CONTENTS

ABSTRACT **6**

1 INTRODUCTION **7**

1.1 Background and motivation 7

1.2 Related Works 8

1.2.1 System Modeling 8

1.2.2 Control Algorithms 10

1.3 Thesis Outline 12

2 ROBOT MATHEMATICAL MODELING **14**

2.1 Kinematic Models and Constraints 14

2.2 Dynamic Model and Analysis 19

2.2.1 Kinetic Energy Analysis 20

2.2.2 Potential Energy Analysis 22

2.2.3 Lagrange Equations 23

2.2.4 Nonlinear Presentatation 24

2.2.5 Linear State Space Representation 25

2.3 Summary 26

3 CONTROL ALGORITHM SUBSTANTIATION **28**

3.1 Algorithm Theoretical Basis 28

3.1.1 Linear Quadratic Regulator 28

3.1.2 Sliding Mode Control 29

3.1.3 Hierarchical Sliding Mode Control 32

3.2 Algorithm Unit Test 35

3.2.1 Linear Mass-spring System 35

3.2.2 Nonlinear Mass-spring System 36

3.2.3 Simplified Inverted Pendulum 37

3.3 Summary 39

4 CONTROLLER DESIGN AND VERIFICATION **40**

4.1 Problem Formulation 40

4.1.1 System Characteristics Analysis 40

4.1.2 Plant Implementation 42

4.2 LQR Controller Experiments 45
4.2.1 Controller Design 45
4.2.2 Linear Plant Simulation and Analysis 46
4.2.3 Nonlinear Plant Simulation and Analysis 47
4.3 Nonlinear HSMC Controller Experiments 49
4.3.1 Controller Design 49
4.3.2 Nonlinear Plant Simulation and Analysis 50
4.4 Linear HSMC Controller Experiments 52
4.4.1 Controller design 52
4.4.2 Linear Plant Simulation and Analysis 53
4.4.3 Nonlinear Plant Simulation and Analysis 54
4.5 Summary 57

5 Conclusion and Future Work **58**
5.1 Conclusion 58
5.2 Future Work 59

REFERENCES **61**

ABSTRACT

In contemporary times, numerous service robots are constructed with the basis of two-wheeled, self-balancing designs, due to their petite size and flexible movement, making them a suitable option for narrow spaces. These robots exhibit great potential in various domains like elderly care, disabled assistance, hazardous operations, education and entertainment, logistics handling, security, inspection, and more. Nevertheless, the obstacle-crossing ability, when faced with complex terrain such as rubble grounds, is insufficient and remains a challenge. This limits the overall applications of the robot. The objective of this thesis is to tackle the obstacle-crossing weakness by formulating a robust controller. The research content is as follows:

Firstly, the self-balancing robot structure, based on the double inverted pendulum, is proposed. This is a highly coupled nonlinear and underactuated system. The kinematic and dynamic models for the system are developed, and the system is linearized around the upright equilibrium position.

Secondly, theoretical studies are performed on three types of control algorithms; LQR, basic SMC, and hierarchical sliding mode control algorithms are tested through different experiments. The pros and cons of the three control algorithms are compared, providing a theoretical basis for the design of the subsequent controller.

Finally, the design for the controller and MATLAB simulation are completed. Experiments are conducted on linear and nonlinear plant, under varied disturbances. Through experimentation, it is proved that the hierarchical sliding mode controller exhibits better robustness than LQR controllers while also being efficient in suppressing external disturbances.

Keywords: Two-wheeled self-balancing robot, Double inverted pendulum, Nonlinear model, Hierarchical sliding mode control, Robustness

1 INTRODUCTION

1.1 Background and motivation

With the development of semiconductor and computer science technologies, the application scope of robots has expanded to the service field, gradually entering the public's field of vision, and no longer limited to industrial application scenarios. Currently, the society is facing deepened aging and a decline in birth rates, which have accelerated the development of service robots. Many service robot designs are based on two-wheeled self-balancing robots, which are characterized by their small size and flexible movement, making them suitable for narrow environments [1]. For example, Boston Dynamics developed the two-wheel self-balancing robot Handle specifically for logistics scenarios, which received widespread attention for its efficient autonomous handling, loading and unloading of goods, and high mobility. The prospect of service robots is promising, especially with great application potential in elderly care, assisting the disabled, hazardous operations, education and entertainment, logistics handling, security and inspection, and other fields.

Two-wheeled self-balancing robots represent a nonlinear and underactuated system, with the core issue being motion balance control. Two-wheeled self-balancing robots have different structures for different application scenarios, making it difficult to establish precise and universal kinematics and dynamics models. Therefore, the control system needs to have strong robustness and a certain degree of versatility. At present, the main developmental trends for two-wheeled self-balancing robots are improving stability, adaptability, and load-carrying capacity. However, the problem of insufficient obstacle-crossing ability has not yet found a satisfactory solution, which poses challenges when traversing complex terrains, such as rubble grounds, and thus limits the application scenarios.

This thesis aims to address the insufficient obstacle-crossing ability of two-wheeled self-balancing robots by designing a controller with strong robustness. In terms of production and society, this controller can enhance the robots' production efficiency, allowing two-wheeled self-balancing robots to replace hu-

man workers in more potentially dangerous environments, such as mines, thus improving production safety and reducing personnel injuries and accident occurrences. Simultaneously, more stable self-balancing robots can better promote the construction of smart cities. In terms of research, the model established by this thesis can serve as an experimental system to verify new control theories, and the designed controller can also be applied to other high-tech control fields, such as rotary-wing aircraft and space robots, demonstrating universality.

Service robots have strong technological radiation and economic benefits, and are a strategic emerging industry with intense technological competition [2]. It is not only an important part of future advanced manufacturing and modern service industries, but also a significant opportunity for the development of world high-tech industries. Therefore, this research has good application prospects and research value.

1.2 Related Works

Researchers often model the well-designed two-wheeled self-balancing robots as an inverted pendulum system [3]. The nonlinear model of the inverted pendulum makes control quite complex, with very high requirements for the design of the controller. As research deepens, the traditional two-wheeled self-balancing robot system gradually reveals its limitations. In order to improve the performance of two-wheeled self-balancing robots to meet the demands of different scenarios, researchers in China and abroad have carried out in-depth studies on categories such as body structure, system modeling, control methods, and sensor optimization.

1.2.1 System Modeling

Two-wheeled self-balancing robots can be categorized into fixed-center-of-mass robots and variable-center-of-mass robots. The fixed-center-of-mass robot is the basis for research, but due to its unchangeable structure, it is limited in terms of extending functionality, thus restricting the application scenarios of two-wheeled self-balancing robots. Therefore, the variable-center-of-mass two-wheeled self-balancing robot emerged. Many researchers have designed different

structures to control the center-of-mass position, such as sliders, flexible [4] or rigid swinging joints [5], or the addition of robotic arms to enhance the interaction with the environment and improve the performance of self-balancing robots. In order to design efficient control algorithms, it is necessary to establish as accurate a kinematic and dynamic model as possible for the designed robot.

Kinematic equations describe the motion characteristics of the robot, without considering the effects of forces, and the process is relatively simple. Furthermore, the equations are basically the same in different structures. The dynamic equation describes the relationship between forces and object motion. When researchers model it, they mainly use two approaches: the Euler-Lagrange approach or the Newton-Euler approach:

(1) The Lagrange method is the most commonly used method. Starting from Hamilton's principle, it mainly analyzes and models the system energy, and is suitable for multi-joint or high-degree-of-freedom systems. For example, Qian Qingwen designed a two-wheeled self-balancing robot with a freely swinging pendulum at the top [6], making the simplified model a double inverted pendulum and established a dynamic model using the Lagrange method.

(2) The Newton method applies the Newton's laws of motion to each part of the robot, fully considering the internal force terms between the rigid bodies. The resulting model is easy to design control strategies, but the mathematical derivation process is quite cumbersome. Saeed Ebrahimi used the Newton method to perform dynamic modeling analysis for a self-balancing robot with a robotic arm that he designed [7].

In past research, there were errors in modeling. Sangtae Kim and others discussed the reasons for modeling errors in past research[8], modeled by Lagrangian method and Kane's method, and used numerical simulation to illustrate the impact of error terms on attitude stability.

The dynamic equations of a two-wheeled self-balancing robot contain trigonometric functions of the pitch angles and other nonlinear terms, making it a complex nonlinear system. Some researchers linearize the system near the equilibrium point[9] and use the simplified system for subsequent controller design. However, linearization is only valid within a small angular range, and

the pitch angle of the two-wheeled self-balancing robot changes significantly during obstacle crossing. Therefore, this study focuses on the nonlinear model.

1.2.2 Control Algorithms

Traditional designs use two controllers to separately control the motion and balance and finally superpose the obtained control quantities. There are also many optional control strategies for the two controllers. So far, researchers have proposed many excellent control methods, and have designed more advanced control algorithms for improved two-wheeled self-balancing robots. Control algorithms are usually divided into two categories: linear control and nonlinear control. Robust control, adaptive control, and the popular intelligent control all belong to nonlinear control.

(1) Linear control algorithms

The main linear controls include PID control algorithm, pole assignment algorithm, LQR control algorithm, and their corresponding improved control algorithms [10–12]. The two-wheeled self-balancing robot is a typical nonlinear control system. The traditional control method is to linearize the relationship between the pitch angles and acceleration near the equilibrium point. If the robot's state is outside the linearizable range, the control effect cannot be guaranteed.

(2) Robust control

Many control systems do not have strong anti-interference capability, and in reality, many external factors also affect the stability of the control system. The accuracy of the robot's dynamic model is limited, so model errors are difficult to avoid. Enhancing the system's robustness can improve the system's anti-interference capability, including sliding mode control, H∞, H2 control [13, 14]. Yang Menglu once applied sliding mode control to a stair-climbing two-wheeled self-balancing robot, and built a physical robot test after the joint simulation of ADAMS and MATLAB, achieving a good effect [15].

(3) Adaptive control

The adaptive control algorithm can design adaptive parameters according to uncertain external conditions, thus effectively dealing with parameter changes of the robot caused by load and external conditions, and obtaining better control effect. In many cases, adaptive control methods are often combined with some algorithms, such as fuzzy adaptive, neural network adaptive [16], sliding mode adaptive, etc.

(4) Intelligent Control

Various improved two-wheeled self-balancing robots have led to an increase in complexity, making it difficult to model the robots and introducing numerous uncertain parameters. Compared with traditional algorithms, schemes represented by fuzzy logic, probabilistic reasoning, neural networks, genetic algorithms, and chaos [17, 18] are more robust, user-friendly, and low-cost. As a result, more and more research is starting to use these control schemes. For example, Laurent Vermeiren and others, in order to eliminate the interference of terrain and passenger to maintain passenger balance in the problem of self-balancing robot carrying passengers [19], established a Takagi-Sugeno fuzzy control model, adopted real-time robust PDC control, and carried out experiments through a prototype with good results. Mohammad and others designed a control strategy with three control loops to deal with uphill situations and designed a combination controller of PID and fuzzy control to control a two-wheeled self-balancing robot with extendable parts. The performance was good in MATLAB numerical simulation [20].

Based on past research results, Magdi Mahmoud and others designed balance controllers based on four different methods. The schemes include Linear Quadratic Regulator (LQR), Linear Quadratic Gaussian (LQG) control, H2 control, and H∞ control [21]. After comparing these schemes through simulation, it is believed that H∞ control works best. Some linear control algorithms are still able to provide superior control performance under specific conditions in the current popularity of various nonlinear control methods, so when designing the control algorithm for a two-wheeled self-balancing robot, it is necessary to select a suitable algorithm according to the specific situation.

In conclusion, to improve the motion balance control performance of a

two-wheeled self-balancing robot, the use of a combination control algorithm is a trend. Moreover, a suitable control algorithm should be selected for the nonlinear model.

1.3 Thesis Outline

This thesis mainly studies the control algorithms for two-wheeled self-balancing robots suitable for complex terrains. The thesis intends to build a two-wheeled self-balancing robot based on a double inverted pendulum structure, establish the corresponding kinematic and dynamic models, and design a robust controller. The performance of the controller will be verified using MATLAB simulation, aiming to solve the problem of insufficient obstacle-crossing ability of traditional two-wheeled self-balancing robots. The chapter arrangement of the thesis is as follows:

In Chapter 1, the research background and significance of the inverted pendulum are first introduced, followed by a description of the characteristics and application fields of two-wheeled self-balancing robots. The research situation of this robot by domestic and foreign scholars is summarized, including research methods such as model establishment and control strategy. Finally, the research content and chapter arrangement of this paper are reviewed.

In Chapter 2, thesis focuses on the two-wheeled self-balancing robot's structure based on the double inverted pendulum, and establishes the system coordinate system. The system kinematic characteristics are analyzed, and its kinematic model is established. The Lagrange equation method is then used to establish the system dynamic model. In the end, the obtained dynamic equations are linearized, and both linear and nonlinear system state space models are established.

In Chapter 3, the theoretical research on LQR, basic SMC, and hierarchical sliding mode algorithms is conducted respectively, and different experiments are used to test the corresponding controllers. The advantages and disadvantages of the three control algorithms are compared, providing theoretical basis for the subsequent controller design.

In Chapter 4, system controller design is performed. LQR and HSMC controllers are designed, and linear and nonlinear models are simulated using

MATLAB. Fixed and random disturbances are applied to verify and compare the performance of the controllers.

Chapter 5 summarizes the work from the aspects of system modeling and controller design, and proposes future works for the idealized assumptions and approximate treatment made in the modeling process and the shortcomings of controller design in this thesis.

2 ROBOT MATHEMATICAL MODELING

Accurate mathematical models are the prerequisite for precise system control, so establishing system models is of great importance. The modeling of the two-wheeled self-balancing robot based on the double inverted pendulum structure mainly consists of kinematic modeling and dynamic modeling. The relationship between the system's velocity, yaw angle, and left and right wheel angular velocities is described by the kinematic model, while the dynamic model represents the relationship between the system motor's output torque, left and right wheel angular velocities, and cart and pendulum tilt angles. This chapter first defines the system coordinate system and sets modeling conditions. Then, the system kinematics is analyzed and the kinematic model is established. The dynamic model is developed using the Lagrange method.

2.1 Kinematic Models and Constraints

The structure model of the two-wheeled self-balancing robot is shown in the figure 1. Compared with the traditional two-wheeled self-balancing robot, a single pendulum that can freely swing in the plane is added to the top of the cart. The rigid two-wheeled self-balancing robot consists of three main parts: the wheel system, the cart body, and the pendulum. The wheel system consists of two coaxial parallel drive wheels, each driven by its motor, with the left and right wheels subject to rolling constraints. In reality, the cart is usually used to install batteries, controllers, sensors, and other equipment, but in the thesis, it is treated as a rigid thin rod with uniform mass.

As shown in figure 1, the two-wheeled self-balancing robot has four coordinate systems: the ground coordinate system $O(X, Y, Z)$, the shaft(axle) axis coordinate system $O_s(X_s, Y_s, Z_s)$, the cart coordinate system $O_c(X_c, Y_c, Z_c)$, and the pendulum coordinate system $O_p(X_p, Y_p, Z_p)$. The ground coordinate system has its Z axis pointing in the negative direction of the gravity acceleration, and its X and Y axes are initially consistent with the axle coordinate system.

The axle coordinate system $O_s(X_s, Y_s, Z_s)$ is a Cartesian coordinate sys-

tem with the origin O_s at the center of the robot's wheel axle. The Z_s axis is perpendicular to the axle, pointing from the origin O_s in the opposite direction of the gravity acceleration. The Y_s axis is along the wheel axle, pointing from the origin towards the left wheel center, and the X_s axis is perpendicular to both Y_s and Z_s, pointing from the origin O_s forward. Similarly, the cart and pendulum coordinate systems have the same axis directions as the axle coordinate system, but their origins are located at the centers of mass of the cart and pendulum respectively.

