


Article

Target Tracking of Snake Robot with Double-Sine Serpentine Gait Based on Adaptive Sliding Mode Control

Zhifan Liu ¹ , Wu Wei ^{1,2,*}, Xiongdin Liu ¹ and Siwei Han ¹¹ School of Automation Science and Engineering, South China University of Technology, Guangzhou 510641, China² The Key Laboratory of Autonomous Systems and Networked Control, Ministry of Education, Unmanned Aerial Vehicle Systems Engineering Technology Research Center of Guangdong, South China University of Technology, Guangzhou 510641, China

* Correspondence: weiwu@scut.edu.cn; Tel.: +86-139-2601-3970

Abstract: This paper studies the target tracking control strategy of a snake robot and proposes an adaptive sliding mode control method. The strategy ensures the robot follows the target path by controlling the joint angle through feedback, pushing the robot to reach the target position through gait function. In order to achieve target tracking, a kinematic model of a snake robot was first established in this paper. Then, we used double-sine serpentine gait to solve the problem of low steering efficiency caused by regular serpentine gait, and we explored the relationship between control parameters and robot steering. On the basis of gait, in order to further improve the efficiency of target tracking for the snake robot, an adaptive sliding mode control method, based on a new sliding mode reaching law, was proposed. Finally, the effectiveness and practicability of the proposed strategy was demonstrated by comparative analysis and simulation experiments.

Keywords: snake robot; double-sine serpentine gait; target tracking; adaptive sliding mode control



Citation: Liu, Z.; Wei, W.; Liu, X.; Han, S. Target Tracking of Snake Robot with Double-Sine Serpentine Gait Based on Adaptive Sliding Mode Control. *Actuators* **2023**, *12*, 38. <https://doi.org/10.3390/act12010038>

Academic Editor: Hicham Chaoui

Received: 27 November 2022

Revised: 6 January 2023

Accepted: 8 January 2023

Published: 10 January 2023



Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

As one of the hotspots of robot research, the bionic robot plays an important role in engineering applications [1–4]. Inspired by the biological snake, the snake robot has strong flexibility and high stability. Compared with wheeled, tracked, and legged robots, the snake robot is more adaptable to narrow and unstructured environments [5–9]. Therefore, it is widely used in post-disaster rescue, terrain reconnaissance and pipeline inspection [10–15].

At present, research on the snake robot mainly includes mechanical structure, motion gait, motion control method and algorithms related with robot vision information. The snake robot usually adopts a modular structure, which drives the robot forward by controlling the joint angle. In the 1940s, Gray [16] studied the locomotion of biological snakes, and used mathematical methods to describe the movement mechanism. Hirose developed the first snake robot prototype, ACM-III, in 1972 [17]. By observing a large quantity of biological snake motions, Hirose [18] proposed a serpentine gait, and added a constant offset to change the motion direction of the snake robot. Ye et al. [19,20] analyzed the serpenoid curve of the snake robot, and proposed the amplitude modulation method, the phase modulation method and the side movement modulation method to complete the turning motion of a snake robot. Dai et al. [21] referred to geometric mechanics to control a snake robot moving across a particle surface. He established an empirical model of the local relationship between change in the robot shape and position change, which could give an efficient turning gait. However, it is difficult to achieve a sharp turn of the fuselage in a narrow space with a single wave using the above studies. Inspired by the postures of *Caenorhabditis elegans* (*C. elegans*), Wang et al. [22,23] introduced a template with two coplanar traveling waves, and proposed the omega turn method to realize robot turning in place.

Motion control determines whether a snake robot can work accurately in narrow and unstructured environments, so it is necessary to study the control strategy of the snake robot [24–26]. Liljebäck et al. [27,28] proposed a Line-of-Sight (LOS) guidance law, and controlled the motion direction angle of a snake robot by adjusting the joint offset to realize the robot's path tracking. Kelasidi et al. [29] proposed an integrated line-of-sight guidance law, and designed a direction controller with serpentine gait to design a trajectory tracking controller for an underwater snake robot. Ariizumi et al. [30] designed a controller based on the minimization of cost function to realize the head trajectory tracking control of a snake robot. Zhang et al. [31,32] proposed a path edge guidance strategy with the angular symmetry adjustment method to control a snake robot following the centerline of desired path. Xiao [33] adopted a sliding mode controller, based on a power reaching law, and integrated target detection technology to achieve path tracking. Cao et al. [34] analyzed the relationship between phase offset of the pitch joint and the direction of the snake robot, and proposed a control method with an adaptive LOS guidance law, which improved the tracking accuracy of the snake robot.

Although researchers have made many achievements in the motion control of the snake robot, the following problems still need to be solved: the traditional serpentine gait limiting the turning range of the robot, and the imperfect control strategy of the snake robot in target tracking, causing slow error convergence.

In order to improve the efficiency of robot target tracking, this paper proposes a target tracking control strategy based on an adaptive sliding mode controller. This control strategy drives the snake robot to change its heading angle by designing an effective feedback control law, so that the robot can quickly and stably converge to the desired path containing the target point. Meanwhile, this strategy controls the joint angle of the robot to reach the target position through the gait function. First, according to the structure of the snake robot, the kinematic model was obtained, based on screw theory. Second, considering that the target tracking process requires a lot of direction adjustment, and it is difficult to control a snake robot with many control parameters in tracking the target point, we established a double-sine serpentine gait. Third, based on the double-sine serpentine gait, an adaptive sliding mode controller was designed to achieve target tracking, and the stability of the controller was analyzed theoretically. Finally, through a simulation experiment, the tracking characteristics of the robot under different controllers were analyzed, and the effectiveness and superiority of the adaptive tracking controller, based on the double-sine serpentine gait, were verified.

Compared with existing works, the main contributions of this paper are as follows:

(1) A snake robot in regular serpentine gait, with the offset of joint angle, generates jitter when adjusting the direction angle, which causes inefficiency in controlling changes in motion direction of the snake robot. To solve the problem, this paper referred to the superposition of the two sine waves method, and established a double-sine serpentine gait model. In addition, we also analyzed the influence of each control parameter in the gait model on the movement form of the snake robot in detail.

(2) In order to further improve the tracking accuracy and convergence speed of the controller, we proposed an adaptive sliding mode controller, based on a new sliding mode reaching law, which effectively realized target tracking by controlling the steering of the snake robot.

The remainder of this paper is organized as follows. Section 2 mainly expounds the construction of the kinematic model of the snake robot. Section 3 establishes the double-sine serpentine gait model and analyzes the influence of gait control parameters on the morphology of the snake robot. Section 4 describes the design process and overall structure of the target tracking controller for the snake robot. Section 5 verifies the effectiveness of the gait control function and adaptive sliding mode controller proposed in this paper through simulation experiments. Finally, we conclude and propose future research in Section 6.

2. Model Description

This paper builds a snake robot with 16 joints, and its simulation model is shown in Figure 1. Adjacent joint modules are connected by means of orthogonal connection, where the odd joints are pitch joints, and the even joints are yaw joints. During the movement of the serpentine gait, the pitch joint angle is set to zero, and the yaw joint produces a sine wave-like rotation [10,35]. Before analyzing the serpentine gait and controller design, this paper gives a partial definition for the simple structure of the snake robot shown in Figure 2, and describes the kinematics of the snake robot.

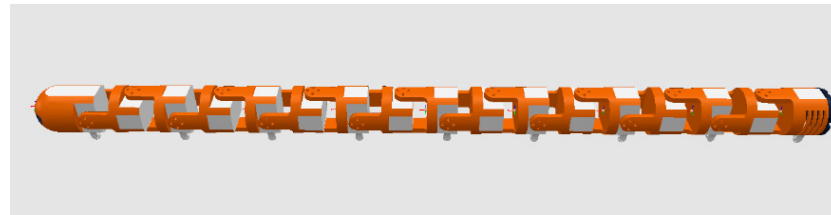


Figure 1. Simulation model of snake robot.

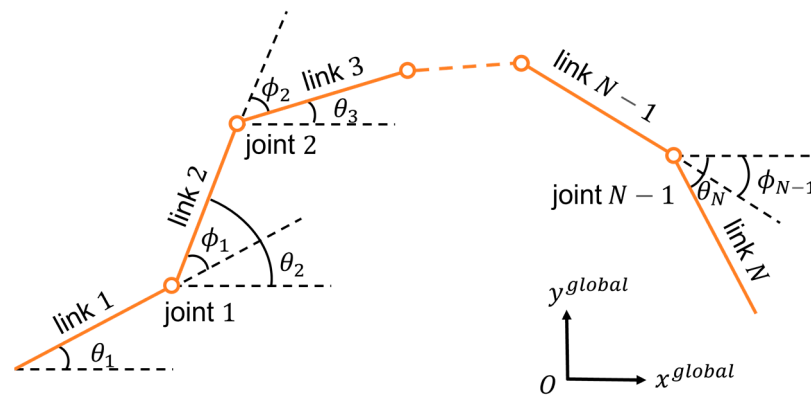


Figure 2. Simple structure of snake robot with $N - 1$ yaw joints.

2.1. The Parameters of Snake Robot

Table 1 shows the basic symbols used in this paper, conducive to describing the kinematic model of the snake robot.

Table 1. Parameters that describe the snake robot.

