

轮圈腿式平衡机器人多模态运动控制研究

摘 要

随着科学技术在移动机器人中的发展，移动机器人的构型设计与控制算法需要随着人类的需求环境变换而不断迭代。在常见的城市及野外的作业环境中，台阶，起伏崎岖地形对移动机器人的结构及控制算法都有较大的挑战。目前针对复杂的地形，已经发展处许多轮式及足式的移动机器人，但受限于结构的复杂度及控制算法的设计，移动机器人往往在移动速度，地形通过能力，续航能力等方面需要取舍。除此之外，移动机器人在意外翻到后能否自我恢复也是评价移动机器人运行稳定性的重要标准。

本文设计的轮圈腿式平衡机器人由机体，一对轮圈腿结构组成，其中轮圈腿结构由内轮和外轮组成。根据结构特点，轮圈腿式平衡机器人可以实现坐姿，站姿，跳跃，翻倒恢复多模态运动。依靠多模态运动的特点，轮圈腿式平衡机器人可以实现在平坦地面，起伏路面，台阶等常见地形运动，拥有较强的地形通过能力。依靠轮圈与机体的结构和重心分布设计，轮圈腿式机器人可以实现在正向及侧向翻倒时的自我恢复，拥有较强的运行稳定性。本文的主要研究内容如下：

(1) 机器人的运动学及动力学分析：本文通过将轮圈腿式平衡机器人分为机体，内轮，外轮三部分，规定轮圈腿式机器人各部分坐标系，通过作运动学和牛顿动力学分析，获得机器人在世界坐标系下的运动学方程及动力学方程。

(2) 设计机器人控制算法及在仿真平台验证：通过计算得到的运动学和动力学方程，计算机器人的状态空间方程，通过设计线性二次规划（LQR）算法获得机器人控制率。在 MATLAB 和 Simscape 仿真环境中验证了得到的动力学方程及状态空间方程的合理性。在 Webots 仿真环境中设计机器人各动作的控制算法，验证机器人各模态运动的合理性。

(3) 搭建实物样机实验平台并进行实验验证：通过对电机，传感器，控制板选型，搭建机器人实物样机实验平台。将控制算法迁移至控制板运算平台，实现实物样机控制。在实验部分，设计了机器人平地行走，交叉原地跳跃，向前跳跃，连续上台阶，侧向翻到恢复实验。验证了机器人的多模态运动性能。最终实验效果为：机器人样机重 9.8kg，平坦地形最大运行速度 4.5m/s，最大原

地跳跃高度 200mm，最大可通过单级台阶 200mm，最大可通过连续台阶高度 100mm。

关键词：双轮平衡机器人，动力学建模，LQR 控制，多模态运行

Research on multimodal motion control of a rim-legged balancing robot

Abstract

With the development of science and technology in mobile robotics, the configuration design and control algorithms of mobile robots need to be iterated with the changing environment of human needs. In the common urban and field operating environments, steps, undulating and rugged terrains pose greater challenges to the structure and control algorithms of mobile robots. Many wheeled and footed mobile robots have been developed for complex terrain, but due to the complexity of the structure and the design of the control algorithms, mobile robots often need to make trade-offs in terms of movement speed, terrain accessibility, and endurance. In addition, the ability of mobile robot to recover itself after accidental overturning is also an important criterion for evaluating the stability of mobile robot operation.

The rim-legged balancing robot designed in this paper consists of a body, a pair of rim-legged structures, where the rim-legged structure consists of inner and outer wheels. According to the structural characteristics, the rim-legged balancing robot can achieve multi-modal motion in sitting, standing, jumping and tipping recovery. Relying on the characteristics of multimodal movement, the rim-legged balanced robot can achieve movement in the common terrain such as flat ground, undulating road, steps, etc., and has a strong ability to pass through the terrain. Relying on the structure and center of gravity distribution design of the wheels and the body, the wheel-legged robot can achieve self-recovery in forward and sideways tipping over, and has strong operational stability. The main research content of this paper is as follows:

(1) Kinematics and dynamics analysis of the robot: In this paper, we divide the wheel-legged balancing robot into three parts: body, inner wheel and outer wheel, specify the coordinate system of each part of the wheel-legged robot, and obtain the kinematics and dynamics equations of the robot under the world coordinate system by making kinematics and Newtonian dynamics analysis.

(2) The design of the robot control algorithm and its subsequent validation in a

simulation platform involves the calculation of the robot's kinematics and dynamics equations, the computation of the robot's state space equations, and the design of the Linear Quadratic Programming (LQR) algorithm to control the robot's rate. The reasonableness of the obtained kinematic and state space equations is verified in both the MATLAB and Simscape simulation environments. The control algorithm for each action of the robot is then designed in the Webots simulation environment to verify the reasonableness of the robot's motion in each mode.

(3) The construction of a physical prototype experimental platform was undertaken, with experimental verification being conducted through the selection of motors, sensors and control boards. The control algorithm was migrated to the control board computing platform, thus enabling the control of the physical prototype. The experimental phase involved the robot being designed to walk on flat ground, cross in situ jumping, forward jumping, continuous up steps, and lateral flip to recovery experiments. The multimodal motion performance of the robot was verified. The final experimental results are as follows: the robot prototype weighs 9.8Kg, the maximum running speed is 4.5m/s in flat terrain, the maximum in-situ jumping height is 200mm, the maximum passable single step is 200mm, and the maximum passable continuous step height is 100mm.

Key words: Two-wheeled balancing robot, Dynamics modelling, LQR control, multimodal operation

第 1 章 绪 论

1.1 课题研究背景及意义

移动机器人是机器人的重要组成之一，在工厂巡检，无人勘探，野外探索等人类常见生活场景中起着重要的作用。执行这些特种任务时，移动机器人往往面临着移动环境危险、复杂、未知等难题。在面对狭小空间场景，例如管道，沟壑类的场景，机器人运行灵活性如转弯半径，移动的自由度决定了机器人能否应对这类地形。在野外或城市场景中，常出现台阶，碎石，杂草等崎岖地形，这对机器人的地形通过能力有较高的要求。在面对外力冲击或在未知环境中意外翻倒后，机器人能否只依赖自身构型恢复到正常运行状态，这对机器人的运行稳定性提出了较大的要求。

陆地移动机器人按照移动机构的常见类型可以划分为轮式移动机器人，足式移动机器人，履带式移动机器人，轮足复合式移动机器人。其中每种构型的移动机构都有其独特的适宜运行的场景。对于轮式移动机器人，其是移动机器人中发展时间最久，技术最成熟的一类构型，也是目前应用最广泛的一类。轮式移动以及的主要优势有运行效率高，平坦地形运行速度快，动作灵活，负载能力强，但是在地形复杂的情况，如遇到台阶等有垂直高度差的地形，传统轮式移动机器人往往无法跨越，需要增添辅助的结构。履带式移动机器人的运动特点有转弯半径小，在沙地泥泞的地形不打滑，越野能力强，负载能力强。但是缺点是机体重量大，速度慢，功耗高。足式机器人是近些年来逐渐成熟的一种移动机构，不同轮式机器人，足式机器人由于其仿生的特点，具有较强的地形跨越能力，尤其是在面对类似于台阶，废墟等传统轮式机器人难以通过的地形，有较好的地形通过能力。但是足式机器人的缺点有运行速度慢，负载能力低，运行效率等。同时在控制足式机器人的算法设计上，往往需要复杂的控制算法维持机器人的平稳运行，对运算平台也有很高的要求。轮足复合式移动机器人是将轮式移动机器人与足式机器人相结合，在相对平坦的地形运行时使用轮式的移动机构，在面对复杂地形情况下发挥足式机器人的优势。这类轮足复合式的机器人往往结构较为复杂，需求的主动电机数量角度，成本较高。

综合以上分析，发现轮式，足式，履带机器人都有其较为擅长的运行环境同时也具有明显的不足，轮足式机器人兼顾的轮式与足式移动机器人优点，但同时存在结构复杂的不足。本论文沿着轮足式移动机器人的思路，设计并提出

一款新型的移动机构，该移动机构可以满足在简单环境下高速灵活运行，在复杂环境中可以克服障碍，续航能力强，运行稳定性高。

1.2 国内外发展现状

近年来为应对日益复杂的环境要求，增强移动机器人的非结构化地形适应性已成为热门研究方向之一。如前文所述，单一运动模式已无法满足工作要求，因此，能提高越障成功率和复杂地形适应能力的复合式移动机器人应运而生。

目前根据复合式机构移动机器人可以分为轮腿式机器人，异性轮式机器人和变形轮式机器人。异性轮式和变形轮式机器人通过改变传统轮式移动机构外形，通过结构特征实现对复杂地形的适应。轮足式机器人保留轮式机器人的外观特征，并在轮式移动机构中串联腿式移动机构，使轮式移动机构拥有更多可移动自由度实现对复杂地形的适应。以下重点介绍上述复合式机构移动机器人的研究发展现状。

1.2.1 国外发展现状

近年来，双轮平衡机器人因其结构简单、运动灵活，广泛应用于移动机器人领域。2016 年，日本广岛大学的研究团队开发了一种双轮平衡机器人，采用一对大型轮子、齿形带和带轮以及机体组成，如图 1.1 所示。该机器人仅通过两个轮子的驱动电机即可实现移动，无需额外的驱动器。当轮子接触到台阶时，机器人通过调整上层机体的重心，实现了攀爬高度为 120mm 至 130mm 的台阶。这一设计提供了一种全新的越障思路，展示了双轮平衡机器人在复杂地形中的应用潜力。

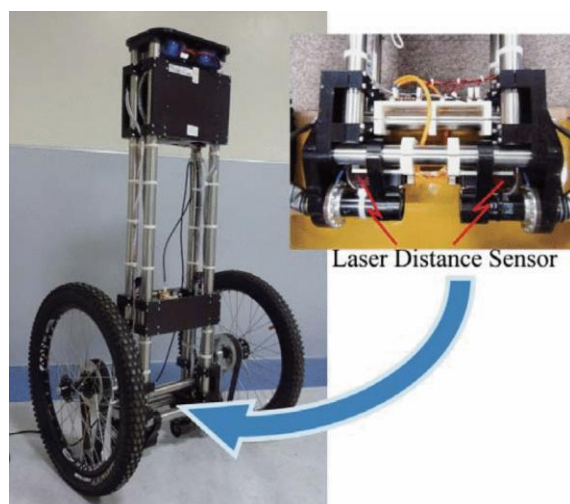


图 1.1 广岛大学楼梯攀爬机器人

Fig. 1.1 Hiroshima University Stair Climbing Robot

瑞士的苏黎世联邦理工学院机器人系统实验室在 2018 年 IEEE ICRA 的会议中提出了一种新型构型的双轮足式平衡机器人 Ascento，这个机器人由四个电机组成，通过拓扑优化的腿部结构配合扭簧组成了结构紧凑的轮足结构。

Ascento 机器人可以通过跳跃姿势完成 100mm 高台阶的跳跃，通过 LQR 控制，这台机器人有很高的平衡稳定性即使在受到外力冲击和翻倒情况下也可以恢复正常运行状态。在 2021 年，Ascento 机器人更新结构，在通过在腿部膝关节和髋关节添加主动电机拓展了机器人腿部的活动自由度，实现了控制机体高度，在更复杂的地形平稳运行以及在多种翻倒的姿势下恢复运行的功能。如图 1.2 所示。



(a) 2018 年 Ascento 样机



(b) 2021 年 Ascento 样机

图 1.2 苏黎世联邦理工学院 Ascento 机器人两代样机

Fig. 1.2 Two generations of prototype Ascento robots at ETH Zurich

苏黎世联邦理工大学的机器人系统实验室在 2019 首次推出 ANYmal 四轮

足机器人，该机器人将传统四足机器人的行走机构替换为轮足机构，通过 ZMP 算法和 WBC 控制算法实现对该机器人的控制。最终达到最大运行速度 4m/s，相较四足机器人降低了 83% 的行走成本，验证了轮足机器人在行走方面相较于四足机器人的优越性。如图 1.3 所示。在 2023 年的 IEEE ICRA 会议中提出使用强化学习算法训练出的四轮足机器人，该机器人通过在仿真环境中的学习策略可以在现实环境中做到跨越复杂地形，躲避障碍物，规划路线，在各种运行模式中实现自主切换。四轮足机器人相较于四足机器人，在行走机构上通过更高效的轮足机构代替足式行走机构，提高了运行速度以及运行效率，同时在面对障碍环境，四轮足可以发挥足式机器人高自由度的优势实现地形的跨越。



图 1.3 苏黎世联邦理工大学 ANYmal 四轮足机器人

Fig. 1.3 ANYmal quadrupedal foot robot at ETH Zurich

1.2.2 国内发展现状

重庆大学机械传动国家重点实验室 2019 年研制的 LDR 被动或主动变形轮腿移动机器人。该轮腿机器人最大的特点是，可以被动变形，这种转换是由轮辋与障碍物之间的摩擦力被动触发，没有任何执行装置。但是被动模式切换机构高度依赖于轮辋与接触面之间的摩擦力，变形成功率低。所以又加入了主动变形的执行机构，作为补充。变形机构是通过内辐条与外辐条相对位置的变化，车轮会呈现不同的构型。增加车轮的直径，跨越圆轮状态下跨越不了的障碍。该轮腿机器人可以攀爬高于车轮半径的 2.8 倍的障碍高度。如图 1.4 所示。

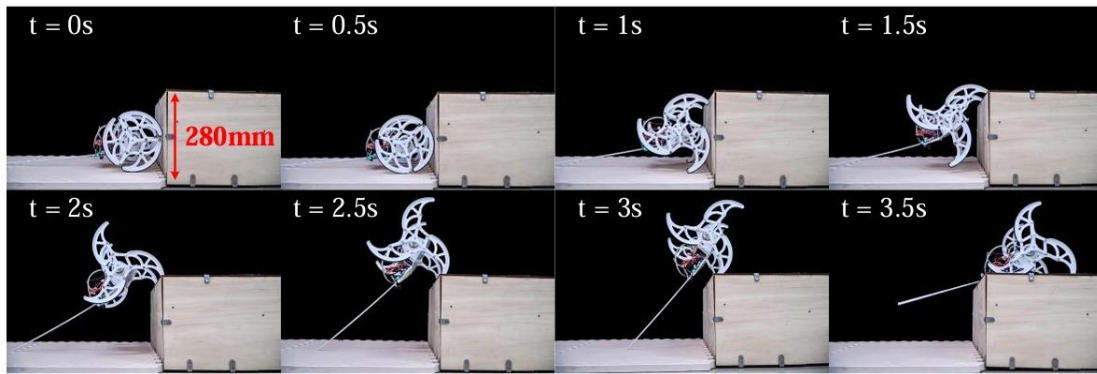


图 1.4 重庆大学被动或主动变形轮腿移动机器人

Fig. 1.4 Passive or active deformation of wheel-legged mobile robots at Chongqing University

由国内腾讯公司 Robotics X 部门在 2021 研发的并联双轮平衡机器人 Ollie 实现了机器人从初始角度 36° 的形态恢复并以误差在 1° 内稳定在水平姿态，可以通过控制并联腿部关节实现机体高度从 0.37m 到 0.7m 之间的变化。该机器人通过使用 LQR 控制算法和 IDA-PBC 的非线性控制算法对机器人的直立以及抗干扰能力做控制，并在 2023 年将算法迭代为使用 WBC 算法对机器人进行控制，实现了机器人空翻，上下台阶等动作。如图 1.5 所示。



图 1.5 腾讯公司的 Ollie 双轮足机器人

Fig. 1.5 Tencent's Ollie bi-pedal robot

燕山大学在 2024 年研发的一种轮足结构一体化复合式移动机器人，该机器人的移动通过融合折扇机构与剪叉机构设计一种可收折轮式结构，并通过螺旋理论对腿部结构进行设计，将两种机构整合，完成对变形轮机构的设计。如图 1.6 所示。该构型机器人通过实验在轮形移动机构状态下可以达到 2.19m/s 的运行速度，在轮形态下可以通过 100mm 高的台阶，在足形态下可以通过 150mm 高的台阶，在沙地，冰面，腐叶地，草地等复杂环境中均进行测试并且

轮式和足式模态均可通过。

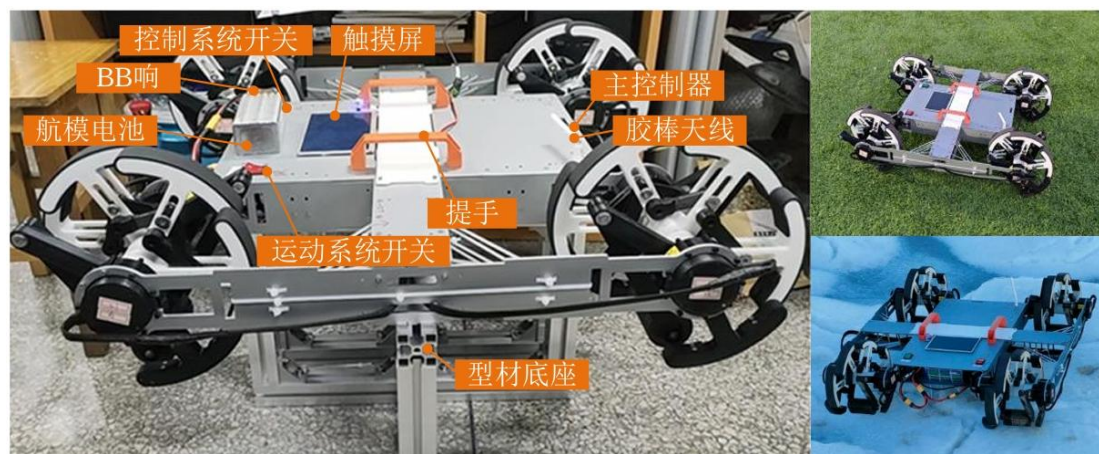


图 1.6 燕山大学轮足结构一体化复合式移动机器人二代样机

Fig. 1.6 Yanshan University Wheel-foot structure integrated composite mobile robot II prototype

1.3 主要研究内容

通过分析现有移动机器人结构优势，设计一款轮圈腿式双轮平衡机器人。该结构具有结构简单，需要的电机数量少，运动敏捷灵活，越障能力强，稳定性强的特点。本文的主要内容是针对轮圈腿式机器人进行系统分析并设计控制器，搭建样机平台并实验验证机器人性能。具体研究内容如下：

(1) 对轮圈腿式机器人作系统分析。通过对轮圈腿式移动结构作结构抽象，建立机体坐标系与世界坐标系，在约定坐标系下作运动学分析，分析各电机转动与机器人机体重心在世界坐标系下的运动学关系。使用牛顿动力学对机器人各抽象部分作动力学分析，得到各电机驱动力与机器人重心在世界坐标系下的加速度关系。通过联立运动学方程与动力学方程得到机器人状态空间方程。

