To appear in *Advanced Robotics*
Vol. 00, No. 00, March 2021, 1–16

FULL PAPER

Local body shape control of an articulated mobile robot and an application for recovery from a stuck state

Mizuki Nakajima[a]*, Shinnosuke Fukumura[a], Kazuo Tanaka[a], and Motoyasu Tanaka[a]

[a]*Department of Mechanical and Intelligent Systems Engineering, The University of Electro-Communications, 1-5-1 Chofugaoka, Chofu, Tokyo, Japan*;

(*v1.0 released October 2020*)

This paper proposes a method for an articulated mobile robot to change its local body shape in a complex environment. By coexisting a heuristic approach which is easy to design a motion in a complex three-dimensional environment and a model-based approach which can utilize the redundancy of the robot, the robot achieves propulsion in a complex environment and the local body shape control simultaneously. A part of the robot's body shape is changed based on a kinematic model, whereas the rest of it is controlled to follow the original shape or maintain its original position based on a heuristic method. As an application of the proposed method, the recovery motion from a stuck state is proposed. The robot is recovered from a stuck state by controlling the part of the robot that is in collision with an obstacle to move it away from the obstacle. The effectiveness of the proposed method is verified by experiments on an actual robot.

Keywords: Redundant Robots, Articulated Mobile Robot, Recovery from a Stuck State.

1. Introduction

Articulated robots are built to inspect narrow spaces and conduct exploration activities because of their elongated body shape. Moreover, these robots have a high degree of freedom of movement and can operate in various environments. Yamada et al. proposed a continuous curve approximation method as a control method for a cord-like robot with active joints [1]. The body shape of the robot was approximated to an arbitrary curve, and the robot was propelled by changing the curve like a wave. Takemori et al. proposed a method to design complex target shapes by connecting simple curves, and applied this method to realize ladder climbing [2] and propulsion on uneven terrain [3]. Propulsion inside and outside the cylinder is achieved by controlling the body shape of the entire robot to wrap around the cylinder [4–6]. Kamegawa et al. achieved propulsion inside and outside a pipe by changing the body shape of the robot into a spiral shape and rotating it around the trunk axis [5]. Additionally, the branches on the pipes are overcome by changing the robot's body shape to send a wave in the longitudinal direction of the robot [6]. Articulated mobile robots, which are composed of modules with propulsion mechanisms, such as wheels and crawlers, have high mobility. The articulated robot can be propelled in various environments, such as uneven terrain, rubble, steps, and stairs [7–11]. ACM-R8 consists of large mono-tread wheels and active joints, and achieved step and stair climbing [8]. T^2 Snake-4 climbed steep stairs and high steps, by operating valves with the arm attached to the head of the robot [9], and opening a door [12].

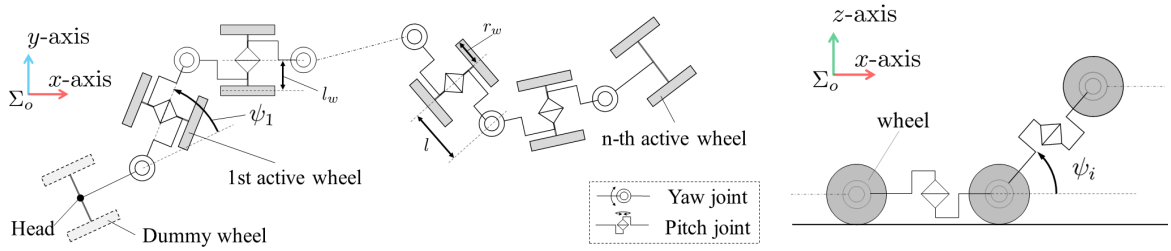
*Corresponding author. Email: mizuki.nakajima@rc.mce.uec.ac.jp

The method of controlling the robot body shape to follow the target curve, represented by the continuous curve approximation method [1], makes it possible to design the three-dimensional motion of the robot easily. In contrast, these methods cannot utilize the redundancy of the robot as in the method based on kinematic or dynamic model [9, 12, 13]. In addition, kinematics or dynamics based control methods have only been applied to simple environments such as flat surfaces and steps, because the modeling of the interaction between the robot and the environment is very complex. If these methods can coexist, various motions utilizing redundancy in complex three-dimensional environments can be expected to realize. For example, one of the problems in propelling an articulated mobile robot in a complex environment is a stuck state of the robot due to interference between the robot and the environment. Due to the length of the robot and large number of joints, there is a high risk of the robot getting stuck by collision between the robot and environment. To avoid this, obstacle avoidance using kinematic redundancy has been proposed [13]. The robot is prevented from getting stuck by avoiding collisions with obstacles based on information from attached distance sensors. However, this method assumes only two-dimensional motion, and it is difficult to completely avoid getting stuck in an unknown three-dimensional environment. Additionally, some researches use contact with the environment actively [14, 15]. Propulsion in an environment with obstacles is achieved by pushing obstacles with the robot's body. In these methods, the outline of the robot is designed as smooth for pushing the obstacles. The propulsion mechanisms of articulated mobile robots inhibit the smoothness of the robot outline. Therefore, the application of this method to articulated mobile robots is difficult. In addition, not all contact with the environment contributes to propulsion, and some contact inhibits movement. Hirose et al. developed the ACM-R3 with large wheels to prevent the robot becomes a stuck state with the environment [16]. The gap of exterior that may cause stuck with the environment are eliminated by installing large wheels. However, adopting such a structure limits the movable limit angle of the rotational joints. As described above, the recovery method for stuck articulated mobile robots is important because it is difficult to completely avoid getting stuck. Genbu 3 [17], which consists of active wheels and passive joints, uses internal sensors to detect the stuck state and recovers the robot by lifting its body. KOHGA 2 [18], which consists of crawler modules connected in series, lifts the robot's body by rotating the crawler arm and recovers from the stuck state, which is caused by obstacles at the bottom of the robot. The application of these methods in other robots is difficult because they use dedicated hardware to detect or recover from the stuck state.

This study presents a method to realize local body shape change in the complex environment. By simultaneously using the continuous curve approximation method, which is easy to design 3D motions, and method based on kinematic model, which can make explicit use of redundancy, local body shape changes in complex environments can be realized. In addition, this study presents recovery operation from a stuck state as an application of the proposed method. By controlling a part of the body shape of the robot which causes the stuck state, the robot recovers from the stuck state by moving the robot away from the obstacle that interferes with its motion.

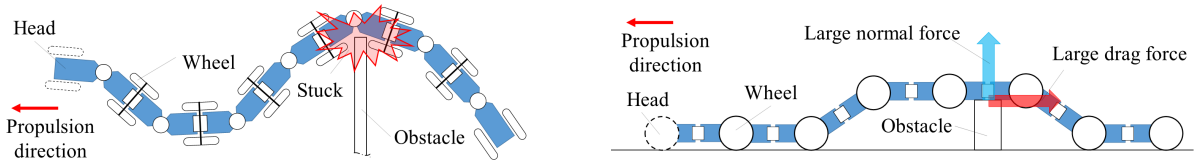
2. Problem setting

Figure 1 shows the schematics of the articulated mobile robot used in this study. The robot consists of $2n$ links connected in series by rotating joints. The rotation axes of the adjacent joints are rotated by $\pi/2$ deg around the trunk, as shown in Fig. 1. Let l be the length of each link. The yaw and pitch joints are the odd-numbered joints from the head and even-numbered joints, respectively. Let ψ_i be the i-th joint angle, and $\boldsymbol{\psi} \in \mathrm{R}^{2n-1,1}$ be the vector of all joint angles. The wheels are attached to the tail of the robot and coaxial axis of the pitch joint. Let l_w and r_w be the length between the link and center of the wheel, and the radius of the wheel, as shown in Fig. 1. The wheels are active and their rotation generates thrust. We assume that the frictional coefficient of the wheel is high and the wheel does not slide in the axle direction.