Symbol	Description	Vector
N	The total number of joints	
n	The joint index	
l	The length of a link	
$\bar{\theta}$	The forward direction angle of snake robot	
$\bar{\theta}_{exp}$	The expected direction angle of snake robot	
θ_n	Angle between the link n and the global x -axis	$\theta \in \mathbb{R}^{N+1}$
ϕ_n	Angle of joint n	$\phi \in \mathbb{R}^N$
r_n	The center point of joint n	$r \in \mathbb{R}^N$
ω_n	The unit vector in the direction of the rotation axis of joint n	$\omega \in \mathbb{R}^N$
ξ_n	The twist of joint n	$\xi \in \mathbb{R}^N$
(p_x, p_y)	Global coordinates of the center of mass of snake robot	$p \in \mathbb{R}^2$

2.2. Definition of Snake Robot Kinematics

Definition 1. The link angle θ_n of the snake robot is defined as the angle between the link n and the global x -axis [27] (p. 42), where counterclockwise rotation is specified as the positive direction, $n \in \{1, \dots, N\}$.

Definition 2. The joint angle ϕ_n of the snake robot is defined as the angle between the link n and the link $n + 1$ [27] (p. 42). The relationship between the joint angle and the link angle is as follows:

$$\phi_n = \theta_n - \theta_{n+1}. \quad (1)$$

Definition 3. The forward direction angle $\bar{\theta}$ of the snake robot is defined as the average value of the link angle θ_n [27] (p. 43), namely:

$$\bar{\theta} = \frac{1}{N} \sum_{n=1}^N \theta_n. \quad (2)$$

2.3. Kinematic Model of Snake Robot

To realize the kinematic model of a snake robot with multi-joint redundancy, the screw theory method needs to establish the screw coordinate system of each joint relative to the base coordinate system. Compared with the D–H parameter method, this method avoids the coordinate transformation between lots of transfer matrices and simplifies the solution process of the kinematic equations [36–38]. Therefore, we established kinematic model of the snake robot based on the screw theory.

In this paper, we took the front end of the snake robot as the base coordinate system $\{S\}$, where the coordinate origin O_S was set on the central axis of the snake head, the X_S axis along the direction of the snake head to the snake tail, and the Z_S axis was the same as the axis direction of the first joint. The tool coordinate system $\{T\}$ was established at the end of the robot, where the X_T axis was in the same direction as the X_S axis, the Z_T axis was different from the axis direction of the adjacent joint by $\pi/2$, and the Y_T axis could be determined by the right-hand rule. The screw coordinate model of the snake robot is shown in Figure 3, where the distance of O_S from the first joint is l_s , the length of O_T from its adjacent joint is l_e , and the distance between adjacent joints is l . The value r_n represents the center point of joint n , and ω_n represents the unit vector in the same direction as the rotation axis of joint n .



Figure 3. The screw coordinate model of snake robot.

When all joints are in the initial state, as shown in Figure 3, the pose transformation matrix ${}^S_T g(0)$ of the tool coordinate system $\{T\}$ relative to the base coordinate system $\{S\}$ is as follows:

$${}^S_T g(0) = \begin{bmatrix} l_s + l_e + (N-1)l & & & \\ {}^S_T R & 0 & & \\ & 0 & & \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad (3)$$

where N is the total number of joints, $l_s + l_e + (N-1)l$ represents the distance between the origins of the two coordinate systems, and ${}^S_T R$ is the rotation matrix of the tool coordinate system $\{T\}$ relative to the base coordinate system $\{S\}$. The general solution of the rotation matrix ${}^S_T R$ in Equation (3) is shown in Equation (4). According to the periodicity of the rotation axes of joints and the total number of joints, when the total number of joints equals $4j$ ($j = 1, 2, 3, \dots$), the tool coordinate system $\{T\}$ is parallel to the base coordinate

system $\{S\}$, so we have ${}^S_T R$ as the unit matrix. When the total number of joints equals $4j + 1$ ($j = 1, 2, 3, \dots$), the difference between the tool coordinate system $\{T\}$ and the base coordinate system $\{S\}$ along the x -axis is $\frac{\pi}{2}$, we have ${}^S_T R = \text{Rot}(x, \frac{\pi}{2})$. By analogy, when the total number of joints equals $4j + 2$ ($j = 1, 2, 3, \dots$), we have ${}^S_T R = \text{Rot}(x, \pi)$. When the total number of joints equals $4j + 3$ ($j = 1, 2, 3, \dots$), we have ${}^S_T R = \text{Rot}(x, -\frac{\pi}{2})$. In summary, the mathematical expression of ${}^S_T R$ is as follows:

$${}^S_T R = \begin{cases} \text{Rot}(x, \frac{\pi}{2}) & N \bmod 4 = 1, \\ \text{Rot}(x, \pi) & N \bmod 4 = 2, \\ \text{Rot}(x, -\frac{\pi}{2}) & N \bmod 4 = 3, \\ E_{3 \times 3} & N \bmod 4 = 0, \end{cases} \quad (4)$$

where \bmod represents the remainder, $\text{Rot}(x, \theta)$ represents the rotation matrix when rotating θ around the x -axis, as in the Equation (5).

$$\text{Rot}(x, \theta) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\theta & -\sin\theta \\ 0 & \sin\theta & \cos\theta \end{bmatrix}. \quad (5)$$

According to Figure 3, the unit vector ω_n can be written as Equation (6). The distance d_n between the n -th joint center r_n and the origin O_s of the base coordinate system is shown by Equation (7).

$$\omega_n = \begin{cases} [0 \ 0 \ 1]^T & n \bmod 4 = 1, \\ [0 \ -1 \ 0]^T & n \bmod 4 = 2, \\ [0 \ 0 \ -1]^T & n \bmod 4 = 3, \\ [0 \ 1 \ 0]^T & n \bmod 4 = 0. \end{cases} \quad (6)$$

$$d_n = \begin{bmatrix} (n-1)l + l_s \\ 0 \\ 0 \end{bmatrix}. \quad (7)$$

From this, the twist for the joint n can be obtained as:

$$\begin{aligned} \xi_n &= \begin{bmatrix} \omega_n \\ v_n \end{bmatrix} = \begin{bmatrix} \omega_n \\ d_n \times \omega_n \end{bmatrix} \\ &= \begin{cases} \begin{bmatrix} 0 & 0 & 1 & 0 & -(i-1)l - l_s & 0 \end{bmatrix}^T & n \bmod 4 = 1, \\ \begin{bmatrix} 0 & -1 & 0 & 0 & 0 & -(i-1)l - l_s \end{bmatrix}^T & n \bmod 4 = 2, \\ \begin{bmatrix} 0 & 0 & -1 & 0 & (i-1)l + l_s & 0 \end{bmatrix}^T & n \bmod 4 = 3, \\ \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & (i-1)l + l_s \end{bmatrix}^T & n \bmod 4 = 0. \end{cases} \end{aligned} \quad (8)$$

Since the joints of snake robot are revolute joints, and $\|\omega\| = 1$, the product of exponential (POE) can be obtained as [37]:

$$\begin{aligned} e^{\theta \hat{\xi}} &= \sum_{n=0}^{\infty} \frac{(\theta \hat{\xi})^n}{n!} \\ &= E_4 + \theta \hat{\xi} + \hat{\xi}^2 (1 - \cos\theta) + \hat{\xi}^3 (\theta - \sin\theta) \\ &= \begin{bmatrix} e^{\theta \hat{\omega}} & (E - e^{\theta \hat{\omega}})(\omega \times v) + \theta \omega \omega^T v \\ 0 & 1 \end{bmatrix}, \end{aligned} \quad (9)$$

where $e^{\theta \hat{\omega}}$ can be obtained by Rodriguez formula:

$$e^{\theta \hat{\omega}} = E + \hat{\omega} \sin\theta + \hat{\omega}^2 (1 - \cos\theta). \quad (10)$$

Combining Equations (6)–(10), the POE equation of each joint can be obtained as follows:

$$e^{\theta_n \hat{\xi}_n} = \begin{cases} \begin{bmatrix} \cos\theta_n & -\sin\theta_n & 0 & (1 - \cos\theta_n)((n-1)l + l_s) \\ \sin\theta_n & \cos\theta_n & 0 & -\sin\theta_n((n-1)l + l_s) \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} & n \bmod 4 = 1, \\ \begin{bmatrix} \cos\theta_n & 0 & -\sin\theta_n & (1 - \cos\theta_n)((n-1)l + l_s) \\ 0 & 1 & 0 & 0 \\ \sin\theta_n & 0 & \cos\theta_n & -\sin\theta_n((n-1)l + l_s) \\ 0 & 0 & 0 & 1 \end{bmatrix} & n \bmod 4 = 2, \\ \begin{bmatrix} \cos\theta_n & \sin\theta_n & 0 & (1 - \cos\theta_n)((n-1)l + l_s) \\ -\sin\theta_n & \cos\theta_n & 0 & \sin\theta_n((n-1)l + l_s) \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} & n \bmod 4 = 3, \\ \begin{bmatrix} \cos\theta_n & 0 & \sin\theta_n & (1 - \cos\theta_n)((i-1)l + l_s) \\ 0 & 1 & 0 & 0 \\ -\sin\theta_n & 0 & \cos\theta_n & \sin\theta_n((i-1)l + l_s) \\ 0 & 0 & 0 & 1 \end{bmatrix} & n \bmod 4 = 0. \end{cases} \quad (11)$$

Therefore, the kinematic model of the snake robot with N joints is as follows:

$${}^S_T g(\theta) = e^{\theta_1 \hat{\xi}_1} e^{\theta_2 \hat{\xi}_2} \dots e^{\theta_{N-1} \hat{\xi}_{N-1}} {}^S_T g(0). \quad (12)$$

According to Equation (12), we can obtain the transformation matrix between the tool coordinate system $\{T\}$ established on the N -th joint of the snake robot and the base coordinate system $\{S\}$ established on the head of the snake robot. Through the feedback data of the angle sensors installed at each joint of the robot, the transformation matrix of any joint on the robot, relative to the base coordinate system, can be obtained. With the GPS positioning data of the snake head, the position and attitude data of any joint on the entire robot, relative to the world coordinate system, can be obtained.