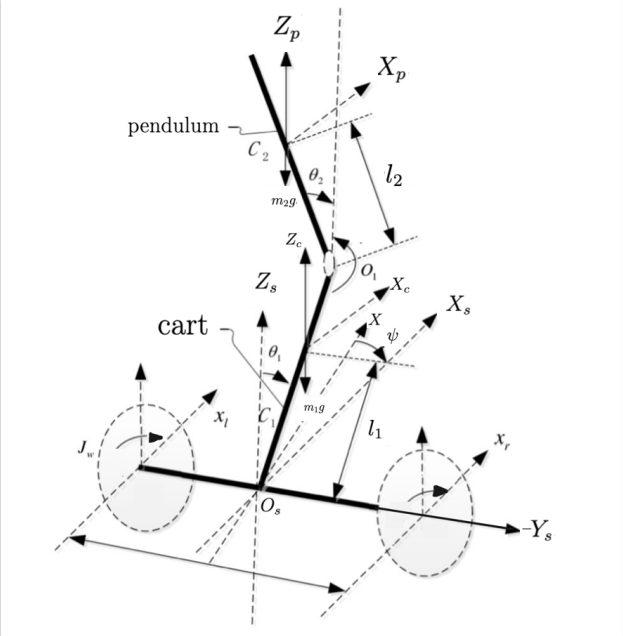


Figure 1 — Structure modeling diagram

The above two-wheeled self-balancing robot is a typical multi-rigid-body system. However, in practice, such robots are very complex and difficult to establish completely accurate mathematical models. Therefore, the following assumptions are made in this study when modeling:

(1) Both the body and pendulum of the two-wheeled self-balancing robot are considered as homogeneous rigid bodies

(2) The wheels roll on the ground without slipping or friction

(3) Friction loss in the transmission system is ignored

(4) In the pendulum motion, the friction between the pendulum and the cart is ignored

(5) Robot climbing, descending, and turning situations are not considered

(6) The specific motor servo system is not considered, only the output torque is considered

The notation and definitions of the robot parameters are shown in Tables 1 and 2.

Table 1 — Robot System Parameters

Parameter Name	**Symbol**	**Unit**
Cart Mass	m_1	kg
Pendulum Mass	m_2	kg
Wheel Mass	m_w	kg
Cart Half Length	l_1	m
Pendulum Half Length	l_2	m
Wheel Radius	R_w	kg
Axle Length	D	kg
Wheel Moment of Inertia	J_w	$kg \cdot m^2$
Cart Y-axis Moment of Inertia	J_{θ_1}	$kg \cdot m^2$
Pendulum Y-axis Moment of Inertia	J_{θ_2}	$kg \cdot m^2$

Table 2 — Robot System Variables

Parameter Name	**Symbol**	**Unit**
Cart Pitch Angle	θ_1	rad
Pendulum Pitch Angle	θ_2	rad
Cart Yaw Angle	ψ	rad
Wheel Angular Displacement	$\theta_L,\ \theta_R$	rad
Wheel angular velocity	$\dot{\theta}_L,\ \dot{\theta}_R$	rad/s
Wheel linear displacement	$x_L,\ x_R$	m
Wheel linear velocity	$\dot{x}_L,\ \dot{x}_R$	m/s
Single motor output torque	τ	$N \cdot m$
System kinetic energy	T	$kg \cdot m^2/s^2$
System potential energy	V	$kg \cdot m^2/s^2$
Lagrangian	L	$kg \cdot m^2/s^2$

The kinematic model mainly studies the motion relationship, without considering the cause of the motion or internal action, and is a mathematical expression describing the kinematic characteristics of the system. This section separately analyzes the three parts of the system and derives the planar motion constraint.

(1) Wheel velocity analysis

The relationship between the angular velocity and linear velocity $v = R\omega$ can easily derive:

$$\begin{cases} \dot{x}_L &= R_w \dot{\theta}_L \\ \dot{x}_R &= R_w \dot{\theta}_R \\ \dot{x} &= \frac{1}{2} \cdot (\dot{x}_L + \dot{x}_R) \end{cases} \tag{1}$$

(2) Cart velocity analysis

Decompose the velocity of the cart in the Z_s-O_s-X_s plane, as shown in Figure 2:

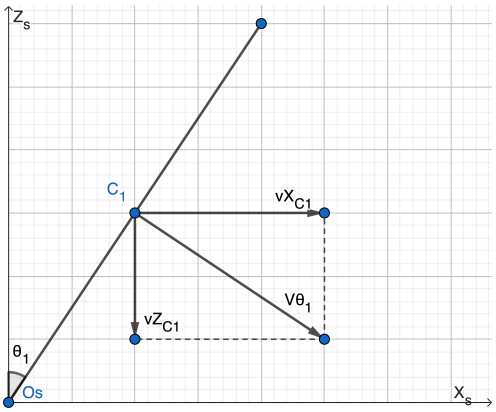


Figure 2 — Cart velocity decomposition

Here, point C_1 is the centroid of the cart, and vX_{C1} and vZ_{C1} are the components of the angular velocity $\dot{\theta}_1$ in the X_s and Z_s directions. And since steering is not considered, the analysis on the Y_s-O_s-X_s plane is omitted.

$$\mathbf{V_{C1}} = \begin{bmatrix} vX_{C1} & vY_{C1} & vZ_{C1} \end{bmatrix}^{\mathsf{T}} = \begin{bmatrix} \dot{x} + l_1 \dot{\theta}_1 \cos\theta_1 \\ l_1 \dot{\psi} \sin\theta_1 \\ -l_1 \dot{\theta}_1 \sin\theta_1 \end{bmatrix} \tag{2}$$

(3) Pendulum velocity analysis

Similar to the cart, decompose the velocity of the pendulum in the Z_s-O_s-X_s plane, as shown in Figure 3:

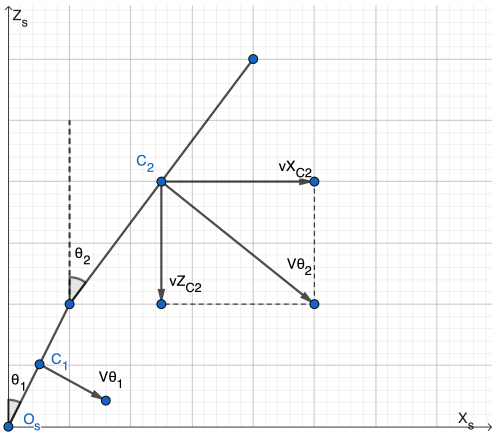


Figure 3 — Pendulum velocity decomposition

Here, point C_2 is the centroid of the pendulum, and vX_{C2} and vZ_{C2} are the components of the angular velocity $\dot{\theta}_2$ in the X_s and Z_s directions.

$$\mathbf{V_{C2}} = \begin{bmatrix} vX_{C2} & vY_{C2} & vZ_{C2} \end{bmatrix}^{\mathsf{T}} = \begin{bmatrix} \dot{x} + l_2\dot{\theta}_2\cos\theta_2 + 2l_1\dot{\theta}_1\cos\theta_1 \\ 2l_1\dot{\psi}\sin\theta_1 + l_2\dot{\psi}\sin\theta_2 \\ -l_2\dot{\theta}_2\sin\theta_2 - 2l_1\dot{\theta}_1\sin\theta_1 \end{bmatrix} \tag{3}$$

(4) Planar Motion Constraint

As shown in Fig.4, the overall robot motion in the Y-O-X plane can be analyzed, and the differential equations of yaw angle and advancing velocity can be obtained from the wheel's angular velocities:

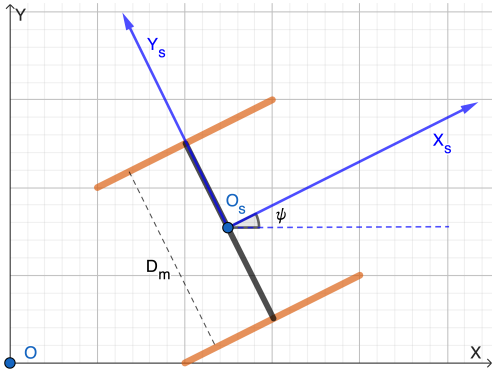


Figure 4 — Planar motion diagram

$$\begin{cases} \dot{x}_s = \frac{R_w}{2}(\dot{\theta}_L + \dot{\theta}_R) \\ \dot{\psi} = \frac{R_w}{D_m}(\dot{\theta}_R - \dot{\theta}_L) \end{cases} \tag{4}$$

The inverse relationship can be derived in matrix form:

$$\begin{bmatrix} \dot{\theta}_L \\ \dot{\theta}_R \end{bmatrix} = \frac{1}{R_w} \begin{bmatrix} -D_m/2 & 1 \\ D_m/2 & 1 \end{bmatrix} \begin{bmatrix} \dot{x}_s \\ \dot{\psi} \end{bmatrix} \tag{5}$$

At this point, the kinematic model has been completely established, and the relationship between the velocities of each subsystem, the robot's left and right wheel velocities, advancing velocities, and yaw angles has been obtained. Using the kinematic model, the actual left and right wheel velocities required for control can be derived from the target velocity, thus achieving control.

2.2 Dynamic Model and Analysis

The dynamic model describes the relationship between external input and object movement. Considering the applicability of the modeling method, the Lagrange method is chosen for modeling. Therefore, the Lagrange equations have been established by analyzing the kinetic and potential energies of the three subsystems.

The Lagrangian approach is based on Hamilton's principle and solves the energy functions of the system. This method is more suitable for rigid body systems, significantly reducing the mechanical analysis and the amount of mathematical derivation.

In the generalized coordinate system q:

$$\frac{\mathrm{d}}{\mathrm{d}t}(\frac{\partial L}{\partial \dot{q}_i}) - \frac{\partial L}{\partial q_i} = Q_i \tag{6}$$

where Q_i is the generalized force in coordinate system q_i, and L is the system's Lagrangian, defined as the difference between the system's kinetic energy T and potential energy V.

The translational kinetic energy formula of an object is:

$$T = \frac{1}{2}mv^2 \tag{7}$$

The formula for the rotational kinetic energy of an object is:

$$P = \frac{1}{2} I \omega^2 \tag{8}$$

Where I is the moment of inertia of the object being analyzed about the axis of rotation.

2.2.1 Kinetic Energy Analysis

In this subsection, we analyze the kinetic energy of the wheels, the cart, and the pendulum (T_1, T_2, and T_3) to obtain the total kinetic energy of the system.

(1) Wheel Kinetic Energy Analysis

The kinetic energy of the wheels consists of the translational kinetic energy T_L, T_R, and the rotational kinetic energy of the tire P_L, P_R. The translational kinetic energy of the left wheel can be easily obtained:

$$T_L = \frac{1}{2} m_w \dot{x}_L^2 = \frac{1}{2} m_w R_w^2 \dot{\theta}_L^2 \tag{9}$$

Similarly, the translational kinetic energy of the right wheel can be obtained:

$$T_R = \frac{1}{2} m_w \dot{x}_R^2 = \frac{1}{2} m_w R_w^2 \dot{\theta}_R^2 \tag{10}$$

The rotational kinetic energy of both wheels is:

$$\begin{cases} P_L &= \frac{1}{2} J_w \dot{\theta}_L^2 \\ P_R &= \frac{1}{2} J_w \dot{\theta}_R^2 \end{cases} \tag{11}$$

Adding the above terms, the total kinetic energy of the wheels can be obtained:

$$\begin{aligned} T_1 &= T_L + T_R + P_L + P_R \\ &= \tfrac{1}{2} m_w (\dot{x}_L^2 + \dot{x}_R^2) + \tfrac{1}{2} J_w (\dot{\theta}_L^2 + \dot{\theta}_R^2) \\ &= \tfrac{1}{2} (m_w R_w^2 + J_w)(\dot{\theta}_L^2 + \dot{\theta}_R^2) \end{aligned} \tag{12}$$

Since steering scenarios are not considered , so $\dot{\theta}_L = \dot{\theta}_R$, and formula (12) can be simplified to:

$$\begin{aligned} T_1 &= \tfrac{1}{2}(m_w R_w^2 + J_w)\tfrac{2\dot{x}^2}{R_w^2} \\ &= \tfrac{(mR^2+J_w)}{R^2}\dot{x}^2 \end{aligned} \tag{13}$$

(2) Body Kinetic Energy Analysis

The translational kinetic energy of the body can be obtained using forumla (7):

$$\begin{aligned} T_c &= \tfrac{1}{2}m_1 V_{C1}^{\intercal} V_{C1} \\ &= \tfrac{1}{2}m_1[(\dot{x} + l_1\dot{\theta}_1\cos\theta_1)^2 + (-l_1\dot{\theta}_1\sin\theta_1)^2] \\ &= \tfrac{1}{2}m_1[l_1^2\dot{\theta}_1^2 + 2l_1\dot{\theta}_1\dot{x}\cos\theta_1 + \dot{x}^2] \end{aligned} \tag{14}$$

The rotational kinetic energy of the body about the wheel axis Y_s is:

$$P_{Yc} = \frac{1}{2}J_{\theta_1}\dot{\theta}_1^2 \tag{15}$$

Adding the terms above, we can obtain the total kinetic energy of the vehicle body as:

$$\begin{aligned} T_2 &= T_c + P_{Yc} \\ &= \tfrac{1}{2}m_1[l_1^2\dot{\theta}_1^2 + 2l_1\dot{\theta}_1\dot{x}\cos\theta_1 + \dot{x}^2] + \tfrac{1}{2}J_{\theta_1}\dot{\theta}_1^2 \\ &= \tfrac{1}{2}(m_1 l_1^2 + J_{\theta_1})\dot{\theta}_1^2 + m_1 l_1\dot{x}\dot{\theta}_1\cos\theta_1 + \tfrac{1}{2}m_1\dot{x}^2 \end{aligned} \tag{16}$$

(3) Pendulum kinetic energy analysis

The translational kinetic energy of the pendulum can be derived from equation (7) as:

$$\begin{aligned} T_p &= \tfrac{1}{2}m_2 V_{C2}^{\intercal} V_{C2} \\ &= \tfrac{1}{2}m_2[(\dot{x} + l_2\dot{\theta}_2\cos\theta_2 + 2l_1\dot{\theta}_1\cos\theta_1)^2 + (-l_2\dot{\theta}_2\sin\theta_2 - 2l_1\dot{\theta}_1\sin\theta_1)^2] \\ &= \tfrac{1}{2}m_2[l_2^2\dot{\theta}_2^2 + 4l_1^2\dot{\theta}_1^2 + (2l_2\dot{\theta}_2\cos\theta_2 + 4l_1\dot{\theta}_1\cos\theta_1)\dot{x} \\ &\quad +4l_1 l_2\dot{\theta}_1\dot{\theta}_2(\sin\theta_1\sin\theta_2 + \cos\theta_1\cos\theta_2) + \dot{x}^2] \end{aligned} \tag{17}$$

According to equation (8), the rotational kinetic energy of the pendulum

around the Y-axis at the joint with the vehicle body can be obtained as:

$$P_{Yp} = \frac{1}{2} J_{\theta_2} \dot{\theta}_2^2 \tag{18}$$

Adding the terms above, we can obtain the total kinetic energy of the pendulum as:

$$\begin{aligned} T_3 &= T_p + P_{Yp} \\ &= \tfrac{1}{2} m_2 [l_2^2 \dot{\theta}_2^2 + 4 l_1^2 \dot{\theta}_1^2 + (2 l_2 \dot{\theta}_2 \cos\theta_2 + 4 l_1 \dot{\theta}_1 \cos\theta_1) \dot{x} \\ &\quad + 4 l_1 l_2 \dot{\theta}_1 \dot{\theta}_2 (\sin\theta_1 \sin\theta_2 + \cos\theta_1 \cos\theta_2) + \dot{x}^2] + \tfrac{1}{2} J_{\theta_2} \dot{\theta}_2^2 \\ &= \tfrac{1}{2} (m_2 l_2^2 + J_{\theta_2}) \dot{\theta}_2^2 + 2 m_2 l_1^2 \dot{\theta}_1^2 \\ &\quad + m_2 (l_2 \dot{\theta}_2 \cos\theta_2 + 2 l_1 \dot{\theta}_1 \cos\theta_1) \dot{x} \\ &\quad + 2 m_2 l_1 l_2 \dot{\theta}_1 \dot{\theta}_2 (\sin\theta_1 \sin\theta_2 + \cos\theta_1 \cos\theta_2) \\ &\quad + \tfrac{1}{2} m_2 \dot{x}^2 \end{aligned} \tag{19}$$

The total kinetic energy T of the system is the sum of the wheel kinetic energy T_1, the cart kinetic energy T_2, and the pendulum kinetic energy T_3, i.e.:

$$\begin{aligned} T &= T_1 + T_2 + T_3 \\ &= [\tfrac{1}{2}(m_1 + m_2) + \tfrac{(mR^2 + J_w)}{R^2}] \dot{x}^2 \\ &\quad + \tfrac{1}{2} (m_1 l_1^2 + J_{\theta_1} + 4 m_2 l_1^2) \dot{\theta}_1^2 + \tfrac{1}{2} (m_2 l_2^2 + J_{\theta_2}) \dot{\theta}_2^2 \\ &\quad + ((m_1 l_1 + 2 m_2 l_1) \dot{\theta}_1 \cos\theta_1 + m_2 l_2 \dot{\theta}_2 \cos\theta_2) \dot{x} \\ &\quad + 2 m_2 l_1 l_2 \dot{\theta}_1 \dot{\theta}_2 \cos(\theta_1 - \theta_2) \end{aligned} \tag{20}$$