(a) Top view (b) Side view

Figure 1. Articulated mobile robot



(a) Stuck I: stuck because of geometric relationships

(b) Stuck II: stuck because of the force acting on the robot

Figure 2. Stuck state because of collision between the robot and obstacle

Furthermore, a dummy wheel with the same radius as the active wheel is attached to the head. It is assumed that the frictional coefficient of the dummy wheel is sufficiently low to ignore the frictional force caused by the contact between the dummy wheel and environment. The robot is propelled by a combination of bending of the entire robot with the joints and rotation of the active wheels.

The assumed environment is not limited in detail, but it is assumed a complex environment that requires three-dimensional motion of the robot such as steps and slopes. The robot is assumed to be operated based on the continuous curve approximation method [1] in the assumed environment. The robot and the environment is assumed to be visible, and the motion of the robot is directed by the operator. The operator instructs the direction and speed of movement of the head of the robot, and the robot propels by propagating the head's motion backward [19]. In the proposed method, a part of the body shape is controlled based on a kinematic model, and the redundancy of the robot is utilized to change the body shape locally. A part of the robot which is not controlled by the kinematic model follows the original body shape based on the continuous curve approximation method. The proposed method achieves local body shape change control by utilizing redundancy in complex environments by coexisting the continuous curve approximation method and the kinematic model-based control.

This paper also presents the recovery motion from a stuck state as an application of the proposed method. A stuck state is defined as a state in which the robot cannot to continue propelling due to the interference of parts of the robot other than its wheels with the environment, as shown in Figure 2. Fig. 2(a) shows a stuck state caused by the geometric relationship between the robot and the environment, and Fig. 2(b) shows a stuck caused by the frictional forces acting between the robot and the environment. A part of the robot which includes the stuck point is controlled based on a kinematic model, and the goal is to recover from the stuck state by moving the robot away from the obstacle using redundancy.

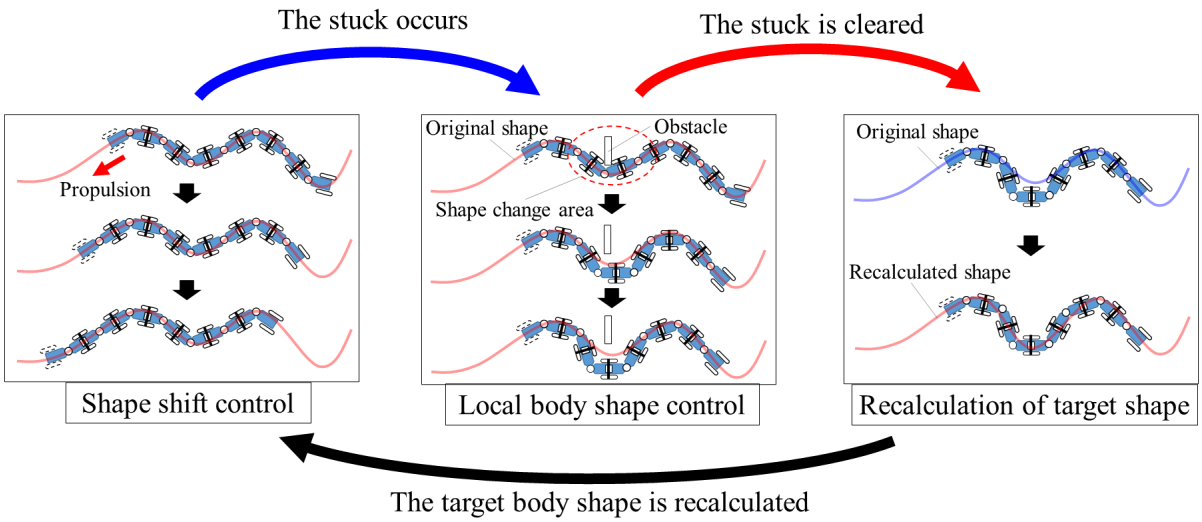


Figure 3. System overview

3. Control method

Figure 3 shows an overview of the proposed method, and the motion flow of the application for recovery motion from a stuck state. As shown in Fig. 3, the robot is propelled while switching propulsion by the curve approximation method and local body shape control method. The robot is propelled in a three-dimensional environment by using the curve approximation method, which is easy to design three-dimensional motions, and changes a part of body shape by using local shape control method in scenes where local body shape changes are required. Then, the target continuous curve is reconstructed using the joint angles after local body shape change [20], and propulsion by the curve approximation method is resumed. The details of each operation are described below. As shown in Fig. 3, the part of the robot including the stuck point is controlled based on the kinematic model, and the body shape is controlled to move away from the obstacle by using redundancy.

3.1 *Shape shift control*

In the shape shift control, the robot is controlled such that the body shape approximates the target continuous curve based on the body approximation method to a continuous curve [1]. The robot is then propelled by changing the target curve by sending a wave from the head to the tail side. This approximation method determines the joint angles such that the body shape approximates the target curve. The continuous curve is defined as the backbone curve by considering the joint configuration of the robot [5], as shown in Figure 4(a). Let s be the mediation variable in the length direction of a continuous curve, and let $c(s) = [x(s), y(s), z(s)]^\top$ be the three-dimensional position of the curve at position s. The backbone curve is defined as the following differential equation.

$$\begin{cases} d\boldsymbol{c}(s)/ds = \boldsymbol{e}_r(s), \\ d\boldsymbol{e}_r(s)/ds = \kappa_y(s)\boldsymbol{e}_p(s) - \kappa_p(s)\boldsymbol{e}_y(s), \\ d\boldsymbol{e}_p(s)/ds = -\kappa_y(s)\boldsymbol{e}_r(s), \\ d\boldsymbol{e}_y(s)/ds = \kappa_p(s)\boldsymbol{e}_r(s), \end{cases} \tag{1}$$

where s represents the position on the curve, $\boldsymbol{e}_r(s) \in \mathrm{R}^{3,1}$ is the unit vector parallel to the tangent of the curve at s, $\boldsymbol{e}_y(s) \in \mathrm{R}^{3,1}$ is the unit vector orthogonal to $\boldsymbol{e}_r$ and pointing upward

at s in the model, $\boldsymbol{e}_p(s) \in \mathrm{R}^{3,1}$ is the unit vector denoted by $\boldsymbol{e}_p(s) = \boldsymbol{e}_y \times \boldsymbol{e}_r(s)$, $\kappa_y(s)$ and $\kappa_p(s)$ are arbitrary functions that determine the shape of the target continuous curve. The target backbone curve is uniquely determined by designing the $\kappa_y(s)$ and $\kappa_p(s)$. Additionally, a lateral rolling motion is generated by rotating the backbone curve around e_r. The curvature of the target continuous curve is limited by the limit angle of the joint and link length of the robot. In addition, if the curvature is large, the error between the target curve and the robot's body shape will be large due to the effect of discretizing the curve [1]. Let θ be the rotating angle of the backbone curve around e_r, κ'_p is κ_p after rotation, and κ'_y is κ_y after rotation. κ'_p and κ'_y are then represented as

$$\begin{bmatrix} \kappa'_p(s) \\ \kappa'_y(s) \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} \kappa_p(s) \\ \kappa_y(s) \end{bmatrix}. \tag{2}$$

Let s_h be the position of the head on the continuous curve, and $s_i = s_h - il$ be the position of the i-th joint on the continuous curve. The robot's body shape is approximately $s_h - 2nl \leq s \leq s_h$ of the target continuous curve by determining the joint angle based on the following equations:

$$\psi_i = -\int_{s_i-l}^{s_i+l} f_\kappa(s,i)\mathrm{ds}, \tag{3}$$

$$f_\kappa(s,i) = \begin{cases} \kappa_y(s), & (i : \text{odd}) \\ \kappa_p(s). & (i : \text{even}) \end{cases} \tag{4}$$