This paper verified the kinematic model of the snake robot based on the simulation platform of CoppeliaSim and MATLAB. CoppeliaSim is used to build simulation models and receives control commands sent by MATLAB. MATLAB was used to calculate, analyze and verify the basic theory and gait algorithm of the robot. Meanwhile, it sent control commands through the interface function provided by CoppeliaSim [39].

In this paper, the snake robot was controlled by the above simulation software to achieve a serpentine gait. Table 2 shows the technical specifications of our snake robot. The pose matrix of the robot tail could be calculated according to the established kinematic model and the base coordinate pose fed back by CoppeliaSim. The actual position of the robot tail was compared with the position calculated by the kinematic model, shown in the Figure 4. The calculated data of the pose matrix was compared with the actual pose matrix, as shown in Table 3. It could be found that the error between the calculated data and the actual data was very small, so the kinematic model of snake robot was correct.

Table 2. Overview of snake robot specifications in simulation.

Items	Description
Mass	1.5 kg
Length	$l_s = 0.09$ m, $l = 0.07$ m, $l_e = 0.10$ m
Diameter	0.05 m
Sensing	Angle sensor, Vision sensor, Proximity sensor
Number of modules	17
Number of joints	16
Connection mode between modules	Orthogonal connection

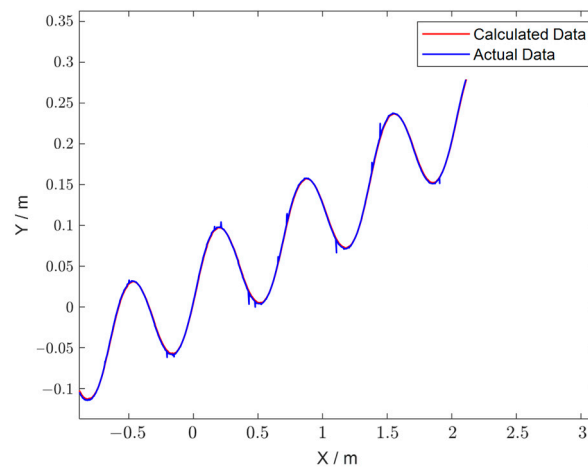


Figure 4. Comparison between calculated position and actual position of the robot tail.

Table 3. Comparison of calculated data and actual data of pose matrix.

Pose Matrix Time	Calculated Data	Actual Data
2.20 s	$\begin{bmatrix} 0.0009 & 0.0697 & -0.9976 & -0.8325 \\ -0.0023 & -0.9976 & -0.0697 & -0.1103 \\ -1.0000 & 0.0023 & -0.0008 & 0.0534 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$\begin{bmatrix} 0.0005 & 0.0696 & -0.9976 & -0.8368 \\ -0.0023 & -0.9976 & -0.0696 & -0.1117 \\ -1.0000 & 0.0023 & -0.0003 & 0.0534 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
3.70 s	$\begin{bmatrix} 0.0009 & 0.4387 & -0.8987 & -0.6972 \\ -0.0019 & -0.8986 & -0.4387 & -0.0973 \\ -1.0000 & 0.0021 & 0.0001 & 0.0534 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$\begin{bmatrix} 0.0005 & 0.4386 & -0.8987 & -0.7012 \\ -0.0019 & -0.8987 & -0.4386 & -0.0991 \\ -1.0000 & 0.0020 & 0.0004 & 0.0534 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
12.95 s	$\begin{bmatrix} 0.0005 & -0.2357 & -0.9718 & -0.1946 \\ -0.0017 & -0.9718 & 0.2357 & -0.0506 \\ -1.0000 & 0.0015 & -0.0001 & 0.0534 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$\begin{bmatrix} 0.0001 & -0.2357 & -0.9718 & -0.1988 \\ -0.0017 & -0.9718 & 0.2358 & -0.0507 \\ -1.0000 & 0.0017 & -0.0004 & 0.0534 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
29.75 s	$\begin{bmatrix} 0.0003 & 0.5652 & -0.8249 & 0.6715 \\ 0.0034 & -0.8249 & -0.5652 & 0.0911 \\ -1.0000 & -0.0027 & -0.0022 & 0.0534 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$\begin{bmatrix} 0.0001 & 0.5651 & -0.8250 & 0.6675 \\ 0.0032 & -0.8249 & -0.5651 & 0.0900 \\ -1.0000 & -0.0027 & -0.0018 & 0.0534 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
50.05 s	$\begin{bmatrix} 0.0006 & 0.2771 & -0.9608 & 2.1352 \\ 0.0030 & -0.9608 & -0.2771 & 0.2896 \\ -1.0000 & -0.0027 & -0.0014 & 0.0534 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$\begin{bmatrix} 0.0003 & 0.2770 & -0.9609 & 2.1309 \\ 0.0027 & -0.9609 & -0.2770 & 0.2895 \\ -0.9999 & -0.0025 & -0.0010 & 0.0534 \\ 0 & 0 & 0 & 1 \end{bmatrix}$

3. Gait Analysis

3.1. Traditional Serpentine Gait

By observing the serpentine curve, Hirose [18] proposed a serpentine gait in Equation (13), which can be realized by controlling the yaw joint angle:

$$\phi_n = A \sin(\omega t + kn) + \phi_0. \quad (13)$$

In the above equation, ϕ_n is the n -th joint angle, A determines the wave amplitude of the snake robot, ω is the temporal frequency, k is the spatial frequency, n is the joint index, and ϕ_0 determines the locomotion direction of the snake robot. When $\phi_0 = 0$, the robot moves forward; when $\phi_0 \neq 0$, the robot moves forward and turns.

3.2. Omega Turn

Inspired by the motion waveform of *C. elegans*, Wang et al. [23] realized a serpentine gait by superimposing two coplanar traveling sine waves, as shown in Equation (14):

$$\begin{aligned}\phi_n &= A_f(t)\sin\left(2\pi\omega t + 2\pi k_f \frac{n}{N}\right) + A_t(t)\sin\left(2\pi\omega t + 2\pi k_t \frac{n}{N} + \psi\right) \\ &= a_f\left(\gamma_f + \sin\left(\omega t + \phi_f\right)\right)\sin\left(2\pi\omega t + 2\pi k_f \frac{n}{N}\right) \\ &\quad + a_t\left(\gamma_t + \sin\left(\omega t + \psi + \phi_t\right)\right)\sin\left(2\pi\omega t + 2\pi k_t \frac{n}{N}\right),\end{aligned}\quad (14)$$

where ϕ_n, ω, n have the same definition as in Equation (13), N is the total number of joints, $k_f, A_f(t)$ determines the spatial frequency and wave amplitude of forward motion, $k_t, A_t(t)$ determines the spatial frequency and wave amplitude of steering motion, $a_f, a_t, \gamma_f, \gamma_t, \phi_f, \phi_t, \psi$ are adjustable parameters. When $A_t(t) = 0$, the robot moves forward without turning and when $A_t(t) \neq 0$, the robot turns and moves.

3.3. Double-Sine Serpentine Gait

Omega turn introduces a large number of parameters. as shown in Equation (14), which makes it more difficult to control the turning of a snake robot. Therefore, we conducted a lot of experiments on the above adjustable parameters, and found that the main parameters that determined the gait of the snake robot were a_f, a_t, k_f, k_t . This paper simplified the omega turn parameters, and adopted a double-sine serpentine gait by controlling yaw joint angle as:

$$\phi_n = a_f \sin(\omega t) \sin(\omega t + k_f n) + a_t \sin(\omega t) \sin(\omega t + k_t n), \quad (15)$$

where $\phi_n, \omega, n, k_f, k_t, a_f, a_t$ have the same definitions as in Equations (13) and (14), the first part of Equation (15) controls forward movement of the snake robot, and the second part of Equation (15) controls the robot's steering.

Each control parameter is the key factor to determine the motion form of the snake robot. The existing literature [10] explained the influence of gait control parameters A, k on the motion form of the snake robot. Parameters a_f, k_f have the same influence, which determines the wave amplitude of the snake robot, but they cannot change the direction of the robot movement. However, the qualitative relationship between the control parameter and the robot steering motion has not been explored by scholars. Therefore, the influence of the steering parameter a_t, k_t on the motion of a snake-like robot is worth discussing, which plays a vital role in the gait control of a snake robot.

First, we kept a_t unchanged, and studied the influence of k_t on the movement of the snake robot. We fixed $a_t = 30^\circ$, select $k_t = \pm 10^\circ, \pm 20^\circ \dots, \pm 90^\circ$ in turn, then calculated the expected angle of each joint and inputted it into the snake model. The simulation results are shown in Figure 5. Experiments showed that when $|k_t|$ was small, the turning angle of the robot was not obvious and when $|k_t| = 90^\circ$, the turning angle was the largest. The effect of $k_t \in [0^\circ, 180^\circ]$ was opposite to the effect of $k_t \in [-180^\circ, 0^\circ]$ on the turning direction.