(2) 设计机器人运动控制算法并在仿真平台中验证。通过已有的状态空间方程设计线性二次规划算法（LQR）对选定的状态变量进行控制。在 MATLAB 中使用 LQR 求解工具获得反馈增益矩阵，在 Simscape 中搭建仿真环境，通过与 MATLAB 求解器的配合验证 LQR 算法的可行性与模型的正确性。通过设计轮圈腿式机器人坐姿，站姿，跳跃，上台阶，翻倒恢复各个状态的控制算法，并在 Webots 仿真软件中搭建带有复杂场景的仿真环境，验证各个模态下的控制算法的合理性。

(3) 搭建实物样机平台。根据仿真以及理论分析数据分析各电机所需性能做电机选型。设计编码器传感器用于反馈机器人关节角度与角速度。选择合适的

控制板并在控制板中编写机器人控制框架。通过上位机与下位机实现机器人无线调试与数据反馈，完成实物样机的搭建。

(4) 完成实物样机实验并做分析。在实验部分根据机器人设计的场景，对机器人坐姿，站姿，跳跃，翻倒恢复四个模态进行实验，验证机器人多模态运动的性能。并将机器人置于水泥地面，草地，斜坡等不同的环境中验证机器人控制算法的鲁棒性。通过实验数据分析当前机器人控制算法的性能并提出改进方向。

第 2 章 轮圈腿式机器人系统分析

2.1 轮圈腿式机器人结构设计

该结构由一对轮圈腿和一个主躯干组成、轮圈腿分布在躯干两侧。轮圈腿由一个内轮和一个外轮组成、其中内轮与躯干相连，可沿躯干内侧轴向移动可沿外轮内侧轴向移动。有四个驱动两个内轮俯仰电机和两个外轮驱动电机。两个外轮驱动电机。电机分布在电机分布在躯干上，通过轴系统设计将扭矩传递到内轮和外轮。



图 2.1 轮圈腿式机器人样机三维模型图

Fig. 2.1 3D model drawing of a prototype robot

该结构可通过驱动齿轮和外轮的配合，驱动整个机器人移动和行走。通过

驱动齿轮和外轮的配合，该结构可驱动整个机器人移动和行走，俯仰轴电机通过链传动使躯干绕驱动电机轴旋转，在机器人运行时保持躯干角度。通过驱动电机和变桨电机的配合，可以改变整个机器人的重心，从而实现坐、站、卧、下坡整体翻滚、上台阶、跳跃、双腿奔跑、两圈腿不在同一轴上运动、侧翻恢复等多模式动作。

机器人由一个躯干和固定在躯干上的两个圆环组成。轮圈腿的轴系与车轮的中心旋转轴相距 160 毫米，使躯干能够相对于外轮做半径为 160 毫米的圆周运动。在保证结构强度的前提下，对车轮内圈进行大面积镂空以减轻重量，并将所有驱动电机和电子元件置于躯干内，当机器人处于坐姿运动情况下，当机器人收到外部冲击时，机体可以被完全置于轮内，使机体免受外界冲击。

轮圈腿式平衡机器人共有四个电机分别驱动机体的俯仰运动和外轮的旋转运动。俯仰电机通过驱动链轮 1，链轮 1 与链轮 2 之间采用链传动，使机体绕旋转轴运动，通过控制俯仰电机，可以控制机体的俯仰角。外轮驱动电机通过轴将扭矩传递到外轮的内齿轮中，外轮通过一组齿轮内啮合驱动外轮圈转动，为机器人提供前进驱动力。

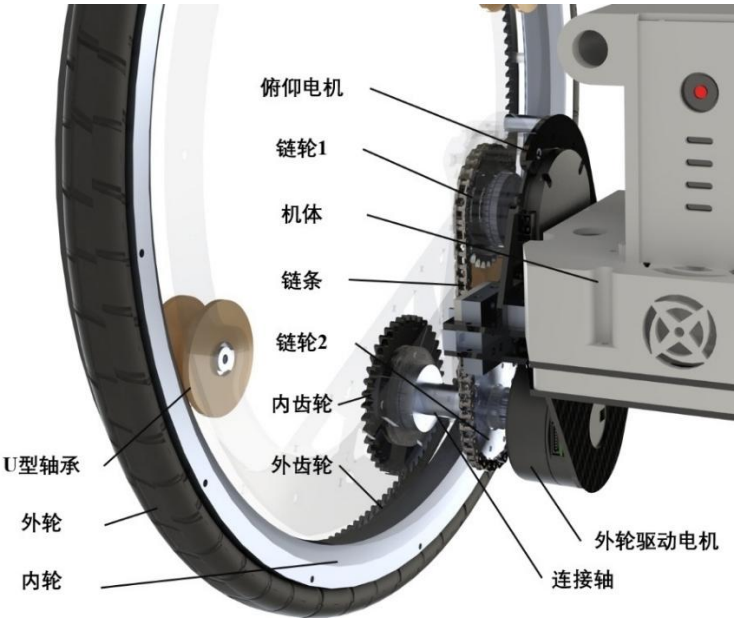


图 2.2 机器人传动示意图

Fig. 2.2 Schematic diagram of robot drive

2.2 轮圈腿式机器人运动学分析

由于机器人内部传动较为复杂，在进行机器人运动学及动力学分析的过程

中，通过将机器人抽象简化，机器人可以分为外轮，内轮，机体三部分。内外轮的齿轮配合可抽象为一组连杆，方便对整体做受力分析。

为实现轮圈腿式平衡机器人多模态运动的理论建模，本研究选择在 xoz 平面内建立二维运动学模型。该简化基于以下假设：

- (1) 机器人对称结构在 y 轴方向无显著动力学耦合。
- (2) 横滚轴（Roll）运动通过独立 PD 控制器实现稳定。
- (3) 地面接触点摩擦力满足非完整约束条件。

本文定义状态空间方程的状态变量为机器人沿前进正方向 x 的位移为 s ，速度为 \dot{s} ，机器人绕竖直轴 z 的旋转角度 ω ，旋转角速度 $\dot{\omega}$ ，两侧内轮的抬升角度 θ_{bl} 、 θ_{br} ，抬升角速度 $\dot{\theta}_{bl}$ 、 $\dot{\theta}_{br}$ ，机体绕旋转轴的俯仰角度 φ ，角速度 $\dot{\varphi}$ 。各变量的正方向均在图中表示。各状态变量写作向量形式为：

$x = [s \quad \dot{s} \quad \omega \quad \dot{\omega} \quad \theta_{bl} \quad \dot{\theta}_{bl} \quad \theta_{br} \quad \dot{\theta}_{br} \quad \varphi \quad \dot{\varphi}]^T$ 。机器人的外部输入为四个电机的扭矩，写作向量形式为： $u = [\tau_{wl} \quad \tau_{wr} \quad \tau_{bl} \quad \tau_{br}]^T$ 。机器人各物理量及坐标系表示如图所示。

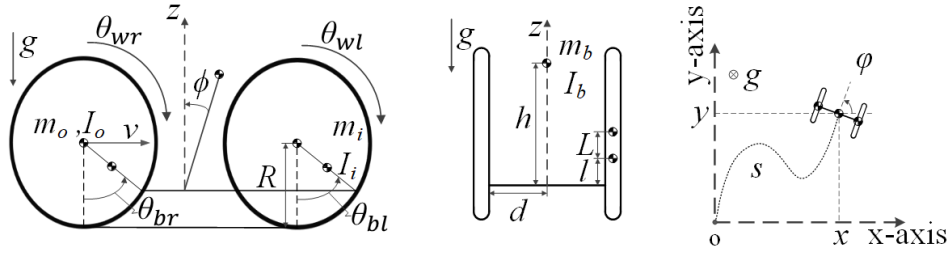


图 2.3 机器人机体坐标系和世界坐标系定义及各物理量符号定义

Fig. 2.3 Definition of the robot body coordinate system and the world coordinate system and definition of the symbols of each physical quantity

图 2.3 中各物理量的具体意义为下表所示：

表 2.1 机器人表达式中各物理量的含义

Table 2.1 Meaning of each physical quantity in the robot expression

物理量	含义
g	重力加速度
m_b	中心机体质量
m_o	外轮质量
m_i	内轮质量
I_o	外轮绕 y 轴旋转惯量
I_i	内轮绕 y 轴旋转惯量
I_b	中心机体绕 y 轴旋转惯量
h	中心机体重心到内轮旋转轴的距离
L	内轮质心到外轮旋转中心的距离
l	内轮质心到内轮旋转轴的距离
d	外轮中心到机体中轴线的距离
R	外轮半径

对于外轮，假设外轮始终与地面接触，且接触点为 $z=0$ 点，对于外轮中心点：

$$x_o = R \cdot \theta_w \quad (2.1)$$

$$z_o = R \quad (2.2)$$

式中 x_o 为外轮中心点在 x 轴方向上的位移， z_o 为外轮中心点在 z 轴方向上的位移， R 为外轮的半径。

对于内轮：

$$x_i = x_o + (R - l) \cdot \sin(\theta_l) \quad (2.3)$$

$$z_i = (R - l) \cdot \cos(\theta_l) \quad (2.4)$$

式中 x_i 为内轮质心点在 x 轴方向上的位移， z_i 为内轮质心点在 z 轴方向上的位移， l 为内轮质心到内轮圆心的距离。

对于机体：

$$x_b = \frac{1}{2}(x_{il} + x_{ir}) - h \cdot \sin(\phi) \quad (2.5)$$

$$z_b = \frac{1}{2}(z_{il} + z_{ir}) - h \cdot \cos(\phi) \quad (2.6)$$

式中 x_b 为机体质心点在 x 轴方向上的位移， z_b 为机体质心点在 z 轴方向上的位移， h 为机体质心到旋转轴的距离。

s 为两轮与地面接触点之间连线的中点，运动学表示为：

$$s = \frac{1}{2}(x_{ol} + x_{or}) \quad (2.7)$$

机器人绕竖直轴 z 的旋转角度 ω 的运动学可表示为：

$$\omega = \frac{1}{2} \frac{x_{or} - x_{ol}}{d} \quad (2.8)$$

式中 d 为两轮轮距的一半。

根据式(2.1)-(2.6)可得机体质心坐标关于各个关节位置表达式为：

$$x_b = \frac{1}{2}[R \cdot (\theta_{wl} + \theta_{wr}) + (R - l) \cdot (\sin(\theta_{bl}) + \sin(\theta_{br})) - h \cdot \sin(\phi)] \quad (2.9)$$

$$z_b = \frac{1}{2}(R - l) \cdot [\cos(\theta_{bl}) + \cos(\theta_{br})] - h \cdot \cos(\phi) \quad (2.10)$$

将式(2.9)(2.10)微分可得质心速度关于各关节转动速度的关系：

$$\dot{x}_b = \frac{1}{2}[R \cdot (\dot{\theta}_{wl} + \dot{\theta}_{wr}) + (R - l) \cdot (\cos(\theta_{bl}) \cdot \dot{\theta}_{bl} + \cos(\theta_{br}) \cdot \dot{\theta}_{br})] - h \cdot \cos(\phi) \cdot \dot{\phi} \quad (2.11)$$

$$\dot{z}_b = \frac{1}{2}(R - l)[- \sin(\theta_{bl}) \cdot \dot{\theta}_{bl} - \sin(\theta_{br}) \cdot \dot{\theta}_{br}] + h \cdot \sin \phi \cdot \dot{\phi} \quad (2.12)$$

2.3 轮圈腿式机器人动力学分析

在对于轮圈腿式平衡机器人，本文在动力学分析中采用牛顿动力学方法。选择牛顿动力学作为分析方法，主要是因为其物理直观性强，能够直接描述机器人系统中力、加速度和质量之间的关系。这种直观性有助于清晰地理解系统的动力学行为，并为后续的控制策略设计提供基础。

因此将机器人分为外轮，内轮，机体三部分并做受力分析，如图所示。

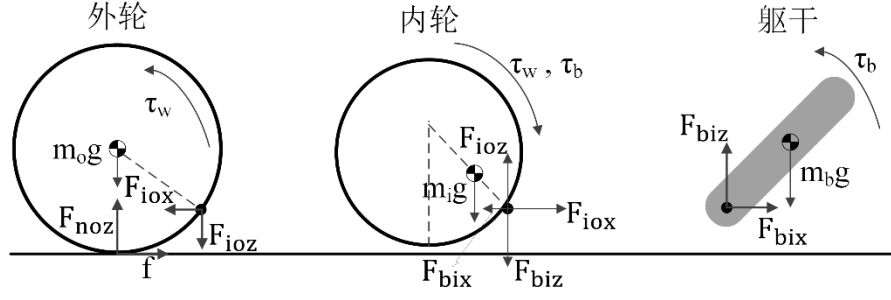


图 2.4 机器人外轮、内轮、躯干三部分受力分析图

Fig. 2.4 Force analysis diagram of the three parts of the robot: outer wheel, inner wheel and torso

对外轮做受力分析：

$$F_{noz} = F_{ioz} + m_o \cdot g \quad (2.13)$$

$$m_o \cdot R \cdot \ddot{\theta}_w = f - F_{iox} \quad (2.14)$$

$$I_o \cdot \ddot{\theta}_w = f \cdot R - F_{noz} \cdot L \cdot \sin(\theta_l) - F_{ioz} \cdot L \cdot \cos(\theta_l) + \tau_w \quad (2.15)$$

其中 m_o 为外轮质量， I_o 为外轮绕旋转中心的转动惯量。

对内轮做受力分析：

$$m_i \cdot \ddot{x}_i = F_{iox} - F_{bix} \quad (2.16)$$

$$m_i \cdot \ddot{z}_i = F_{ioz} - F_{biz} - m_i \cdot g \quad (2.17)$$

$$I_i \cdot \ddot{\theta}_l = (F_{iox} - F_{bix})l \cos(\theta_l) + (F_{ioz} - F_{biz})l \sin(\theta_l) - (\tau_w + \tau_b) \quad (2.18)$$

其中 m_i 为内轮质量， I_i 为内轮绕质心旋转的转动惯量。

对机体做受力分析：

$$m_b \cdot \ddot{x}_b = F_{bixl} + F_{bixr} \quad (2.19)$$

$$m_b \cdot \ddot{z}_b = F_{bizl} + F_{bizr} - m_b \cdot g \quad (2.20)$$

$$I_b \cdot \ddot{\varphi} = (F_{bixl} + F_{bixr})h \cos(\varphi) + (F_{bizl} + F_{bizr})h \sin(\varphi) + (\tau_w + \tau_b) \quad (2.21)$$

其中 m_b 为机体质量， I_b 为机体绕质心选择的转动惯量。

因为假设机体仅在 xoz 平面内运动，可以在机体上得到约束关系：

$$F_{bizl} = F_{bizr} \quad (2.22)$$

第 3 章 控制算法设计及仿真分析

3.1 系统可控性理论验证

在设计系统的控制算法前，例如本机器人所使用的线性二次型规划器（LQR）算法时，需要先验证系统的可控性。只有当系统在线性化平衡点附近是可控的，本文才能利用状态反馈实现对系统设定状态的调节。

3.1.1 系统线性化

在实际机器人运动控制中，轮圈腿式平衡机器人的运动学和动力学模型通常是非线性的，这主要由于存在多自由度、复杂的连杆传动以及非线性受力关系。为了设计有效的状态反馈控制器，需要在平衡点附近对系统进行线性化，从而获得便于分析和设计的线性状态空间模型。根据上文所求得系统的动态特性方程，其一般形式为：

$$\dot{x} = f(x, u) \quad (3.1)$$

其中 $x \in \mathbb{R}^{10}$ 为状态向量，包含机器人沿前进方向的位置、速度、旋转角度及角速度、内轮抬升角及其角速度、机体俯仰角及其角速度等变量； $u \in \mathbb{R}^4$ 为输入向量，对应各电机施加的扭矩。函数 $f(\cdot)$ 则包含机器人运动学、动力学的非线性耦合关系和非线性受力项。

为简化非线性的物理模型，在机器人平衡点附近进行线性化泰勒展开，对于轮圈腿式平衡机器人，机器人在大多运行状态下以坐姿前进为主，如图所示姿势：



图 3.1 机器人坐姿零位置状态

Fig. 3.1 Robot sitting zero position state

以该位置设定为平衡点 (x_0, u_0) 处满足：

$$f(x_0, u_0) = 0 \quad (3.2)$$

这表示在该平衡状态下系统处于静止或稳态工作状态。接下来对 $f(x, u)$ 在该平衡点进行一阶泰勒展开，有：

$$f(x, u) \approx f(x_0, u_0) + \left. \frac{\partial f}{\partial x} \right|_{(x_0, u_0)} (x - x_0) + \left. \frac{\partial f}{\partial u} \right|_{(x_0, u_0)} (u - u_0) \quad (3.3)$$

由于 $f(x_0, u_0) = 0$ ，令状态误差 $\delta x = x - x_0$ 和输入误差 $\delta u = u - u_0$ ，则可得近似的线性化模型：

$$\delta \dot{x} = A \delta x + B \delta u \quad (3.4)$$

其中状态矩阵 A 和输入矩阵 B 分别定义为：

$$A = \left. \frac{\partial f}{\partial x} \right|_{(x_0, u_0)}, B = \left. \frac{\partial f}{\partial u} \right|_{(x_0, u_0)} \quad (3.5)$$

将(2.11)(2.12)得到的表达式按上述方法线性化后可得：

$$\dot{x}_b = \frac{1}{2} [R \cdot (\dot{\theta}_{wl} + \dot{\theta}_{wr}) + (R - l) \cdot (\dot{\theta}_{bl} + \dot{\theta}_{br})] - h \cdot \dot{\phi} \quad (3.6)$$

$$\dot{z}_b = \frac{1}{2} (R - l) [-(\theta_{bl} \cdot \dot{\theta}_{bl} - \theta_{br} \cdot \dot{\theta}_{br})] + h \cdot \phi \cdot \dot{\phi} \quad (3.7)$$

其中由于 $\theta_{bl} \cdot \dot{\theta}_{bl}$ ， $\theta_{br} \cdot \dot{\theta}_{br}$ 和 $\phi \cdot \dot{\phi}$ 为两个小量的乘积，属于二阶小量根据

一阶线性化原则可以忽略，因此可得：

$$\dot{z}_b = 0 \quad (3.8)$$

同理将动力学公式(2.15)(2.18)(2.21)线性化可得：

$$I_o \ddot{\theta}_w = f \cdot R - F_{iox} \cdot L \cdot \theta_l - F_{ioz} \cdot L + \tau_w \quad (3.9)$$

$$I_i \cdot \ddot{\theta}_l = (F_{iox} - F_{bix}) \cdot l - (F_{ioz} - F_{biz}) \cdot l \cdot \theta_l - (\tau_w + \tau_b) \quad (3.10)$$

$$I_b \ddot{\phi} = (F_{bix} \cdot l + F_{bix} \cdot r) \cdot h - (F_{biz} \cdot l + F_{biz} \cdot r) \cdot h \cdot \phi + (\tau_w + \tau_b) \quad (3.11)$$