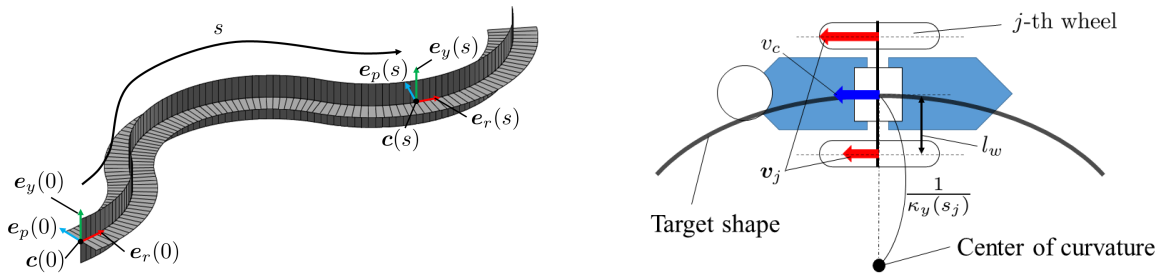
The range of approximation is changed by changing s_h, and motion is generated along the continuous curve. The rotation of the active wheel must be controlled according to the motion of the joint. The translation velocity of the left and right wheels changes depending on the curvature of the target continuous curve, as shown in Fig. 4(b). Let $v_c = \mathrm{d}s_h/\mathrm{d}t$ be the translation velocity of the entire robot, and s_j be the position of the i-th wheel on the continuous curve. The translation velocity of the i-th wheel $\boldsymbol{v}_j \in \mathrm{R}^{2,1}$ is represented by the following equation [20]:

$$\boldsymbol{v}_j = \begin{bmatrix} (1 - l_w\kappa_y(s_j)) \\ (1 + l_w\kappa_y(s_j)) \end{bmatrix} v_c, \tag{5}$$

where the first and second elements of $\boldsymbol{v}_j$ are the translation velocities of the left and right wheels, respectively. The operator controls the propulsion velocity of the robot by instructing v_c, and the direction of the propulsion by instructing κ_p and κ_y at $s = s_h$. The robot is then propelled along the target continuous curve by setting the joint angles and translation velocity of the wheel based on Eq. (3) and Eq. (5).

3.2 *Local body shape control*

In local body shape control, a part of the robot's body shape is controlled based on a kinematic model. The robot's body shape is adaptively changed according to the objective by using kinematic redundancy. As shown in Figure 5, we refer to the part of the body that locally changes its shape as the *deforming part*, and the parts before and after it are the *leading part* and *trailing part*, respectively. If the positions of the leading and trailing parts are fixed, the motion of the deforming part is strongly restricted. Therefore, by pulling the leading or trailing part toward the deforming part, the motion range of the deforming part is increased. When the robot is controlled using shape shift control, it is propelled by propagation of the head motion backward. If the position of the robot's head is changed by the local body shape control, the position of the robot becomes difficult to understand intuitively. Thus, the position of the leading part is fixed, and the trailing part is pulled, as shown in Fig. 5. The feature of this method is that the



(a) Backbone curve

(b) Velocity of wheel

Figure 4. Model of shape shift control

deforming part is controlled based on a kinematic model, which enables the movement to utilize the redundancy explicitly, and the position and posture of the trailing part end, which is the control point, can be controlled. The robot can also change a part of its body shape by using the continuous curve approximation method [1]. By deforming a part of the target continuous curve that the robot is fitting, it is possible to deform a part of the robot's body shape. In this case, the new target continuous curve should take into account not only the position of the trailing part but also the torsion around the trunk. Assuming the use of an articulated mobile robot, the twisting of the trailing part may disrupt the proper contact between the environment and the wheel, and resuming of propulsion after a body shape change becomes difficult. For example, it is difficult to resume propulsion if the body shape is such that the side of the wheel is in contact with the environment due to twisting. However, the design of the curve considering the torsion of the trailing part around the trunk axis is very complicated and difficult. In contrast, the proposed method controls the position and orientation of the trailing part edge based on a kinematic model, which makes it easy to design a target posture which does not include this torsion.

The control point is the rear end wheel of the deforming part, and the joint angles of the deforming part are determined based on a kinematic model such that the control point follows the target trajectory. The appropriate range of the deforming part depends on the posture of the robot and the surrounding environment; it is difficult to determine the appropriate range uniquely. In this study, the range of the deforming part is determined by trial and error. The automatic setting of the range will be done in future work.

Figure 6 shows the schematics of the deforming part. The front end of the deforming part is assumed to be fixed. Let Σ_{base} and Σ_{end} be the coordinate system with the origin at the center of the axle of the front and rear end wheels of the deforming part, respectively. The directions of the axes of Σ_{base} and Σ_{end} are shown in Fig. 6. In the coordinate system Σ_{base}, the x-axis is parallel to the previous link of the front end of the deforming part, the y-axis is parallel to the axle of the front end wheel of the deforming part, and the z-axis is orthogonal to these axes. In the coordinate system Σ_{end}, the x-axis is parallel to the rear link of the rear end of the deforming part, the y-axis is parallel to the axle of the rear end wheel of the deforming part, and the z-axis is orthogonal to these axes. The joints of the deforming part are from the pitch joint coaxial with the front end wheel to the pitch joint coaxial with the rear end wheel. Let n_c be the number of joints in the deforming part, $\psi_{c,i}$ be the joint angle of the i-th joint of the deforming part, $\boldsymbol{\psi}_c \in \mathrm{R}^{n_c,1}$ be the joint angles of the deforming part, and $\boldsymbol{w} \in \mathrm{R}^{6,1}$ be the position and orientation of the rear end wheel of the deforming part. To control the three-dimensional position and orientation of the rear end wheel of the deforming part, at least six degrees of freedom are required. Therefore, $n_c \geq 6$ is a constraint that determines the range of the deformation part. Next, a kinematic model for the deforming part is introduced. The position and orientation of the rear end wheel of the deforming part are uniquely determined by the joint angle of the