Keeping k_t constant, we studied the influence of a_t on the movement of the snake robot. We fixed $k_t = 80^\circ$, and selected $a_t = \pm 10^\circ, \pm 20^\circ \dots, \pm 90^\circ$ in sequence. The simulation results are shown in Figure 6. It was found that the larger $|a_t|$ was, the larger the turning angle was. The effect of $a_t \in [0^\circ, 180^\circ]$ was opposite to the effect of $a_t \in [-180^\circ, 0^\circ]$ on the turning direction.

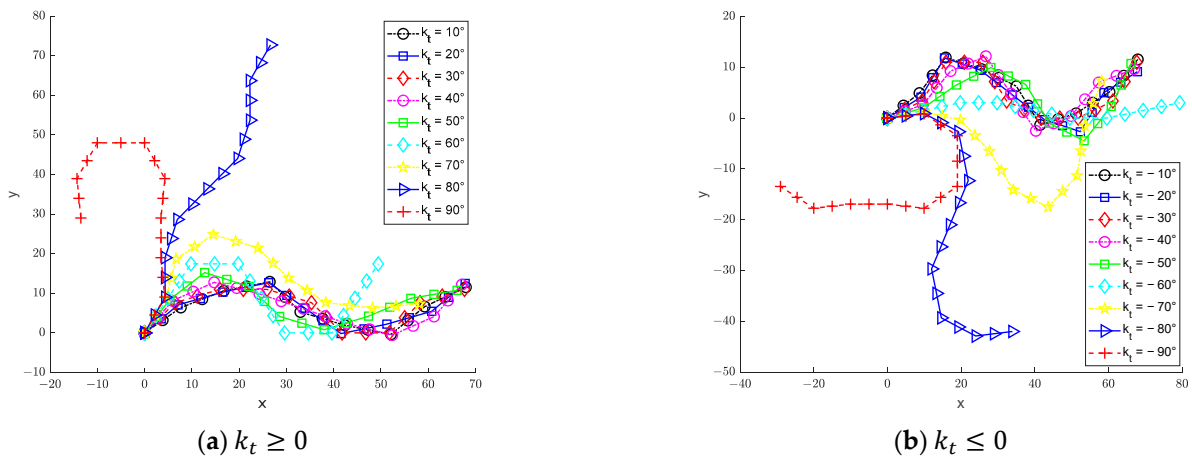


Figure 5. The influence of control parameter k_t on the movement of snake robot.

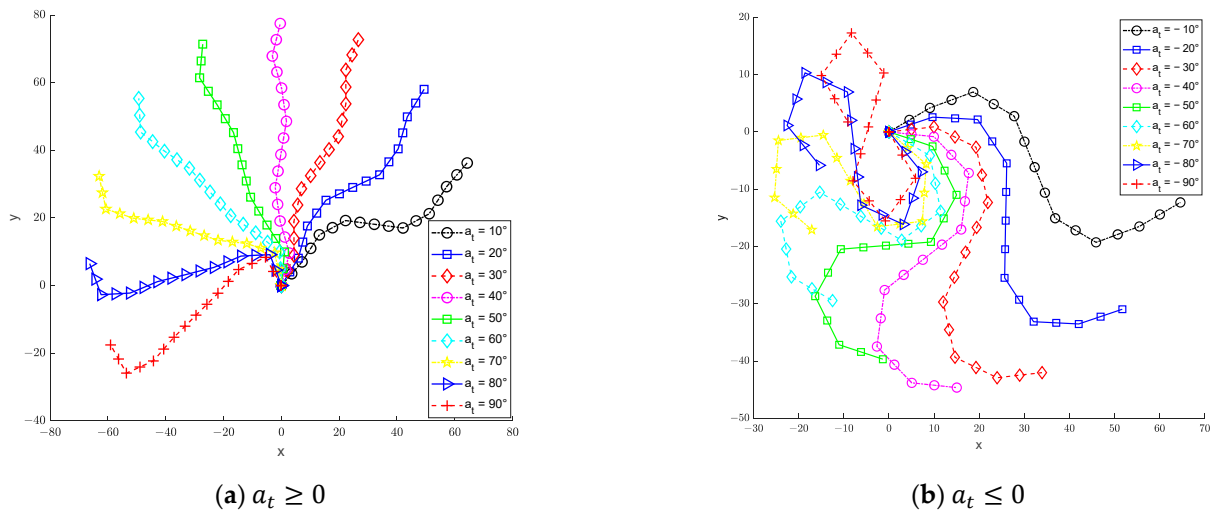


Figure 6. The influence of control parameter a_t on the movement of snake robot.

In summary, when a_t changed very little, it could obviously change the direction of the snake robot. However, when k_t changed very little, its influence on the turning motion of the robot was not obvious. It was found that a small change in the parameter a_t could make the robot reach the desired direction angle. Therefore, in the subsequent control strategy, we only considered changing the control parameter a_t to realize the turning of the snake robot.

Finally, we compared the double-sine serpentine gait with the traditional serpentine gait by controlling the snake robot to turn 60° , 90° and 120° , respectively. In each scenario, the control parameters that determined the forward direction were set as $A = a_f = 45^\circ$, $K = k_f = 55^\circ$, and temporal frequency $\omega = 40^\circ$. The results of the simulation experiments are shown in Figure 7, where the red dotted line represents the movement trajectory of the snake robot under the control of the double-sine serpentine gait, and the green solid line represents the movement trajectory under the control of the traditional serpentine gait.

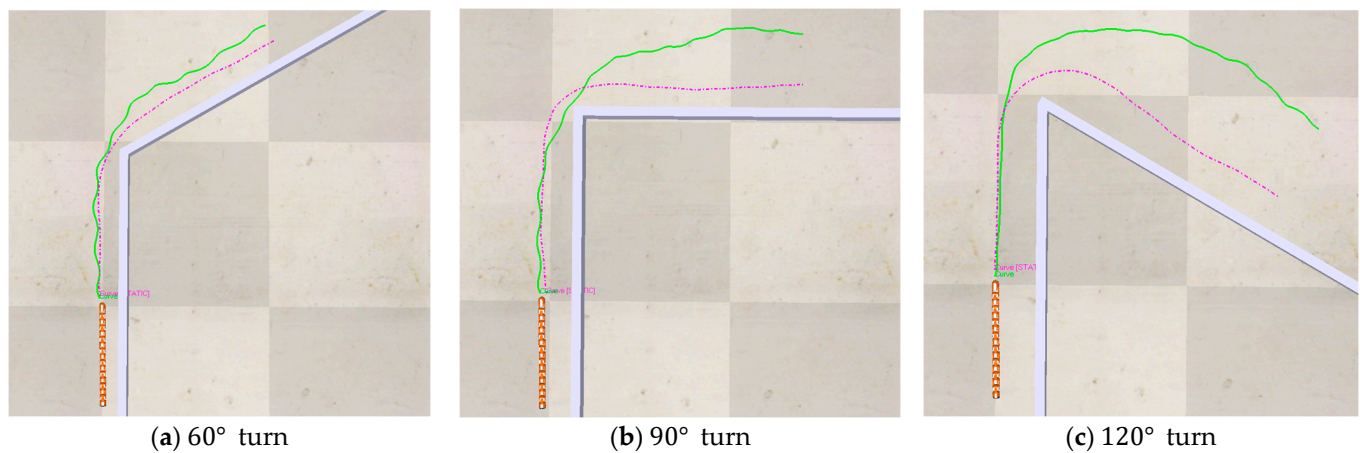


Figure 7. Comparison experiments of double-sine serpentine gait and traditional serpentine gait.

It was found that the double-sine serpentine gait had a smaller turning radius than the traditional serpentine gait, making it more suitable for turning in narrow spaces. Counting the time spent by the two gaits when turning, as shown in Table 4, it was obvious that the double-sine serpentine gait took a short time to turn, which helped in quickly tracking the expected direction.

Table 4. Comparison of turning time.

Gait	Double-Sine Serpentine Gait	Traditional Serpentine Gait
Turning Time		
60° Turning Time	11.15 s	39.55 s
90° Turning Time	16.85 s	52.95 s
120° Turning Time	22.15 s	89.70 s

4. Controller Design

In order to realize target tracking of the snake robot, this paper designed a target tracking controller by combining double-sine serpentine gait and sliding mode control theory. The reasoning behind the design involved controlling the snake robot to track the desired direction angle with a new adaptive sliding mode controller, and pushing the snake robot forward with the double-sine serpentine gait function to reach the target point. The structure of the target tracking controller for the snake robot, based on double-sine serpentine gait, is shown in Figure 8.

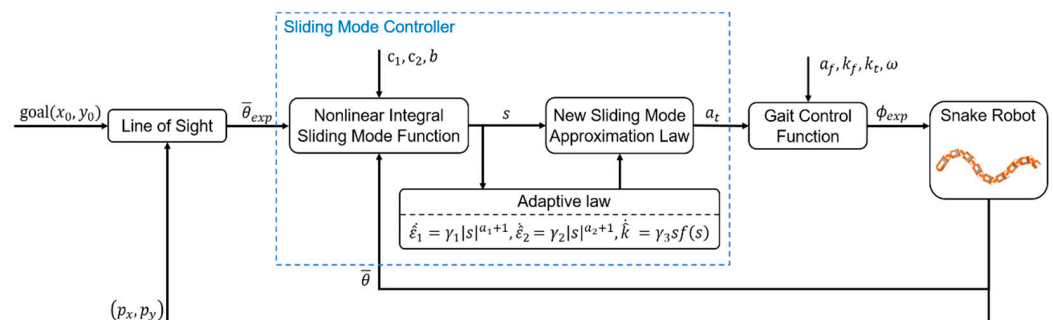


Figure 8. The structure of target tracking controller.