这种线性化处理将原本非线性的系统在平衡点附近近似为线性系统，从而便于利用线性控制理论进行控制器设计和性能分析。

3.1.2 系统状态空间方程

通过对机器人平衡点位置线性展开，可以通过状态空间方程的形式表达系统的动态特性，其一般形式为：

$$\begin{cases} \dot{x} = Ax + Bu \\ y = Cx + Du \end{cases} \quad (3.12)$$

其中 y 是系统的输出向量，表示可观测得到的输出， $A \in \mathbb{R}^{10 \times 10}$ 为系统矩阵，刻画了系统状态变量之间的线性耦合关系以及系统的动态特性； $B \in \mathbb{R}^{10 \times 4}$ 为输入矩阵，反映了输入向量 u 如何作用在各状态变量上； C 与 D 分别为输出矩阵和传递矩阵，由于在此部分只做系统的动态响应验证，与这两个矩阵无关，则取 $C = I^{10}$ ， $D = 0$ 。

通过联立式(2.13)-(2.22)十个方程，其中将(2.28)-(2.33)带入可求得十个未知数 $x = [s \quad \dot{s} \quad \omega \quad \dot{\omega} \quad \theta_{bl} \quad \dot{\theta}_{bl} \quad \theta_{br} \quad \dot{\theta}_{br} \quad \phi \quad \dot{\phi}]^T$ 的线性状态空间方程，其形式如式(3.13)所示。

$$\begin{bmatrix} \dot{s} \\ \ddot{s} \\ \dot{\omega} \\ \ddot{\omega} \\ \dot{\theta}_{bl} \\ \ddot{\theta}_{bl} \\ \dot{\theta}_{br} \\ \ddot{\theta}_{br} \\ \dot{\varphi} \\ \ddot{\varphi} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ A_{21} & 0 & A_{23} & 0 & A_{25} & 0 & A_{27} & 0 & A_{29} & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ A_{41} & 0 & A_{43} & 0 & A_{45} & 0 & A_{47} & 0 & A_{49} & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ A_{61} & 0 & A_{63} & 0 & A_{65} & 0 & A_{67} & 0 & A_{69} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ A_{81} & 0 & A_{83} & 0 & A_{85} & 0 & A_{87} & 0 & A_{89} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ A_{101} & 0 & A_{103} & 0 & A_{105} & 0 & A_{107} & 0 & A_{109} & 0 \end{bmatrix} \begin{bmatrix} s \\ \dot{s} \\ \omega \\ \dot{\omega} \\ \theta_{bl} \\ \dot{\theta}_{bl} \\ \theta_{br} \\ \dot{\theta}_{br} \\ \varphi \\ \dot{\varphi} \end{bmatrix} + \begin{bmatrix} 0 & 0 & 0 & 0 \\ B_{21} & B_{22} & B_{23} & B_{24} \\ 0 & 0 & 0 & 0 \\ B_{41} & B_{42} & B_{43} & B_{44} \\ 0 & 0 & 0 & 0 \\ B_{61} & B_{62} & B_{63} & B_{64} \\ 0 & 0 & 0 & 0 \\ B_{81} & B_{82} & B_{83} & B_{84} \\ 0 & 0 & 0 & 0 \\ B_{101} & B_{102} & B_{103} & B_{104} \end{bmatrix} \begin{bmatrix} \tau_{wl} \\ \tau_{wr} \\ \tau_{bl} \\ \tau_{br} \end{bmatrix} \quad (3.13)$$

将机器人的样机物理参数带入公式中，其中各物理量参数大小如表 3.1 所示。

表 3.1 机器人物理参数计算值

Table 3.1 Calculated values of physical parameters of the robot

物理量	数值
g	9.81 m/s^2
m_b	4.20 kg
m_o	0.67 kg
m_i	1.02 kg
I_o	0.03 kg
I_i	$0.03 \text{ kg} \cdot \text{m}^2$
I_b	$0.02 \text{ kg} \cdot \text{m}^2$
h	0.07 m
L	0.16 m
d	0.18 m
R	0.23 m

经计算在坐姿零位线性展开，状态矩阵 A 与输入矩阵 B 的数值解分别为：

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.911 & 0 & 0.413 & 0 & -2.481 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.413 & 0 & 0.911 & 0 & -2.481 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 30.117 & 0 & 1.163 & 0 & -6.978 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1.163 & 0 & 30.117 & 0 & -6.978 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 16.043 & 0 & 16.043 & 0 & 97.356 & 0 \end{bmatrix} \quad (3.14)$$

$$B = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 8.492 & 0.437 & 0.449 & -0.543 \\ 0 & 0 & 0 & 0 \\ 0.437 & 8.492 & -0.543 & 0.449 \\ 0 & 0 & 0 & 0 \\ 78.886 & 1.229 & 56.269 & -1.528 \\ 0 & 0 & 0 & 0 \\ 1.229 & 78.886 & -1.528 & 56.269 \\ 0 & 0 & 0 & 0 \\ 16.952 & 16.952 & 85.769 & 85.769 \end{bmatrix} \quad (3.15)$$

3.1.3 系统可控性分析

在控制理论中，可控性描述了通过合适的控制输入，能否在有限时间内将系统状态从任一初始状态转移到任一指定目标状态。直观上，若一个系统是可控的，则说明系统的所有状态都能通过设计输入加以调节，从而实现预期的控制目标。针对线性时不变系统，R.E.Kalman 提出了判断系统可控性的判据，其基本思想是利用系统的状态矩阵 A 与输入矩阵 B 构造一个可控性矩阵，通过判断该矩阵的秩是否达到状态维数 n 来确定系统是否可控。定义系统的可控矩阵为：

$$c = [A \quad AB \quad A^2B \quad \cdots \quad A^{n-1}B] \quad (3.16)$$

将上文求得的 A 、 B 式(3.14)和式(3.15)带入式(3.16)，Kalman 可控性判据指出，系统 $\dot{x} = Ax + Bu$ 是可控的当且仅当可控性矩阵 c 满秩，也就是：

$$\text{rank}(c) = 10 \quad (3.17)$$

在 MATLAB 中，利用奇异值分解（SVD），将矩阵分解为奇异值，并绘制奇异值曲线图，如错误!未找到引用源。所示。X 轴为奇异值的索引，Y 轴为奇异值的大小，若所有奇异值均明显大于零，表明该矩阵是满秩的。计算将所求得的状态矩阵检验后满足可控性判据，即该系统为可控系统。

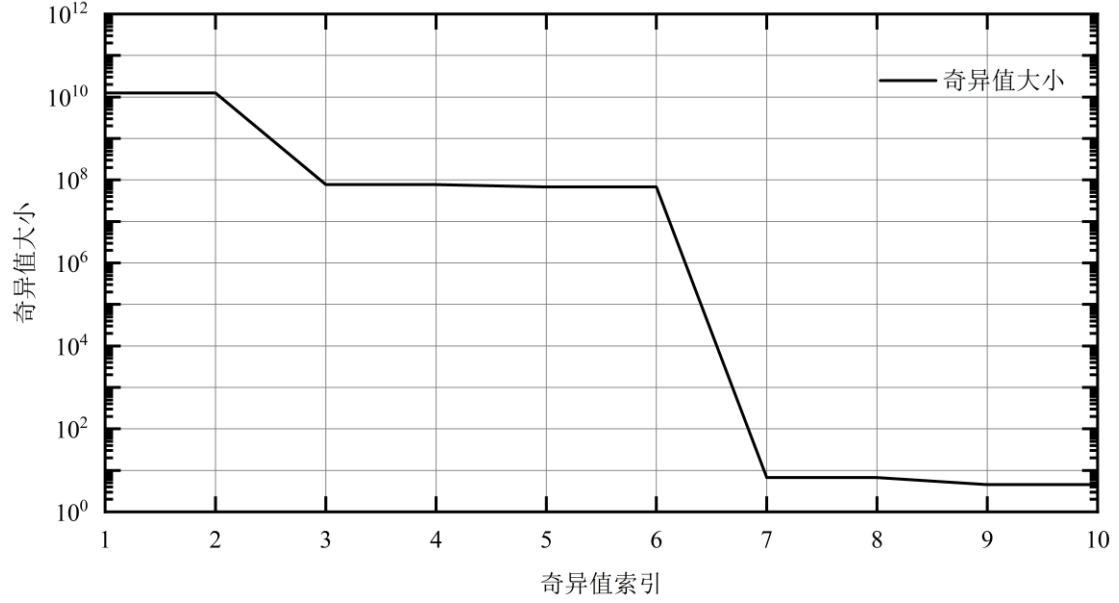


图 3.2 可控矩阵 SVD 图

Fig. 3.2 Controllable Matrix SVD Diagram

3.2 平衡算法设计

3.2.1 线性二次规划器设计

由于轮圈腿式机器人在空间中具有六自由度，而主动控制电机只有 4 个，属于欠驱模型，在控制过程中，需要对多输入多输出的模型进行控制。选择线性二次型调节器，可以处理多变量控制问题，适合于有多个状态变量和输入变量的复杂系统。它通过状态反馈实现对全局状态的优化，能够同时控制多个状态变量，如位置、速度、角度等。同时，LQR 通过使用精确的系统模型，能够更加精确地调节系统状态，尤其是在欠驱系统中，能够考虑到系统的复杂性，提供更全面的控制。在系统优化方面，通过最小化代价函数（通常是状态误差和控制能量的加权和）来实现全局优化。它能确保控制系统在满足控制性能的同时，尽量减少控制能量的消耗，特别适合控制输入受限的欠驱系统。

LQR 的目标是找一组最优控制律 u ，使得系统达到稳定状态，并且对于代价函数：

$$J = \frac{1}{2} \int_0^{\infty} (x^T Q x + u^T R u) dt \quad (3.18)$$

其中， $Q \geq 0$ ，是状态权重矩阵，反应了对状态偏离零的惩罚。 $R \geq 0$ ，

是控制权重矩阵，反映了对使用控制输入的惩罚。为了最小化代价函数 J ，本文引入动态规划中的 Hamilton-Jacobi-Bellman (HJB) 方程：

$$\frac{dV}{dt} + x^T Q x + u^T R u = 0 \quad (3.19)$$

由于 $V(x)$ 通常选取二次型形式：

$$V(x) = x^T P x \quad (3.20)$$

根据系统的动态方程 $\dot{x} = Ax + Bu$ 和式(3.20)带入(3.19)得：

$$x^T A^T P x + u^T B^T P x + x^T P A x + x^T P B u + x^T Q x + u^T R u = 0 \quad (3.21)$$

为了最小化该方程，对 u 求导求得：

$$u = -R^{-1} B^T P x \quad (3.22)$$

定义反馈增益矩阵 K ：

$$K = R^{-1} B^T P \quad (3.23)$$

为了求解反馈增益矩阵 K ，需要求解矩阵 P 。通过将最优控制律代入状态方程和代价函数，可以得到矩阵黎卡提方程（ARE, Algebraic Riccati Equation）：

$$A^T P + P A - P B R^{-1} B^T P + Q = 0 \quad (3.24)$$

通过数值求解方法求解 P ，即可得到最优控制律。

在平衡控制中，本文将平衡点设置为机体的 Pitch 轴倾角 φ 为 0，机体重心在两侧外轮与地面接触点连线上，此时机器人可以维持静止平衡。在移动过程中，由于在调节 Q 矩阵时，机体的 Pitch 轴倾角 φ 的误差权重大于内轮抬升角，当有速度指令时，内轮会向速度方向抬升，为机器人整体提供方向加速度达到目标速度，而 φ 角在此过程中会始终使机体保持竖直状态。

3.2.2 水平轴控制器设计

在机器人动力学建模中，通常只考虑了 Pitch 轴（俯仰轴）和 Yaw 轴（偏航角）方向的控制。而对于机器人 Roll 轴（横滚轴）的控制，则需要设计单独的控制器。为此，本文采用了 PD（比例-微分）反馈控制器，该控制器用于平行于 LQR 平衡控制系统，独立地控制机器人 Roll 轴角度，以维持机器人在横滚轴方向的平衡。在实际应用中，尤其是在从坐姿过渡到站姿的过程中，机器

人横滚轴的稳定性对整体稳定性有着显著影响。因此，控制横滚轴的平衡是确保机器人稳定运行的关键。为此，使用 PD 控制器来计算横滚轴的平衡力矩，具体公式为：

$$\tau_{roll} = K_p \phi + K_d \dot{\phi} \quad (3.25)$$

其中 K_p 和 K_d 分别为比例增益和微分增益， ϕ 和 $\dot{\phi}$ 分别为横滚轴角度和横滚轴角速度。通过调整这些增益，可以灵活控制横滚轴的角度偏差和角速度，确保机器人保持平稳的横滚状态。在实际控制过程中，PD 控制器计算出的横滚轴平衡力矩将与 LQR 控制器计算出的内轮力矩叠加。具体来说，总的控制力矩为：

$$\tau_{total} = \tau_{LQR} + \tau_{roll} \quad (3.26)$$

其中 τ_{LQR} 是 LQR 控制器计算出的内轮力矩，旨在维持机器人在坐姿或站姿下的平衡。通过这种叠加方式，机器人可以保持两侧内轮的高度一致性，确保机器人横滚轴的平衡，从而提升整体稳定性。通过使用这种组合控制方法，机器人能够在运动过程中，特别是在从坐姿过渡到站姿时，有效地控制横滚轴，避免由于横滚角度的变化而导致机器人失去平衡。

3.2.3 重心补偿控制器设计

在双轮机器人移动转向过程中，由于向心加速度的作用，机器人整体的受力平衡点发生了变化，且该平衡点不再是固定在机器人中心，而是随着机器人前进速度和转向角速度的大小与方向的变化而移动。特别是在较高的转向角速度和前进速度下，向心加速度的作用变得显著，可能导致机器人产生翻倒的风险。因此，考虑参考零力矩点（ZMP）的思路成为控制机器人动态稳定性的重要方法，尤其是在转向和动态行走过程中。

ZMP（Zero Moment Point，零力矩点）是指在机器人与地面接触的特定地点，在该点处机器人受到的重力、惯性力和地面反作用力等所有外力和力矩之和为零。ZMP 通常用于足式机器人运动控制中，它为保证机器人稳定性提供了一个直观的参考点。具体来说，ZMP 是机器人重心与地面反作用力的合力矩的平衡点。

对于轮圈腿式机器人，在转向过程中受力情况为，机器人的重力 $F_{gravity}$ ，转向产生的惯性力 $F_{inertial}$ ，地面的反作用力 $F_{reaction}$ 。ZMP 的定义为：

$$\sum M_{ZMP} = 0 \quad (3.27)$$

假设机器人的重心位置为 r_{CG} ，并机器人转向半径为 R 和角速度 $\dot{\theta}$ ，当机器人转向时机器人会受到向心加速度 $a_{centripetal}$ ，而这个加速度产生的惯性力作用在机器人的重心上。向心加速度的计算公式为：

$$a_{centripetal} = R\dot{\theta}^2 \quad (3.28)$$

惯性力的计算公式为：

$$F_{inertial} = ma_{centripetal} = mR\dot{\theta}^2 \quad (3.29)$$

其中 m 为机器人质量。

ZMP 位置由机器人所受的合力矩决定。为了保证 ZMP 点的平衡，重力、惯性力和地面反作用力对 ZMP 点的力矩之和必须为零。通过力矩平衡的关系，本文可以得到 ZMP 点的计算公式：

$$r_{ZMP} = \frac{F_{reaction} \cdot d}{F_{gravity} + F_{inertial}} \quad (3.30)$$

其中 d 是从机器人重心到 ZMP 点的水平距离。

机器人转向时，ZMP 会随着转向速度和惯性力的变化而变化。当转向速度较大时，ZMP 位置会偏移，从而导致机器人可能失去平衡。为了确保机器人稳定，ZMP 必须始终保持在支撑多边形内。本文通过横滚角度 ϕ 的变化来调整 ZMP 的位置。设定横滚角度的控制力矩，使得 ZMP 保持在安全区域内。因此，随着转向速度的增大，ZMP 的偏移量会增大，从而需要调整横滚角度或控制力矩以确保 ZMP 始终保持在支撑多边形内。

3.3 站立控制

为了实现机器人在站立姿态下的精确控制，必须解决一个关键问题：原有的线性化模型是在机器人坐姿状态下建立的，其平衡点以内轮抬升角为 0 为基准。然而，当机器人处于站立状态时，内轮抬升角并不再接近零，而存在一个 $\pm\pi$ 的角度差。这意味着，如果继续采用坐姿状态下的模型进行控制设计，系统的线性化误差将明显增大，从而导致控制输出不准确，甚至影响系统稳定性。

为了解决这一问题，本文提出了以下改进措施：

首先，将站立姿态下的稳定状态重新定义为新的平衡点。在站立状态下，

内轮抬升角、机体俯仰角等状态变量的实际数值作为新的零点进行线性化处理。对系统状态变量做如下变换：

$$\tilde{x} = x - x_s, \quad (3.31)$$

其中 x_s 表示站立状态下的平衡状态向量。在这一新的平衡点处，对非线性模型进行一阶泰勒展开，可以得到：

$$\dot{\tilde{x}} = A_s \tilde{x} + B_s \tilde{u} \quad (3.32)$$

其中矩阵 A_s 和 B_s 是在站立状态下重新计算的系统矩阵和输入矩阵。其次，由于站立状态下的动态特性与坐姿状态有所不同，为了确保控制器能够在两个不同状态下均达到较好的性能，需要相应地调整 LQR 设计中的状态权重矩阵 Q 和输入权重矩阵 R ，可以使得线性化模型在站立状态下更能体现实际的状态误差和控制目标，从而计算出新的反馈增益矩阵 K_s 。

最后，在实际运行过程中，机器人需要完成从坐姿到站姿的过渡。为了平稳切换控制策略，本文设计了增益切换机制：当机器人检测到状态变量逐渐接近站立状态即内轮抬升角达到预设的阈值附近，系统会平滑切换使用新的反馈增益 K_s 而不是原始坐姿状态下的反馈增益 K 。这种切换可以通过增益调度策略实现，确保整个姿态转换过程中控制输出的连续性和稳定性。具体的站立过程演示如图 3.3 所示。

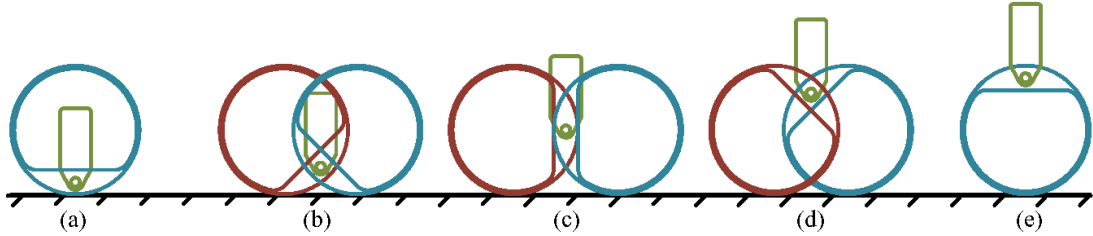


图 3.3 从坐姿过渡到站姿的五个部分。(a)初始站姿。(b)两侧的抬升电机向反方向运动。(c)提升过程。(d)到达顶端时，切换 LQR 参数矩阵。(e)完成站立。

Fig. 3.3 The transition from sitting to standing five section. Initial pose (a). Lift pitch motors in different direction (b). The process of lifting (c). When reaching top, switch LQR parameter matrix (d). Finish stand (e).