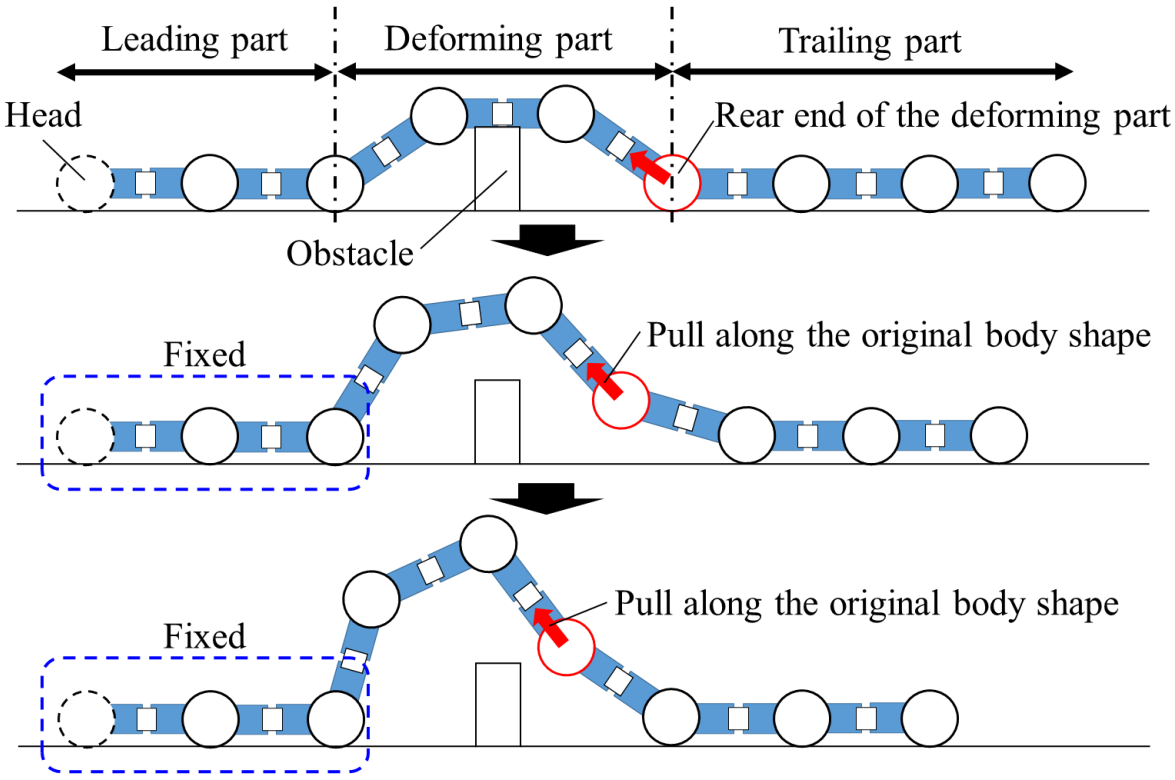


Figure 5. Overview of recovery from the stuck state

deforming part. The kinematic relationship between $\boldsymbol{w}$ and $\boldsymbol{\psi}_c$ is represented as

$$\dot{\boldsymbol{w}} = \boldsymbol{J}\dot{\boldsymbol{\psi}}_c, \tag{6}$$

where $\boldsymbol{J} \in \mathrm{R}^{6,n_c}$ is the jacobian. Let $\boldsymbol{w}_d$ be the target of the control variable. The angular velocity of the joint $\boldsymbol{u} = \dot{\boldsymbol{\psi}}_c$ is calculated as

$$\boldsymbol{u} = \boldsymbol{u}_{\mathrm{ctrl}} + \boldsymbol{u}_{\mathrm{evl}}, \tag{7}$$

$$\boldsymbol{u}_{\mathrm{ctrl}} = \boldsymbol{J}^{\dagger}\left\{\dot{\boldsymbol{w}}_d + \boldsymbol{K}\left(\boldsymbol{w}_d - \boldsymbol{w}\right)\right\}, \tag{8}$$

$$\boldsymbol{u}_{\mathrm{evl}} = k_v\left(\boldsymbol{I} - \boldsymbol{J}^{\dagger}\boldsymbol{J}\right)\boldsymbol{\eta}, \tag{9}$$

where $k_v < 0$ is the control gain for redundancy, $\boldsymbol{K} \in \mathrm{R}^{6,6}$ is the control gain with positive constant for diagonal element for convergence of the controlled variables, and $\boldsymbol{\eta} \in \mathrm{R}^{n_c,1}$ is an arbitrary vector. The $\boldsymbol{u}_{\mathrm{ctrl}}$ and $\boldsymbol{u}_{\mathrm{evl}}$ are the input components for converging the controlled variables to the target and the input component due to redundancy, respectively. The decrease of the arbitrary evaluation function $V(\boldsymbol{\psi}_c)$ is expected by determining $\boldsymbol{\eta}$ using the following equation [21]:

$$\boldsymbol{\eta} = \left[\frac{\partial V(\boldsymbol{\psi}_c)}{\partial \psi_{c,1}}, \cdots, \frac{\partial V(\boldsymbol{\psi}_c)}{\partial \psi_{c,n_c}}\right]^{\top}. \tag{10}$$

By designing the evaluation function appropriately according to the objectives, it is possible to realize secondary control objectives such as obstacle avoidance and joint movement limit avoidance [22], and singular configuration avoidance [23]. The details of the evaluation function for the recovery from stuck state, which is an application example in this research, will be

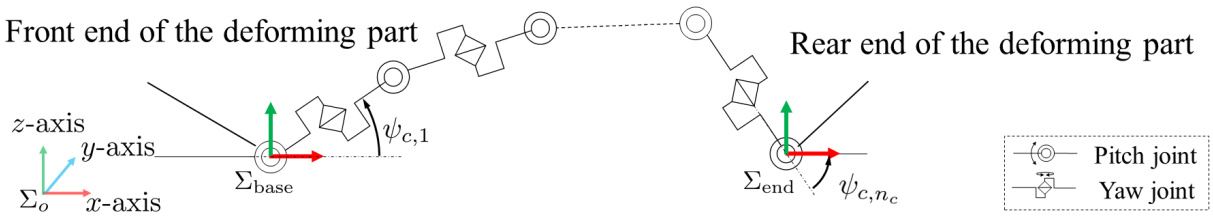


Figure 6. Deforming part

described later. When $\boldsymbol{J}$ is a full row rank, the closed loop of the system is represented as

$$\dot{\boldsymbol{w}}_d - \dot{\boldsymbol{w}} + \boldsymbol{K}(\boldsymbol{w}_d - \boldsymbol{w}) = \boldsymbol{0}. \tag{11}$$

Thus, the control variable $\boldsymbol{w}$ converges to the target value $\boldsymbol{w}_d$ in $t \to \infty$. The target value of the controlled variable $\boldsymbol{w}_d$ is given along the link so that the rear end of the trailing part follows the original body shape. Let $\boldsymbol{w}_0 = [\boldsymbol{p}_0^\top, \boldsymbol{\Theta}_0^\top]^\top$ be the controlled variable at $t = 0$, $T > 0$ be an arbitrary constant, $\boldsymbol{w}_{d,\text{mid}} = [\boldsymbol{p}_{d,\text{mid}}^\top, \boldsymbol{\Theta}_{d,\text{mid}}^\top]^\top$ and $\boldsymbol{w}_{d,\text{end}} = [\boldsymbol{p}_{d,\text{end}}^\top, \boldsymbol{\Theta}_{d,\text{end}}^\top]^\top$ be the position and orientation of the yaw and pitch joints, which is one before the rear end wheels of the deforming part, respectively. The target of the controlled variable $\boldsymbol{w}_d$ is defined as

$$\boldsymbol{w}_d(t) = \begin{bmatrix} \boldsymbol{p}_d(t) \\ \boldsymbol{\Theta}_d(t) \end{bmatrix} \tag{12}$$

$$\boldsymbol{p}_d(t) = \begin{cases} \boldsymbol{p}_0 + (\boldsymbol{p}_{d,\text{mid}} - \boldsymbol{p}_0)\tau, & \tau < 1 \\ \boldsymbol{p}_{d,\text{mid}} + (\boldsymbol{p}_{d,\text{end}} - \boldsymbol{p}_{d,\text{mid}})(\tau - 1), & 1 \le \tau < 2 \\ \boldsymbol{p}_{d,\text{end}}, & \tau \ge 2 \end{cases} \tag{13}$$

$$\boldsymbol{\Theta}_d(t) = \begin{cases} \boldsymbol{\Theta}_0 + \frac{(\boldsymbol{\Theta}_{d,\text{end}} - \boldsymbol{w}_0)}{2}\tau, & \tau \le 2 \\ \boldsymbol{\Theta}_{d,\text{end}}, & \tau > 2 \end{cases} \tag{14}$$

$$\tau = \frac{2t}{T}, \tag{15}$$

$$\dot{\boldsymbol{w}}_d(t) = \frac{\mathrm{d}\boldsymbol{w}_d(t)}{\mathrm{d}t}. \tag{16}$$

After changing body shape, the robot transits the shape shift control mode and resumes propulsion. There is a gap between the original body shape and that after the local body shape control. Thus, the target body shape curve must be reconstructed according to the current body shape. However, the target body shape curve cannot be uniquely determined from the joint angle of the robot. In this study, $\kappa_y(s)$ within $s_h - 2il < s \le s_h - 2(i-l)l$ and $\kappa_p(s)$ within $s_h - (2i+1)l < s \le s_h - (2i-l)l$ are assumed to be constant values, and $\kappa_{y,i}$ and $\kappa_{p,i}$ are $\kappa_y(s)$ and $\kappa_p(s)$, respectively. From Eq. (3), $\kappa_{y,i}$, and $\kappa_{p,i}$ are represented by the following equation [20]:

$$\kappa_{y,i} = -\frac{\psi_{2i-1}}{2l}, \ \kappa_{p,i} = -\frac{\psi_{2i}}{2l}. \tag{17}$$

By reconstructing the target body shape from the current joint angles based on Eq. (17), the robot can resume propulsion by shape shift control with the current body shape. If the joint configuration is similar, the proposed method can be applied to snake robot with passive wheel or without wheels. However, devices to satisfy the assumption that the position of the front end of the connection part does not change are needed.