4.1. Control Objectives

In this paper, we assumed that there was no obstacle. As shown in Figure 9, in order to achieve the target tracking task, it was only necessary to control the snake robot to track the expected path containing the target point. We supposed that the actual position and actual

direction angle of the snake robot were set as (p_x, p_y) and $\bar{\theta}$, respectively, and the target position was set as (x_0, y_0) , and the expected direction angle recorded as $\bar{\theta}_{exp}$. Therefore, the control objective was that the position error $e_x = p_x - x_0$, $e_y = p_y - y_0$ and direction angle error $e_{\bar{\theta}} = \bar{\theta} - \bar{\theta}_{exp}$ all converged to the $e_x = 0, e_y = 0, e_{\bar{\theta}} = 0$ limit cycle.

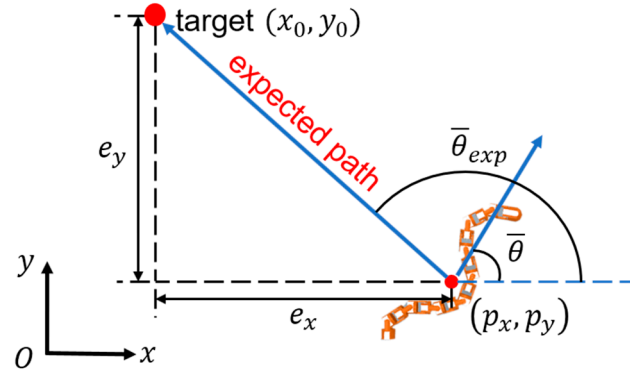


Figure 9. Schematic diagram of target tracking control of snake robot.

Since $\bar{\theta}$ determines the forward direction, by controlling direction angle $\bar{\theta}$, the snake robot can move in the forward direction where the target point is located to quickly track the target point.

In order to accurately locate the target point, we established the quantitative relationship between the position error of the snake robot and its direction angle, which is shown in Equation (16):

$$\bar{\theta}_{exp} = \begin{cases} \arctan\left(\frac{e_y}{e_x}\right) & e_x < 0, \\ \arctan\left(\frac{e_y}{e_x}\right) + \pi & e_x > 0. \end{cases} \quad (16)$$

According to the analysis in Section 3, the double-sine serpentine gait can adjust the direction angle of the snake robot in a short time, which is conducive to making the snake robot quickly reach the desired direction. From the experiment in Section 3, it was found that the control parameters a_t and k_t could realize the steering of the robot. However, when the control parameter k_t was small, the effect on the turning was not obvious. Therefore, we only considered controlling the parameter a_t to achieve steering of the snake robot. There is a quantitative relationship between heading direction $\bar{\theta}$ and the control parameter a_t according to literature [33] which is as follows:

$$\dot{\bar{\theta}} = bu(t) + d(t) = ba_t(t) + d(t), \quad (17)$$

where b is a constant related to the mechanical structure of the robot, $d(t)$ is the external disturbance of the system, and a_t is the joint controller input $u(t)$.

4.2. Sliding Mode Controller Design

Sliding mode control is a nonlinear control method that can force the system to follow the designed motion trajectory by purposefully changing the state of the system. Since the sliding mode is independent of object parameters and disturbances, SMC has the advantages of quick response speed and insensitive disturbance [40].

The sliding surface of the traditional sliding mode control method is shown in Equation (18). In order to reduce the steady state error of the control system, the integral term $\int_0^t e d\tau$ was introduced [40], as shown in Equation (19):

$$s(t) = ce(t) + \dot{e}(t). \quad (18)$$

$$s(t) = ce(t) + \dot{e}(t) + \int_0^t e d\tau. \quad (19)$$

However, when the initial error was particularly large, adopting Equation (19) would cause the deterioration of the transient response, and even cause the system to be unstable. In order to improve the above situation, this paper replaced the integral term in Equation (19) with an appropriate potential energy function and selected the sliding surface of the snake robot control system as follows:

$$s(t) = c_1 e(t) + \dot{e}(t) + c_2 \int_0^t \psi(e) d\tau. \quad (20)$$

The tracking target error is $e = \bar{\theta} - \bar{\theta}_{exp}$, and $c_1, c_2 \in \mathbb{R}^+$, $\psi(e)$ is a nonlinear function [41], as shown in Equation (21):

$$\psi(e) = \begin{cases} \alpha \sin \frac{\pi e}{2\alpha} & |e| < \alpha, \\ \alpha & e \geq \alpha, \\ -\alpha & e \leq -\alpha. \end{cases} \quad (21)$$

In the above equation, $\alpha \in \mathbb{R}^+$ is an adjustable parameter. Figure 10 shows the function curve of $\psi(e)$ when $\alpha = 3$. We found that when the absolute value of the error was $|e| < \alpha$, $|\psi(e)| > |e|$ and when the absolute value of the error was $|e| \geq \alpha$, $|\psi(e)| \leq |e|$, it could be concluded that the error would be amplified when it was small, and saturated at $\pm\alpha$ when it exceeded the design parameter α . The value $\psi(e)$ could obtain the expected error state by designing the parameter α .

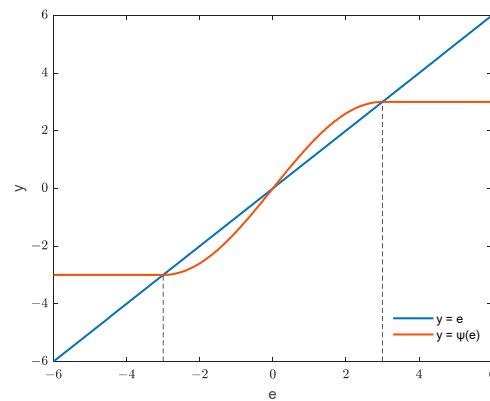


Figure 10. The function curve of $\psi(e)$ when $\alpha = 3$.

Considering that the snake robot system would be far away from the sliding surface, adopting the exponential reaching law could approach the sliding surface quickly, but the system shook violently when approaching the sliding surface. Although the jitter could be reduced by using the power reaching law, it took a long time to approach the sliding surface [42]. In order to improve the deficiencies of traditional reaching law, based on double power reaching law [43], this paper added an improved exponential approach term and designed the reaching law as follows:

$$\dot{s} = -\varepsilon_1 |s|^{a_1} \text{sgn}(s) - \varepsilon_2 |s|^{a_2} \text{sgn}(s) - k f(s), \quad (22)$$

where $\varepsilon_1, \varepsilon_2, k \in \mathbb{R}^+, a_1 > 1, 0 < a_2 < 1$. $f(s)$ is a nonlinear function, as shown in Equation (23), where $q \in \mathbb{R}^+$.

$$f(s) = \begin{cases} s & |s| \leq q, \\ \text{sgn}(s) & |s| > q. \end{cases} \quad (23)$$

From Equations (20)–(23), the sliding mode control law could be designed as:

$$\dot{\bar{\theta}} = -c_1^{-1} \left(c_2 \psi(e) + \ddot{e}(t) + \hat{\varepsilon}_1 |s|^{a_1} \text{sgn}(s) + \hat{\varepsilon}_2 |s|^{a_2} \text{sgn}(s) + \hat{k} f(s) \right) + \dot{\bar{\theta}}_{exp}(t), \quad (24)$$

where $\hat{\varepsilon}_1, \hat{\varepsilon}_2, \hat{k}$ is the estimated value of the reaching law parameters $\varepsilon_1, \varepsilon_2, k$, and the adaptive rate is shown in Equation (25).

$$\begin{cases} \dot{\hat{\varepsilon}}_1 = \gamma_1 |s|^{a_1+1}, \\ \dot{\hat{\varepsilon}}_2 = \gamma_2 |s|^{a_2+1}, \\ \dot{\hat{k}} = \gamma_3 s f(s). \end{cases} \quad (25)$$

In the above equation, $\gamma_1, \gamma_2, \gamma_3 \in \mathbb{R}^+$ are adaptive parameters.

4.3. Controller Stability Analysis

Select the Lyapunov function as follows:

$$V = \frac{1}{2}s^2 + \frac{1}{2\gamma_1}\tilde{\varepsilon}_1^2 + \frac{1}{2\gamma_2}\tilde{\varepsilon}_2^2 + \frac{1}{2\gamma_3}\tilde{k}^2, \quad (26)$$

where $\tilde{\varepsilon}_1 = \varepsilon_1 - \hat{\varepsilon}_1, \tilde{\varepsilon}_2 = \varepsilon_2 - \hat{\varepsilon}_2, \tilde{k} = k - \hat{k}$ represents the estimation error of $\varepsilon_1, \varepsilon_2, k$.