3.4 跳跃控制

在轮圈腿式平衡机器人的跳跃动作中，机器人通过控制内轮的抬升角来产

生所需的竖直加速度，进而实现跳跃。跳跃动作可以分为两种类型：两侧轮交叉跳跃和两侧轮同向跳跃。

在跳跃过程中，机器人通过控制内轮的抬升角，使机体产生竖直方向的加速度。当内轮抬升时，由于其在地面接触点的高度变化，机器人会向上加速，克服重力产生跳跃。当机器人达到内轮的最高点时，内轮的抬升角度开始减小，机体的重心会逐渐收回，从而产生跳跃的高度。控制内轮的抬升角度变化是跳跃动作的核心。

在两侧轮交叉跳跃这种跳跃方式中，左右内轮的抬升角度是交替变化的，即左轮和右轮抬升角度交替加大，以产生竖直方向的力。这种方式适合较为复杂的地面环境下的跳跃，能够确保机器人保持较好的稳定性。

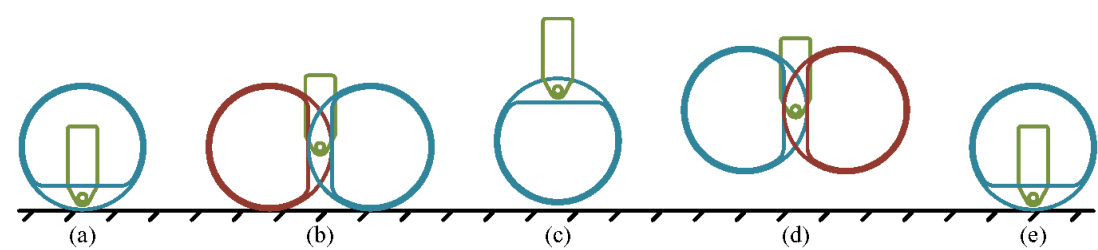


图 3.4 交叉跳跃的过程，(a)初始姿势；(b)-(d)两侧抬升电机绕外轮反向运动；(e)落地
Fig. 3.4 The process of cross jumping. Initial pose (a). Lifting leg motors around outer wheels in opposite direction (b) to (d). Land (e).

在两侧轮同向跳跃这种跳跃方式中，左右内轮的抬升角度同步增加，确保机器人在跳跃时的稳定性和速度。这种方式适合要求高效跳跃的任务，可以在短时间内产生较高的竖直加速度。

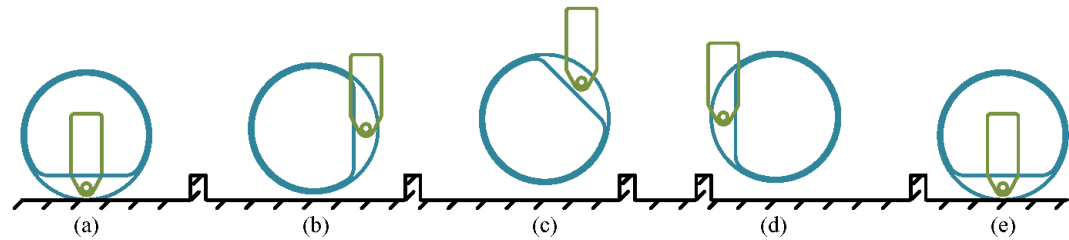


图 3.5 同向跳跃的过程，(a)初始姿势；(b)-(d) 两侧抬升电机绕外轮同向运动；(e)落地
Fig. 3.5 The process of forward jumping. Initial pose (a). Lifting leg motors around outer wheels (b) to (d). Land (e).

为了实现有效的跳跃控制，机器人必须在跳跃过程中生成足够的竖直加速度，以克服重力并迅速达到所需的跳跃高度。控制策略的核心是通过设计两侧

内轮抬角变化曲线，调节两侧内轮的抬升角度，从而实现所需的加速度。通过精确地控制内轮抬升角度，机器人能够在合适的时机产生足够的竖直力，使机器人完成跳跃。

3.5 打滑恢复控制

在轮式机器人运动过程中，特别是在复杂地形或摩擦系数变化较大的环境下，机器人可能会出现打滑现象。当机器人与地面接触的摩擦力无法满足需求时，外轮会高速旋转，并导致速度失真。由于机器人的前进速度是基于两侧外轮的旋转速度计算的，如果外轮的旋转速度与实际速度不匹配，模型会产生误差，导致系统无法准确估计机器人实际速度。因此，为了提高速度估计的精度，特别是在出现打滑现象时，本文采用了卡尔曼滤波器来对机器人加速度和轮速计数据进行融合。

在理想情况下，轮式机器人行驶时，前进速度是由两侧外轮的旋转速度计算得到的。假设机器人的前进速度 v_{robot} 与外轮的角速度 ω_{left} 和 ω_{right} 存在如下关系：

$$v_{robot} = \frac{R}{2}(\omega_{left} + \omega_{right}) \quad (3.33)$$

其中 R 是外轮半径， ω_{left} 和 ω_{right} 分别为左轮和右轮的角速度。然而，在实际行驶中，由于摩擦力不足、地面不平或其他因素，可能会导致外轮的高速旋转无法有效推动机器人前进，进而产生打滑现象。当出现打滑时，机器人前进速度的计算将不再准确，模型计算出的速度与实际速度之间存在偏差。这种偏差会导致误差积累，从而影响系统的性能。

为了更准确地估计机器人实际的行驶速度，并有效预测和修正打滑现象，本文引入卡尔曼滤波器对加速度计和轮速计数据进行融合。卡尔曼滤波器是一种递归算法，通过结合模型预测和测量值来提供最优的状态估计。卡尔曼滤波器的基本形式可以表示为：

$$\begin{aligned} \hat{x}_{k|k-1} &= A\hat{x}_{k-1|k-1} + Bu_k \\ \hat{y}_k &= C\hat{x}_{k|k-1} + v_k \end{aligned} \quad (3.34)$$

其中， $\hat{x}_{k|k-1}$ 是上一时刻预测的状态， A 是状态转移矩阵，描述了系统动态， B 是控制矩阵，表示控制输入 u_k 对系统状态的影响， C 是观测矩阵，表示如何将状态转化为测量值 \hat{y}_k ， v_k 是测量误差。通过卡尔曼滤波器，本文可以将轮速

计和加速度计的测量值进行加权融合，得到一个更加精确的速度估计。在卡尔曼滤波的框架下，控制和观测数据的误差通过模型的预测与实际测量之间的比较得到最小化。

在机器人打滑的短时间内，本文可以假设机器人的加速度是固定的，即在短时间内机器人符合匀加速运动模型，根据这一模型，机器人的速度和位置随着时间的推移呈现出匀加速运动的特征。设定机器人的速度为 $v(t)$ ，位置为 $x(t)$ ，加速度为 $a(t)$ ，根据匀加速运动的基本公式，本文可以得到以下关系：

$$v(t) = v_0 + a \cdot t \quad (3.35)$$

$$x(t) = x_0 + v_0 t + \frac{1}{2} a t^2, \quad (3.36)$$

其中， v_0 和 x_0 分别是初始速度和初始位置， a 为常数加速度。机器人的加速度 $a(t)$ 来自机器人中心机体上的加速度计传感器，通常会有噪声影响。为了减少测量误差，卡尔曼滤波器结合加速度计的测量数据和机器人的运动模型，提供更准确的速度和位置估计。

本文将卡尔曼滤波器应用于速度和位置的估计中。设定状态向量为：

$$X(t) = \begin{bmatrix} x(t) \\ v(t) \end{bmatrix} \quad (3.37)$$

状态转移矩阵 A 描述了系统的动态演变，控制矩阵 B 描述了加速度对状态的影响。对于匀加速模型，状态转移矩阵和控制矩阵分别为：

$$A = \begin{bmatrix} 1 & \Delta t \\ 0 & 1 \end{bmatrix}, B = \begin{bmatrix} \frac{1}{2} \Delta t^2 \\ \Delta t \end{bmatrix} \quad (3.38)$$

其中， Δt 是时间间隔， B 矩阵中的项表示加速度对位置和速度的影响。

测量向量 $z(t)$ 由加速度计和轮速计提供，测量方程为：

$$z(t) = \begin{bmatrix} a(t) \\ v(t) \end{bmatrix} + w(t) \quad (3.39)$$

其中 $a(t)$ 是加速度计测量的加速度， $v(t)$ 是轮速计测量的速度， $w(t)$ 为测量噪声。

卡尔曼滤波通过预测和更新步骤对机器人状态进行估计。在每一时刻，卡尔曼滤波器根据当前的控制输入（即加速度）和先前的状态估计预测下一时刻的状态，并使用测量值进行修正。卡尔曼滤波的基本步骤如下：

预测步骤：根据系统的动态模型预测当前时刻的状态和协方差：

$$\begin{aligned}\hat{\mathbf{X}}^-(t) &= \mathbf{A}\hat{\mathbf{X}}(t-1) + \mathbf{B}a(t) \\ \mathbf{P}^-(t) &= \mathbf{A}\mathbf{P}(t-1)\mathbf{A}^T + \mathbf{Q}\end{aligned}\quad (3.40)$$

其中， $\hat{\mathbf{X}}^-(t)$ 为预测状态， $\mathbf{P}^-(t)$ 为预测协方差矩阵， \mathbf{Q} 为过程噪声协方差矩阵。

更新步骤：

$$\begin{aligned}\mathbf{K}(t) &= \mathbf{P}^-(t)\mathbf{C}^T(\mathbf{C}\mathbf{P}^-(t)\mathbf{C}^T + \mathbf{R})^{-1} \\ \hat{\mathbf{X}}(t) &= \hat{\mathbf{X}}^-(t) + \mathbf{K}(t)(z(t) - \mathbf{C}\hat{\mathbf{X}}^-(t)) \\ \mathbf{P}(t) &= (\mathbf{I} - \mathbf{K}(t)\mathbf{C})\mathbf{P}^-(t)\end{aligned}\quad (3.41)$$

其中， $\mathbf{K}(t)$ 是卡尔曼增益， \mathbf{C} 是测量矩阵， \mathbf{R} 是测量噪声协方差矩阵。如图 3.6 所示为采集机器人测量加速度并使用卡尔曼滤波进行处理的两个速度的对比，可以明显看到在机器人换向时产生打滑现象，并且两个曲线有较大的差异。

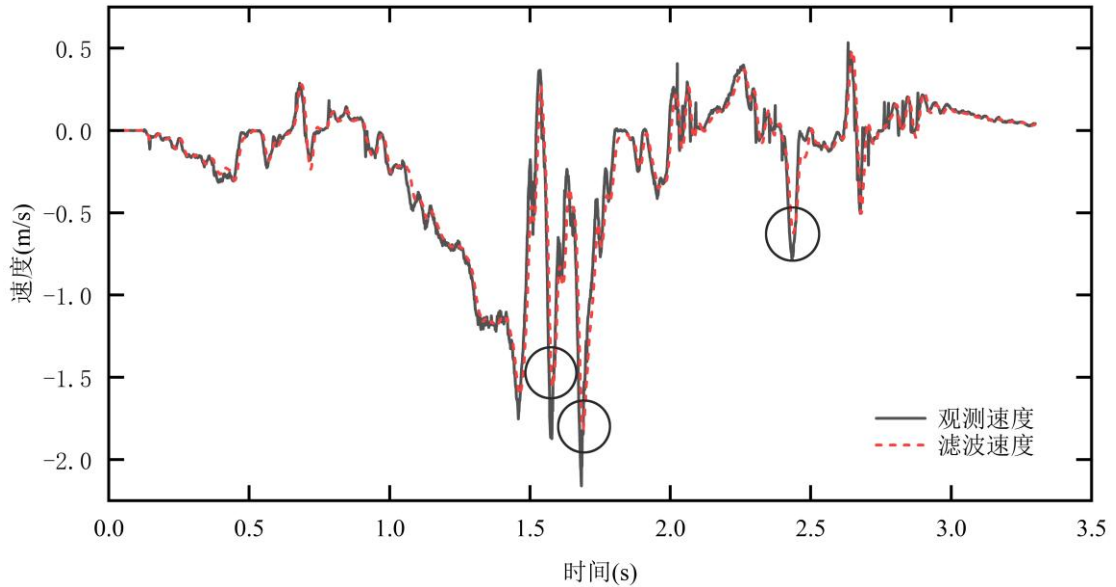


图 3.6 机器人的测量速度和卡尔曼滤波后的滤波速度对比，在机器人急转向时发生打滑现象，此时两个速度出现明显差异可以判断打滑

Fig. 3.6 Comparison between the measured speed of the robot and the filtered speed after Kalman filtering, slipping occurs when the robot turns sharply, at this time there is a significant difference between the two speeds can be judged to be slipping

在机器人行驶过程中，尤其是打滑现象发生时，机器人与地面之间的摩擦力不足，导致实际速度和计算出的速度之间存在较大差异。卡尔曼滤波器通过

不断地调整状态估计，在加速度和轮速计数据之间实现数据融合，能够有效减少由打滑引起的速度突变。

卡尔曼滤波器的状态估计和速度修正可以有效预测打滑发生的时刻。具体来说，当加速度计的加速度和轮速计的测量速度之间的差异大于设定阈值时，卡尔曼滤波器将快速修正速度估计，从而避免由于打滑带来的速度突变。当检测到打滑时，状态观测器将卡尔曼滤波后的速度作为当前速度，可以有效降低因为打滑带来的计算发散。

3.6 倒地恢复控制

在双轮机器人实际运行中，由于地面摩擦、重心分布等因素，机器人可能发生翻覆，其倾覆情况可分为四种：坐姿、俯卧、跨轮和侧翻。对于坐姿、俯卧和跨轮这三种状态，由于重心分布的特点，恢复过程通常只需将 LQR 控制器计算出的转矩施加到内轮俯仰电机上，而外轮电机则不输出转矩；当躯干角度恢复到预定的稳定范围后，所有电机的控制均恢复为 LQR 控制，从而在尽可能小的空间内实现平衡恢复。

而对于侧翻情况，由于重心分布的限制，大多数双轮平衡机器人难以在滚动轴方向产生足够的转动力矩，从而无法调整重心位置以恢复平衡。相比之下，轮圈腿式平衡机器人能够通过调整两个俯仰电机的角度，产生滚轴方向的旋转趋势，从而将机器人侧翻时产生的滚动力矩引导至有利于恢复平衡的方向，这也是该机器人在翻覆恢复方面表现出较高稳定性的一个重要因素。

当机器人检测到翻覆状态时即在机器人躯体上的姿态传感器反馈的姿态角度超出正常行驶范围：

$$|\phi| > \phi_{th} \quad (3.42)$$

此时系统状态已不再符合基于 LQR 控制的线性化建模条件。一旦检测到翻覆，机器人内轮控制由 LQR 控制切换为 PD 位置控制，以便更快速地调整机器人中心位置恢复平衡。

在倒地状态下，本文采用 PD 控制器对机器人内轮进行位置控制，目的是将机器人的中心位置迅速调整回预设的平衡位置。PD 控制器的控制律可表示为：

$$\tau_{PD} = K_p(\theta_{des} - \theta) + K_d(\dot{\theta}_{des} - \dot{\theta}) \quad (3.43)$$

当机器人中心调整至预定平衡区域且躯干角度恢复到安全范围内后，系统切换回 LQR 控制。此时，LQR 控制器重新计算并输出内轮力矩 τ_{LQR} 维持机器人整体平衡，总的控制力矩为：

$$\tau_{total} = \tau_{LQR} + \tau_{PD} \quad (3.44)$$

其中在恢复完成后 τ_{PD} 会趋近于零，系统完全由 LQR 控制维持稳定。

3.7 MATLAB 仿真验证

在机器人系统的设计与控制中，仿真验证是一个重要的环节。通过仿真可以在没有物理样机的情况下，验证控制策略、算法及系统设计的有效性和鲁棒性。为此，本文在 MATLAB 环境中使用 Simscape 仿真工具进行机器人系统的建模和验证。使用 MATLAB 仿真验证的主要目的是验证动力学方程的合理性以及控制算法的可行性，为编写算法程序提供指导。

3.7.1 Simscape 环境搭建

在 MATLAB 中本文使用 Simscape 仿真环境作为机器人的仿真环境进行搭建。Simscape 是 MATLAB 中的一个用于建模和仿真物理系统的工具箱，它提供了一个多领域的建模平台，支持机械、电气、液压、热学等多种物理域的建模与仿真。在本文的研究中，Simscape 被用来搭建机器人模型，并进行各项控制策略的验证。

3.7.1.1 机器人模型的构建

在 Simscape 环境中，本文首先根据机器人系统的实际结构与功能要求构建机器人模型。该模型包括机器人的各个物理部件，如内外轮电机、内外轮圈腿、机体等。每个部件使用 Simscape 的预定义组件或自定义组件进行建模。

其中各部分实体建模使用 File Solid 模块将已有三维模型的 STL 格式文件导入如图 3.7 所示，既可以节省建模时间，又可以尽量还原物理结构的各项物理参数如质量，转动惯量等重要仿真参数。