3.3 *Recovery from a stuck state*

In the application example, recovery from a stuck state, the robot is moved away from the obstacle that caused the stuck state by designing the evaluation function appropriately, and recovers from the stuck state. In addition, the evaluation function is designed to take into account the joint motion limit avoidance and singular posture avoidance at the same time. When the input is calculated using Eq. (7), if the evaluation function is designed as a function that decreases as the joint angle decreases, a decrease in the evaluation function can prevent the joint angle from exceeding its limit [22]. If it is designed as a function that decreases as the singular value of $\boldsymbol{J}$ increases, the singular configuration is expected to be avoided. The robot can recover from the stuck state by moving it away from the obstacle that causes the stuck. If the evaluation function decreases as the distance between the robot and obstacle increases, there is a risk of interference by other obstacles or the ground by moving the robot in unexpected directions. Therefore, we designed an evaluation function that decreases as the distance between the deforming part and an arbitrary target point decreases, and set the target point to a point away from the obstacle. This evaluation function prevents new interference and recovers the robot from the stuck state. The target point for the evaluation function is referred to as the *shape target point*. The shape target point is determined by trial and error based on the relationship between the robot and obstacles. The position and orientation of the front end of the deforming part are fixed, and those of the rear end are controlled as the controlled variables. Thus, the joints around the front and rear ends of the deformation part cannot be moved significantly. In this study, the entire deforming part is moved away from the obstacle by increasing the distance between the obstacle and middle joint of the deforming part, $(n_c+1)/2$-th joint of the deforming part. Let d_c be the distance between the $(n_c+1)/2$-th joint of the deforming part and the shape target point, and the evaluation function be a function that decreases as d_c decreases. The shape target point should be set appropriately considering the range of the deforming part and the environment around the stuck point.

From the above, the evaluation function is defined as

$$V = a_1 V_1 + a_2 V_2 + a_3 V_3, \tag{18}$$

$$V_1 = \frac{1}{n_c} \sum_{i=1}^{n_c} f(\psi_{th}, |\psi_{c,i}|), \tag{19}$$

$$V_2 = \frac{1}{\det\left(\boldsymbol{J}\boldsymbol{J}^\top\right)}, \tag{20}$$

$$V_3 = d_c^2, \tag{21}$$

$$f(x,y) = \begin{cases} (y-x)^3, & \text{if } x < y \\ 0, & \text{otherwise} \end{cases} \tag{22}$$

where $a_1 > 0$, $a_2 > 0$, and $a_3 > 0$ are weights for each evaluation function, and $\psi_{th} > 0$ is the threshold for the avoidance of the joint limit. The influence of each evaluation function can be adjusted by setting the weights of each function. The weights are determined by trial and error so that the order of each evaluation function is equal. The evaluation function V_1 is the function for the avoidance of the joint limit, V_2 is the function for the avoidance of the singular configuration, and V_3 is the function for the recovery from the stuck state. When the input is calculated using Eq. (7), the time derivative of the evaluation function V is represented as

$$\dot{V} = \boldsymbol{\eta}^\top \boldsymbol{u}_{\text{ctrl}} + k_v \boldsymbol{\eta}^\top \left(\boldsymbol{I} - \boldsymbol{J}^\dagger \boldsymbol{J}\right) \boldsymbol{\eta}, \tag{23}$$

where the first term on the right side of the equation is the input for the convergence of the

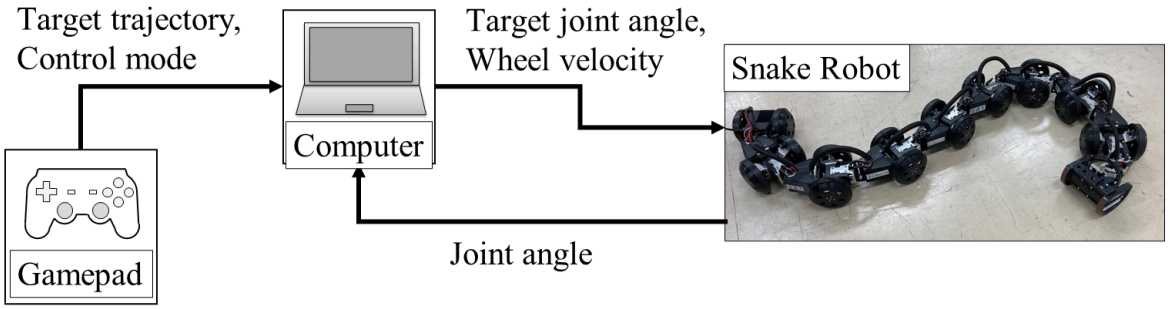


Figure 7. System overview of experiment

controlled variable to the target, and the second term is not positive if $k_v < 0$. Thus, the input calculated using Eq. (7) contributes to a reduction in the evaluation function V. By decreasing V_1, the joint angle decreases and the joint limit is expected to be avoided. By decreasing V_2, the singular value of $\boldsymbol{J}$ increases, and the avoidance of the singular configuration is expected. By decreasing V_3, the distance between the shape target point and deforming part decreases, and the deforming part moves away from the obstacle that causes the stuck state; recovery from the stuck state is expected. These functions are determined based on previous researches [22, 23]. Therefore, the realization of the subtasks is expected by calculating the input based on Eq. (18).

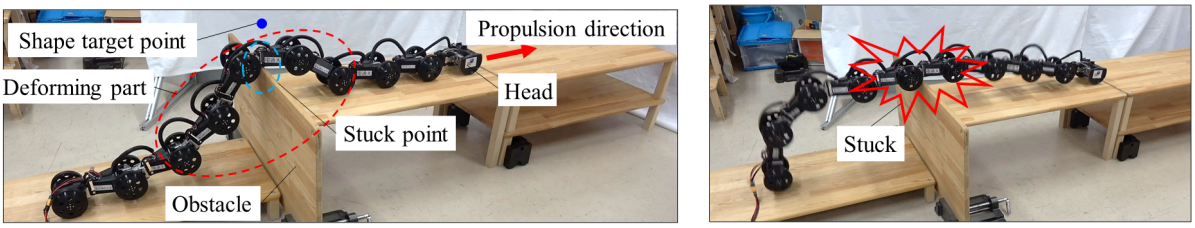
4. Experiments

The experiments on recovery motion from a stuck state were conducted to verify the effectiveness of the proposed method. Experiments in different conditions were conducted: one is the initially stuck state and the other is in a random step field, which is closer to the real environment. Figure 7 shows the robot used in the experiment and the overview of the experimental system. The robot used in the experiment was the T^2 Snake-3.1, with all the wheels of T^2 Snake-3 [20] changed to active wheels. The structural parameters of the robot used are the number of wheels $n = 9$, link length $l = 0.0905$ m, limit angle of the yaw joint $\psi_{\mathrm{y,lim}} = 13\pi/36$ rad, limit angle of the pitch joint $\psi_{\mathrm{p,lim}} = 11\pi/18$ rad, wheel radius $r_w = 0.050$ m, and distance from the link center to wheel $l_{w,j} = 0.054$ m. All the wheels are active, and the robot can be propelled by the rotation of the wheel. The environment is assumed to be an unknown environment, and the positions of the robot and obstacle are not measurable. The control parameters are as follows: the control period $t = 0.050$ s, gain for trajectory tracking $\boldsymbol{K} = \mathrm{diag}([10, 10, 10, 10, 10])$, gain for redundant input $k_v = -4$, and weights for the evaluation function $\boldsymbol{a}_v = \begin{bmatrix} a_1, a_2, a_3 \end{bmatrix} = \begin{bmatrix} 1.5, 0.0, 2.0 \end{bmatrix}$. These parameters were determined by trial and error through preliminary experiments. In these experiments, we set $T = 10$ and transited the robot to the shape shift control mode at approximately $t = 15$ s. Because $T = 10$, the redundant input is large compared to the input for convergence in $t > 10$.