Combined with Equation (20), the above equation can be derived as follows:

$$\begin{aligned} \dot{V} &= s(c_1\dot{e} + \ddot{e} + c_2\psi(e)) - \frac{1}{\gamma_1}\tilde{\varepsilon}_1\dot{\hat{\varepsilon}}_1 - \frac{1}{\gamma_2}\tilde{\varepsilon}_2\dot{\hat{\varepsilon}}_2 - \frac{1}{\gamma_3}\tilde{k}\dot{\hat{k}} \\ &= \left(c_1\left(\dot{\bar{\theta}} - \dot{\bar{\theta}}_{exp}\right) + \ddot{e} + c_2\psi(e)\right) - \frac{1}{\gamma_1}\tilde{\varepsilon}_1\dot{\hat{\varepsilon}}_1 - \frac{1}{\gamma_2}\tilde{\varepsilon}_2\dot{\hat{\varepsilon}}_2 - \frac{1}{\gamma_3}\tilde{k}\dot{\hat{k}}. \end{aligned} \quad (27)$$

Putting Equation (24) into the above equation, we get:

$$\dot{V} = -\hat{\varepsilon}_1 |s|^{a_1+1} - \hat{\varepsilon}_2 |s|^{a_2+1} - \hat{k} s f(s) - \frac{1}{\gamma_1}\tilde{\varepsilon}_1\dot{\hat{\varepsilon}}_1 - \frac{1}{\gamma_2}\tilde{\varepsilon}_2\dot{\hat{\varepsilon}}_2 - \frac{1}{\gamma_3}\tilde{k}\dot{\hat{k}}. \quad (28)$$

Combined with the adaptive rate Equation (25), we obtain:

$$\begin{aligned} \dot{V} &= -\hat{\varepsilon}_1 |s|^{a_1+1} - \hat{\varepsilon}_2 |s|^{a_2+1} - \hat{k} s f(s) - \frac{1}{\gamma_1}(\varepsilon_1 - \hat{\varepsilon}_1)\dot{\hat{\varepsilon}}_1 - \frac{1}{\gamma_2}(\varepsilon_2 - \hat{\varepsilon}_2)\dot{\hat{\varepsilon}}_2 \\ &\quad - \frac{1}{\gamma_3}(k - \hat{k})\dot{\hat{k}} \\ &= -\hat{\varepsilon}_1 |s|^{a_1+1} - \hat{\varepsilon}_2 |s|^{a_2+1} - \hat{k} s f(s) - \frac{1}{\gamma_1}(\varepsilon_1 - \hat{\varepsilon}_1)\gamma_1 |s|^{a_1+1} \\ &\quad - \frac{1}{\gamma_2}(\varepsilon_2 - \hat{\varepsilon}_2)\gamma_2 |s|^{a_2+1} - \frac{1}{\gamma_3}(k - \hat{k})\gamma_3 s f(s) \\ &= -\varepsilon_1 |s|^{a_1+1} - \varepsilon_2 |s|^{a_2+1} - k s f(s). \end{aligned} \quad (29)$$

For Equation (29), when $|s| \leq q$, the above equation can be written as:

$$\dot{V} = -\varepsilon_1 |s|^{a_1+1} - \varepsilon_2 |s|^{a_2+1} - k s^2 \leq 0. \quad (30)$$

In the above equation, if and only if $s = 0$, then $\dot{V} = 0$.

Similarly, when $|s| > q$, Equation (29) can be written as:

$$\dot{V} = -\varepsilon_1 |s|^{a_1+1} - \varepsilon_2 |s|^{a_2+1} - k |s| \leq 0. \quad (31)$$

In the above equation, if and only if $s = 0$, then $\dot{V} = 0$.

According to LaSalle's invariance principle, when $t \rightarrow \infty, s \rightarrow 0$. If and only if $e = 0$, we have $\psi(e) = 0$. If and only if $e = 0$ and $\dot{e} = 0$, we have $s = 0$. From $\lim_{t \rightarrow \infty} s(t) = 0$, we can

obtain $\lim_{t \rightarrow \infty} e(t) = \lim_{t \rightarrow \infty} \dot{e}(t) = 0$. When $t \rightarrow \infty$, we have $\bar{\theta} - \bar{\theta}_{exp} \rightarrow 0$, and $\dot{\bar{\theta}} - \dot{\bar{\theta}}_{exp} \rightarrow 0$. Therefore, the error of the tracking control system eventually tends to zero. The controller that we designed is stable.

5. Simulation Results

In this part, we used the CoppeliaSim robot simulator software and MATLAB [39] to verify the effectiveness of the above control strategy. The initial position and the initial direction angle of the snake robot were set to (0, 0) and 90° , respectively. The target was set to a cylinder, and its position was (5, 5). The initial state of the snake robot is shown in Figure 11.

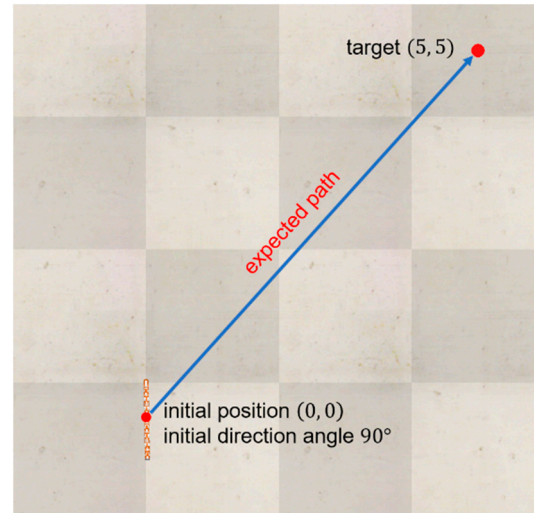


Figure 11. The initial state of the snake robot.

The control parameters in the gait control function were set to $a_f = 30^\circ$, $k_f = 60^\circ$, $k_t = 90^\circ$, $\omega = 40^\circ$, and the initial value of a_t was set to zero. The control parameters in the proposed adaptive sliding mode controller (ASMC) were set to $c_1 = 150$, $c_2 = 5$, $a_1 = 1.1$, $a_2 = 0.1$, $q = 1$, $\alpha = 0.01$, $\gamma_1 = -0.001$, $\gamma_2 = 0.01$, $\gamma_3 = 0.01$. The results of the simulation experiment are shown in Figure 12.

According to Figure 12a,b, the snake robot with the proposed control strategy could effectively reach the target. As shown in Figure 12c,d, the snake robot tracked the expected direction in about 30 s, and, finally, tracked the target in about 81 s. Note that since the direction angle of the snake robot was a sine wave affected by a serpentine gait, its direction angle error was also a sinusoidal waveform [27]. The closer the error was to zero, the better the tracking control effect would be.

In order to further explore the superiority of ASMC in the proposed control strategy, it was compared with a sliding mode controller (SMC) and a PD controller. The expressions and parameters of each controller were set as follows:

ASMC:

$\dot{\bar{\theta}} = -c_1^{-1}(c_2\psi(e) + \ddot{e}(t) + \hat{\varepsilon}_1|s|^{a_1}\text{sgns} + \hat{\varepsilon}_2|s|^{a_2}\text{sgns} + \hat{k}f(s)) + \dot{\bar{\theta}}_{exp}(t)$, where the parameter setting is the same as above.

PD controller:

$\dot{\bar{\theta}} = b^{-1}(k_p e + k_d \dot{e})$, where $b = 0.05$, $k_p = 1$, $k_d = 0.26$.

SMC:

$\dot{\bar{\theta}} = -c_1^{-1}(\varepsilon|s|^a\text{sgns} + \ddot{e} + c_2 e) + \dot{\bar{\theta}}_{exp}(t)$. We adopt the power reaching law, and the sliding surface is $s(t) = ce(t) + \dot{e}(t) + \int_0^t e d\tau$. Set parameters $c_1 = 300$, $c_2 = 1$, $\varepsilon = 100$, $a = 0.55$.