2.2.2 Potential Energy Analysis

Considering the height of the vehicle axle as the zero potential energy surface, the system potential energy is:

$$\begin{aligned} V &= m_1 g l_1 \cos\theta_1 + m_2 g (2 l_1 \cos\theta_1 + l_2 \cos\theta_2) \\ &= (m_1 + 2 m_2) g l_1 \cos\theta_1 + m_2 g l_2 \cos\theta_2 \end{aligned} \tag{21}$$

2.2.3 Lagrange Equations

Combining equations (20) and (21), the Lagrange quantity of the system can be obtained:

$$\begin{aligned} L &= T - V \\ &= [\tfrac{1}{2}(m_1+m_2) + \tfrac{(mR^2+J_w)}{R^2}]\dot{x}^2 \\ &\quad +\tfrac{1}{2}(m_1 l_1^2 + J_{\theta_1} + 4m_2 l_1^2)\dot{\theta}_1^2 + \tfrac{1}{2}(m_2 l_2^2 + J_{\theta_2})\dot{\theta}_2^2 \\ &\quad +((m_1 l_1 + 2m_2 l_1)\dot{\theta}_1 \cos\theta_1 + m_2 l_2 \dot{\theta}_2 \cos\theta_2)\dot{x} \\ &\quad +2m_2 l_1 l_2 \dot{\theta}_1 \dot{\theta}_2 (\sin\theta_1 \sin\theta_2 + \cos\theta_1 \cos\theta_2) \\ &\quad -(m_1 + 2m_2) g l_1 \cos\theta_1 - m_2 g l_2 \cos\theta_2 \end{aligned} \tag{22}$$

Simplifying the expression, define the variable M_{ij} to replace the constant coefficient:

$$\begin{aligned} L &= T - V \\ &= M_{11}\dot{x}^2 \\ &\quad +M_{21}\dot{\theta}_1^2 + M_{22}\dot{\theta}_2^2 \\ &\quad +(M_{31}\dot{\theta}_1 \cos\theta_1 + M_{32}\dot{\theta}_2 \cos\theta_2)\dot{x} \\ &\quad +M_{41}\dot{\theta}_1 \dot{\theta}_2 (\sin\theta_1 \sin\theta_2 + \cos\theta_1 \cos\theta_2) \\ &\quad -M_{51}\cos\theta_1 - M_{52}\cos\theta_2 \end{aligned} \tag{23}$$

Where,

$$\begin{bmatrix} M_{11} \\ M_{21} \\ M_{22} \\ M_{31} \\ M_{32} \\ M_{41} \\ M_{51} \\ M_{52} \end{bmatrix} = \begin{bmatrix} \frac{1}{2}(m_1+m_2) + \frac{(mR^2+J_w)}{R^2} \\ \frac{1}{2}(m_1 l_1^2 + J_{\theta_1} + 4m_2 l_1^2) \\ \frac{1}{2}(m_2 l_2^2 + J_{\theta_2}) \\ m_1 l_1 + 2m_2 l_1 \\ m_2 l_2 \\ 2m_2 l_1 l_2 \\ (m_1 + 2m_2) g l_1 \\ m_2 g l_2 \end{bmatrix} \tag{24}$$

The generalized coordinates are selected as $q = [\ x\ \theta_1\ \theta_2\ ]$, with the system input being the motor torque τ and the vehicle body subjected to the reaction torque of two motors -2τ. Since there is no external generalized force acting between the vehicle body and the pendulum, the Lagrange equation of

the system is:

$$\begin{cases} \frac{d}{dt}(\frac{\partial L}{\partial \dot{x}}) - \frac{\partial L}{\partial x} &= \tau \\ \frac{d}{dt}(\frac{\partial L}{\partial \dot{\theta_1}}) - \frac{\partial L}{\partial \theta_1} &= -2\tau \\ \frac{d}{dt}(\frac{\partial L}{\partial \dot{\theta_2}}) - \frac{\partial L}{\partial \theta_2} &= 0 \end{cases} \tag{25}$$

Expanding the equations yields the system's dynamic equations:

$$\begin{cases} \begin{aligned} 2M_{11}\ddot{x} - M_{31}\dot{\theta}_1^2 \sin\theta_1 - M_{32}\dot{\theta}_2^2 \sin\theta_2 \\ +M_{31}\ddot{\theta}_1 \cos\theta_1 + M_{32}\ddot{\theta}_2 \cos\theta_2 \end{aligned} &= \tau \\ \begin{aligned} 2M_{21}\ddot{\theta}_1 - M_{51}\sin\theta_1 + M_{31}\ddot{x}\cos\theta_1 \\ -M_{41}\dot{\theta}_2^2 \cos\theta_1 \sin\theta_2 + M_{41}\dot{\theta}_2^2 \cos\theta_2 \sin\theta_1 \\ +M_{41}\ddot{\theta}_2 \cos\theta_1 \cos\theta_2 + M_{41}\ddot{\theta}_2 \sin\theta_1 \sin\theta_2 \end{aligned} &= -2\tau \\ \begin{aligned} 2M_{22}\ddot{\theta}_2 - M_{52}\sin\theta_2 + M_{32}\ddot{x}\cos\theta_2 \\ +M_{41}\dot{\theta}_1^2 \cos\theta_1 \sin\theta_2 - M_{41}\dot{\theta}_1^2 \cos\theta_2 \sin\theta_1 \\ +M_{41}\ddot{\theta}_1 \cos\theta_1 \cos\theta_2 + M_{41}\ddot{\theta}_1 \sin\theta_1 \sin\theta_2 \end{aligned} &= 0 \end{cases} \tag{26}$$

2.2.4 Nonlinear Presentatation

To solve the corresponding state variable differential equations, we transform the formula (26) into matrix form:

$$M_n \cdot \begin{bmatrix} \ddot{x} \\ \ddot{\theta}_1 \\ \ddot{\theta}_2 \end{bmatrix} + M_{naux1} \cdot \begin{bmatrix} \dot{x} \\ \dot{\theta}_1 \\ \dot{\theta}_2 \end{bmatrix} + M_{naux2} = \begin{bmatrix} \tau \\ -2\tau \\ 0 \end{bmatrix} \tag{27}$$

where:

$$M_n = \begin{bmatrix} 2M_{11} & M_{31}\cos\theta_1 & M_{32}\cos\theta_2 \\ M_{31}\cos\theta_1 & 2M_{21} & M_{41}\cdot\cos(\theta_1 - \theta_2) \\ M_{32}\cos\theta_2 & M_{41}\cdot\cos(\theta_1 - \theta_2) & 2M_{22} \end{bmatrix} \tag{28}$$

$$M_{naux1} = \begin{bmatrix} 0 & -M_{31}\dot{\theta}_1 \sin\theta_1 & -M_{32}\dot{\theta}_2 \sin\theta_2 \\ 0 & 0 & -M_{41}\dot{\theta}_2 \sin(\theta_1 - \theta_2) \\ 0 & M_{41}\dot{\theta}_1 \sin(\theta_1 - \theta_2) & 0 \end{bmatrix} \tag{29}$$

$$M_{naux2} = \begin{bmatrix} 0 \\ -M_{51}\sin\theta_1 \\ -M_{52}\sin\theta_2 \end{bmatrix} \tag{30}$$

Assuming the inverse of matrix M_n exists, we can obtain:

$$\begin{bmatrix} \ddot{x} \\ \ddot{\theta}_1 \\ \ddot{\theta}_2 \end{bmatrix} = f_q(q) + b_q(q)\tau \tag{31}$$

where,

$$f_q(q) = M_n^{-1} \cdot (-M_{naux1}) \cdot \begin{bmatrix} \dot{x} \\ \dot{\theta}_1 \\ \dot{\theta}_2 \end{bmatrix} + M_n^{-1} \cdot (-M_{naux2}) \tag{32}$$

$$b_q(q) = M_n^{-1} \begin{bmatrix} 1 \\ -2 \\ 0 \end{bmatrix} \tag{33}$$

2.2.5 Linear State Space Representation

As the robot is under normal working conditions, both pitch angles of cart and pendulum are near the vertical position with $\theta_1 = 0, \theta_2 = 0$. During normal conditions, the angle between the cart and the pendulum is small. Although the controllable angle range of nonlinear control is larger, linearizing the equation in the neighborhood of the equilibrium point $[x, 0, 0]$ will not affect the overall system performance.

When the θ_1 and θ_2 vary within a small range near the equilibrium point, the linearization error of the system is small. Based on $\lim_{\theta\to 0}\cos\theta = 1$, $\lim_{\theta\to 0}\sin\theta = \theta$ and ignoring second-order small quantities, equation (26) can be simplified to:

$$\begin{cases} 2M_{11}\ddot{x} + M_{31}\ddot{\theta}_1 + M_{32}\ddot{\theta}_2 = \tau \\ M_{31}\ddot{x} + 2M_{21}\ddot{\theta}_1 + M_{41}\ddot{\theta}_2 - M_{51}\theta_1 = -2\tau \\ M_{32}\ddot{x} + M_{41}\ddot{\theta}_1 + 2M_{22}\ddot{\theta}_2 - M_{52}\theta_2 = 0 \end{cases} \tag{34}$$

Converting the equation set to the matrix form:

$$M \cdot \begin{bmatrix} \ddot{x} \\ \ddot{\theta}_1 \\ \ddot{\theta}_2 \end{bmatrix} = -M_{aux} \cdot \begin{bmatrix} x \\ \theta_1 \\ \theta_2 \end{bmatrix} + \begin{bmatrix} \tau \\ -2\tau \\ 0 \end{bmatrix} \tag{35}$$

where $M = \begin{bmatrix} 2M_{11} & M_{31} & M_{32} \\ M_{31} & 2M_{21} & M_{41} \\ M_{32} & M_{41} & 2M_{22} \end{bmatrix}$, $M_{aux} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & -M_{51} & 0 \\ 0 & 0 & -M_{52} \end{bmatrix}$

Assuming the inverse of the matrix M exists, we can obtain:

$$\begin{bmatrix} \ddot{x} \\ \ddot{\theta}_1 \\ \ddot{\theta}_2 \end{bmatrix} = M^{-1} \cdot (-M_{aux}) \cdot \begin{bmatrix} x \\ \theta_1 \\ \theta_2 \end{bmatrix} + M^{-1} \cdot \begin{bmatrix} \tau \\ -2\tau \\ 0 \end{bmatrix} \tag{36}$$

Let $M_f = M^{-1} \cdot (-M_{aux})$, $M_b = M^{-1} \cdot [\ 1 \ \ -2 \ \ 0\ ]^\intercal$, we can obtain the state-space expression:

$$\begin{bmatrix} \dot{x} \\ \dot{\theta}_1 \\ \dot{\theta}_2 \\ \ddot{x} \\ \ddot{\theta}_1 \\ \ddot{\theta}_2 \end{bmatrix} = \begin{bmatrix} 0_{3\times3} & I_{3\times3} \\ M_f & 0_{3\times3} \end{bmatrix} \cdot \begin{bmatrix} x \\ \theta_1 \\ \theta_2 \\ \dot{x} \\ \dot{\theta}_1 \\ \dot{\theta}_2 \end{bmatrix} + \begin{bmatrix} 0_{3\times1} \\ M_b \end{bmatrix} \cdot \tau \tag{37}$$

2.3 Summary

In this chapter, the proposed two-wheeled self-balancing robot is set as a multi-rigid body system, and kinematic analysis is conducted after establishing the coordinates. The Lagrangian method is chosen for dynamic modeling. From the nonlinear model, it can be seen that the motion state variables of the two-wheeled self-balancing robot, such as the cart and pendulum pitch angles and the angular velocity of the two wheels, are mutually coupled. The high degree of coupling greatly increases the difficulty in designing the subsequent control system. However, considering the small swing angles of the system

when balanced, linearization is performed near the balance point to establish a linear state-space model.

3 CONTROL ALGORITHM SUBSTANTIATION

An appropriate control algorithm must be selected for the model obtained to achieve the required control effect. For the linear model obtained, it is planned to use the LQR algorithm for control, and the HSMC method is planned to control the nonlinear model. This chapter briefly introduces the basic concepts of these algorithms and builds a mass-spring model and a single inverted pendulum model to test the corresponding control algorithms.

3.1 Algorithm Theoretical Basis

This subsection mainly presents the theoretical background for LQR, SMC and HSMC, and why certain strategies are selected.

3.1.1 Linear Quadratic Regulator

Linear Quadratic (LQ) control is a linear state feedback control method and is the most mature and systematic method in optimal control theory. The Linear Quadratic Regulator (LQR) is relatively simple to control compared to other regulators, and its control effect is effective and easy to implement. It also has a certain degree of robustness for model uncertainties and disturbances within a certain range, so it is widely used.

The LQR method is based on minimizing the quadratic cost function of system states and control inputs, which is constructed by weighting matrices of system states and control inputs. By solving the minimization problem of the cost function, the optimal control input of the system can be obtained, making the system state converge to a stable point or track a given reference trajectory.

The optimal feedback control rate $u = -Kx$ is sought to minimize the control index J, usually:

$$J = \frac{1}{2}\int_{t_0}^{\infty} [x^{\intercal}Qx + u^{\intercal}Ru]dt \tag{38}$$

Where Q, R are the weight matrices of the system state variable and the control variable, respectively, and $Q > 0, R > 0$.

The optimal feedback control law is:

$$u = -Kx = -R^{-1}B^{\intercal}Px \tag{39}$$

Where, P is the solution of the Riccati equation:

$$A^{\intercal}P + PA - PBR^{-1}B^{\intercal}P + Q = 0 \tag{40}$$

The feedback gain K is uniquely determined by the weight matrices Q and R, so the choice of Q and R is particularly important. Q represents the emphasis on state variables, and R represents the demand for control variables. The larger the coefficient, the more important the corresponding variable in the performance function, and the higher the constraint requirement for that state.

3.1.2 Sliding Mode Control

Sliding Mode Control (SMC) [22] is a nonlinear control method with strong robustness, capable of resisting external disturbances and model errors to a certain extent. It has good stability and can respond quickly to changes in system states, making it very useful for control systems requiring rapid response. Compared to the LQR control algorithm, sliding mode control has a better tracking ability for dynamic targets.

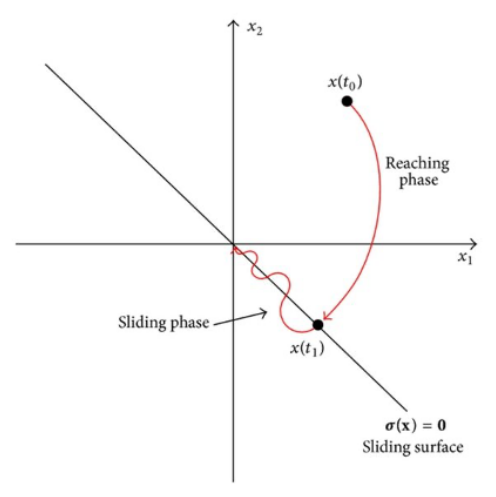


Figure 5 — Phase portarit example

The core idea of sliding mode control is to divide the system's state space

into two parts, one is the sliding surface, and the other is the complement of the sliding surface. The control of system states is realized on the sliding surface. The sliding surface is a hyperplane that can be considered as a springboard. When the system state enters the sliding surface, the sliding mode controller[23] generates a strong control force, pushing the system state towards the target state, thus achieving the control goal. When the system state deviates from the sliding surface, the sliding mode controller applies a sufficiently large control force to forcibly pull the system state back onto the sliding surface, like figure 5.