4.1 *Experiments with stuck in the initial state*

In the experiment with stuck in initial state, the robot recovers from the stuck state by the local body shape control. After recovery, the robot transits to the shape shift control mode and resumes propulsion. The timing of switching mode is determined visually by the operator.

We assumed that the robot was stuck in the initial state by the collision between the obstacle and the bottom of the robot between the fourth and fifth wheels, as shown in Figure 8(a). In this case, the vertical drag force acting on the bottom of the robot increased because the load is concentrated at the contact point. Thus, the excessive friction force acts in the direction that



(a) Initial state (b) Propulsion using shape shift control

Figure 8. Initial state and stuck state caused by the collision

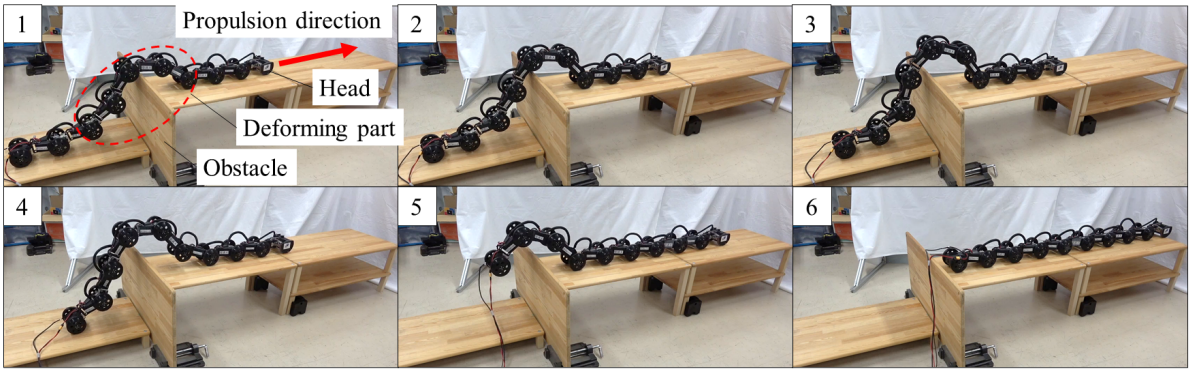
inhibits the propulsion, and the robot cannot be propelled by the shape shift control because the propulsive force is less than the frictional force, as shown in Figure 8(b). The robot may be overloaded if it tries to continue its motion in the stuck state as shown in Figure 8(b). Therefore, it is safer to stop the propulsion once, clear the stuck state, and then resume the propulsion as in the proposed method. The range of the deforming part was set from the third to the seventh wheel. The joints of the deforming part were from the 6th joint to the 14th joint, and the number of deforming part joints was $n_c = 9$. Additionally, we set the shape target point $\boldsymbol{p}_{st} = [0.25, 0.0, 0.25]^\top$. Note that the shape target point is relative to Σ_{base}. Fig. 8(a) shows the position of the shape target point. In this setting of the shape target point, the deforming part is lifted by the redundant input.

Figure 9 shows the experimental results; Fig. 9(a) shows the motion in the experiment; Fig. 9(b) shows the error between the control variable and its target; and Fig. 9(c) shows V_2, the evaluation function for the avoidance of the singular configuration, and d_c, the distance between the middle joint of the deforming part and the shape target point. The error $e_i (i = 1, 2, \cdots, 6)$ in Fig. 9(b) represents i-th element of $\boldsymbol{e}_w = \boldsymbol{w}_d - \boldsymbol{w}$, which is the error between the control variable and its target. As shown in Fig. 9(a), the deforming part was moved away from the obstacle by the proposed motion, and the robot was recovered from the stuck state and resumed propulsion. Fig. 9(b) shows that the error between the controlled variables and its target were very small and the controlled variables converged to its target. From Fig. 9(c), d_c decreases as the motion and the middle joint of the deforming part approach the shape target point, and the robot did not become a singular configuration because V_2 did not diverge to infinity, and the excessive input, which occurs when the robot's posture is close to the singular configuration, was not observed. Additionally, the joint angle did not exceed its limit angle.

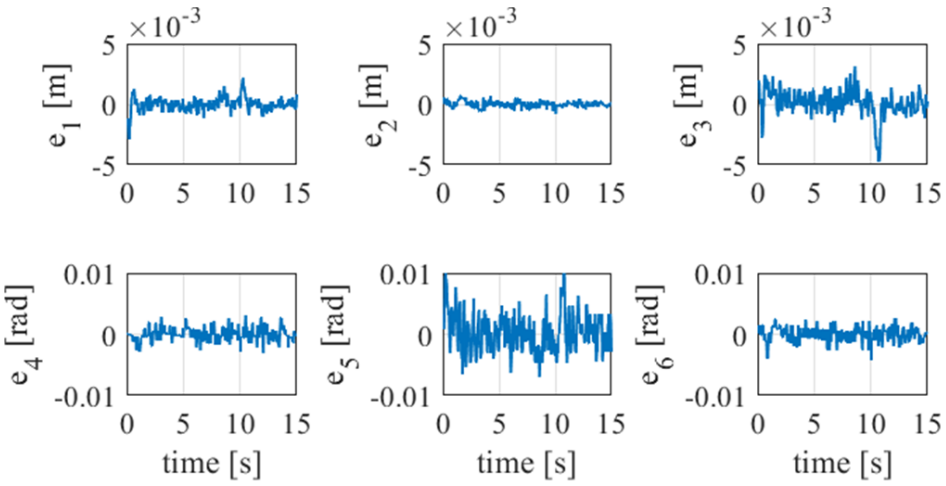
The effectiveness of the proposed method was confirmed from the experimental results; the robot could recover from the stuck state caused by the force acting on the robot using the proposed method.

4.2 *Experiments in the random step field*

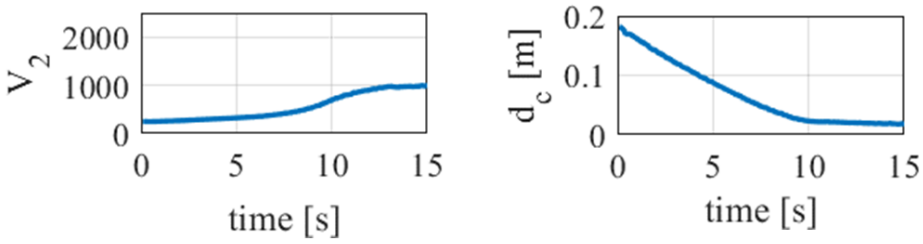
Experiments on a random step were conducted to verify the operation in a setting closer to the real environment. Figure 10 shows the experimental environment. The random step consists of 90 mm square prisms placed at equal intervals, and various environments can be constructed by replacing the prisms. The experiments were conducted in several environments by changing the arrangement of the prisms. The random step field contained steps with a maximum height difference of 180 mm, and the field was not always symmetrical. The robot started its motion before a random step and was propelled over the random step. The robot was operated by the operator, and the operator adjusted the amount and direction of propulsion while visually checking the situation. In addition, when the robot became a stuck state, the range of the deforming part and the shape target point were adjusted based on the state of the stuck. The range of the deforming part was set to include the stuck point, and the shape target point