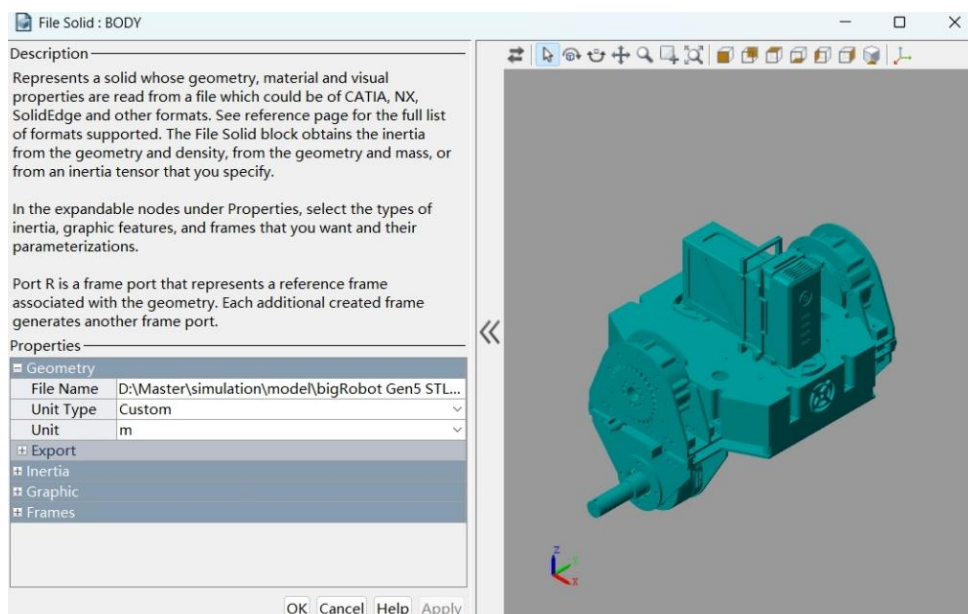


图 3.7 Simscape File Solid 组件导入实体模型窗口

Fig. 3.7 Simscape File Solid component import solid model window

其中各电机使用 **Revolute joint** 模块进行模拟，该模块用于模拟两个实体组件间的旋转关节，且仅有一个自由度，如图。该模块可以驱动关节转动，同时可以模拟位置速度传感器实时反馈当前关节的状态量用于控制算法的计算。除此之外，还可以设置关节的物理惯性参数如平衡位置，摩擦，阻尼等尽可能还原实体电机的性能参数。

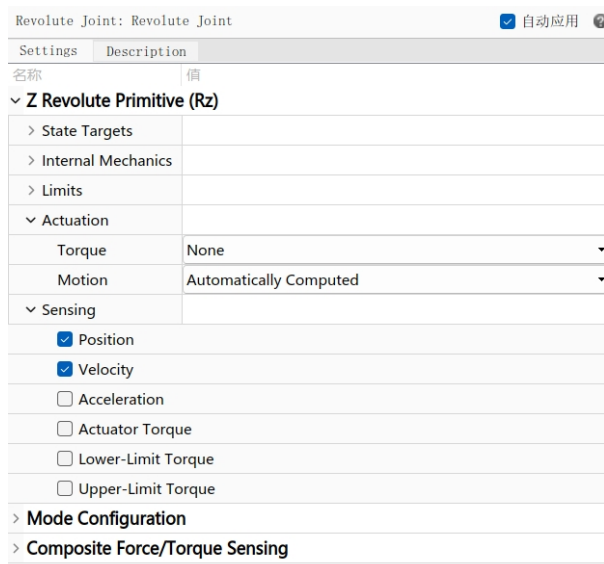


图 3.8 Simscape Revolute joint 组件导入窗口

Fig. 3.8 Simscape Revolute joint import window

在获取机器人姿态位置部分，本文在机器人的机体上设定一个六自由度的浮动基座即 **6-DOF Joint** 模块，该模块输出反馈当前机体的四元数数据，本文通

过 Quaternions to Rotation Angles 模块将获得的四元数数据解算为 Pitch 轴，Yaw 轴，Roll 轴的三轴姿态信息。同时 6-DOF Joint 模块可以反馈机器人机体沿 PYR 三轴的角速度信息，即可以获得机器人三轴角度和角速度信息。

最终在 Simscape 中搭建的仿真效果如图所示：

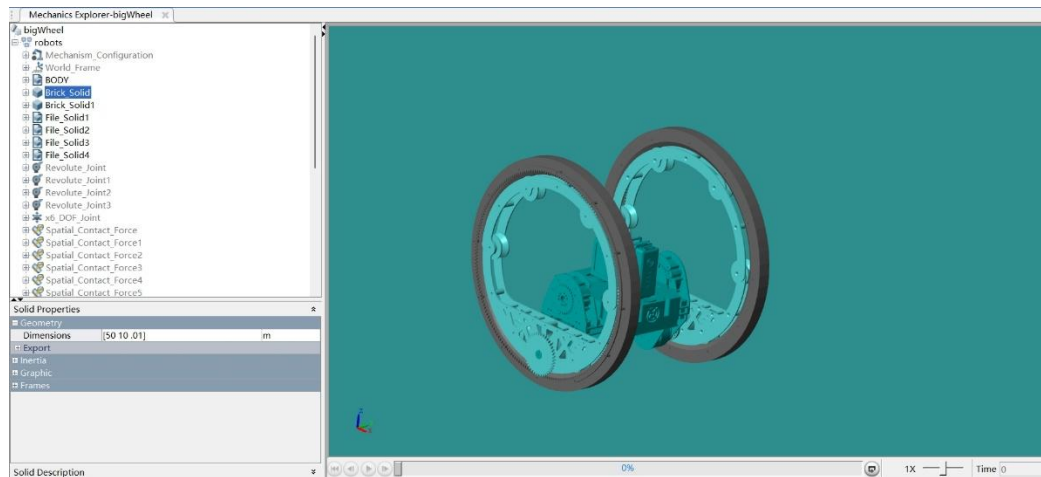


图 3.9 Simscape 仿真环境搭建效果图

Fig. 3.9 Simscape simulation environment construction effect diagram

3.7.1.2 控制环境搭建

为了将 LQR 控制器引入 Simscape 仿真环境中，本文使用了 Simulink 中的 Interpreted MATLAB Function 模块。该模块允许直接调用 MATLAB 脚本函数，并将其嵌入到 Simulink 模型中，实现与 Simscape 物理模型的实时交互。在 Simscape 中，机器人物理模型会实时输出机器人的状态信息。这些状态信息作为输入信号被反馈给 LQR 控制器，通过 Simulink 中的 Interpreted MATLAB Function 模块进行实时计算。这种方法的优势在于，LQR 控制器能够实时根据机器人当前状态做出动态调整，并且与物理系统的集成使得整个系统可以通过仿真验证不同控制策略的效果，确保机器人在动态环境中的稳定性和响应性。

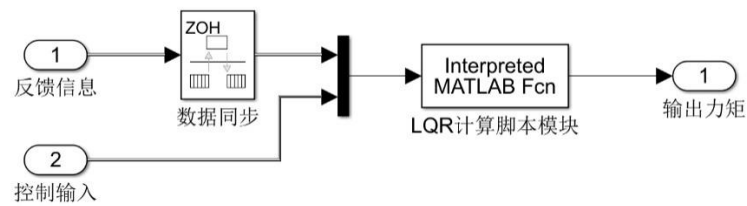


图 3.10 通过 Interpreted MATLAB Function 模块实现 LQR 在仿真环境中实时计算

Fig. 3.10 Real-time computation of LQR in a simulation environment through the Interpreted MATLAB Function module

通过这种集成方式，本文能够在 Simscape 仿真环境中验证 LQR 控制器的效果。仿真过程中，LQR 控制器根据输入的状态信息调整电机的控制力矩，机器人能够实时响应不同的运动模式和外部扰动（如翻覆、加速等）。仿真结果通过 Simulink 中的 Scope 模块进行实时显示，帮助评估控制系统的稳定性和精确性。

3.7.2 实验验证

3.7.2.1 平衡行驶试验

为了验证轮圈腿式平衡机器人在不同运动状态下的平衡与响应能力，设计了平衡行驶实验。本实验的主要目标是测试机器人在设定的目标速度下是否能够平稳行驶，并考察其在减速过程中的表现，以确保机器人在动态行驶和减速过程中的稳定性。

实验过程中，机器人首先在平坦的地面上启动，目标速度设定为 2 m/s，并观察机器人的反馈速度是否能够达到目标值。在初始阶段，机器人加速至目标速度，并保持该速度 2 秒。接着，在 2 秒后，控制系统指令机器人逐渐减速至 0 m/s，并观察其减速过程中的平稳性。其实验过程如图 3.11 所示。

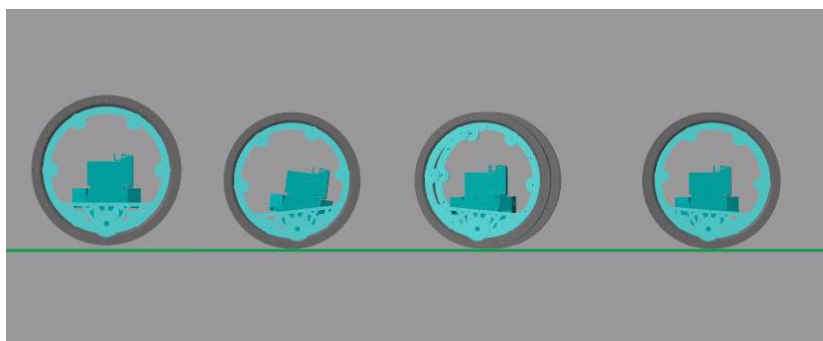


图 3.11 机器人平衡行走 MATLAB 仿真实验

Fig. 3.11 Robot balanced walking MATLAB simulation experiments

实验数据结果如图 3.12 所示，图中展示了机器人反馈速度（蓝色曲线）与目标速度（红色曲线）的变化过程。在图中，本文可以看到：在加速阶段：从 0 秒到约 1 秒，机器人的速度逐渐增加，并稳定达到 2 m/s。反馈速度紧跟目标速度（，系统响应稳定且快速。在减速阶段：在 2 秒时，机器人开始减速，目标速度迅速降至 0 m/s。反馈速度在减速过程中逐渐降低，并平稳收敛。值得注意的是，虽然在减速过程中，反馈速度出现了轻微的波动，但整体趋势平滑，没有出现过度震荡或失稳现象。

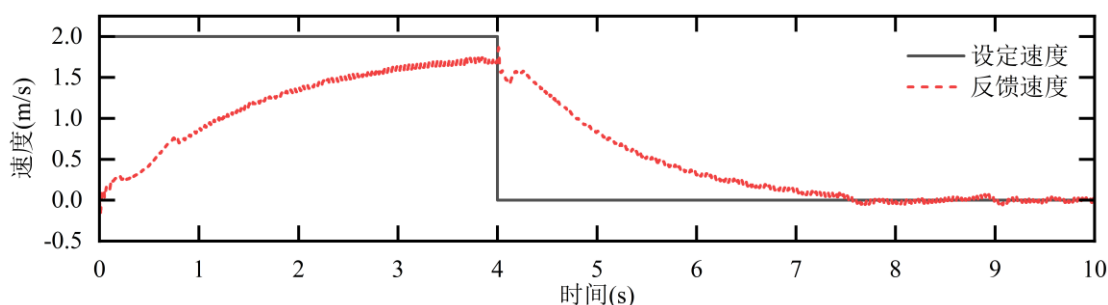


图 3.12 平衡行走仿真实验数据图（其中绿色实线为设定速度，蓝色短划线为反馈速度）

Fig. 3.12 Plot of experimental data from balanced walking simulation (where the green solid line is the set speed and the blue dashed line is the feedback speed)

3.7.2.2 站立实验

为了验证轮圈腿式平衡机器人在站立过程中的姿态控制能力，本文设计了一个站立姿态实验。在该实验中，本文通过调节机器人两侧内轮的抬升角，控制机器人从坐姿过渡到站立姿态，并测试其站立稳定性和反馈速度的表现。实验的主要目标是评估机器人在站立过程中，内轮角度调整对机器人姿态的影响，并验证控制系统是否能够平稳地调整 and 保持站立姿态。

实验中，本文设置了机器人的目标状态为站立姿态，即机器人两侧内轮的抬升角分别为 π 和 $-\pi$ 。在该目标设定下，本文希望机器人在2秒内通过内轮角度的变化达到站立位置。控制系统将在此过程中根据内轮的角度变化实时调整机器人姿态，确保机器人能够平稳地从坐姿转移到站立状态。试验过程视频如图 3.13 所示。

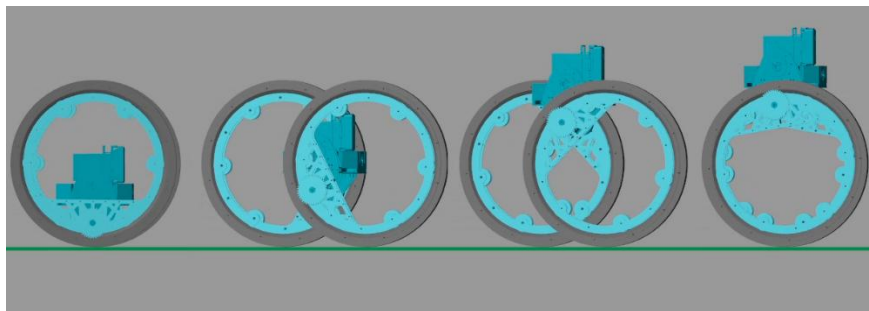


图 3.13 机器人站立平衡 MATLAB 仿真实验

Fig. 3.13 Robot standing balance MATLAB simulation experiments

实验数据结果如图 3.14 所示，图中展示了机器人单侧内轮反馈角度（蓝色曲线）与目标角度（红色曲线）的变化过程。在该图中：从0秒到约1秒，机器人保持静止状态，从1秒开始站立，在1秒到3秒期间，机器人以匀速抬升内轮，在3秒时到达最高点，此时机器人反馈角度与目标角度几乎完全重合，系统表现出良好的跟踪能力和稳定性，同时也验证了在最高点切换 LQR 控制参数的控制策略的可行性。

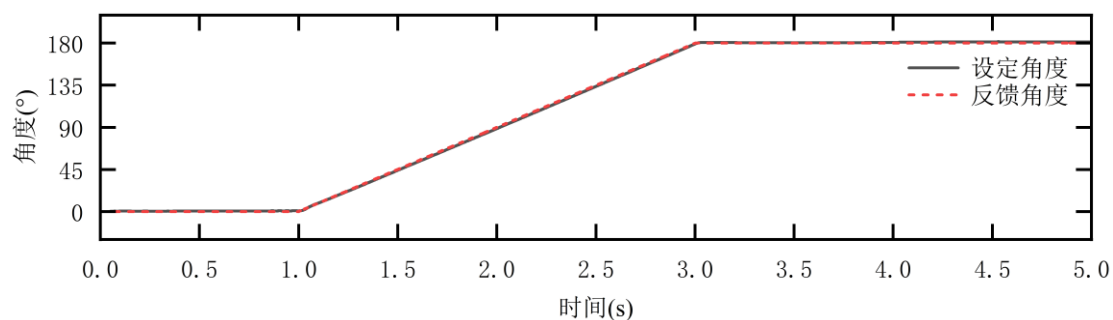


图 3.14 站立实验仿真数据图（其中绿色实线为设定内轮角度，蓝色短划线为反馈角度）

Fig. 3.14 Simulation data plot of standing experiment (where the green solid line is the set inner wheel angle and the blue dashed line is the feedback angle)

3.8 Webots 仿真验证

为了进一步验证轮圈腿式平衡机器人在复杂运动环境中的控制性能与稳定性，本文使用 Webots 仿真平台进行了多种运动实验。Webots 是一款常用的机

器人仿真工具，提供了丰富的物理引擎和控制接口，可以高效地模拟机器人的动态行为。在这一章中，本文将介绍在 Webots 中进行的四个仿真实验：行走实验、站立实验、跳跃实验以及上台阶实验。

3.8.1 Webots 仿真环境搭建

在进行机器人的仿真验证时，Webots 作为一款广泛使用的机器人仿真平台，提供了一个高效的虚拟环境，能够模拟机器人在不同运动模式下的表现。为了构建一个符合实际机器人设计的仿真环境，需要进行一系列的步骤，包括导入机器人模型、定义物理属性、添加传感器与执行器模块等。以下是基于 Webots 的环境搭建过程，具体描述了如何通过 SolidWorks 三维模型导入、添加铰链关节、模拟电机和传感器。

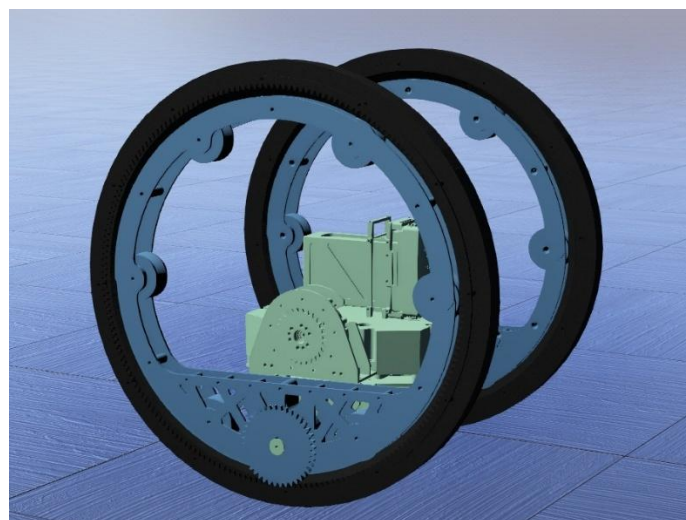


图 3.15 Webots 仿真环境下的机器人

Fig. 3.15 Robots in Webots Simulation Environment

3.8.1.1 导入三维模型

在 Webots 中，首先需要创建机器人的三维外形。为了实现这一目标，本文通过 SolidWorks 设计了机器人的 3D 模型，并将其导出为 STL 格式文件。STL 格式是一种常用于三维打印和仿真中描述几何形状的文件格式。在 SolidWorks 中，设计完成后导出为 STL 文件。该文件包含机器人的三维外形数据，可以被 Webots 识别和使用。通过 Webots 中的“Mesh”组件，将导出的 STL 文件导入到仿真环境中。这样，Webots 可以渲染并显示机器人外形，确保仿真中的机器人

与实际机器人在视觉上的一致性。通过添加实体属性以及物理属性，修改模型的惯性属性确保和机器人模型一致。

3.8.1.2 定义机器人关节与传感器

在 Webots 中，为了模拟机器人的动态行为，需要合理地定义机器人各部分之间的物理连接。这通常通过在机器人机体中添加适当的关节和连接器来实现。为了模拟机器人运动中的铰链转动，使用了 Webots 中的 HingeJoint 组件。通过在机器人机体内部添加两个铰链关节，可以模拟机器人的关节连接。铰链关节能够模拟旋转运动，类似于机器人的腰部或其他旋转部分，允许机器人根据控制信号进行转动。为了让机器人能够执行具体的运动任务，如平衡、转向等，在铰链关节中加入电机模块。电机模块能够通过提供旋转力矩来驱动关节，控制机器人的姿态和运动。

为了实现机器人的精确控制，尤其是在进行平衡和路径规划时，需要实时反馈机器人的位置和姿态。为此，在铰链关节上添加了位置传感器模块。位置传感器模块能够实时测量机器人的角度和转动情况，并将数据传递给控制系统，用于闭环控制。除此之外添加了 IMU 传感器以及角速度计传感器模拟机器人上的板载陀螺仪模块实时反馈机器人的三轴角速度信息和三轴角度位置信息。