For a second-order system:

$$\begin{cases} x_1 &= x \\ \dot{x}_1 &= x_2 \\ \dot{x}_2 &= f_1(x) + b_1(x)u + d_1 \end{cases} \tag{41}$$

Here, $f_1(x), b_1(x)$ are polynomial functions of state x, u is the control input, while assuming $b_1(x) \neq 0$ and d_1 is a bounded external disturbance. At the same time, we define the error e:

$$e = x_1 - x_d$$

where x_d is the target state value. In this thesis, only the case where x_d is a constant is considered, but when it is a function of time, the mathematical proof are similar. We define the sliding surface as:

$$s = ce + \dot{e},\ c > 0 \tag{42}$$

The goal of the control is to make the system state move along the sliding surface, that is:

$$\begin{cases} s \to 0 \\ \dot{s} \to 0 \end{cases} \Rightarrow c\dot{e} + \ddot{e} = 0 \tag{43}$$

The control input is divided into two parts: the equivalent control u_{eq} of the system state on the sliding surface and the switching control u_{sw} to bring it back when it deviates from the sliding surface:

$$u = u_{eq} + u_{sw} \tag{44}$$

The equivalent control u_{eq} can be derived from Equation (43):

$$c \cdot x_2 + f_1(x) + b_1(x) \cdot u + d_1 = 0$$

$$u_{eq} = -\frac{f_1(x) + cx_2 + d_1}{b_1(x)} \tag{45}$$

The switching control u_{sw} can be set through the analysis of the Lyapunov function. Select the positive definite Lyapunov function, and when its derivative $\dot{V}$ is negative definite, the system trajectory will converge with the sliding surface to the origin:

$$\begin{cases} V(t) &= \frac{1}{2}s^2 \\ \dot{V}(t) &= s \cdot \dot{s} \leq 0 \end{cases} \tag{46}$$

Substituting formula (45) into formula (46), we get:

$$\begin{aligned} \dot{V} &= s(cx_2 + \dot{x}_2) \\ &= s(cx_2 + f_1 + b_1(u_{eq} + u_{sw}) + d_1) \\ &= s(cx_2 + f_1 + b_1u_{eq} + b_1u_{sw} + d_1) \\ &= s(b_1u_{sw} + d_1) \end{aligned} \tag{47}$$

According to the exponential reaching law, let:

$$b_1u_{sw} = -\eta\ sgn(s) - k \cdot s \tag{48}$$

where $\eta > 0,\ k > 0$. Substituting into the original equation, we get:

$$\begin{aligned} \dot{V} &= s(-\eta sgn(s) - ks + d_1) \\ &= -\eta|s| - ks^2 + sd_1 \\ &\leq -\eta|s| - ks^2 + |s| \cdot |d_1| \end{aligned}$$

Since d_1 is bounded, we define $|d_1| \leq d_M$, we can get:

$$\begin{aligned} \dot{V} &\leq -\eta|s| - ks^2 + |s| \cdot d_M \\ &\leq -ks^2 - |s| \cdot (\eta - d_M) \end{aligned} \tag{49}$$

It can be easily seen that when $\eta > d_M$, $\dot{V}$ is negative definite:

$$\dot{V} \leq 0$$

Thus, the system is stable on the sliding surface.

3.1.3 Hierarchical Sliding Mode Control

The previous section briefly introduced the basic principle of sliding mode control, but it was limited to controlling a single state variable. In this section, we will introduce hierarchical sliding mode control to control multiple state variables, that is, HSMC[24, 25]. HSMC is a hierarchical sliding mode control method, the basic idea of which is to divide the system into multiple subsystems, each subsystem having its own sliding surface, and the subsystems exchanging information through sliding surfaces to achieve system control.

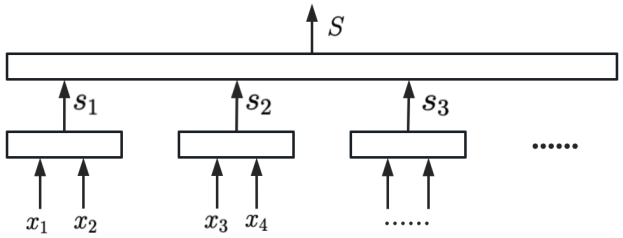


Figure 6 — Sliding surface structure of HSMC

For a single-input multi-output (SIMO) nonlinear coupled system, it can be expressed in the following form:

$$\begin{cases} \dot{x}_{2i-1} = x_{2i} \\ \dot{x}_{2i} = f_i(x) + b_i(x)u \end{cases}, \ i = 1, 2, \ldots n \tag{50}$$

For each state variable, we set the corresponding first-layer sliding surface:

$$s_i = c_i \cdot e_i + \dot{e}_i \tag{51}$$

Accordingly, we can obtain n equivalent controls:

$$u_{eq_i} = -\frac{f_i(x) + c_i \cdot x_{2i}}{b_i(x)} \tag{52}$$

The selected input is:

$$u = \sum_{i=1}^{n} u_{eq_i} + u_{sw} \tag{53}$$

As show in figure 6, the second-layer sliding surface is defined as:

$$\begin{aligned} S &= w_1 s_1 + w_2 s_2 + w_3 s_3 + \cdots + w_n s_n \\ &= \textstyle\sum_{i=1}^{n} w_i \cdot s_i \end{aligned} \tag{54}$$

Choose the Lyapunov function as:

$$V = \frac{1}{2} S^2 \tag{55}$$

Taking the time derivative of the Lyapunov function, analyzing the positive and negative properties of $\dot{V}$, first derive the case where $n = 3$.

$$\begin{aligned} \dot{V} &= S\dot{S} \\ &= S(\alpha \dot{s}_1 + \beta \dot{s}_2 + \gamma \dot{s}_3) \\ &= S[\alpha(c_1 x_2 + \dot{x}_2) + \beta(c_2 x_4 + \dot{x}_4) + \gamma(c_3 x_6 + \dot{x}_6)] \\ &= S[\alpha(c_1 x_2 + f_1 + b_1(u_{eq1} + u_{eq2} + u_{eq3} + u_{sw}) + d_1) \\ &\quad +\beta(c_2 x_4 + f_2 + b_2(u_{eq1} + u_{eq2} + u_{eq3} + u_{sw}) + d_2) \\ &\quad +\gamma(c_3 x_6 + f_3 + b_3(u_{eq1} + u_{eq2} + u_{eq3} + u_{sw}) + d_3)] \\ &= S[\alpha b_1(u_{eq2} + u_{eq3} + u_{sw}) + \beta b_2(u_{eq1} + u_{eq3} + u_{sw}) \\ &\quad +\gamma b_3(u_{eq1} + u_{eq2} + u_{sw}) + (\alpha d_1 + \beta d_2 + \gamma d_3)] \\ &= S[(\alpha b_1 + \beta b_2 + \gamma b_3) u_{sw} + (\alpha d_1 + \beta d_2 + \gamma d_3) \\ &\quad +[(\beta b_2 + \gamma b_3) u_{eq1} + (\alpha b_1 + \gamma b_3) u_{eq2} + (\alpha b_1 + \beta b_2) u_{eq3}]] \end{aligned} \tag{56}$$

Generalizing to the universal form:

$$\begin{aligned} \dot{V} &= S\dot{S} \\ &= S(\textstyle\sum_{i=1}^{n} w_i \cdot \dot{s}_i) \\ &= S[\textstyle\sum_{i=1}^{n} w_i \cdot (c_i \dot{e}_i + \ddot{e}_i)] \\ &= S[\textstyle\sum_{i=1}^{n} w_i \cdot (c_i \cdot x_{2i} + \dot{x}_{2i})] \\ &= S[\textstyle\sum_{i=1}^{n} w_i (c_i x_{2i} + f_i + b_i(\sum_{j=1}^{n} u_{eq_j} + u_{sw}) + d_i)] \\ &= S[(\textstyle\sum_{i=1}^{n} w_i b_i (u - u_{eqi})) + \sum_{i=1}^{n} w_i d_i] \\ &= S[(\textstyle\sum_{i=1}^{n} ((\sum_{j=1}^{n} w_j b_j) - w_i b_i) u_{eq_i}) \\ &\quad +(\textstyle\sum_{i=1}^{n} w_i b_i) u_{sw} + (\sum_{i=1}^{n} w_i d_i)] \end{aligned} \tag{57}$$

In a similar manner as in the previous section, let $\dot{S} - (\sum_{i=1}^{n} w_i d_i) = -\eta sgn(S) - kS$:

$$(\sum_{i=1}^{n}((\sum_{j=1}^{n} w_j b_j) - w_i b_i)u_{eq_i}) + (\sum_{i=1}^{n} w_i b_i)u_{sw} \;=\; -\eta \cdot sgn(S) - kS$$

It can be inferred that the swithcing control u_{sw} is as follows:

$$u_{sw} \;=\; -(\sum_{i=1}^{n} w_i b_i)^{-1}[(\sum_{i=1}^{n}((\sum_{j=1}^{n} w_j b_j) - w_i b_i)u_{eq_i}) + \eta \cdot sgn(S) + kS] \tag{58}$$

Substitute into the original equation:

$$\begin{aligned} \dot{V} &= S[-\eta \cdot sgn(S) - kS + (\sum_{i=1}^{n} w_i d_i)] \\ &= -\eta|S| - kS^2 + S(\sum_{i=1}^{n} w_i d_i) \\ &\leq -\eta|S| - kS^2 + |S| \cdot |\sum_{i=1}^{n} w_i d_i| \end{aligned} \tag{59}$$

Since d_i are all bounded, define $|\sum_{i=1}^{n} w_i d_i| \leq D_M$, when $\eta > D_M$

$$\begin{aligned} \dot{V} &\leq -\eta|S| - kS^2 + |S| \cdot |\sum_{i=1}^{n} w_i d_i| \\ &\leq -kS^2 - |S| \cdot (\eta - |\sum_{i=1}^{n} w_i d_i|) \\ &\leq -kS^2 - (\eta - D_M)|S| \leq 0 \end{aligned} \tag{60}$$

Thus, the system is Lyapunov stable on the second-layer sliding surface. The final control law is:

$$\begin{aligned} u &= u_{eq1} + u_{eq2} + u_{eq3} + \cdots + u_{sw} \\ &= (\sum_{i=1}^{n} w_i b_i)^{-1}[\sum_{i=1}^{n} w_i b_i u_{eq_i} - \eta \cdot sgn(S) - kS] \end{aligned} \tag{61}$$

Now, we have proved the asymptotic stability of the second-layer sliding surface S. Next, we will prove the stability of all first-layer sliding surfaces s_i when the control law is given by formula (61).

We have:

$$\lim_{t \to \infty} S = 0 \tag{62}$$

Proof by contradiction, suppose s_i is not stable, that is:

$$\lim_{t \to \infty} s_i \neq 0 \tag{63}$$

Calculate the limit of the left and right sides of equation (54) as follows:

$$\begin{aligned} \lim_{t\to\infty} S &= \lim_{t\to\infty} \sum_{i=1}^{n} w_i \cdot s_i \\ &= \sum_{i=1}^{n} \lim_{t\to\infty} w_i \cdot s_i \\ &= \sum_{i=1}^{n} w_i \cdot (\lim_{t\to\infty} s_i) \neq 0 \end{aligned} \tag{64}$$

Equation (64) contradicts the previously obtained equation (62), so the assumption does not hold and s_i is asymptotically stable.

3.2 Algorithm Unit Test

To ensure the mentioned strategies are capable of controlling the robot system studied in the thesis, different problem are formulated in this section to test the performance of the control algorithms.

3.2.1 Linear Mass-spring System

To illustrate the application of LQR algorithm, a classic mass-spring system is used as an example, and the control target is set as an constant in the experiment.

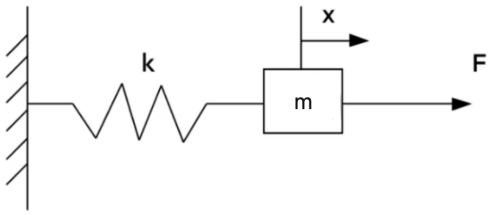


Figure 7 — Mass-Spring System

Let the input be $F = u$, and the spring be linear with $F_s = -k_s x$. By applying Newton's second law, the dynamics of the system are given by:

$$\begin{cases} x_1 &= x \\ \dot{x}_1 &= x_2 \\ \dot{x}_2 &= -\frac{k_s}{m} x_1 + \frac{1}{m} u \end{cases}$$

Transforming this into a state-space representation gives:

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -\frac{k}{m} & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{1}{m} \end{bmatrix} u$$

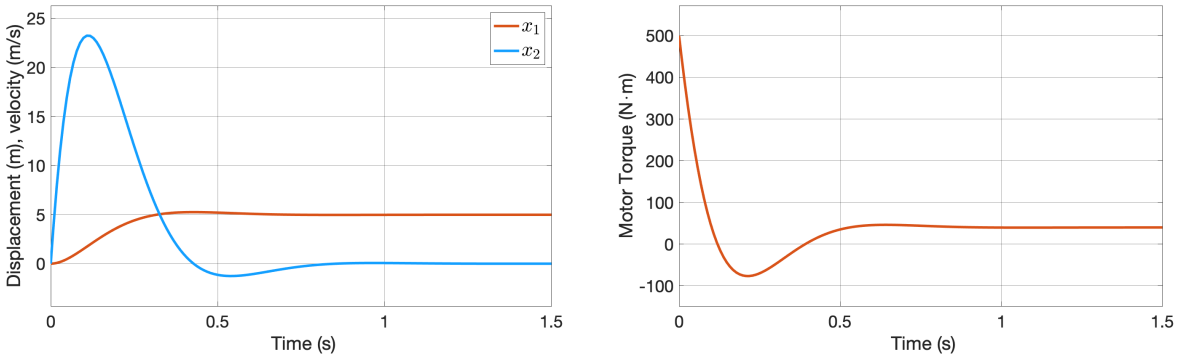
Let

$$Q = \begin{bmatrix} 10000 & 0 \\ 0 & 100 \end{bmatrix}, \quad R = 1$$

The constants of the system are $m = 1$, $k_s = 8$. Substituting these values into the Riccati equation, and after solving for P, inserting the solution into equation (39) yields:

$$K = \begin{bmatrix} 92.3195 & 13.6249 \end{bmatrix}$$

The control target is set to $x_d = 5$. The simulation results are shown in Figure 8.



(a) State variables (b) Control input

Figure 8 — LQR simulation results

It can be seen that the controlled system quickly stabilizes to the target point.

3.2.2 Nonlinear Mass-spring System

Again taking the classic mass-spring system as shown in Figure 7 for the controller design experiment, but this time assuming a nonlinear spring $F_s = -k_s x^3$. Set the control target as an sinusoidal function. It is easy to get by Newton's method, the dynamics model of the system is:

$$\begin{cases} x_1 &= x \\ \dot{x}_1 &= x_2 \\ \dot{x}_2 &= -\frac{k_s}{m} x_1^3 + \frac{1}{m} u \end{cases}$$

At this time:

$$\begin{cases} e &= x_1 - x_{1d} \\ \dot{e} &= x_2 - \dot{x}_{1d} \end{cases}$$

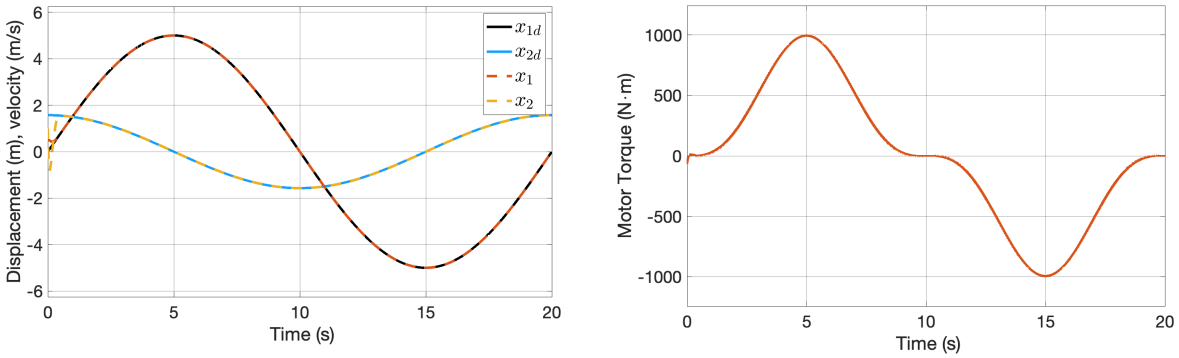
From equations (45) and (48), we get:

$$\begin{cases} u_{eq} &= -m(-\frac{k_s}{m}x_1^3 - \ddot{x}_{1d} + c(x_2 - \dot{x}_{1d})) \\ u_{sw} &= -m(\eta \cdot sgn(s) + ks) \end{cases}$$

Thus

$$u = m(\frac{k_s}{m}x_1^3 + \ddot{x}_{1d} - c(x_2 - \dot{x}_{1d}) - \eta \cdot sgn(s) - ks)$$

The experiment are conducted under condition $m = 1$, $k_s = 8$, $k = 10$, $\eta = 15$, and the control tracking target is set to $5 \cdot \sin(\frac{\pi}{10} \cdot t)$. The simulation results are shown in Figure 9.



(a) State Variables (b) Control Input

Figure 9 — SMC simulation results

As can be seen from the figure that the system is stable with good tracking performance, but there exists chattering in the input.

3.2.3 Simplified Inverted Pendulum

Taking the classic single inverted pendulum system as an example [26], as the inverted pendulum showed in figure 10 is a simplification of the research object in this thesis, the simulation can effectively verify the feasibility of HSMC controller.

It can easily be obtained that under the ideal condition of no friction or damping, the linearized state space expression near the equilibrium point of the system can be obtained.