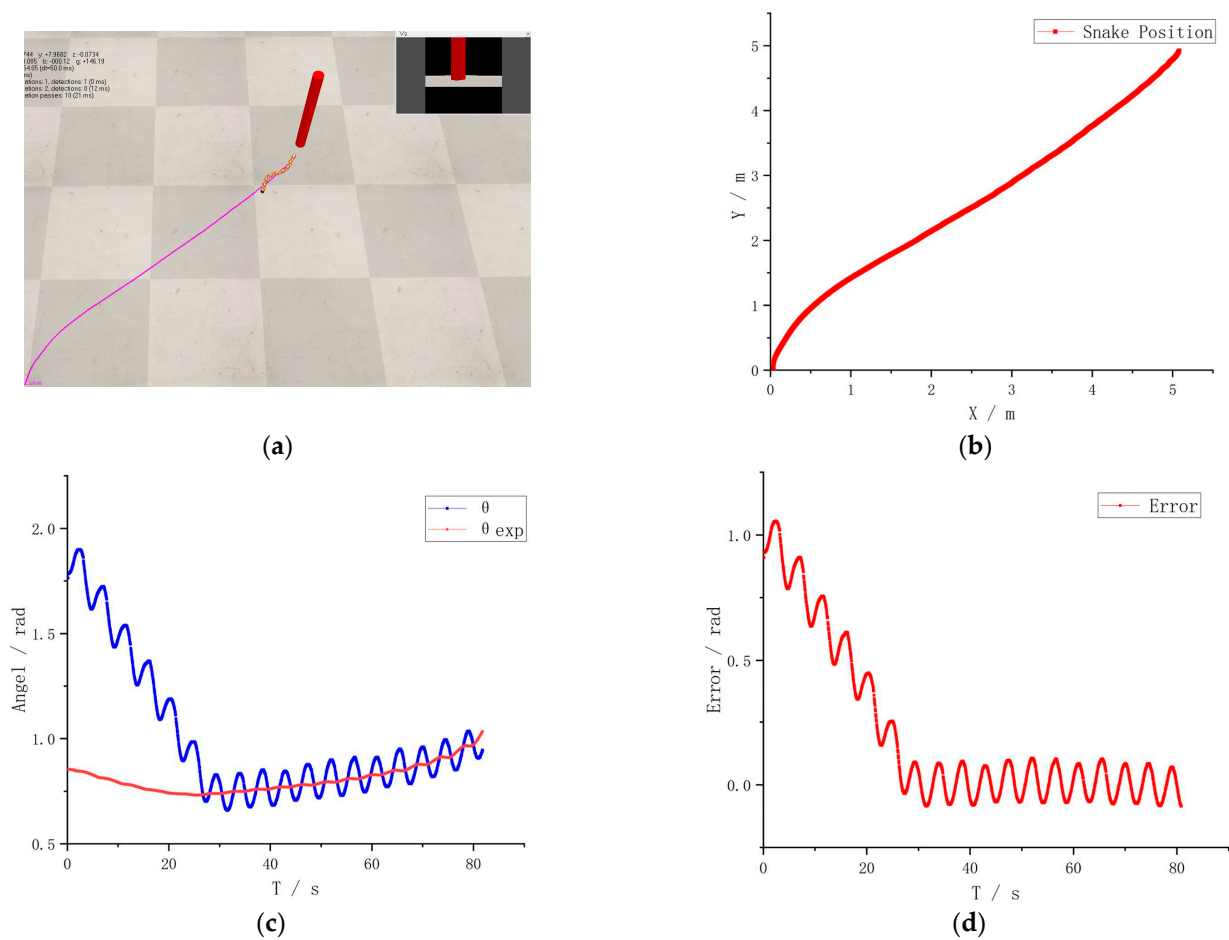


Figure 12. Simulation experiment results of target tracking control: (a) The trajectory of snake robot under tracking control; (b) Snake robot mass centric coordinates; (c) Desired direction angle and actual direction angle curve; (d) The error curve of the direction angle of snake robot.

To highlight the robustness and accuracy of the controllers in the process of tracking the desired target, we added external disturbance $d(t) = \frac{2}{\sqrt{2\pi}}e^{-2(t-5)}$ in the experiments. The error of the direction angle with the above controllers is shown in Figure 13a, and the stabilized data of error, drawn as a box plot, is shown in Figure 13b.

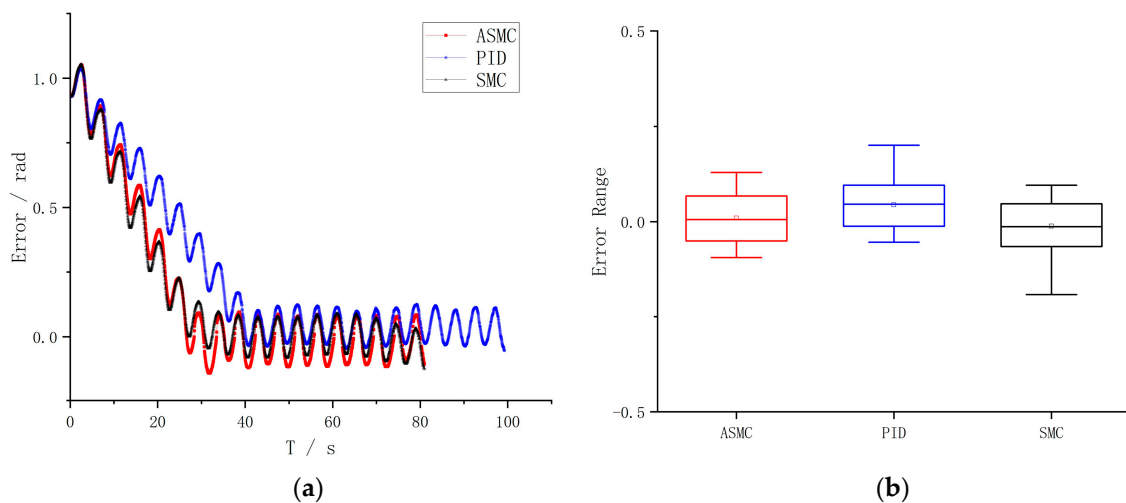


Figure 13. Comparison of control error results of each controller: (a) The error curve of the direction angle; (b) Box plot of state error.

According to Figure 13a, it can be seen that ASMC made the snake robot track the expected direction in 30 s, SMC made snake robot track the expected direction in 36 s, and PID made snake robot track the expected direction in 41 s. Compared with the PD controller and SMC, it was found that the proposed controller could quickly reduce the error and track the expected direction in a short time. Combined with Figure 13b, it could be seen that the error of ASMC was concentrated around 0.002 rad, the error of SMC was concentrated around -0.015 rad, and the error of PID was concentrated around 0.047 rad. Compared with the other two controllers, the state error of ASMC was more concentrated around 0. In brief, ASMC obviously converged faster than the other controllers in the case of external disturbance, and ASMC had higher tracking accuracy. Therefore, the tracking performance of ASMC was superior.

Finally, in order to verify the efficiency of the double-sine serpentine gait in our control strategy, we compared the result of the double-sine serpentine gait (see in Figure 12) with the result of regular gait [18] (see in Figure 14) for target tracking control. Figure 14 shows that the robot with regular gait tracked the expected direction in about 90 s, and tracked the target in about 160 s. Compared with the double-sine serpentine gait, the regular gait converged slower, and it took longer to reach the target point, which further verified that the double-sine serpentine gait in our control strategy helped in quickly tracking the expected direction.

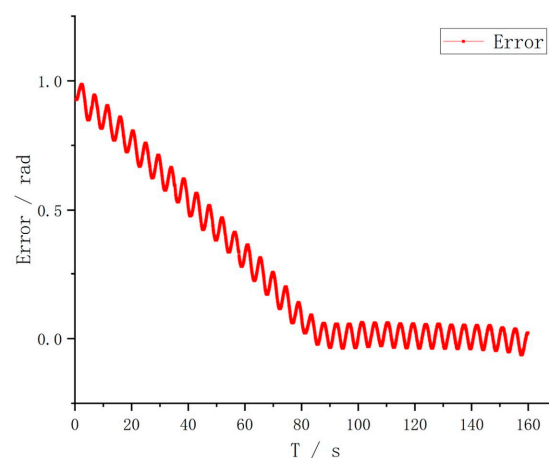


Figure 14. Simulation results of target tracking based on regular gait.

6. Conclusions

In this paper, an adaptive target tracking controller, based on the double-sine serpentine gait, was proposed. Considering traditional serpentine gait with its low movement efficiency, we established a double-sine serpentine gait, a superposition of two coplanar sine waves, which has a smaller turning radius and faster turning speed. After verification through simulation experiments, the double-sine serpentine gait was found to rapidly adjust the forward direction of the snake robot. Furthermore, an adaptive sliding mode controller was proposed in this paper to improve the robustness and convergence speed of the snake robot. Compared with a PD controller and SMC, the proposed controller enabled the snake robot to track the target point more quickly and accurately. However, the control strategy proposed in this paper was based on joint angle control which is only applicable to an ideal environment. In the future, considering robot motion with accuracy in the real environment, it is necessary to introduce joint torque control. This paper only studied target tracking control with no obstacles, so we need to further add an obstacle-avoiding strategy based to this control strategy. To improve the adaptability of the snake robot in complex environments, we will further explore the robust tracking of the snake robot on complex paths (including straight and curved paths) based on the research obtained in this paper.

Author Contributions: All authors contributed to this work. Conceptualization, Z.L. and W.W.; methodology, Z.L. and X.L.; software, Z.L. and S.H.; validation, Z.L. and W.W.; writing—original draft, Z.L.; writing—review and editing, Z.L., W.W., X.L. and S.H. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the National Natural Science Foundation of China [Grant Nos. 61573148, 61603358]; and the Science and Technology Planning Project of Guangdong Province, China [Grant Nos. 2015B010919007, 2019A050520001].

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data that support the findings of this study are available on request from the corresponding author. The data are not publicly available due to privacy and ethical restrictions.

Conflicts of Interest: The authors declare no conflict of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript, or in the decision to publish the results.