(a) Motion using the proposed method



(b) Time responses of $\boldsymbol{e}_w$



(c) Time responses of V_2 and d_c

Figure 9. Experimental result

was set to be far away from the stuck point. Then, in order to check how the robot behaves with the set parameters, the behavior of the robot was simulated on a PC before sending control inputs to the actual robot. After confirming whether excessive input would not occur when input was given under the same conditions, and whether the deforming part would not behave in an unexpected way due to the setting of the shape target point, the control input was sent to the actual robot. The joint angles in the simulations were estimated using the Euler method based

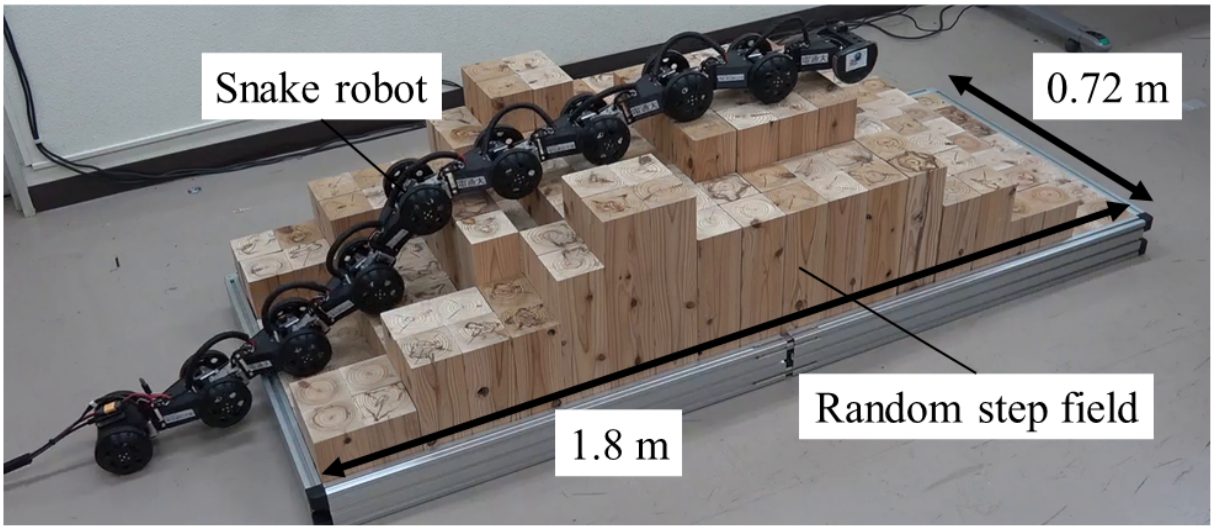


Figure 10. Experimental settings in random step field

on the calculated inputs, and note that the behavior of the robot does not be completely equal that of the actual robot. The robot was often stuck in environments such as crossing valleys via high steps or small footholds. As experimental results, the proposed method was able to recovery the robot from the stuck state in most of the cases where the robot became stuck state. However, the robot could not be recovered from the stuck state in some cases. The details of each case are shown below.

Figure 11 shows the behavior when the recovery motion was successful. As shown in Fig. 11, while propelling the robot on a random step with shape shift control, the bottom of the robot between the 3 and 4 wheels got stuck on the step. Therefore, the range of the deforming part was set the range between 3rd and 7-th wheel and the shape target point to the position where the deforming part is lifted upward. As shown in Fig. 11, it can be confirmed that the stuck state was recovered by lifting the part that caused the stuck state. In addition, the trailing part maintained contact with the environment and resumed propulsion after the local body shape control.

In contrast, there were cases where the recovery from the stuck state failed, as in Figures 12. The robot often failed to recover from the stuck state in the situation which the wheels of the leading or trailing part were not contact with the environment properly. The figure of the process of getting stuck is omitted. Fig. 12(a) shows that the stuck occurred in two places at almost the same time. Since the proposed control method assumes that the stuck occurs at only one place, the robot could not recover from the stuck state when the stuck occurred at multiple places. The cause of failure is thought to be that the other stuck interfered with the movement to leave one of the stuck points, and thus prevented the robot from recovering from the stuck state. In addition, in cases where stuck occurred in multiple points, it was difficult to visually identify the stuck points. To operate in a real environment, it is necessary to identify the stuck part by installing sensors (e.g. a force sensor for measuring the contact force with the environment). In the case of Fig. 12(b), a new stuck was generated at the trailing part during the local body shape control. In the proposed method, the trailing part follows the body shape of the robot based on the continuous curve approximation method. In this process, new stuck was generated at other locations and the robot failed to recover from a stuck state. In these cases where stuck occurred at multiple locations, it is necessary to design methods for local body shape control at multiple locations and evaluation functions that allow the robot to move away from multiple obstacles. Addressing these issues is an issue for the future. Similar problems will occur in more complex environment. In contrast, if the leading part and trailing part are contact with the environment properly, the proposed method is expected to be effective in more complex environment.

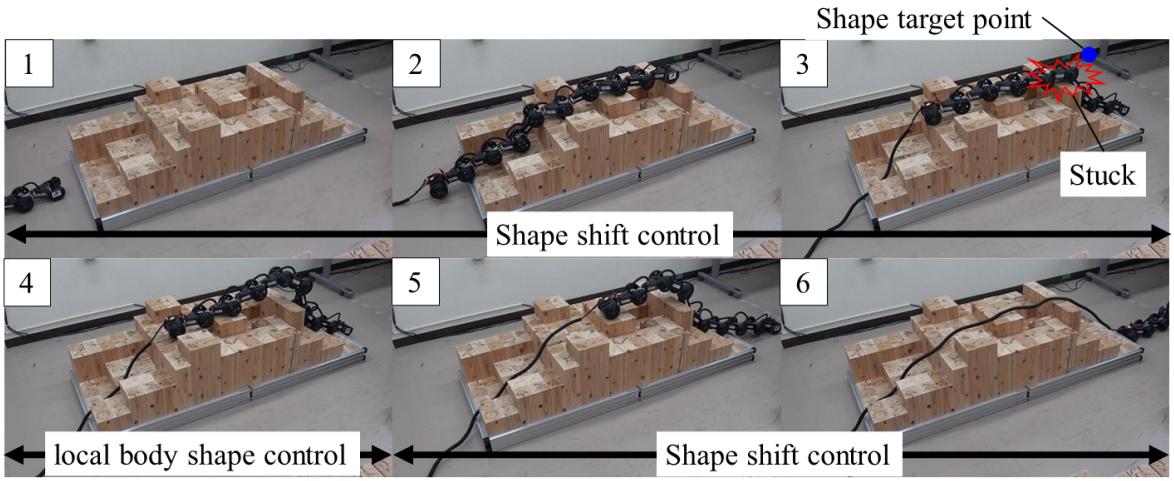
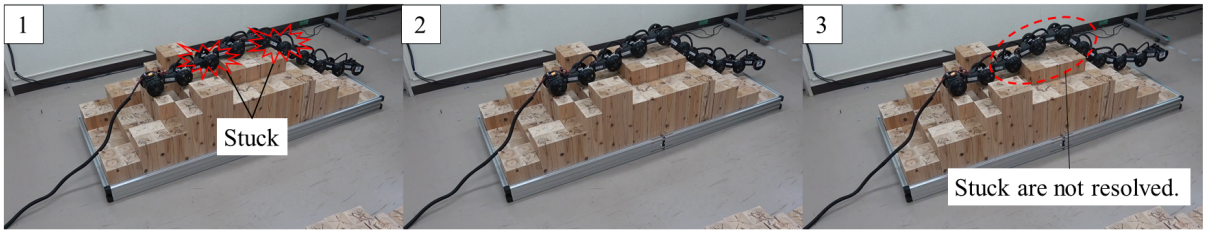
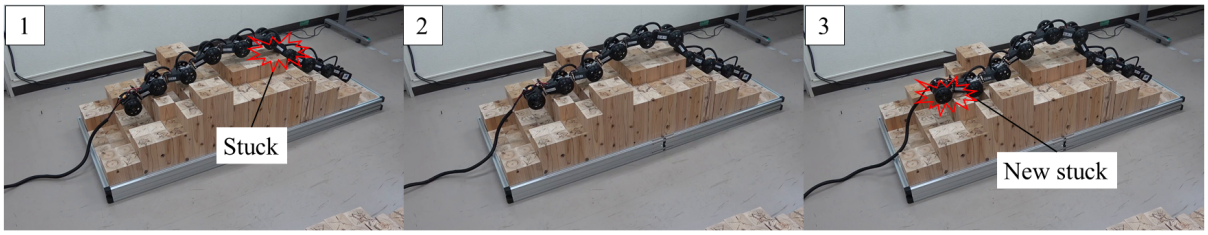