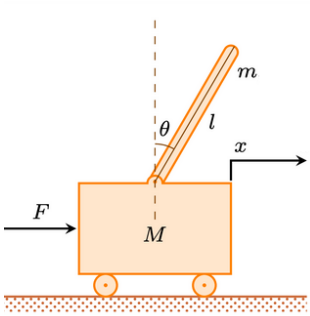


Figure 10 — Inverted pendulum structure

$$\begin{bmatrix} \dot{x} \\ \ddot{x} \\ \dot{\theta} \\ \ddot{\theta} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & -\frac{m^2gl^2}{Den} & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & -\frac{mgl(M+m)}{Den} & 0 \end{bmatrix} \begin{bmatrix} x \\ \dot{x} \\ \theta \\ \dot{\theta} \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{I+ml^2}{Den} \\ 0 \\ \frac{-ml}{Den} \end{bmatrix} u$$

Where I is the moment of inertia of the pendulum rod around the end point, and den is the common denominator:

$$\begin{cases} I &= \frac{1}{3}ml^2 \\ Den &= I(M+m)+Ml^2 \end{cases}$$

At this point:

$$\begin{cases} e_1 &= x - x_d \\ \dot{e}_1 &= \dot{x} \\ e_2 &= \theta \\ \dot{e}_2 &= \dot{\theta} \end{cases}$$

Let the sliding surfaces of the first and second layers be:

$$\begin{cases} s_1 &= c_1e_1 + \dot{e}_1 \\ s_2 &= c_2e_2 + \dot{e}_2 \\ S &= \alpha \cdot s_1 + \beta \cdot s_2 \end{cases}$$

It follows that:

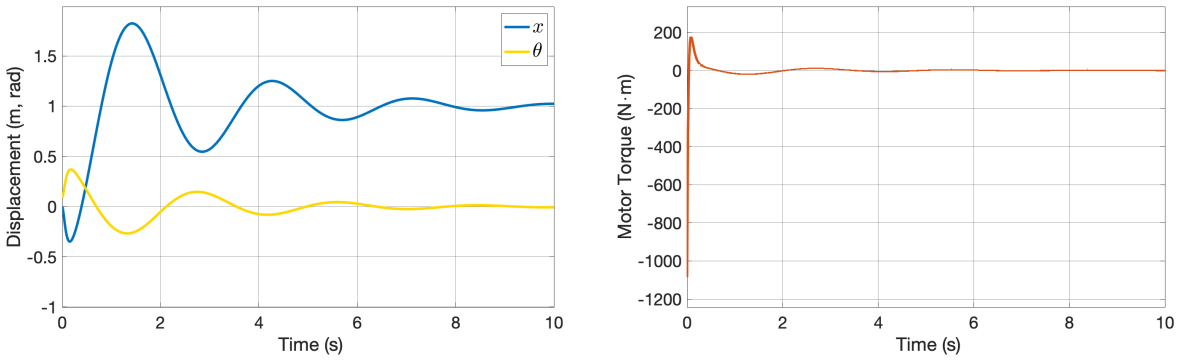
$$\begin{cases} u_{eq1} &= -b_1^{-1}(c_1\dot{x} - \frac{m^2gl^2}{Den} \cdot \theta) \\ u_{eq2} &= -b_2^{-1}(c_2\dot{\theta} - \frac{mgl(M+m)}{Den} \cdot \theta) \\ u_{sw} &= -(\alpha b_1 + \beta b_2)^{-1}[\beta b_2 u_{eq1} + \alpha b_1 u_{eq2} + \eta sgn(S) + kS] \end{cases}$$

Where $b_1 = \frac{I+ml^2}{Den}$, $b_2 = \frac{-ml}{Den}$, thus the final control law can be obtained:

$$u = (\alpha b_1 + \beta b_2)^{-1}[\alpha b_1 u_{eq1} + \beta b_2 u_{eq2} - \eta sgn(S) - kS]$$

The simulation are conducted with condition: $M = 3.5$, $m = 1.5$, $l = 1$, $g = 9.8$, $x_d = 1$, $\theta_0 = 0.0873$rad.

As can be seen from the figure 11, the system can reach a stable state after $8s$, but the input exhibits chattering.



(a) State variables (b) Control input

Figure 11 — HSMC simulation results

3.3 Summary

This chapter briefly describes the concept of three control algorithms and builds different tests for them. The process of designing an LQR controller for linear models is relatively simple and has good control effect for a fixed target, but it is not very effective for dynamic target tracking. Therefore, the SMC algorithm is used to improve it, and the SMC can adapt well to the nonlinear mass-spring model and has excellent performance in dynamic tracking. However, the basic SMC can only control one state variable, so the HSMC algorithm is introduced to control multiple state variables in a hierarchical manner, and the classical single inverted pendulum is tested. HSMC can solve the control problem of the inverted pendulum well, so in the next chapter, LQR and HSMC will be used to design and simulate the controller for the system studied in this thesis.

4 CONTROLLER DESIGN AND VERIFICATION

In this section, we will first analyze the system characteristics and design LQR and HSMC controllers based on linear and nonlinear models, respectively. Then, we will conduct experimental verification and numerical simulation using MATLAB's SIMULINK, and analyze the simulation results.

4.1 Problem Formulation

For the previous mentioned two-wheeled self-balancing robot, the control goal is to drive the robot to the desired position while keeping the cart and pendulum pitch angles around the equilibrium position. To simplify, robot will have nonzero initial pitch angles in the following simulations, and the final goal is to stablized it at position 0.

The basic system constants are shown in the table below.

Table 3 — Simulation System Constants

Parameter Name	Symbol	Simulation Value
Cart mass	m_1	$1kg$
Pendulum mass	m_2	$0.5kg$
Wheel mass	m_w	$0.1kg$
Cart half length	l_1	$0.25m$
Pendulum half length	l_2	$0.5m$
Wheel radius	R_w	$0.1kg$
Wheel center-to-center distance	D	$0.2m$
Gravitational acceleration	g	$9.8m/s^2$
Cart's initial pitch angle	θ_{1_0}	$1deg$
Pendulum initial pitch angle	θ_{2_0}	$1deg$

4.1.1 System Characteristics Analysis

The system characteristics include stability, controllability, and observability. The controllability of the system is a prerequisite for designing a controller, and if the system is unobservable, additional state estimation observers

need to be designed. The simulation scheme will be built based on the obtained results.

(1) System Stability Analysis

System stability analysis can help determine the conditions for stability. When designing a controller, the effect of system parameter changes on system stability should be considered, and the safety margin of system stability should be assessed. By assessing the safety margin, it is possible to avoid system instability, thus ensuring the safety and reliability of the system.

Let $x = 0, \quad \tau = 0$ and solve for the equilibrium state x_e of the system. Substitute into formula (31) to get:

$$\begin{cases} -M_{52} \sin\theta_2 &= 0 \\ M_{51} \sin\theta_1 &= 0 \end{cases}$$

Let $\theta_1, \theta_2 \in [-\pi/2, \ \pi/2]$, we can obtain

$$x_e = [\ x \quad 0 \quad 0\ ]$$

Analyze the eigenvalues of matrix A in the linearized model, which contains multiple zero roots and positive roots, the system is unstable, and the robot cannot autonomously maintain an upright posture in disturbances. Therefore, the system has only one self-balancing state at rest.

(2) System Controllability Analysis

The controllability of linear systems can be judged by the following formula, where n is the number of state variables:

$$rank(Q_c) = rank[\ B \quad AB \quad A^2B \ldots A^{n-1}B\ ] = n$$

Through calculation, it can be obtained that Q_c is of full rank, so the linearized system is fully controllable.

For nonlinear systems, the controllability analysis of linear systems is not applicable. However, nonlinear systems can be linearized near the origin. According to the above analysis, the linearized system is fully controllable, which means that the nonlinear system is locally controllable.

(3) System observability analysis

The observability of the system can be judged by the following formula

$$rank(Q_{ob}) = rank[\ C \quad CA \quad CA^2 \dots CA^{n-1}\ ]^\intercal = n$$

Through calculation, it can be obtained that Q_{ob} is of full rank, so the linearized system is fully observable and does not need additional design of observers.

4.1.2 Plant Implementation

Simulink is a versatile simulation tool that enables multi-domain simulation and model-based design. It facilitates the process of system design, simulation, automatic code generation, as well as continuous testing and verification of embedded systems. Simulink offers a graphical editor, customizable module library, and solvers for modeling and simulating dynamic systems.

(1) Linear model construction

SIMULINK provides a very simple implementation for linear models that can be expressed as state-space equations, using the State-space Block, like figure 12, to achieve the following linear model.

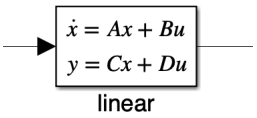


Figure 12 — Linear plant

$$\begin{cases} \dot{x} &= Ax + Bu \\ y &= Cx + Du \\ x|_{t=0} &= x_0 \end{cases}$$

According to the derived formula 37, the specific state matrix and input matrix are obtained, as shown in the following equation:

$$A = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & -10.122 & -0.230 & 0 & 0 & 0 \\ 0 & 40.856 & -4.417 & 0 & 0 & 0 \\ 0 & -8.834 & 10.490 & 0 & 0 & 0 \end{bmatrix}, \quad B = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 5.274 \\ -18.742 \\ 3.512 \end{bmatrix}$$

While matrix C is a 6×6 identity matrix, and D is a 6×1 zero matrix, with input as $u = \tau$.

(2) Nonlinear Model Construction

The S-function provided by SIMULINK can be employed for constructing complex nonlinear models. This function follows a general form and enables the development of continuous, discrete, and hybrid systems. Specifically, the S-function specifies the behavior of a block during different stages of the simulation. At every simulation step, the simulation engine invokes a method to accomplish a designated task.

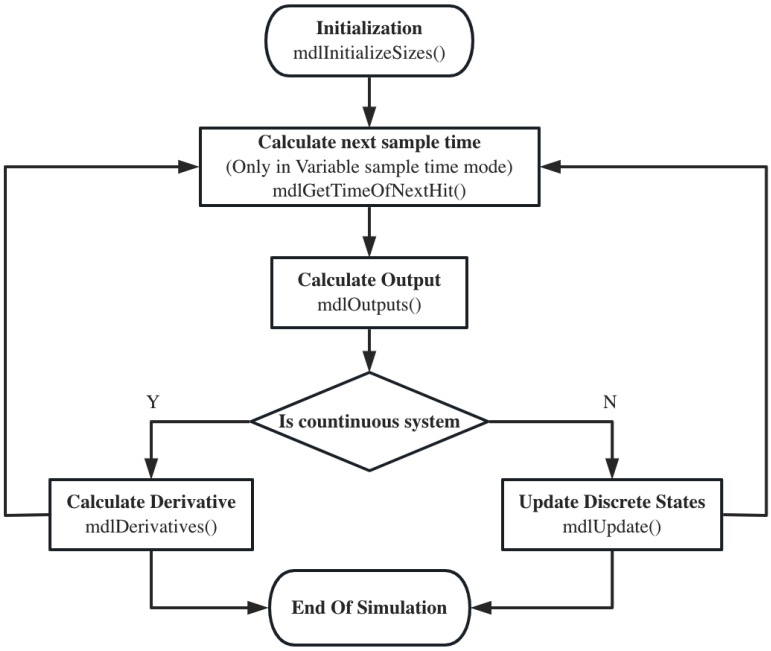


Figure 13 — s-function procedure

MATLAB provides a level-1 s-function template $sfunctml.m$. Based on

this template and the derived equation (31), it is easy to implement the required system model.

Write the following differential equations in code form as an M-file, and then use the S-function Block in SIMULINK to call the corresponding code file.

$$\begin{cases} \ddot{x} &= f_{1x} + b_{1x}\tau \\ \ddot{\theta}_1 &= f_{2x} + b_{2x}\tau \\ \ddot{\theta}_2 &= f_{3x} + b_{3x}\tau \end{cases} \tag{65}$$

where,

$$\begin{aligned} f_{1x} &= [2D_N]^{-1}[(0.5\sin\theta_1(0.0156\cos^2(\theta_1-\theta_2)-0.0790) \\ &\quad -0.125\sin(\theta_1-\theta_2)(0.0677\cos\theta_2-0.0625\cos\theta_1\cos(\theta_1-\theta_2)))\dot{\theta}_1^2 \\ &\quad +(0.25\sin\theta_2(0.0156\cos^2(\theta_1-\theta_2)-0.0790) \\ &\quad +0.125\sin(\theta_1-\theta_2)(0.1458\cos\theta_1-0.0313\cos\theta_2\cos(\theta_1-\theta_2)))\dot{\theta}_2^2 \\ &\quad +4.9\sin\theta_1(0.1458\cos\theta_1-0.0313\cos\theta_2\cos(\theta_1-\theta_2)) \\ &\quad +2.45\sin\theta_2(0.0677\cos\theta_2-0.0625\cos\theta_1\cos(\theta_1-\theta_2))] \end{aligned} \tag{66}$$

$$\begin{aligned} f_{2x} &= [2D_N]^{-1}[(0.5\sin\theta_1(0.1458\cos\theta_1-0.0313\cos\theta_2\cos(\theta_1-\theta_2)) \\ &\quad -0.125\sin(\theta_1-\theta_2)(0.2250\cos(\theta_1-\theta_2)-0.125\cos\theta_1\cos\theta_2))\dot{\theta}_1^2 \\ &\quad +(0.125\sin(\theta_1-\theta_2)(0.0625\cos^2\theta_2-0.525) \\ &\quad +0.25\sin\theta_2(0.1458\cos\theta_1-0.0313\cos\theta_2\cos(\theta_1-\theta_2)))\dot{\theta}_2^2 \\ &\quad +2.45\sin\theta_2(0.2250\cos(\theta_1-\theta_2)-0.125\cos\theta_1\cos\theta_2) \\ &\quad +4.9\sin\theta_1(0.0625\cos^2\theta_2-0.525)] \end{aligned} \tag{67}$$

$$\begin{aligned} f_{3x} &= [2D_N]^{-1}[(0.25\sin\theta_2(0.0677\cos\theta_2-0.0625\cos\theta_1\cos(\theta_1-\theta_2)) \\ &\quad +0.125\sin(\theta_1-\theta_2)(0.2250\cos(\theta_1-\theta_2)-0.125\cos\theta_1\cos\theta_2))\dot{\theta}_2^2 \\ &\quad -(0.125\sin(\theta_1-\theta_2)(0.250\cos^2\theta_1-0.4875) \\ &\quad -0.5\sin\theta_1(0.0677\cos\theta_2-0.0625\cos\theta_1\cos(\theta_1-\theta_2)))\dot{\theta}_1^2 \\ &\quad +4.9\sin\theta_1(0.2250\cos(\theta_1-\theta_2)-0.125\cos\theta_1\cos\theta_2) \\ &\quad +2.45\sin\theta_2(0.250\cos^2\theta_1-0.4875)] \end{aligned} \tag{68}$$

$$
\begin{aligned}
b_{1x} &= [D_N]^{-1}[(0.0156\cos^2(\theta_1-\theta_2)-0.0790)/2 \\
&\quad -(0.1458\cos\theta_1-0.0313\cos\theta_2\cos(\theta_1-\theta_2))] \\
b_{2x} &= [D_N]^{-1}[(0.1458\cos\theta_1-0.0313\cos\theta_2\cos(\theta_1-\theta_2))/2 \\
&\quad -(0.0625\cos^2\theta_2-0.525)] \\
b_{3x} &= [D_N]^{-1}[(0.0677\cos\theta_2-0.0625\cos\theta_1\cos(\theta_1-\theta_2))/2 \\
&\quad -(0.2250\cos(\theta_1-\theta_2)-0.125\cos\theta_1\cos\theta_2)]
\end{aligned} \tag{69}
$$

$$
\begin{aligned}
D_N &= 0.0365\cos^2\theta_1-0.0156\cos\theta_1\cos\theta_2*\cos(\theta_1-\theta_2) \\
&\quad +0.0085\cos^2\theta_2+0.0141\cos^2(\theta_1-\theta_2)-0.0711
\end{aligned}
$$

4.2 LQR Controller Experiments

The primary objective of this section is to develop an LQR controller and evaluate its performance through simulations conducted under diverse conditions, and set its performance as a benchmark for comparison.

4.2.1 Controller Design

The implementation of the LQR controller is relatively simple. After obtaining the corresponding state variable feedback value according to equation (39).

Select:

$$
Q=\begin{bmatrix}
100 & 0 & 0 & 0 & 0 & 0 \\
0 & 5000 & 0 & 0 & 0 & 0 \\
0 & 0 & 100 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 1
\end{bmatrix}, \quad R=1
$$

Solve the Ricatti equation and obtain the feedback matrix K:

$$
K=\begin{bmatrix} 10.000 & 26.581 & 631.783 & 42.085 & 47.669 & 209.072 \end{bmatrix} \tag{70}
$$

4.2.2 Linear Plant Simulation and Analysis

The following figure shows the simulation scheme of the LQR controller applied to the linear model in SIMULINK.