3.8.2 行走实验

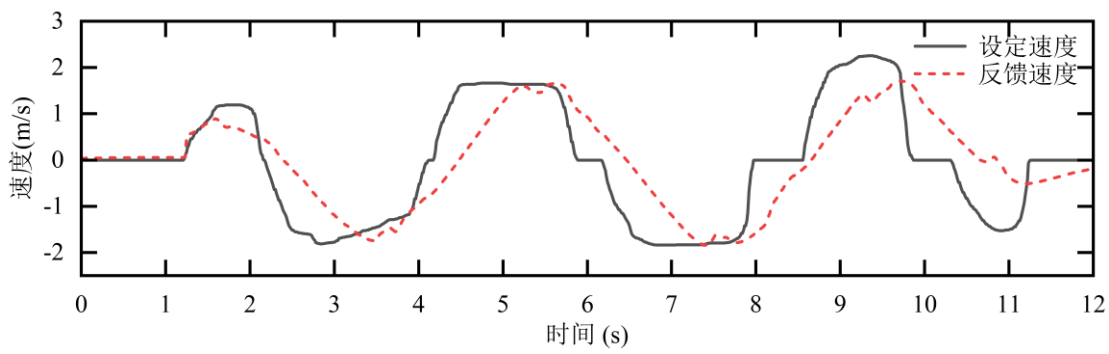


图 3.16

Fig. 3.16

3.8.3 站立实验

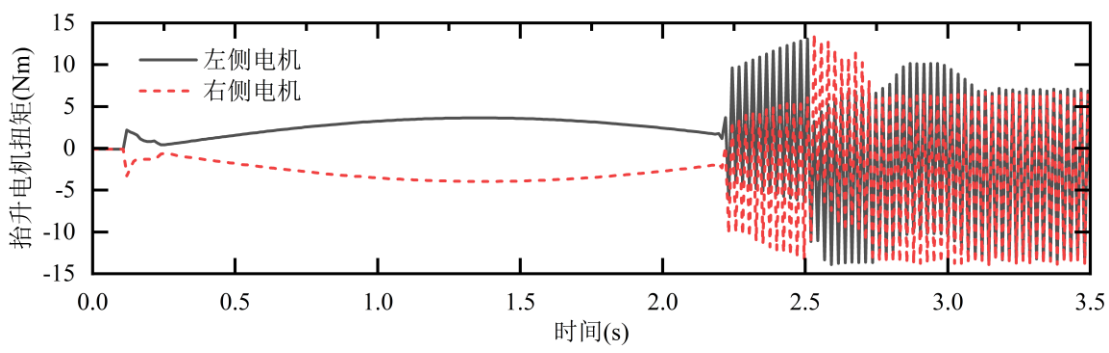


图 3.17

Fig. 3.17

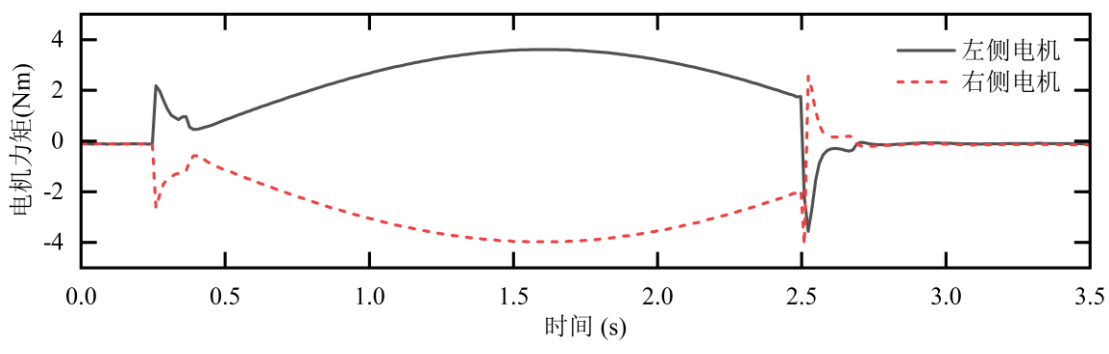


图 3.18

Fig. 3.18

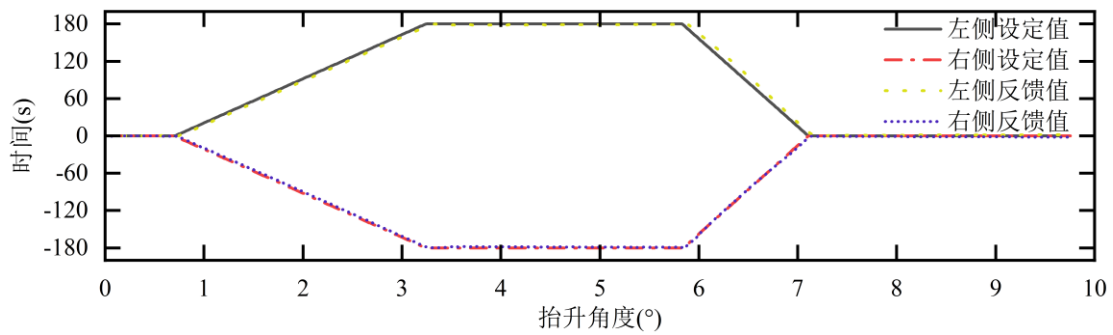


图 3.19

Fig. 3.19

3.8.4 跳跃实验

3.8.5 上台阶实验

第 4 章 实物样机平台搭建

4.1 硬件系统搭建

4.1.1 电机性能需求分析与选型

轮圈腿式平衡机器人的多模态运动特性对驱动系统提出了严苛的功能需求。根据机器人的机械构型与控制目标，轮圈腿式机器人上共有四个电机，分别为两个内轮俯仰关节电机和两个外轮驱动轮电机。内轮电机与外轮电机相配合完成机器人坐姿移动，站姿移动，跳跃等运动功能。

4.1.1.1 内轮电机需求分析

内轮俯仰关节电机主要负责机器人机体姿态的调整，在坐姿-站姿模态切换、跳跃动作执行以及翻倒恢复过程中，需要提供精准且足够的力矩输出。为此，本文需要分别考虑静态和动态两个方面的力矩需求。

在这个运动模式中，机器人需提供最大静态力矩机体处于水平状态时，由

机器人重心高度即为力臂长度 71mm，机器人机体质量为 4.1kg，根据力矩计算公式可得所需的最大静态力矩为：

$$\begin{aligned}\tau_{static} &= m_b \cdot g \cdot h \\ &= 4.1 \times 9.81 \times 0.071 \approx 2.86Nm\end{aligned}\quad (4.1)$$

在机器人跳跃等动态运动过程中，内轮电机不仅要克服静态负载，还需提供足够的动态响应来实现快速加减速。假设机器人内外轮的旋转角加速度为 5rad/s^2 ，根据动态力矩计算公式可得：

$$\begin{aligned}\tau_{dynamic} &= (I_o + I_i) \cdot \alpha \\ &= 0.11 \times 5 \approx 0.55Nm\end{aligned}\quad (4.2)$$

考虑极端情况即在最大静态力矩处两侧轮圈腿进行跳跃动作，理论最大静态力矩与动态力矩叠加可得：

$$\begin{aligned}\tau_{total} &= \tau_{static} + \tau_{dynamic} \\ &= 2.86 + 0.55 = 3.41Nm\end{aligned}\quad (4.3)$$

本文选用的电机是 CyberGear 减速电机，如图 4.1 所示，电机额定扭矩为 4Nm，峰值扭矩为 12Nm，设定安全系数 $k = 2$ ，经对比两个电机的峰值扭矩大于理论扭矩需求之和：

$$\tau_{motor} > k \cdot \tau_{total}\quad (4.4)$$



图 4.1 CyberGear 电机实物图

Fig. 4.1 Physical drawing of CyberGear motor

电机的详细参数如表 4.1 所示，经分析该电机满足理论需求。

表 4.1 内轮电机参数表

Table 4.1 Inner wheel motor parameter table

参数项	描述	参数
型号	电机的型号	CyberGear
额定扭矩	电机在额定工作条件下的输出扭矩 (Nm)	4Nm
峰值扭矩	电机在短时间内能够输出的最大扭矩 (Nm)	12Nm
额定转速	电机在额定工作条件下的转速 (rpm)	240rpm
空载转速	电机在空载下能够达到的最大转速 (rpm)	296rpm
额定功率	电机在额定条件下的功率输出 (W)	156W
额定电压	电机工作的额定电压 (V)	24V
重量	电机的重量 (kg)	0.32kg
外形尺寸	电机的外形尺寸 (mm)	Ø75mm

4.1.1.2 外轮电机需求分析

两个外轮驱动电机通过齿轮组将扭矩传递至外轮圈，主要负责维持机器人的集体平衡，确保机器人在站立、前进、攀爬等不同运动模式下能够稳定运行。外轮电机不仅需要提供机器人平衡所需的推力，还需要应对一些特殊姿态下的动力要求，如在攀爬和跳跃过程中提供足够的支持力矩。以下对外轮电机的需求进行详细分析。

在机器人处于半站立姿态下，内外轮交叉站立时，外轮电机需要支撑两侧内轮和机体的重量，保持机器人竖直状态。此时，外轮电机需要提供最大的静态力矩，以克服机器人重心的倾斜。已知机器人机体质量为 $m_b = 4.1\text{kg}$ ，两侧内轮和机体的质量和为 $m_{total} = m_b + m_i = 6.16\text{kg}$ ，力臂长度即机体转轴到外轮转轴的距离 $L = 0.16\text{m}$ ，根据力矩公式，外轮电机所需的最大静态力矩为：

$$\begin{aligned}\tau_{static} &= m_{total} \cdot g \cdot L \\ &= 6.16 \times 9.81 \times 0.16 \approx 9.66\text{Nm}\end{aligned}\quad (4.5)$$

在动态运动过程中，尤其是机器人执行攀爬、加速等动作时，外轮电机不仅需要提供静态支撑力矩，还需提供足够的动力输出以克服惯性。考虑到外轮的转速和加速度，假设机器人外轮所需的旋转角加速度为 $\alpha = 3\text{rad} / \text{s}^2$ ，外轮

的等效转动惯量为 $I_0 = 0.06 \text{ kg} \cdot \text{m}^2$ ，根据动态力矩公式：

$$\begin{aligned}\tau_{dynamic} &= I_o \cdot \alpha \\ &= 0.06 \times 3 = 0.18 \text{ Nm}\end{aligned}\tag{4.6}$$

外轮电机在机器人执行动态任务时，最大力矩需求是静态力矩和动态力矩的叠加。在机器人进行跳跃、攀爬或急加速等动作时，外轮电机的最大力矩为：

$$\begin{aligned}\tau_{total} &= \tau_{static} + \tau_{dynamic} \\ &= 0.18 + 9.66 = 9.84 \text{ Nm}\end{aligned}\tag{4.7}$$

外轮电机不仅需要足够的力矩输出，还需要具备合适的转速响应能力，尤其是在高速运动和转向时。机器人在转向过程中，外轮的转速和角速度对运动的平稳性至关重要。设定机器人最大运行速度为 6m/s，转换为转每分为：254rpm。因此，外轮电机应具备能够达到或超过此转速的能力，以满足机器人在转向过程中对速度的需求。

轮圈腿式机器人的外轮电机选择的是本文选择的一款 MF7015-24V-10T 直流无刷电机，如图 4.2 所示，该电机的额定扭矩为 1Nm，峰值扭矩为 2Nm，额定转速为 1500rpm，



图 4.2 MF7015 电机实物图

Fig. 4.2 MF7015 motor physical picture

详细参数如表 4.2 所示。满足理论计算需求。

表 4.2 外轮电机参数表

Table 4.2 Outer wheel motor parameter table

参数项	描述	参数
型号	电机的型号	MF7015v2
额定扭矩	电机在额定工作条件下的输出扭矩（Nm）	1Nm
峰值扭矩	电机在短时间内能够输出的最大扭矩（Nm）	2Nm
额定转速	电机在额定工作条件下的转速（rpm）	1500rpm
空载转速	电机在空载下能够达到的最大转速（rpm）	1890rpm
额定功率	电机在额定条件下的功率输出（W）	199W
额定电压	电机工作的额定电压（V）	24V
重量	电机的重量（kg）	0.36kg
外形尺寸	电机的外形尺寸（mm）	Ø69mm

4.1.2 传感器选型

在轮圈腿式平衡机器人中，为了实时获取机器人当前的状态并确保其稳定性和响应速度，需要配备一系列高性能传感器。传感器提供的数据将直接影响控制系统的精度和响应能力。机器人所需的传感器主要包括陀螺仪、加速度计以及磁编码器。陀螺仪和加速度计用于监测机器人机体的姿态和动态状态，磁编码器用于精确测量内轮的位置。接下来，本文将详细分析每种传感器的选型依据及其在机器人中的应用。

4.1.2.1 陀螺仪与加速度计选型

陀螺仪和加速度计是用于测量机器人角速度和加速度的重要传感器。陀螺仪通过测量机器人的旋转速度来推算姿态变化，而加速度计则通过测量机器人的线性加速度来推算其运动状态。对于轮圈腿式机器人来说，陀螺仪和加速度计共同工作，可以提供实时的六轴加速度和角速度信息，同姿态结算算法可以实时解算三轴姿态数据，包括横滚角度、俯仰角度以。

陀螺仪用于测量机器人的角速度，进而推算角度变化。在机器人控制系统中，角速度是评估机器人姿态变化的重要指标。为保证控制系统的高效性和准确性，陀螺仪需要具备较高的精度和稳定性。常见的陀螺仪选型标准包括量程、灵敏度和噪声水平。

加速度计用于测量机器人各个方向上的线性加速度，特别是在机器人运动过程中，它可以提供重要的加速和减速信息。根据加速度计的量程和精度要求。

在本文机器人中选择的是 BMI088 六轴惯性单元，BMI088 是博世推出的一款高性能六轴传感器，集成了三轴加速度计和三轴陀螺仪，专为需要高精度和低功耗的应用而设计。其主要的性能参数如表 4.3 所示。

表 4.3 BMI088 详细参数列表

Table 4.3 BMI088 Detailed Parameter List

参数项	具体参数
传感器类型	六轴惯性测量单元（IMU），集成三轴加速度计和三轴陀螺仪
加速度测量范围	±2g、±4g、±8g、±16g
角速度测量范围	±125°/s、±250°/s、±500°/s、±1000°/s、±2000°/s
供电电压	1.8V 至 3.6V
接口类型	I2C/SPI
工作温度范围	-40°C 至 85°C
特殊特性	小型化封装、低功耗、高抗干扰

4.1.2.2 磁编码器选型

磁编码器是一种常用于精确测量角度的传感器，特别是在内轮位置和角度测量中，磁编码器能够提供高精度和高响应速度的数据。磁编码器的选型依据主要包括分辨率、线型范围和输出方式。

本文选择的磁编码器的型号为 AS5047P，AS5047P 是一款高分辨率旋转位置传感器，广泛应用于角度测量，能够提供精确的位置反馈，特别是在内轮位置和角度测量中，具有极高的精度和快速响应能力。磁编码器的精确控制对于轮圈腿式平衡机器人保持稳定运行和高效控制至关重要。其主要参数如表 4.4

所示。

表 4.4 AS5048P 详细参数列表
Table 4.4 AS5048P Detailed Parameter List

参数项	具体参数
传感器类型	磁编码器，提供绝对角度位置测量
测量范围	14 位分辨率（0°~360°）
供电电压	3V 至 3.6V 或 4.5V 至 5.5V
接口类型	SPI
工作温度范围	-40°C 至 125°C
特殊特性	高分辨率、高速测量（最高 28,000 RPM）、低动态角度误差

4.1.3 其他硬件搭建

4.1.3.1 电池选型

在本文中，机器人系统采用 24V 的直流电机作为驱动装置。为了确保电机和控制系统的正常运行并满足功率需求，本文选择了 DJI 公司的 TB48S 型号航模电池作为机器人动力源。TB48S 电池具备较高的能量密度和放电能力，能够提供持续且稳定的电力支持，确保机器人能够高效执行各项任务，如行走、站立、跳跃等。该电池容量 5700mAh，能量为 129.96Wh，额定放电功率为 5C，经计算最大放电功率为：

$$P_{\max} = 22.4V \times 5.7A \times 5 = 633.7W \tag{4.8}$$

满足机器人电机功率运行的需求。

4.1.3.2 嵌入式控制板选型

本文中，为了实现高效的控制与数据处理，机器人系统选择了 RoboMaster C 型控制板 作为嵌入式主控单元。该控制板具备丰富的接口、强大的计算能力以及可靠的性能，能够满足轮圈腿式平衡机器人对实时计算、通信和传感器数据处理的需求。RoboMaster C 型控制板搭载的主控芯片为 STM32F407，它是 STMicroelectronics 生产的一款高性能 32 位 ARM Cortex-M4 微控制器。

STM32F407 提供了强大的计算能力和丰富的外设支持，特别适用于高性能嵌入式系统。RoboMaster C 型控制板具有多种通信接口，如：串口（USART），SPI，I2C，CAN。这些接口为机器人系统的各个模块之间的高效通信提供了便利。同时 RoboMaster C 型控制板板载了 BMI088 陀螺仪芯片提供了三轴加速度计和三轴陀螺仪的功能，能够实时获取机器人的姿态信息。STM32F407 的高频率和强大的浮点计算能力使得主控板能够快速处理来自各传感器的数据，并通过高效的控制算法实时调整电机输出和姿态控制。在需要频繁调整姿态或执行复杂运动的情况下，主控板的计算能力和实时性能够确保系统快速响应外部变化，并稳定运行。通过结合实时操作系统（RTOS），RoboMaster C 型控制板能够有效管理多任务操作，确保系统中各个模块的协调运行。在复杂的运动控制（如平衡、跳跃、行走等）和姿态调整过程中，主控板可以高效地执行任务调度，并实现精确的控制。