References

1. Yang, X.; Zheng, L.; Lü, D.; Wang, J.; Wang, S.; Su, H.; Wang, Z.; Ren, L. The snake-inspired robots: A review. *Assem. Autom.* **2022**, *42*, 567–583. [\[CrossRef\]](#)
2. Liu, Z.; Wang, Y.; Wang, J.; Fei, Y.; Du, Q. An obstacle-avoiding and stiffness-tunable modular bionic soft robot. *Robotica* **2020**, *40*, 2651–2665. [\[CrossRef\]](#)
3. Li, D.; Deng, H.; Pan, Z.; Xiu, Y. Collaborative obstacle avoidance algorithm of multiple bionic snake robots in fluid based on IB-LBM. *ISA Trans.* **2022**, *122*, 271–280. [\[CrossRef\]](#) [\[PubMed\]](#)
4. Chen, J.; Qiao, H. Muscle-synergies-based neuromuscular control for motion learning and generalization of a musculoskeletal system. *IEEE Trans. Syst. Man Cybern. Syst.* **2021**, *51*, 3993–4006. [\[CrossRef\]](#)
5. Bhandari, G.; Raj, R.; Pathak, P.; Yang, J. Robust control of a planar snake robot based on interval type-2 Takagi–Sugeno fuzzy control using genetic algorithm. *Eng. Appl. Artif. Intell.* **2022**, *116*, 105437. [\[CrossRef\]](#)
6. Jiang, X.; Yang, F.; Shi, S. Design and full-link trajectory tracking control of underwater snake robot with vector thrusters under strong time-varying disturbances. *Ocean. Eng.* **2022**, *266*, 113012. [\[CrossRef\]](#)
7. Mahdi, H. Dynamics and computed-muscle-force control of a planar muscle-driven snake robot. *Actuators* **2022**, *11*, 194.
8. Virgala, I.; Kelemen, M.; Prada, E.; Sukop, M.; Kot, T.; Bobovský, Z.; Varga, M.; Ferenčík, P. A snake robot for locomotion in a pipe using trapezium-like travelling wave. *Mech. Mach. Theory* **2021**, *158*, 104221. [\[CrossRef\]](#)
9. Zhu, L.; Yang, P.; Li, F.; Wang, K.; Shui, L.; Chen, X. On the snake-like lateral undulatory locomotion in terrestrial, aquatic and sand environments. *J. Mech. Phys. Solids* **2021**, *157*, 104629. [\[CrossRef\]](#)
10. Liu, X.; Lin, G.; Wei, W. Adaptive transition gait planning of snake robot based on polynomial interpolation method. *Actuators* **2022**, *11*, 222. [\[CrossRef\]](#)
11. Bae, J.; Kim, M.; Song, B.; Yang, J.; Kim, D.; Jin, M.; Yun, D. Review of the latest research on snake robots focusing on the structure, motion and control method. *Int. J. Control Autom. Syst.* **2022**, *20*, 3393–3409. [\[CrossRef\]](#)
12. Qin, G.; Wu, H.; Cheng, Y.; Pan, H.; Zhao, W.; Shi, S.; Song, Y.; Ji, A. Adaptive trajectory control of an under-actuated snake robot. *Appl. Math. Model.* **2022**, *106*, 756–769. [\[CrossRef\]](#)
13. Han, S.; Chon, S.; Kim, J.; Seo, J.; Shin, D.G.; Park, S.; Kim, J.T.; Kim, J.; Jin, M.; Cho, J. Snake Robot Gripper Module for Search and Rescue in Narrow Spaces. *IEEE Robot. Autom. Lett.* **2022**, *7*, 1667–1673. [\[CrossRef\]](#)
14. Baysal, Y.; Altas, I. Modelling and simulation of a wheel-less snake robot. In Proceedings of the 2020 7th International Conference on Electrical and Electronics Engineering (ICEEE), Antalya, Turkey, 14–16 April 2020; pp. 285–289.
15. Huang, Z.; Ma, S.; Lin, Z.; Zhu, K.; Wang, P.; Ahmed, R.; Ren, C.; Marvi, H. Impact of caudal fin geometry on the swimming performance of a snake-like robot. *Ocean. Eng.* **2022**, *245*, 110372. [\[CrossRef\]](#)
16. Gray, J. The mechanism of locomotion in snakes. *J. Exp. Biol.* **1946**, *23*, 101–120. [\[CrossRef\]](#) [\[PubMed\]](#)
17. Mori, M.; Hirose, S. Locomotion of 3D snake-like robots—shifting and rolling control of active cord mechanism ACM-R3. *J. Robot. Mechatron.* **2006**, *18*, 521–528. [\[CrossRef\]](#)
18. Hirose, S. *Biologically Inspired Robots: Snake-Like Locomotors and Manipulators*; Oxford University Press: London, UK, 1993.
19. Ye, C.; Ma, S.; Li, B.; Wang, Y. Turning and side motion of snake-like robot. In Proceedings of the IEEE International Conference on Robotics and Automation, 2004. Proceedings. ICRA '04. 2004, New Orleans, LA, USA, 26 April–1 May 2004; pp. 5075–5080.
20. Ye, C.; Ma, S.; Li, B.; Wang, Y. Study on turning and sidewise motion of a snake-like robot. *Chin. J. Mech. Eng.* **2004**, *40*, 119–123, 128. [\[CrossRef\]](#)
21. Dai, J.; Faraji, H.; Gong, C.; Hatton, R.; Goldman, D.; Choset, H. Geometric swimming on a granular surface. In Proceedings of the Robotics: Science and Systems 2016, Ann Arbor, MI, USA, 18–22 June 2016.

22. Wang, T.; Chong, B.; Deng, Y.; Fu, R.; Choset, H.; Goldman, D. Generalized omega turn gait enables agile limbless robot turning in complex environments. In Proceedings of the 2022 International Conference on Robotics and Automation (ICRA), Philadelphia, PA, USA, 23–27 May 2022; pp. 1–7.
23. Wang, T.; Chong, B.; Diaz, K.; Whitman, J.; Lu, H.; Travers, M.; Goldman, D.; Choset, H. The omega turn: A biologically-inspired turning strategy for elongated limbless robots. In Proceedings of the 2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Las Vegas, NV, USA, 24 October 2020–24 January 2021; pp. 7766–7771.
24. Liu, X.; Onal, C.; Fu, J. Reinforcement learning of a CPG-regulated locomotion controller for a soft snake robot. *arXiv* **2022**, arXiv:2207.04899.
25. Takemori, T.; Tanaka, M.; Matsuno, F. Adaptive helical rolling of a snake robot to a straight pipe with irregular cross-sectional shape. *IEEE Trans. Robot.* **2022**, 1–15. [\[CrossRef\]](#)
26. Jiang, Z.; Otto, R.; Bing, Z.; Huang, K.; Knoll, A. Target tracking control of a wheel-less snake robot based on a supervised multi-layered SNN. In Proceedings of the 2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Las Vegas, NV, USA, 24 October 2020–24 January 2021; pp. 7124–7130.
27. Liljebäck, P.; Pettersen, K.; Stavdahl, Ø.; Grasdahl, J.T. *Snake Robots Modelling, Mechatronics, and Control*; Springer Science & Business Media: Berlin/Heidelberg, Germany, 2012.
28. Liljebäck, P.; Pettersen, K. Waypoint guidance control of snake robots. In Proceedings of the 2011 IEEE International Conference on Robotics and Automation, Shanghai, China, 9–13 May 2011; pp. 937–944.
29. Kelasidi, E.; Liljebäck, P.; Pettersen, K.; Grasdahl, J. Integral line-of-sight guidance for path following control of underwater snake robots: Theory and experiments. *IEEE Trans. Robot.* **2017**, *33*, 610–628. [\[CrossRef\]](#)
30. Ariizumi, R.; Takahashi, R.; Tanaka, M.; Asai, T. Head-trajectory-tracking control of a snake robot and its robustness under actuator failure. *IEEE Trans. Control Syst. Technol.* **2019**, *27*, 2589–2597. [\[CrossRef\]](#)
31. Zhang, D.; Li, B.; Chang, J. Path following method for snake robot based on the angle symmetry adjustment. *Robot* **2019**, *41*, 788–794, 833.
32. Zhang, D. A path tracking method for the snake robot based on the path edge guidance strategy. *Robot* **2021**, *43*, 36–43.
33. Xiao, W. Design and Research of Advanced Gaits on Snake Robot. Master's Thesis, South China University of Technology, Guangzhou, China, 2021.
34. Cao, Z.; Zhang, D.; Zhou, M. Direction control and adaptive path following of 3-D snake-like robot motion. *IEEE Trans. Cybern.* **2022**, *52*, 10980–10987. [\[CrossRef\]](#)
35. Wang, C.; Peng, Y.; Li, D.; Pan, Z.; Deng, H.; Li, D.; Li, B. Turning strategy of snake-like robot based on serpenoid curve under cloud assisted smart conditions. *Clust. Comput.* **2019**, *22*, 13041–13053. [\[CrossRef\]](#)
36. Craig, J. *Introduction to Robotics: Mechanics and Control*, 4th ed.; China Machine Press: Beijing, China, 2018.
37. Wei, W.; Li, Y.; Liao, Z.; Zhang, J. Kinematics modeling of snake-like robot based on screw theory. *J. South China Univ. Technol. Nat. Sci. Ed.* **2019**, *47*, 1–8.
38. Wei, W.; Li, Y.; Gao, Y.; Xu, J. Safe climbing of snake-like robot based on screw and envelope theory. *J. South China Univ. Technol. Nat. Sci. Ed.* **2019**, *47*, 13–23.
39. Tursynbek, I.; Shintemirov, A. Modeling and Simulation of Spherical Parallel Manipulators in CoppeliaSim (V-REP) Robot Simulator Software. In Proceedings of the 2020 International Conference Nonlinearity, Information and Robotics (NIR), Innopolis, Russia, 3–6 December 2020; pp. 1–6.
40. Liu, J. *Sliding Mode Control Design and Matlab Simulation: The Basic Theory and Design Method*, 4th ed.; Tsinghua University Press: Beijing, China, 2019.
41. Li, P.; Zheng, Z. Sliding mode control approach with nonlinear integrator. *Control Theory Appl.* **2011**, *28*, 421–426.
42. Li, T. Nonlinear integral sliding mode control method based on new reaching law. *Control Eng. China* **2019**, *26*, 2031–2035.
43. Yue, C.; Qian, L.; Chen, L.; Tian, L.; Yang, H. Adaptive control for artillery chain driving powder based on double power reaching law. *J. Southeast Univ. Nat. Sci. Ed.* **2017**, *47*, 1135–1140.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.