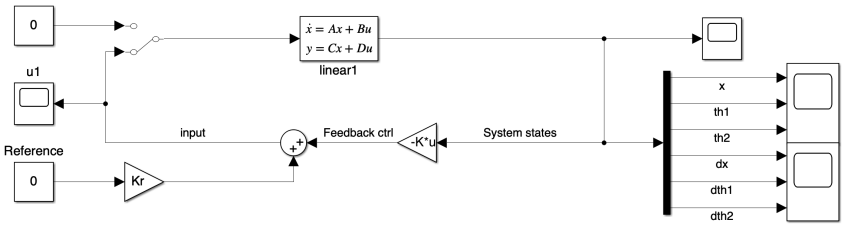


Figure 14 — LQR applied to linear model

The following figures show the simulation results.

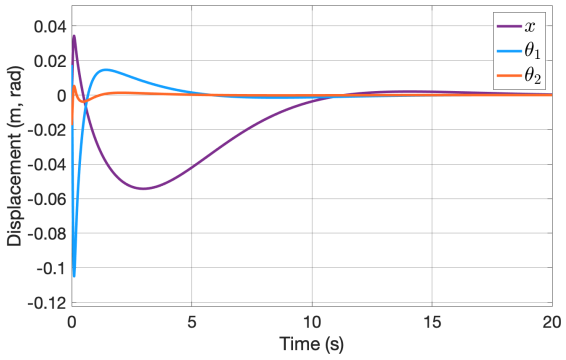


Figure 15 — State transient

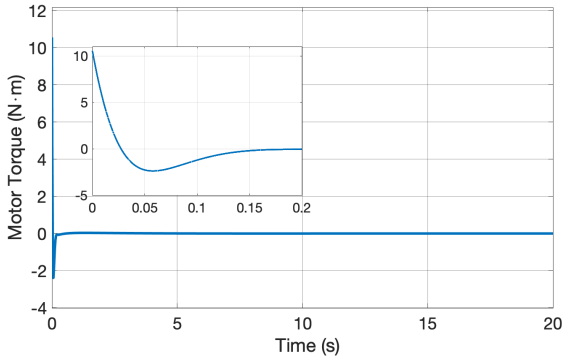


Figure 16 — Control input

From the figure 15 and 16, it can be seen that through LQR control, the transient time of the system is about $20s$, and the maximum single-direction pitch angle is $0.1rad$. The initial control input is as high as $11N \cdot m$.

4.2.3 Nonlinear Plant Simulation and Analysis

In the figure 17, the LQR controller is applied to the nonlinear plant in SIMULINK under three different disutrbance condisions.

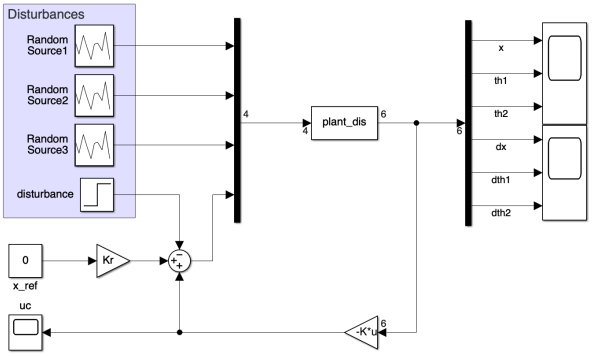


Figure 17 — LQR simulation on nonlinear plant

(1) No Disturbance

The following figure shows the simulation results.

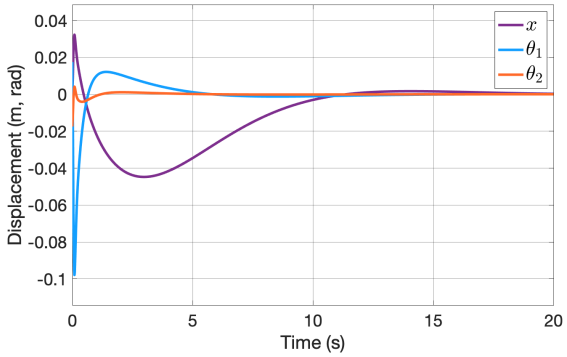


Figure 18 — State transient without disturbance

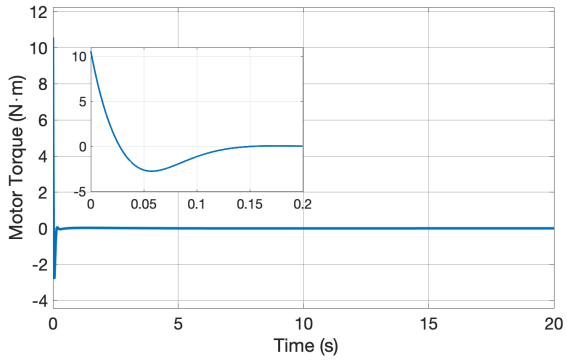


Figure 19 — Control input without disturbance

As can be seen from the figure, the designed LQR controller also has good control performance for a nonlinear model with a small initial angles, and the

system response and generated control quantity are very similar to the linear model simulation results.

(2) Constant Input Disturbance

Based on the nonlinear model, a constant disturbance of $-0.5Nm$ was added at the input at 10s. The simulation results are shown in the following figures.

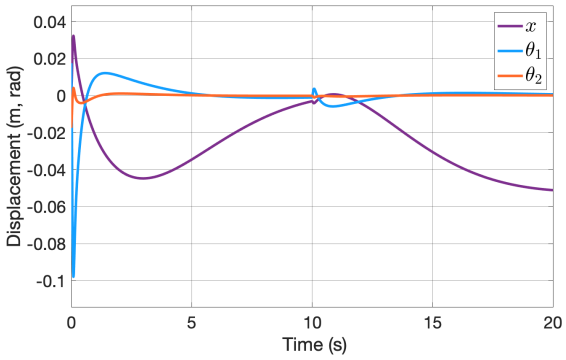


Figure 20 — State transient with constant disturbance

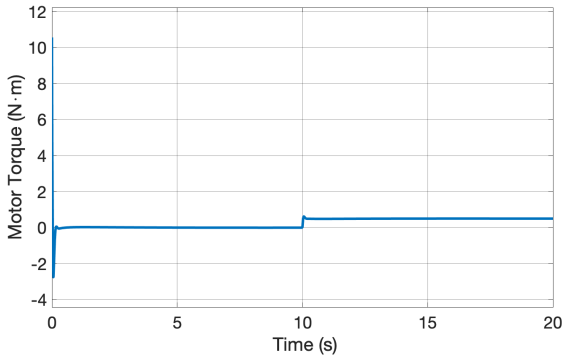


Figure 21 — Control input with constant disturbance

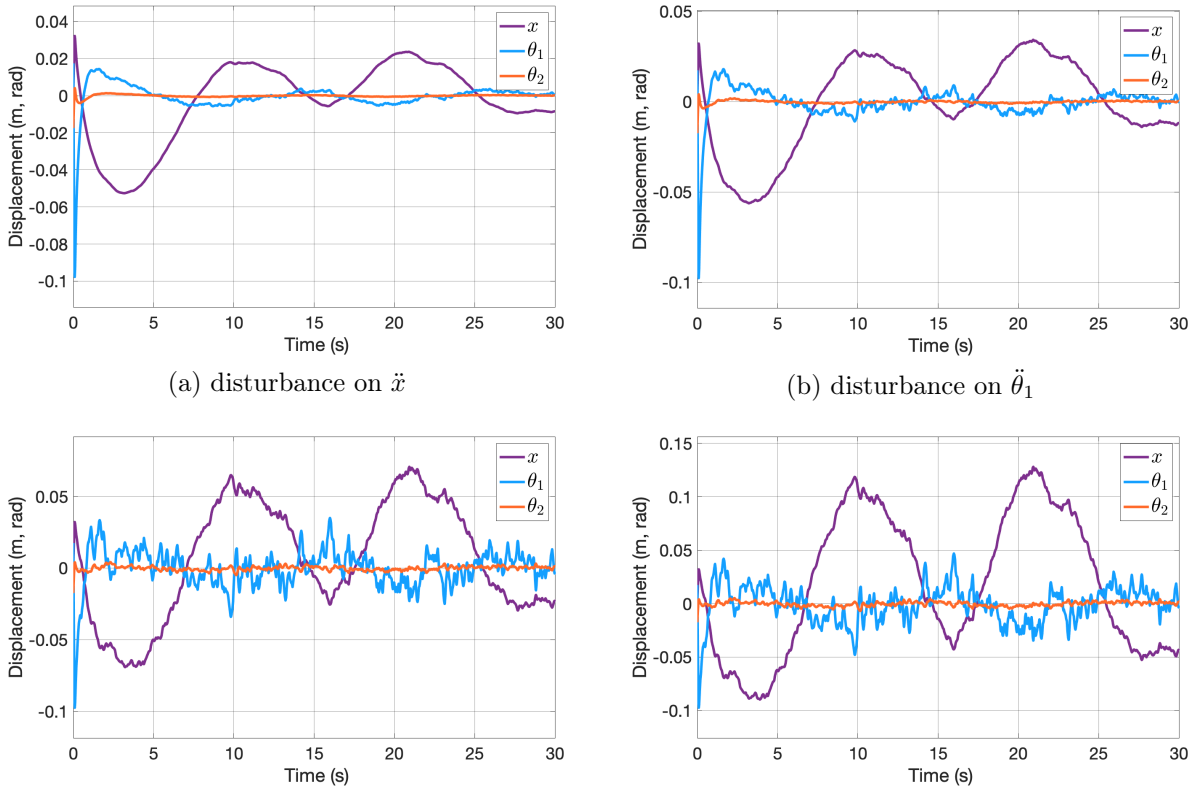
As can be seen, the system response to the constant disturbance mainly manifests in the displacement, and for the constant disturbance, the system eventually stabilizes at the position of -0.05.

(3) Random Disturbance

The simulation is based on the nonlinear model, and random disturbances with mean value 0 and variance 0.1 are applied to $\ddot{x}, \ddot{\theta}_1, \ddot{\theta}_2$, respectively. In the end, all three disturbances are applied to the system simultaneously.

From the figure 22, it can be concluded that among the three state variables, $\ddot{\theta}_2$ has the greatest impact on the system due to the disturbance, followed

by $\ddot{\theta}_1$. When all three random disturbances are applied simultaneously, the system exhibits large amplitude and high frequency oscillations.



(a) disturbance on $\ddot{x}$

(b) disturbance on $\ddot{\theta}_1$

(c) disturbance on $\ddot{\theta}_2$

(d) disturbance all

Figure 22 — LQR simulation on nonlinear plant, random disturbance

4.3 Nonlinear HSMC Controller Experiments

In this section, the nonlinear HSMC controller is designed based on the nonlinear model, and the simulation is performed in MATLAB.

4.3.1 Controller Design

The control objective is 3 state variables and x_d is a constant, with $\theta_{1d} = 0, \theta_{2d} = 0$. According to equation (51), we set 3 first-layer sliding surfaces and

1 second-layer sliding surface:

$$\begin{cases} s_1 &= c_1\theta_1 + \dot{\theta}_1 \\ s_2 &= c_2\theta_2 + \dot{\theta}_2 \\ s_3 &= c_3(x - x_d) + \dot{x} \\ S &= w_1 s_1 + w_2 s_2 + w3 s_3 \end{cases} \tag{71}$$

Where $c_1, c_2, c_3 > 0$ are the slopes of the sliding surfaces, and $w_1, w_2, w_3 > 0$ are the weights of the sliding surfaces.

According to equations (52) and the nonlinear model (31), we obtain the 3 equivalent control:

$$\begin{cases} u_{eq1} &= -b_{1x}^{-1}(f_{1x} + c_1 \cdot \dot{\theta}_1) \\ u_{eq2} &= -b_{2x}^{-1}(f_{2x} + c_2 \cdot \dot{\theta}_2) \\ u_{eq3} &= -b_{3x}^{-1}(f_{3x} + c_3 \cdot \dot{x}) \end{cases} \tag{72}$$

From equation (61), we obtain the HSMC control input based on the nonlinear model:

$$u = (w_1 b_{1x} + w_2 b_{2x} + w_3 b_{3x})^{-1}[w_1 b_{1x} u_{eq1} + w_2 b_{2x} u_{eq2} + w_3 b_{3x} u_{eq3} - \eta \cdot sgn(S) - kS] \tag{73}$$

4.3.2 Nonlinear Plant Simulation and Analysis

The figure below shows the simulation scheme of the HSMC controller. Expriments are conducted under condition: $c_1 = 4,\ c_2 = 2,\ c_3 = 4,\ w_1 = 2,\ w_2 = 2.5,\ w_3 = 3.5,\ \eta = 0.1,\ k = 40.$

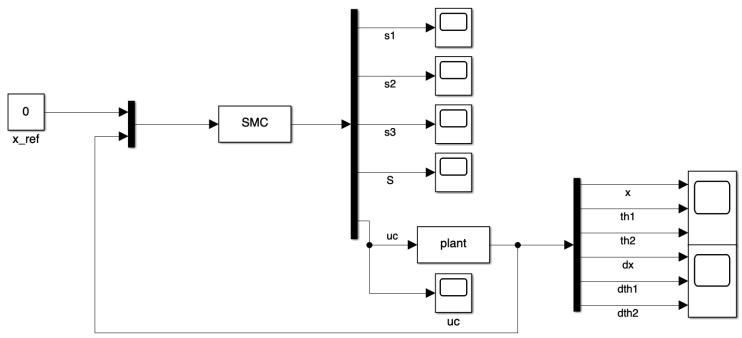
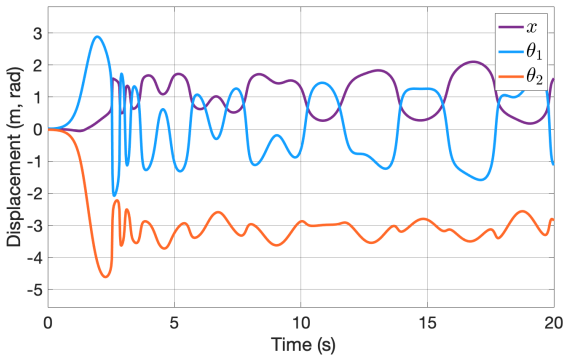
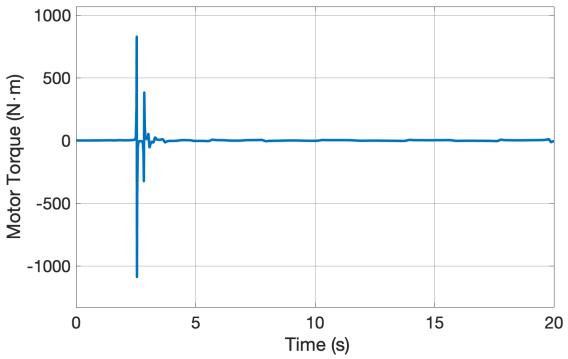


Figure 23 — HSMC Simulation Based on Nonlinear Model

The following figures show the simulation results.

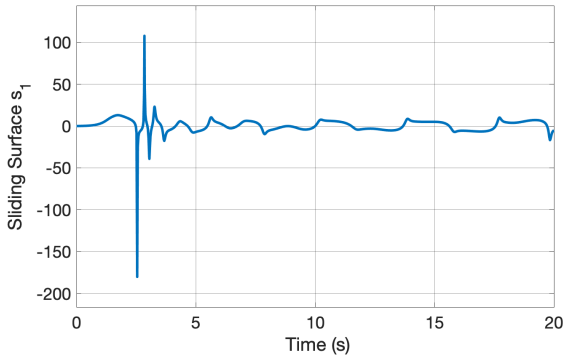


(a) State variables

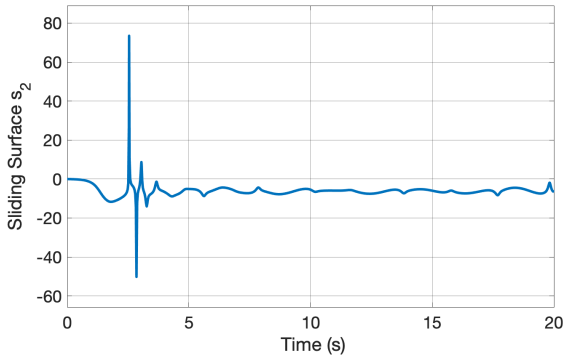


(b) Control input

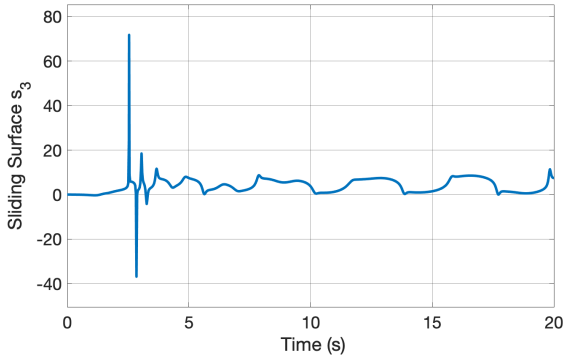
Figure 24 — Nonlinear HSMC simulation results



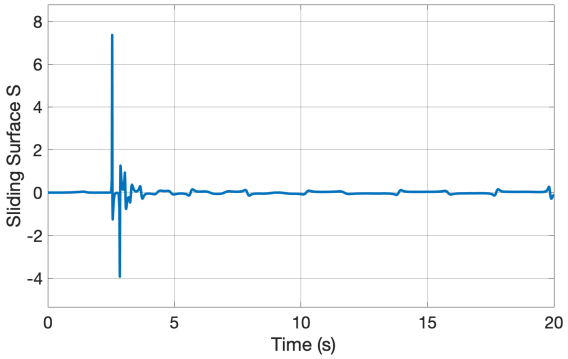
(a) Sliding Surface s1



(b) Sliding Surface s2



(c) Sliding Surface s3



(d) Sliding Surface S

Figure 25 — Sliding surface conditions

As can be seen, since the nonlinear HSMC controller does not handle the coupling problem, the system cannot reach balance. Although sliding surface s_1 oscillates near the expected range, s_2 and s_3 oscillate near positive and negative values, respectively. So the control expectation of $S = 0$ is still achieved in the second-layer sliding surface. The parameters are difficult to tune, and the

system control generated at this time is impractical. The design of nonlinear HSMC controller failed.