Figure 11. Experimental result in the random step



(a) Failure case due to collision in multiple locations



(b) Failure case due to occurrence of a new collision

Figure 12. Experimental results of failed recovery from stuck state

5. Conclusion

This paper proposed a local body shape control method and an application for recovering an articulated mobile robot from a stuck state. A propulsion in a complex environment and the local body shape control were achieved simultaneously by coexisting the continuous curve approximation method which is easy to design a motion in a complex three-dimensional environment and the kinematic model based method which can utilize the redundancy of the robot. A part of the robot's body shape which needs local body shape change was changed based on the kinematic model and the other parts followed its original body shape based on the continuous curve approximation method. In addition, this paper proposed the method for recovering the robot from a stuck state as an application of the proposed method. The robot's body shape around

the stuck point was controlled based on the kinematic model, and the robot was recovered from the stuck state and resumed propulsion. To verify the effectiveness of the proposed method, the experiments with stuck state in the initial state and in the random step field, and it was verified that the robot was recovered from a stuck state by the proposed method. However, in the random step field, the robot could not be recovered from a stuck state when the robot was a stuck state in multiple points. In addition, in the proposed method, the robot was controlled under the assumption that the stuck points were visible. Future work will include the development of hardware that can identify the stuck point and an automatic setting method of the parameters for the control, and the method for recovery from the stuck state with multiple points.

Acknowledgment

This work was partially supported by the JAEA Nuclear Energy S&T and Human Resource Development Project through concentrating wisdom Grant Number JPJA19B19209927, and the ImPACT Program of Council for Science, Technology and Innovation (Cabinet Office, Government of Japan).

References

[1] Yamada H, Hirose S. Study of active cord mechanism: approximations to continuous curves of a multi-joint body (Japanese). Journal of Robotics Society of Japan. 2008;26(1):110–120.
[2] Takemori T, Tanaka M, Matsuno F. Ladder Climbing with a Snake Robot. In: Proc. IEEE/RSJ Int. Conf. Intelligent Robots and Systems. 2018. p. 1–9.
[3] Takemori T, Tanaka M, Matsuno F. Gait Design of a Snake Robot by Connecting Curve Segments and Experimental Demonstration. IEEE Trans Robotics. 2018;34(5):1384–1391.
[4] Wright C, Buchan A, Brown B, Geist J, Schwerin M, Rollinson D, Tesch M, Choset H. Design and Architecture of the Unified Modular Snake Robot. In: Proc. IEEE Int. Conf. Robotics and Automation. 2012. p. 4347–4354.
[5] Kamegawa T, Harada T, Gofuku A. Realization of cylinder climbing locomotion with helical form by a snake robot with passive wheels. In: Proc. IEEE Int. Conf. Robotics and Automation. 2009. p. 3067–3072.
[6] Qi W, Kamegawa T, Gofuku A. Helical wave propagation motion for a snake robot on a vertical pipe containing a branch. Journal of Artificial Life and Robotics. 2018;23(4):515–522.
[7] Pfotzer L, Staehler M, Hermann A, Roennau A, Dillmann R. KAIRO 3: Moving Over Stairs & Unknown Obstacles with Reconfigurable Snake-Like Robots. In: Proc. European Conf. on Mobile Robots. 2015. p. 1–6.
[8] Komura H, Yamada H, Hirose S. Development of snake-like robot ACM-R8 with large and monotread wheel. Advanced Robotics. 2015;29(17):1081–1094.
[9] Tanaka M, Kon K, Nakajima M, Matsumoto N, Fukumura S, Fukui K, Sawabe H, Fujita M, Tadakuma K. Development and field test of the articulated mobile robot T^2 Snake-4 for plant disaster prevention. Advanced Robotics. 2020;34(2):70–88.
[10] Borenstein J, Hansen M, Nguyen H. The OmniTread OT-4 serpentine robot for emergencies and hazardous environments. In: International Joint Topical Meeting: Sharing Solutions for Emergencies and Hazardous Environments. 2006. p. 12–15.
[11] Kouno K, Yamada H, Hirose S. Development of Active-Joint Active-Wheel High Traversability Snake-Like Robot ACM-R4.2. Journal of Robotics and Mechatronics. 2013;25(3):559–566.
[12] Matsumoto N, Tanaka M, Nakajima M, Fujita M, Tadakuma K. Development of a folding arm on an articulated mobile robot for plant disaster prevention. Advanced Robotics. 2020;34(2):89–103.
[13] Tanaka M, Kon K, Tanaka K. Range-sensor-based Semiautonomous Whole-body Collision Avoidance of a Snake Robot. IEEE Trans on Control Systems Technology. 2015;23(5):1927–1934.
[14] Transeth AA, Leine RI, Glocker C, Pettersen KY, Liljebäck P. Snake Robot Obstacle-Aided Locomotion: Modeling, Simulations, and Experiments. IEEE Trans on Robotics. 2008;24(1):88–104.

[15] Liljebäck P, Pettersen KY, Stavdahl Ø, Gravdahl JT. Snake Robot Locomotion in Environments With Obstacles. IEEE/ASME Trans on Mechatronics. 2012;17(6):1158–1169.
[16] Mori M, Hirose S. Three-dimensional serpentine motion and lateral rolling by active cord mechanism ACM-R3. In: Proc. IEEE/RSJ Int. Conf. Intelligent Robots and Systems. 2002. p. 829–834.
[17] Kimura H, Hirose S, Shimizu K. Stuck evasion control for Active-Wheel Passive-Joint snake-like mobile robot 'Genbu'. In: IEEE Int. Conf. Robotics and Automation. 2004. p. 5087–5092.
[18] Miyanaka H, Wada N, Kamegawa T, Sato N, Tsukui S, Igarashi H, Matsuno F. Development of an unit type robot "KOHGA2" with stuck avoidance ability. In: IEEE Int. Conf. Robotics and Automation. 2007. p. 3877–3882.
[19] Yamada H, Takaoka S, Hirose S. A snake-like robot for real-world inspection applications (the design and control of a practical active cord mechanism). Advanced Robotics. 2013;27(1):47–60.
[20] Tanaka M, Nakajima M, Suzuki Y, Tanaka K. Development and Control of Articulated Mobile Robot for Climbing Steep Stairs. IEEE/ASME Transactions on Mechatronics. 2018;23(3):531–541.
[21] Nakamura Y, Hanafusa H, Yoshikawa T. Task-Priority Based Redundancy Control of Robot Manipulators. The International Journal of Robotics Research. 1987;6(2):3–15.
[22] Tanaka M, Tanaka K. Shape Control of a Snake Robot with Joint Limit and Self-collision Avoidance. IEEE Trans Control Systems Technology. 2017;25(4):1441–1448.
[23] Tanaka M, Tanaka K. Singularity Analysis of a Snake Robot and an Articulated Mobile Robot With Unconstrained Links. IEEE Trans on Control Systems Technology. 2016;24(6):2070–2081.