller. Regarding the gains, our controller has numerous gain parameters to be tuned. We will work on not only the application of robust control but also data-driven gain tuning methods.

Funding This work was supported by JSPS KAKENHI under Grant JP18K04011 and Grant JP21H01285.

References

- Hirose S (1993) Biologically inspired robots snake-like locomotors and manipulators. Oxford University Press, Oxford, pp 31–99
- Kamegawa T, Harada T, Gofuku A (2009) Realization of cylinder climbing locomotion with helical form by a snake robot with passive wheels. In: 2009 IEEE International Conference on Robotics and Automation, 3067/3072
- Ariizumi R, Takahashi R, Tanaka M, Asai T (2019) Head trajectory tracking control of a snake robot and its robustness under actuator failure. *IEEE Trans Control Syst Technol* 27:2589–2597
- Hirose S (2010) Machine design of mobile snake-like robots. *J Robot Soc Jpn* 28:151–155
- Wang K, Gao W, Ma S (2017) Snake-like robot with fusion gait for high environmental adaptability: design. *Model Exp Appl Sci* 7:1133
- Tanaka M, Nakajima M, Ariizumi R, Tanaka K (2020) Three-dimensional steering for an articulated mobile robot with prismatic joints with consideration of hardware limitations. *Adv Robot*. <https://doi.org/10.1080/01691864.2020.1753568>
- Sato H, Tanaka M, Matsuno F (2006) Trajectory tracking control of snake robots based on dynamic model. *Trans Soc Instrum Control Eng* 42:651–658
- Liljebäck P, Pettersen KY, Stavadahl Ø, Gravadahl JT (2011) Controllability and stability analysis of planar snake robot locomotion. *IEEE Trans Autom Control* 56:1365–1380
- Harada K, Ariizumi R, Tanaka M, Asai T, Azuma S (2020) Head trajectory tracking control of a snake-like robot with linear actuators. In: *The 7th Multi-symposium on Control Systems 2020*, pp 1–7
- Ariizumi R, Tanaka M (2019) Manipulability analysis of a snake robot without a lateral constraint for head position control. *Asian J Control*. <https://doi.org/10.1002/asjc.2118>
- Matsuno F, Suenaga K (2003) Experimental study on control of redundant 3-d snake robot based on a kinematic model. In: *Proceedings of the 2nd International Symposium on Adaptive Motion of Animals and Machines*, Kyoto
- Open Dynamics Engine (2021). <https://www.ode.org/>. Accessed 21 Dec 2021

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.