4.4 Linear HSMC Controller Experiments

As nonlinear HSMC controller did not reach expectation. This section is mainly for designing linear HSMC controller and conducting simulation under various conditions.

4.4.1 Controller design

The sliding surface is set as formula (71). To address the issue of high coupling degree, a hierarchical sliding mode controller can be designed based on the linearized model.

According to equation (52) and linear model (37), three equivalent controls can be obtained:

$$\begin{cases} u_{eq1} &= -b_1^{-1}[\frac{M_{51}(M_{32}^2-4M_{11}M_{22})}{D_L}\cdot\theta_1+\frac{M_{52}(2M_{11}M_{41}-M_{31}M_{32})}{D_L}\cdot\theta_2+c_2\cdot\dot{\theta}_1] \\ u_{eq2} &= -b_2^{-1}[\frac{M_{51}(2M_{11}M_{41}-M_{31}M_{32})}{D_L}\cdot\theta_1+\frac{M_{52}(M_{31}^2-4M_{11}M_{21})}{D_L}\cdot\theta_2+c_2\cdot\dot{\theta}_2] \\ u_{eq3} &= -b_3^{-1}[\frac{M_{51}(2M_{22}M_{31}-M_{32}M_{41})}{D_L}\cdot\theta_1+\frac{M_{52}(2M_{21}M_{32}-M_{31}M_{41})}{D_L}\cdot\theta_2+c_3\cdot\dot{x}] \end{cases} \tag{74}$$

where,

$$\begin{aligned} D_L &= (2M_{22}M_{31}^2-2M_{31}M_{32}M_{41}+2M_{21}M_{32}^2+2M_{11}M_{41}^2-8M_{11}M_{21}M_{22}) \\ b_1 &= (2M_{22}M_{31}-M_{32}M_{41}-2M_{32}^2+4M_{11}M_{22})/D_L \\ b_2 &= (2M_{21}M_{32}-M_{31}M_{41}-4M_{11}M_{41}+M_{31}M_{32})/D_L \\ b_3 &= (M_{41}^2-4M_{21}M_{22}-4M_{22}M_{31}+2M_{32}M_{41})/D_L \end{aligned}$$

From Equation 61, we can obtain the HSMC control input based on the linear model:

$$u=(w_1b_1+w_2b_2+w_3b_3)^{-1}[w_1b_1u_{eq1}+w_2b_2u_{eq2}+w_3b_3u_{eq3}-\eta\cdot sgn(S)-kS] \tag{75}$$

In the experiment, $c_1=2,\ c_2=5,\ c_3=1.3,\ w_1=1.5,\ w_2=4.4,\ w_3=2.7,\ \eta=0.005,\ k=10$

4.4.2 Linear Plant Simulation and Analysis

The implementation of the HSMC controller involves a large number of equations. To facilitate implementation, the HSMC based on the linear model is implemented using the MATLAB function block. The figure 26 shows the simulation setup of HSMC applied to the linear model in Simulink.

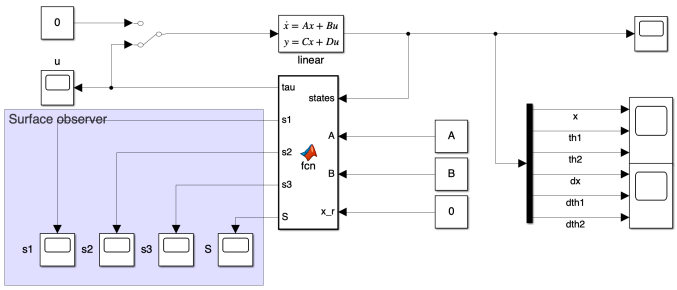
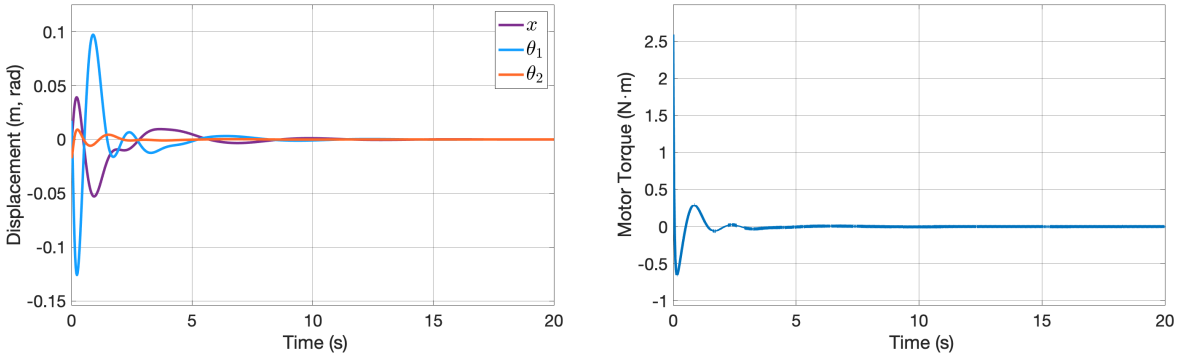


Figure 26 — SMC applied to the linear model

The following figures show the simulation results.

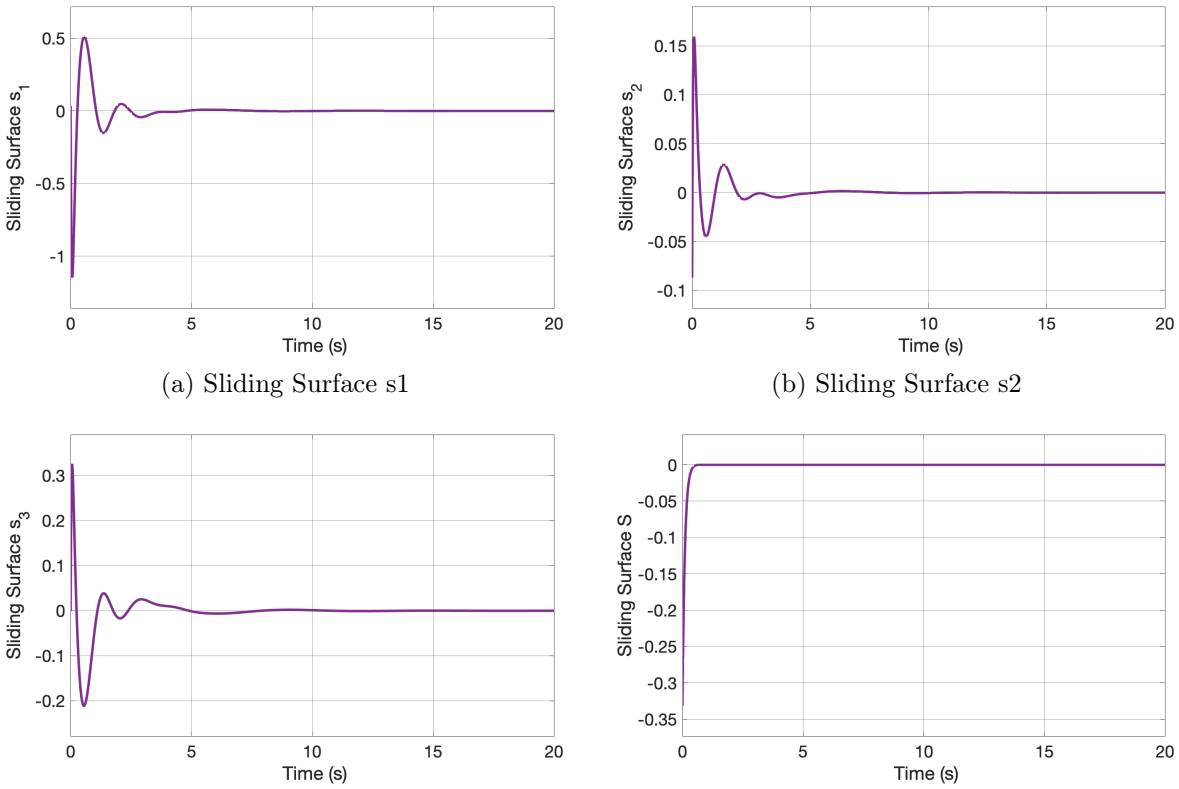


(a) State variables (b) Control input

Figure 27 — HSMC Simulation results on linear plant

As can be seen from the figure 27 and 28, the pendulum is soon stablized upright, and the maximuim unidirectional pitch angle is about 0.01rad. The maximum amplitude of the cart pitch angle is approximately $0.125rad$, the corresponding range is approximately $0.225rad$, the system transient time is approximately $8s$, the maximum control torque value is approximately $2.5Nm$, and the chattering exists. Both the first layer sliding surface $s1, s2, s3$ and the second layer sliding surface S successfully converge to the target state of 0 as expected.

The following figure shows sliding surface conditions.



(a) Sliding Surface s1

(b) Sliding Surface s2

(c) Sliding Surface s3

(d) Sliding Surface S

Figure 28 — Sliding surface condition

4.4.3 Nonlinear Plant Simulation and Analysis

After successfully verifying the HSMC controller in the linear model, the control plant is switched to a nonlinear model in scheme shown in figure 29.

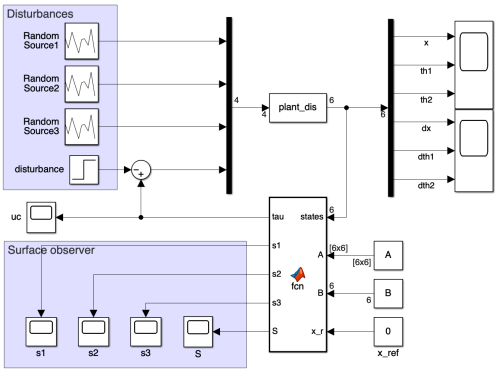
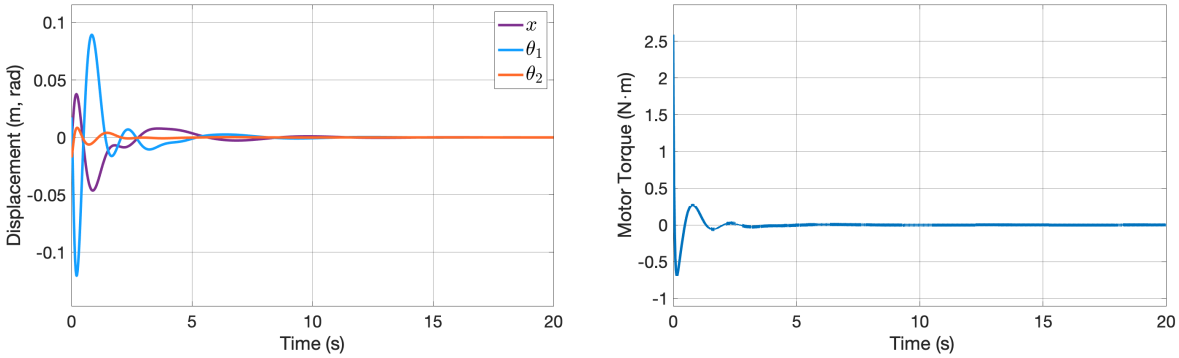


Figure 29 — HSMC applied to the non-linear model

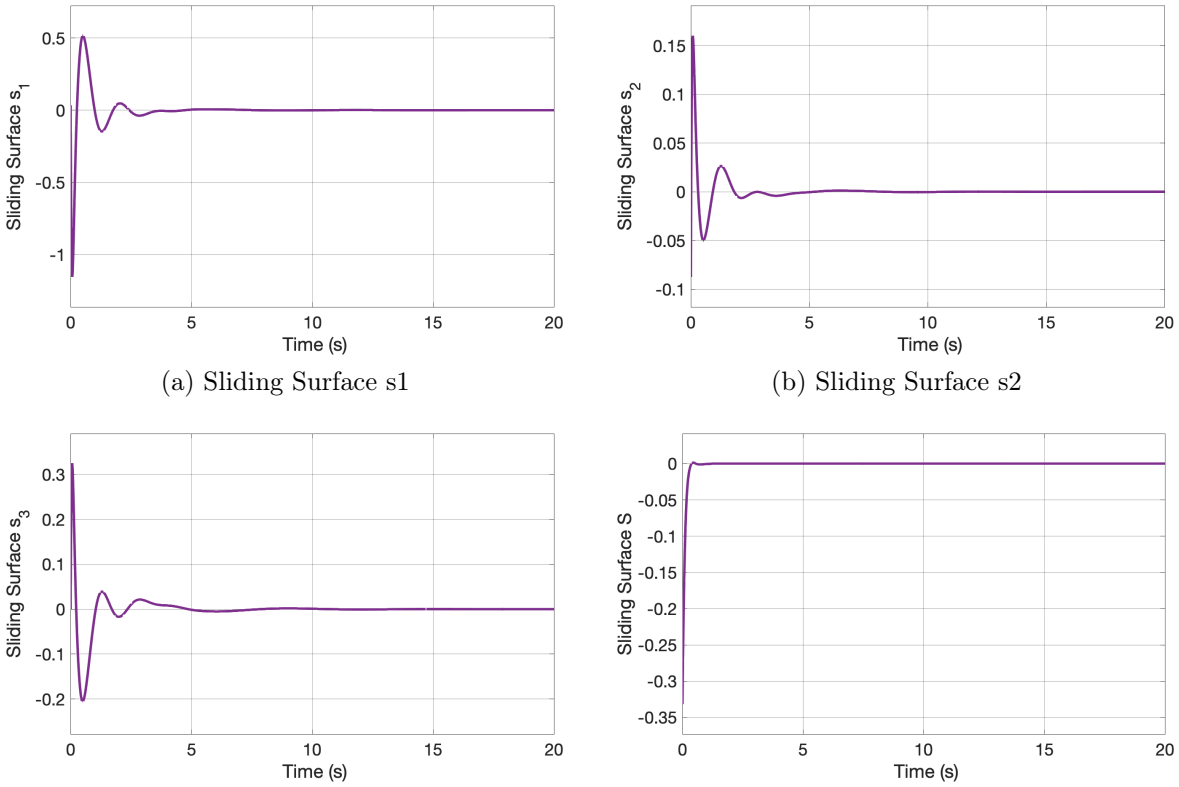
(1) No disturbances

The following figures show the simulation results without disturbances.



(a) State variables (b) Control input

Figure 30 — HSMC simulation results on nonlinear plant



(a) Sliding Surface s1 (b) Sliding Surface s2

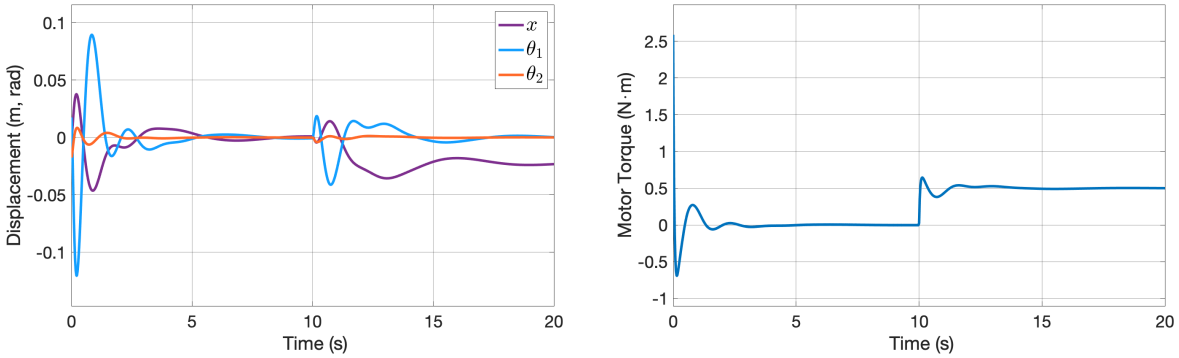
(c) Sliding Surface s3 (d) Sliding Surface S

Figure 31 — Sliding surface conditions

As can be seen, the linear HSMC controller has good performance on the nonlinear model. The system responsed similarly, all states converged.