根据以上硬件选型最终硬件搭建框图如图 4.3 所示：

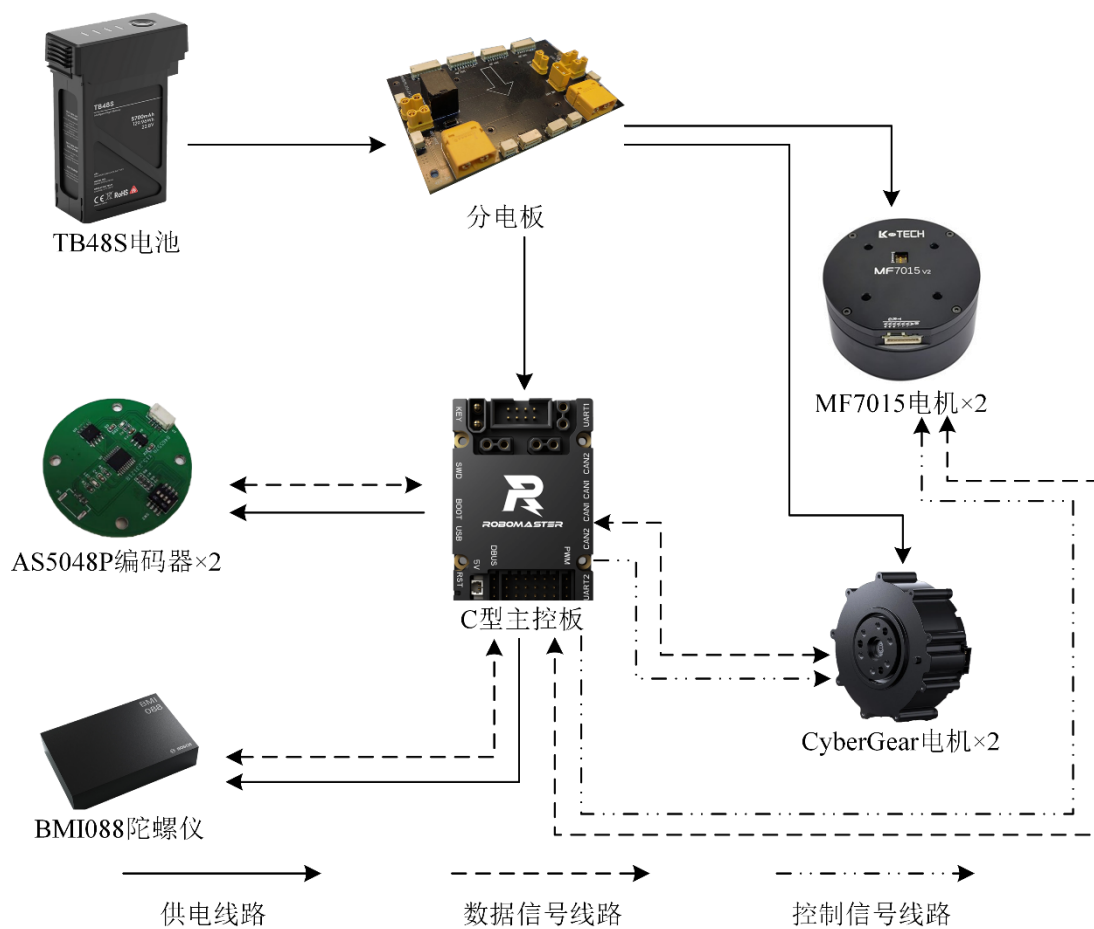


图 4.3 硬件框架示意图

Fig. 4.3 Hardware framework schematic

4.2 控制框架与算法搭建

在机器人的控制系统中，软件框架和算法的搭建是确保机器人能够稳定、高效运行的关键。为了实现机器人复杂任务的控制和实时响应，需要构建一个集成化的软件框架，包括控制框架和调试框架。本章将对控制框架的搭建和调试框架的设计进行详细说明，以确保系统能够在硬件平台上稳定运行，并实现复杂任务的实时控制。

控制框架是机器人的核心软件部分，负责处理传感器输入、执行控制算法并输出控制信号，调节机器人运动和姿态。在本文中，控制框架将分为几个主要模块，每个模块负责机器人系统中的不同功能，确保机器人能够在动态环境中平稳运动。

传感器数据采集模块：负责从机器人的各种传感器如 BMI088 陀螺仪、磁编码器 etc 中获取实时数据，并对数据进行预处理如滤波等，将有效的状态信息

反馈给控制系统。

状态估计与融合模块：根据传感器数据如加速度计、角速度计、内外轮位置与速度等，结合卡尔曼滤波器或其他状态估计算法，估计机器人的当前姿态、速度等状态变量。

控制算法模块：实现机器人的运动控制算法，如 LQR、PID 等控制算法，根据估算的状态信息计算所需的控制力矩或速度，并将计算结果传递给执行模块。

通信与接口模块：管理控制板与各外设（如电机驱动器、传感器等）的通信，确保数据传输的实时性和稳定性。

具体的控制框架如图所示：

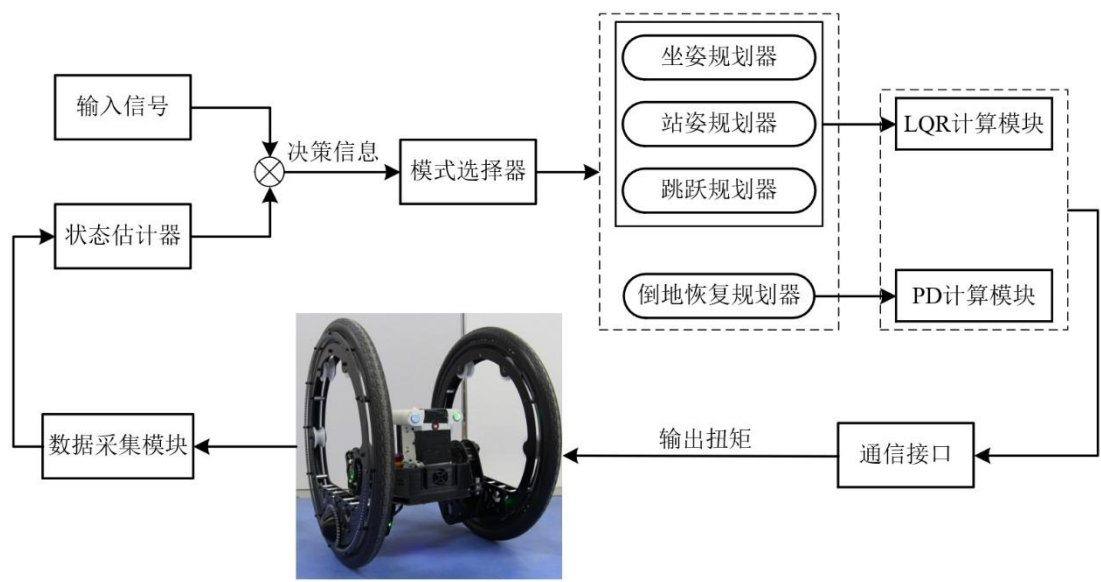


图 4.4 软件控制框架图

Fig. 4.4 Software Control Framework Diagram

其中为获得机器人的三轴角度信息，本文采用了二阶互补滤波算法。加速度计能够提供机器人的重力加速度信息，因此能够精确测量机器人的俯仰角和横滚角。然而，加速度计受到外部加速度和震动的影响，其在动态环境下的表现较差。陀螺仪通过测量角速度来计算角度变化，它在动态情况下表现出色，能够提供实时的角度变化。然而，陀螺仪在长时间使用时会积累漂移误差。为了克服加速度计和陀螺仪的局限性，二阶互补滤波算法结合了两者的优点。加速度计在静态或低速时提供准确的角度估计，而陀螺仪则在动态时提供精确的角速度变化信息。

二阶互补滤波的核心思想是通过对加速度计和陀螺仪信号进行低通滤波和

高通滤波相结合，得到更精确的角度估计。具体步骤为根据加速度计的输即 a_x, a_y, a_z 可以通过重力加速度分量计算机器人的俯仰角（Pitch）和横滚角（Roll）。假设加速度计测得的加速度值在三轴上分别为 a_x, a_y, a_z ，俯仰角和横滚角的计算公式为：

$$\begin{aligned}\theta_{acc} &= \arctan\left(\frac{a_y}{\sqrt{a_x^2 + a_z^2}}\right) \\ \phi_{acc} &= \arctan\left(\frac{-a_x}{a_z}\right)\end{aligned}\quad (4.9)$$

其中， θ_{acc} 为俯仰角， ϕ_{acc} 为横滚角。

通过陀螺仪输出的角速度数据 ω_{gyro} ，可以通过积分计算角度的变化。假设陀螺仪提供的角速度数据为 $\omega_x, \omega_y, \omega_z$ ，时间步长为 Δt ，角度的变化量通过对角速度的积分得到：

$$\theta_{gyro}(t) = \theta_{gyro}(t-1) + \omega_{gyro}(t)\Delta t \quad (4.10)$$

二阶互补滤波器通过融合加速度计和陀螺仪的角度数据来得到最终的估计角度。在二阶互补滤波中，先对加速度计和陀螺仪的信号进行高通和低通滤波，然后将两者的输出进行加权平均，以此来得到最终的角度估计。设定滤波器的时间常数 τ ，计算过程如下：

低通滤波器（加速度计角度信号）：

$$\theta_{acc}(t) = \alpha \cdot \theta_{acc}(t-1) + (1-\alpha) \cdot \theta_{gyro}(t) \quad (4.11)$$

其中 α 是加权系数，决定了加速度计和陀螺仪的融合权重。

高通滤波器（陀螺仪角度信号）：

$$\theta_{gyro}(t) = (1-\alpha) \cdot \theta_{gyro}(t-1) + \alpha \cdot \theta_{acc}(t) \quad (4.12)$$

最终输出的估计角度通过加权平均得到：

$$\theta_{final} = \alpha \cdot \theta_{acc}(t) + (1-\alpha) \cdot \theta_{gyro}(t) \quad (4.13)$$

其中， α 是调节权重的系数，通常取值在 0 到 1 之间。较大的 α 值会使滤波器更倾向于使用加速度计数据，而较小的 α 值则会偏向于使用陀螺仪数据。通过上述滤波步骤，可以得到机器人的三轴角度（俯仰角、横滚角、航向角）信息，并根据需要进行实时更新。最终的估计角度将作为输入反馈到控制系统，用于调整机器人的姿态和运动控制。

第 5 章 实物样机实验并分析

本章主要介绍了基于轮圈腿式平衡机器人完成的实物样机实验。通过不同实验场景的测试，验证了机器人在实际环境中的运动能力、控制精度以及稳定性。实验部分涵盖了多种任务，如坐姿行走、站立、跳跃、上台阶和倒地自起恢复等。通过这些实验，不仅验证了设计方案的可行性，还为后续优化和改进提供了宝贵的数据和分析结果。

为了实时监控机器人在实验过程中的状态，我们使用了无线串口模块，将机器人本体的各种重要数据，如机器人内外轮位置，速度，各电机扭矩，机器人三轴姿态角度等无线传回上位机进行记录。所有这些数据通过无线串口模块实时传输到上位机，用于进一步的分析和记录。上位机通过软件系统接收并存储这些数据，并进行后续的数据处理、可视化展示和实验分析。

为了进一步提高实验数据的准确性，我们采用了运动捕捉设备来采集机器人在实验过程中的位移和速度。运动捕捉系统通过识别机器人机体上的特征点，实时跟踪机器人在实验环境中的运动轨迹。这些特征点通过在机器人机体上粘贴具有高对比度的标记物来实现精确跟踪。通过对比这两类数据，我们可以评估机器人的控制系统是否能够精准地反映机器人的实际运动状态。尤其是在机器人执行快速运动、变向或跳跃等动作时，外部传感器的高精度数据能够作为参考，帮助我们验证内部传感器的表现。

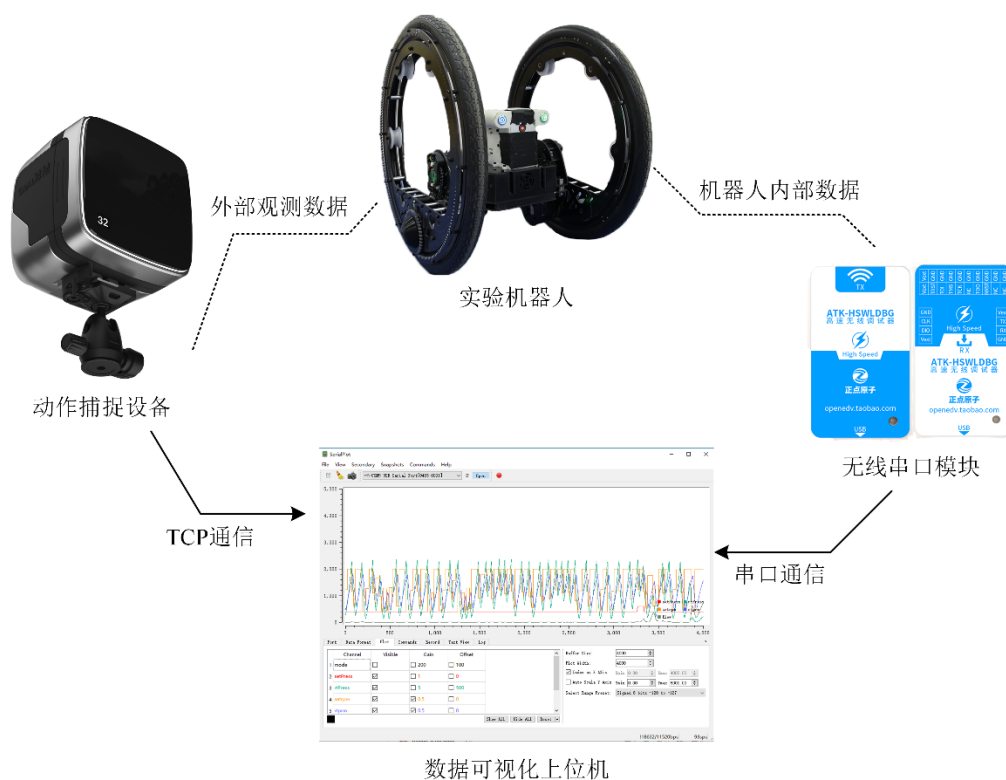
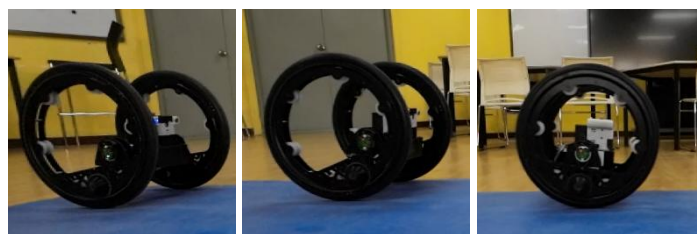


图 5.1 实验流程图

Fig. 5.1 Experimental flow chart

5.1 坐姿行走实验

在坐姿行走实验中，机器人保持较低的重心，以坐姿姿态进行前后往返加速减速运动。为考察其在该姿态下的运动稳定性与姿态控制性能。实验过程图如图 5.2 所示，其中前三张图为机器人加速过程，机器人内轮抬升电机通过带动机身前移重心提供前进加速度实现加速，在加速过程中机身始终维持水平姿态；后三张图为机器人减速过程，在减速过程中抬升电机通过反向抬升机身减速，在整个过程中机体的俯仰角始终在小范围内变化。



(a)初始姿态

(b)抬起身体加速

(c)加速过程



(d)后抬身体减速 (e)减速过程 (f)恢复静止
图 5.2 坐姿行走实验, (a)-(c)为加速过程, (d)-(f)为减速过程

Fig. 5.2 Sitting walking experiment, (a)-(c) are acceleration processes, (d)-(f) are deceleration processes

通过回传机身姿态解算角度和运动捕捉设备, 我们记录了机器人的速度和俯仰角随时间的变化, 如图所示。从速度曲线可见, 机器人在行走过程中速度呈周期性变化, 最高可达约 2.7 m/s , 最低约 -2.7 m/s 。俯仰角曲线则表明, 机器人在行走过程中整体保持了相对稳定的俯仰姿态, 波动范围较小, 约在 -5° 至 5° 之间。这说明在坐姿状态下, 控制系统能够在加减速过程中只通过抬升机体提供足够的加速度并及时响应并修正机器人姿态, 避免过度前倾或后仰。

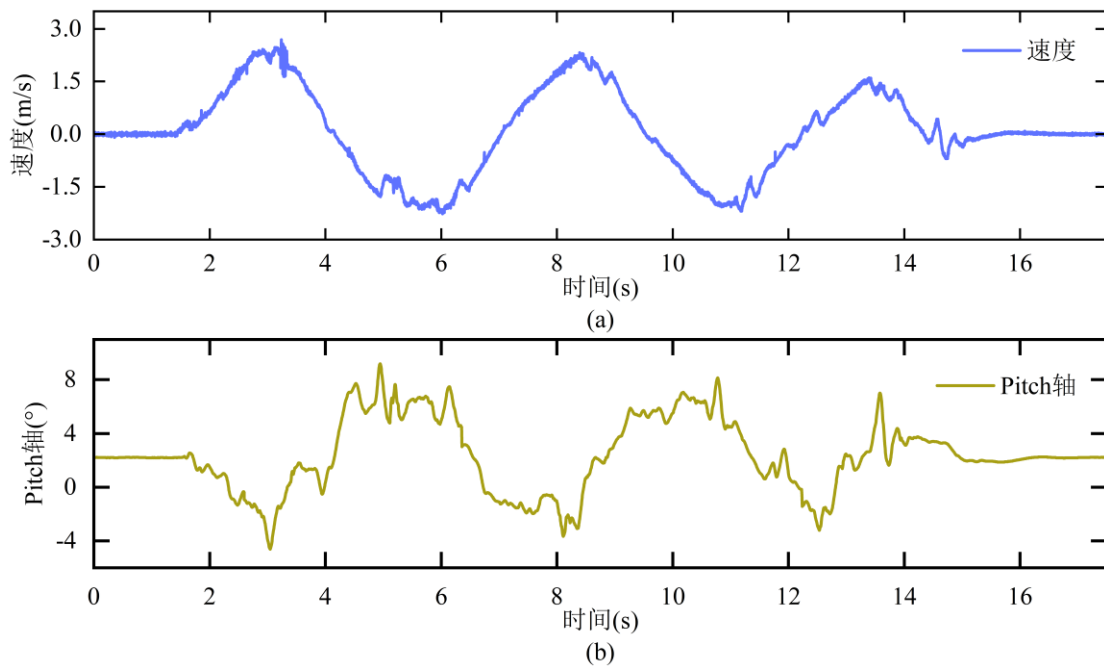


图 5.3 坐姿状态下加速减速实验图

Fig. 5.3 Acceleration and deceleration experiments in sitting position

综合实验结果可知, 坐姿行走不仅保证了机器人在较低重心下的平衡, 而且在速度变化时也能维持基本稳定的姿态。该实验验证了机器人在坐姿模式下的行走能力和控制算法的有效性, 为后续复杂地形或动态场景下的运动控制奠定了基础。

5.2 站立实验

站立实验涉及从坐姿到站立的过渡过程。在这一过程中，我们记录了机器人的高度和俯仰电机的扭矩反馈，如图所示。实验结果表明，轮圈腿式机器人在 0.7 秒内完成了从坐姿到站立的过渡。机器人的机体表面高度从 300mm 升高至 620mm，并且在完成站立后，机体高度稳定保持。