(2) Constant Input Disturbance

A constant disturbance of $-0.5Nm$ is applied at the input at 10s, and the simulation results are shown in the following figure.



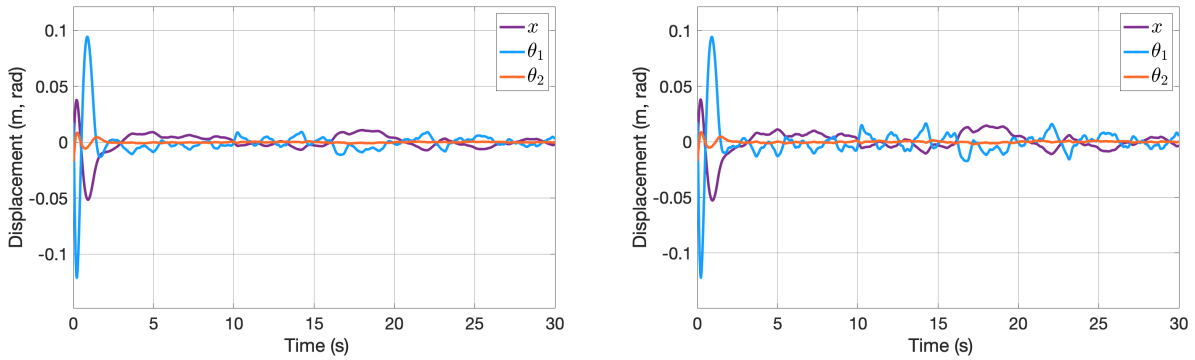
(a) State variables (b) Control input

Figure 32 — HSMC simulation results with constant disturbance

As shown in the figures above, when the system quickly stabilizeed with constant disturbance, the pitch angle of the pendulum undergoes slight changes, but the pitch angle of the cart has little oscillation, with the maximum amplitude of about $0.05rad$, which lasts $4s$, and the system finally stabilizes at the position of $-0.025m$.

(3) Random Disturbance

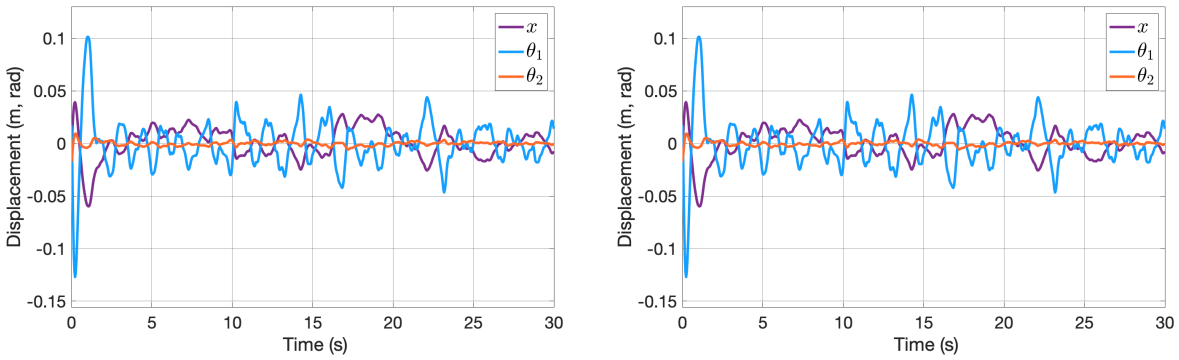
Based on the nonlinear model, random disturbances with a mean value of 0 and a variance of 0.1 are applied separately to $\ddot{x}, \ddot{\theta}_1, \ddot{\theta}_2$, and finally, all three disturbances are applied to the system simultaneously, results are as follows:



(a) disturbance on $\ddot{x}$ (b) disturbance on $\ddot{\theta}_1$

Figure 33 — HSMC simulation on nonlinear plant, random disturbance, part I

Following the previous figure, figure 34 presents and remaining simulation results.



(a) disturbance on $\ddot{\theta}_2$ (b) disturbance all

Figure 34 — HSMC simulation on nonlinear plant, random disturbance, part II

From the figure 33 and 34, it can be concluded that among the three state variables, the disturbance affecting $\ddot{\theta}_2$ has the greatest impact on the system, followed by $\ddot{\theta}_1$. After the implementation of the three random disturbances, the system exhibits a certain degree of oscillation while relatively maintaining pendulum upright, during last disturbed condition, the maximum amplitude is $0.05rad$.

4.5 Summary

This chapter constructs simulations in SIMULINK, verifies the control effects of LQR and HSMC controllers on linear and nonlinear systems, and performs anti-disturbance tests using nonlinear models. The HSMC controller designed based on the nonlinear model cannot control the system successfully, but the HSMC controller and LQR controller designed based on the linearized model have good effects. Both controllers can achieve a certain anti-disturbance effect. However, compared with LQR, the maximum torque required for the HSMC controller is about 1/5 of that required for the LQR controller, and the transient time of the system is reduced to 1/2 of LQR. At the same time, the controller designed based on HSMC has stronger robustness, and the oscillation amplitude and frequency of the system are smaller under the same interference. The designed HSMC controller can enable the self-balancing robot to have a better response capacity while possessing stronger anti-disturbance ability.

5 CONCLUSION AND FUTURE WORK

5.1 Conclusion

The inverted pendulum system, as a typical nonlinear system studied in the control field, has been applied more and more widely in the validation of control theory and practical engineering in recent years. The main work of this paper is summarized as follows:

1. In view of the existing problems of the traditional two-wheel self-balancing robot, in order to improve its performance an improved robot is proposed. Compared with the ordinary ones, an additional pendulum is added, and a double inverted pendulum system model is established. This system has the characteristics of simple structure and intuitive image, which is convenient for the establishment of mathematical models and the verification of control theory.

2. In-depth research is carried out on the motion and characteristics of the two-wheel self-balancing robot. First, a coordinate system is established and kinematic analysis is performed on the designed system, and a kinematic model is established; then, the Lagrangian equation method is used to establish its dynamic model; finally, considering the small-angle oscillation of the system around equilibrium position, a simplified linear system model is established by linearizing the dynamic model reasonably. The characteristics of linear and nonlinear models are analyzed.

3. In the aspect of controller design, linear and nonlinear HSMC controllers are designed, and the performance of LQR controller is used as a benchmark for comparison. MATLAB is used for simulation. The controller designed based on the non-linear model cannot meet the control requirements due to the fact that it does not essentially deal with the coupling problem, while the HSMC and LQR based on the linearized model can maintain the robot in a stable state. In general, the linear HSMC controller can reach a stable state faster than the traditional LQR controller,

achieve better control effects, and has significantly better robustness to random disturbances.

The main innovations of this paper are:

1. An improved two-wheel self-balancing robot system is designed. Compared with the conventional two-wheel self-balancing robot, an additional pendulum that can swing back and forth is added, which to some extent solves the problem that the traditional two-wheel robot cannot maintain the body upright during motion, and provides an effective research platform for solving complex balance control problems.
2. Although researchers have already conducted extensive research on the basic self-balancing robot, there is still relatively little research on the two-wheel self-balancing robot based on double inverted pendulum: some of the existing research directly uses more complicated mechanical structures, such as adding an actuator between the body and the top pendulum to enhance the controllability and practicality of the system, and most of the research tends to use more complex controllers, such as fuzzy controllers, etc. This paper can be considered as a supplement and transition between the two.

5.2 Future Work

In terms of modeling and controller design, this paper still faces many challenges.

1. In system modeling, for the convenience of theoretical research, factors such as internal mechanical friction and external disturbances are ignored, and the changes in the z-axis position of the robot are not considered, making the model less accurate and affecting the design of the controller. Therefore, in subsequent research, these factors need to be taken into account.
2. In linearizing the model near the balance point, the method used is too crude. In subsequent research, the Maclaurin expansion method can be used and the high-order terms can be discarded to make the linearized model more accurate.

3. In the HSMC controller, due to the large number of adjustable parameters, trial and error method is finally used to determine the parameters, resulting in the performance of LQR and HSMC controllers not being optimized. Therefore, in future research, methods such as genetic algorithms can be considered to determine the parameters.

4. In the HSMC controller used in this paper, only the most basic exponential reaching law is used, which involves the sign function $sgn()$, leading to the emergence of chattering phenomenon, and it is impossible to achieve such a high switching frequency in reality. Therefore, in subsequent research, the saturation function can be considered, or higher-order sliding mode can be applied.

5. From the simulation results of controllers designed based on linear and nonlinear models, the HSMC controller based on nonlinear models cannot stabilize the system, which is due to the lack of handling coupling problems in the nonlinear model. The decoupling method in this paper only involves linearization, but this makes the model less accurate. Therefore, in subsequent research, more complex controllers can be considered, such as recursive LQR controllers, PSO optimized LQR controllers [27, 28] or fuzzy controllers[29], etc.

REFERENCES

1. Wang Xiaoyu. *Research on Two-wheeled Self-balanced Robot.* PhD thesis, Harbin Institute of Technology, 2007.

2. Tao Yong Wang Tianmiao and Chen Yang. Research status and development trends of the service robotic technology. *SCIENTIA SINICA Informationis*, 42(1049-1066), 2012. ISSN 1674-7267.

3. Ruan Xiaogang et al. *Research and design of two-wheeled self-balancing robot*, pages 2–5. Beijing: Science Press, 2012.

4. Zhao Jianwei and Ruan Xiaogang. Modeling and control of a flexible two-wheel upright self-balance humanoid robot. *Robot*, 31(179-186), 2009. ISSN 1002-0446. doi: 10.13973/j.cnki.robot.2009.02.003.

5. A. M. Almeshal et al. A new configuration of a two-wheeled double inverted pendulum-like robotic vehicle with movable payload on an inclined plane. In *2012 First International Conference on Innovative Engineering Systems*, pages 97–102, 2012. doi: 10.1109/ICIES.2012.6530852.

6. Qian Qingwen. *Research on modeling and control method of two-wheeled self-balancing vehicle pendulum robot.* PhD thesis, Harbin University of Science and Technology, 2019.

7. Saeed Ebrahimi and Arman Mardany. Dynamic modeling and construction of a two-wheeled mobile manipulator, part i: Self-balancing. In *2015 3rd RSI International Conference on Robotics and Mechatronics (ICROM)*, pages 164–169, 2015. doi: 10.1109/ICRoM.2015.7367778.

8. Sang-Joo K Sangtae K. Dynamic modeling of a two-wheeled inverted pendulum balancing mobile robot. *International Journal of Control, Automation, and Systems*, 4(13):1–8, 2014.

9. Jingtao Li, Xueshan Gao, Qiang Huang, Qinjun Du, and Xingguang Duan. Mechanical design and dynamic modeling of a two-wheeled inverted pendu-

lum mobile robot. In *2007 IEEE International Conference on Automation and Logistics*, pages 1614–1619, 2007. doi: 10.1109/ICAL.2007.4338830.

10. Ali Unluturk, Omer Aydogdu, and Ufuk Guner. Design and pid control of two wheeled autonomous balance robot. In *2013 International Conference on Electronics, Computer and Computation (ICECCO)*, pages 260–264, 2013. doi: 10.1109/ICECCO.2013.6718278.

11. Saibal Manna, Shaili Shaw, and Ashok Kumar Akella. Design of higher-order regulator system using pole-placement technique. In *2020 International Conference on Computational Intelligence for Smart Power System and Sustainable Energy (CISPSSE)*, pages 1–4, 2020. doi: 10.1109/CISPSSE49931.2020.9212270.

12. Changkai Xu, Ming Li, and Fangyu Pan. The system design and lqr control of a two-wheels self-balancing mobile robot. In *2011 International Conference on Electrical and Control Engineering*, pages 2786–2789, 2011. doi: 10.1109/ICECENG.2011.6057680.

13. Nurul Hasanah, Syadza Atika Rahmah, Wulandari Puspita Sari, Niam Tamami, Alrijadjis, and Bambang Sumantri. Movement control of two wheels balancing robot using smc based on lyapunov analysis. In *2019 2nd International Conference on Applied Information Technology and Innovation (ICAITI)*, pages 37–43, 2019. doi: 10.1109/ICAITI48442.2019.8982121.

14. Chao Peng, Jianxiao Zou, Jian Zhang, and Qingze Zou. A mixed h2/h∞ robust inversion based 2dof control approach for system with uncertainty. In *2016 35th Chinese Control Conference (CCC)*, pages 4745–4750, 2016. doi: 10.1109/ChiCC.2016.7554089.

15. Yang Menglu. Research on key technology of two-wheel self-balancing robot with adjustable center of gravity. Master's thesis, Hangzhou Dianzi University, 2021.

16. Maurício Figueiredo and F. Gomide. Adaptive neuro-fuzzy modeling. *Proceedings of 6th International Fuzzy Systems Conference*, 3:1567–1572, 1997.

17. Andrew G. Barto, Richard S. Sutton, and Charles W. Anderson. Neuronlike adaptive elements that can solve difficult learning control problems. *IEEE Transactions on Systems, Man, and Cybernetics*, SMC-13(5):834–846, 1983. doi: 10.1109/TSMC.1983.6313077.

18. Danyang Wang, Mei Wang, and Yuancheng Li. Genetic and fuzzy fusion algorithm for coal-feeding optimal control of coal-fired power plant. In *2020 International Symposium on Computer, Consumer and Control (IS3C)*, pages 500–503, 2020. doi: 10.1109/IS3C50286.2020.00136.

19. Laurent Vermeiren et al. Modeling, control and experimental verification on a two-wheeled vehicle with free inclination: An urban transportation system. *Control Engineering Practice*, 19(7):744–756, 2011. ISSN 0967-0661. doi: https://doi.org/10.1016/j.conengprac.2011.04.002.

20. Agouri et al. Modelling and control of a two-wheeled vehicle with extendable intermediate body on an inclined surface. 02 2013. doi: 10.2316/P.2013.794-049.

21. Magdi S. Mahmoud and Mohammad T. Nasir. Robust control design of wheeled inverted pendulum assistant robot. *IEEE/CAA Journal of Automatica Sinica*, 4(4):628–638, 2017. doi: 10.1109/JAS.2017.7510613.

22. K.D. Young, V.I. Utkin, and U. Ozguner. A control engineer's guide to sliding mode control. *IEEE Transactions on Control Systems Technology*, 7(3):328–342, 1999. doi: 10.1109/87.761053.

23. Olfa Jedda, Jalel Ghabi, and Ali Douik. *Sliding Mode Control of an Inverted Pendulum*, volume 79, pages 105–118. 10 2017. ISBN 978-981-10-2373-6. doi: 10.1007/978-981-10-2374-3_6.

24. Wei Wang, Dongbin Zhao, and Dian-Tong Liu. Design of a stable sliding-mode controller for a class of second-order underactuated systems. *Control Theory and Applications, IEE Proceedings*, 151:683 – 690, 12 2004. doi: 10.1049/ip-cta:20040902.

25. Dianwei Qian, Jianqiang Yi, and Dongbin Zhao. Hierarchical sliding mode

control for a class of simo under-actuated systems. *Control and Cybernetics*, 37, 01 2008. doi: 10.1504/IJMIC.2011.041779.

26. Sonia Mahjoub, Faiçal Mnif, and Nabil Derbel. Second-order sliding mode control applied to inverted pendulum. In *14th International Conference on Sciences and Techniques of Automatic Control & Computer Engineering*, pages 269–273, 2013. doi: 10.1109/STA.2013.6783142.

27. Pang Xinyu. Control system design of double inverted pendulum system based on two-wheeled self-balancing vehicle. Master's thesis, Harbin Normal University, 2020.

28. Maki K. Habib and Samuel A. Ayankoso. Modeling and control of a double inverted pendulum using lqr with parameter optimization through ga and pso. In *2020 21st International Conference on Research and Education in Mechatronics (REM)*, pages 1–6, 2020. doi: 10.1109/REM49740.2020.9313893.

29. Salmiah Ahmad, Muthanna Aminnuddin, and M. A. Sadiq M. Shukor. Modular hybrid control for double-link two-wheeled mobile robot. In *2012 International Conference on Computer and Communication Engineering (ICCCE)*, pages 807–813, 2012. doi: 10.1109/ICCCE.2012.6271329.