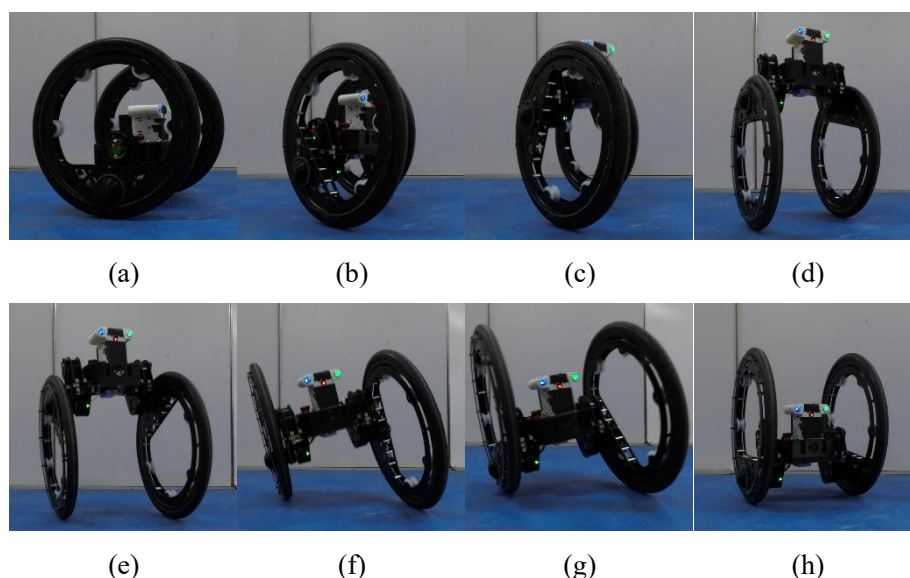


图 5.4 站姿实验，(a)-(d)为站立过程，(e)-(h)为恢复坐姿过程

Fig. 5.4 Standing experiment, (a)-(d) for standing process, (e)-(h) for return to sitting process

根据实验数据，图中的两个曲线分别表示左侧和右侧电机的扭矩反馈。可以看到，在站立过程中，俯仰电机的扭矩呈现出明显的变化，左侧和右侧电机的扭矩差异表明机器人需要通过调整两侧的扭矩输出，以保持平衡并完成站立动作。下方的曲线展示了机器人高度随时间的变化，机器人的高度从 300mm 逐渐上升至 620mm，并在站立后维持在一个稳定值。

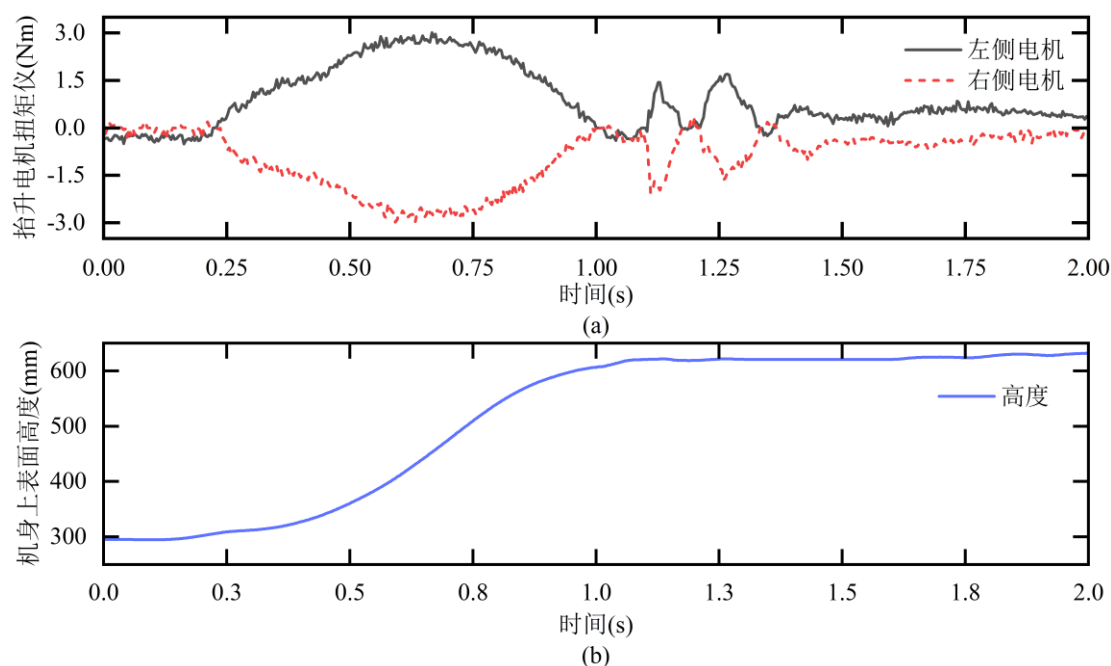


图 5.5 机器人从坐姿切换到站姿状态抬升电机扭矩与机体上表面高度

Fig. 5.5 Robot switching from sitting to standing state lifting motor torque and height of the upper surface of the body

通过这一实验，我们验证了机器人在动态调整中能够顺利完成从坐姿到站立的转换，并且在站立时能够维持稳定的姿态。

5.3 跳跃实验

跳跃实验分为两个部分：一是跑步过程中向前跳跃，二是静止状态下通过两侧轮子交叉跳跃。

在第一个实验中，我们设置两侧内轮的抬升角度，使其同方向旋转一圈。由于角动量守恒，当机体停止旋转时，外轮将围绕机体的轴线旋转，从而实现跳跃。在跳跃过程中，我们记录了机体顶部的高度和腿部电机的扭矩输出，数据如图 14 所示。

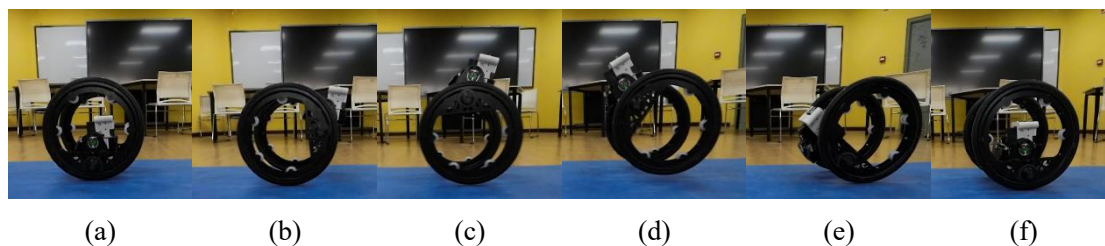


图 5.6 向前跳跃实验

Fig. 5.6 Forward jump experiment

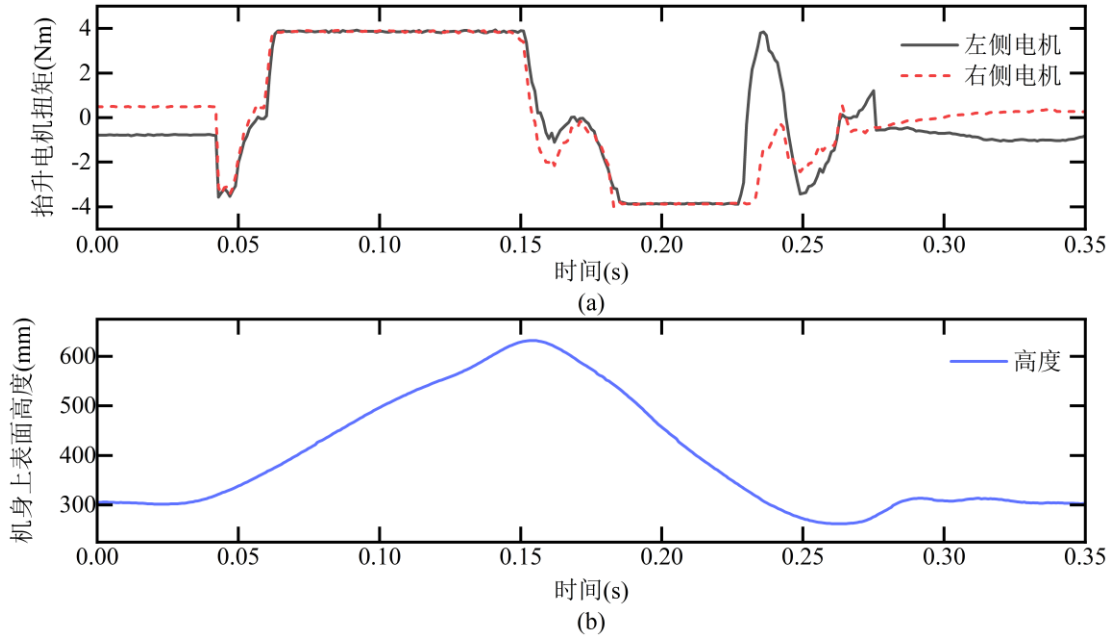


图 5.7 两侧轮圈腿同向跳跃抬升电机与机身上表面高度

Fig. 5.7 Both rim legs jump in the same direction to raise the height of the motor to the upper surface of the fuselage

在第二个实验中，机器人处于静止坐姿状态，我们设置两侧内轮的抬升角度，使其分别逆向旋转一圈。两侧内轮产生的向上抬升扭矩将提升机体，从而产生向上的加速度，实现跳跃。在跳跃过程中，我们记录了与上次实验相同的数据，数据如图 15 所示。

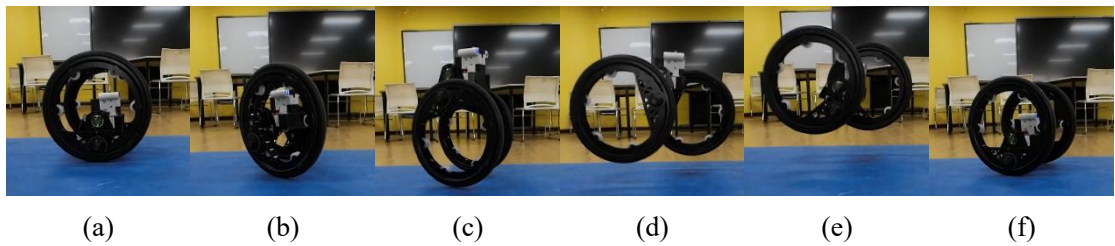


图 5.8 原地交叉跳跃实验

Fig. 5.8 Cross jump experiment

通过这两个实验，我们验证了机器人在不同运动状态下的跳跃能力。第一部分的实验展示了机器人的跑步跳跃能力，外轮的旋转起到了推动机体跳跃的作用；第二部分则验证了机器人在静止状态下利用内轮产生的抬升扭矩进行跳跃的能力。这些实验数据表明，机器人能够在不同的跳跃模式下稳定地完成跳跃动作。

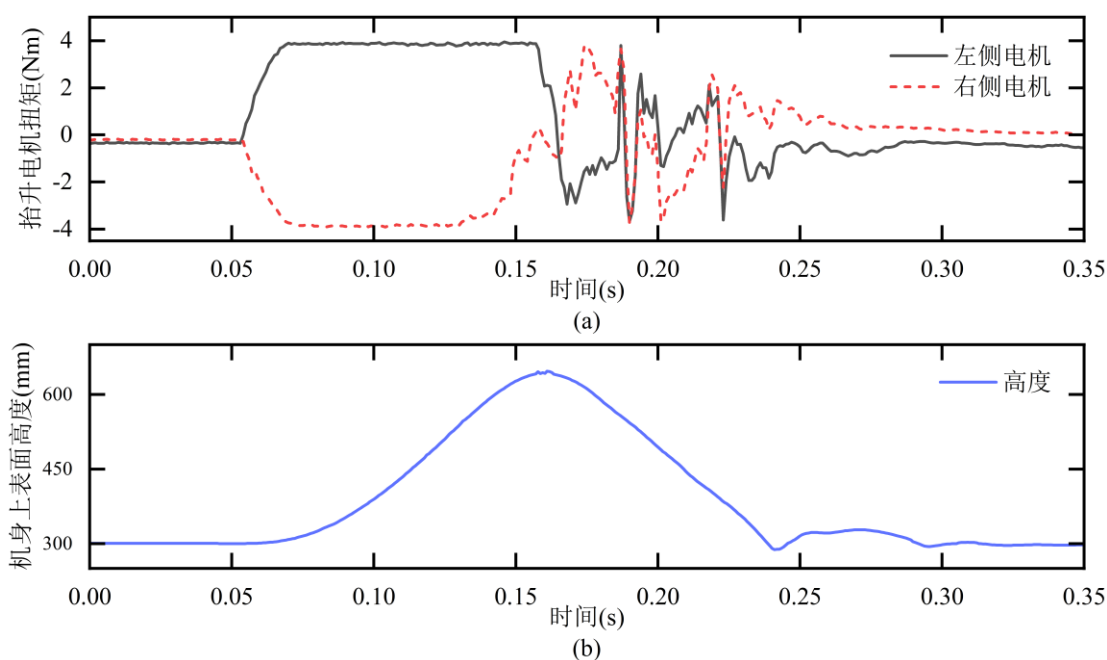


图 5.9 两侧轮圈腿反向跳跃抬升电机扭矩与机身上表面高度

Fig. 5.9 Reverse jumps on both rim legs lift the motor torque to the height of the upper surface of the fuselage

5.4 上台阶实验

在上台阶实验中，我们选择了一个连续的台阶，台阶高度为 100 mm，宽度为 300 mm。机器人通过调整内轮的角度来改变整体重心位置，最终成功爬上台阶。在整个过程中，我们记录了机体的俯仰角度和俯仰电机的扭矩输出，如图所示。实验结果显示，机器人在爬升过程中，机体俯仰角度的最大变化不超过 35° ，而在下台阶过程中，俯仰角度的变化不超过 15° 。

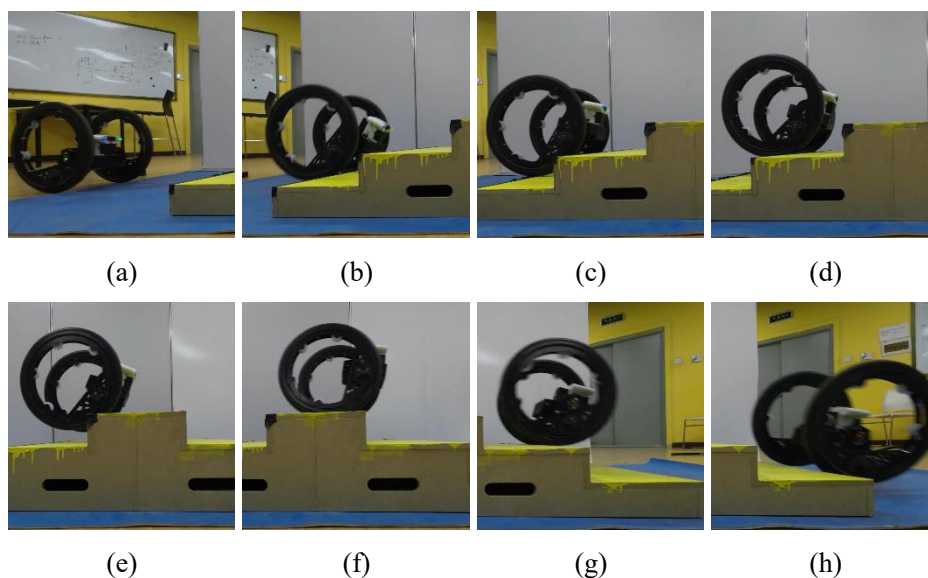


图 5.10 上台阶实验

Fig. 5.10 step-up experiment

这些实验数据表明，机器人能够通过精确控制内轮角度，调整重心位置，从而有效应对台阶的爬升任务。在台阶的爬升过程中，机器人保持了稳定的姿态，且没有出现失衡或无法克服台阶的现象。数据也反映了机器人在不同高度的台阶上执行越障任务的稳定性和精度。

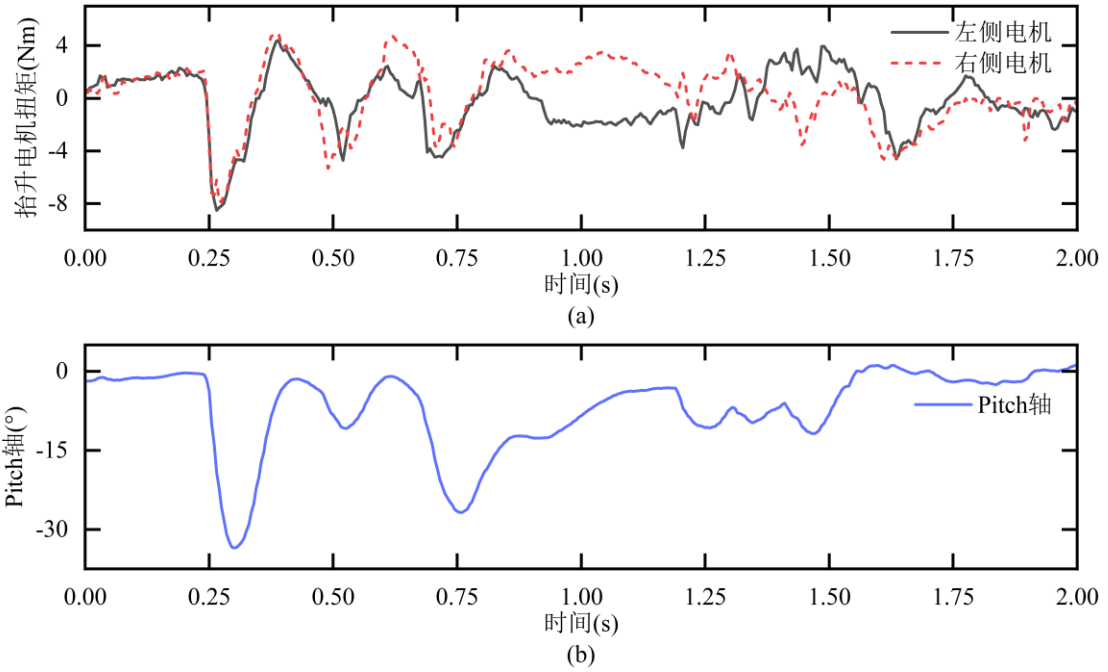


图 5.11 上台阶过程中抬升电机扭矩与机身 Pitch 轴俯仰角

Fig. 5.11 Lift motor torque and fuselage Pitch axis pitch angle during step up.

5.5 倒地自起恢复实验

在倒地自起恢复实验中，我们将机器人置于侧翻状态，并启动自我恢复程序。在这个过程中，我们记录了机器人俯仰角（Pitch）和横滚角（Roll）的变化，数据如图所示。在恢复过程中，首先是俯仰轴开始产生旋转，而横滚轴则在机体接触地面后开始恢复。整个恢复过程持续了约 5.5 秒。



图 5.12 侧向翻倒自起过程

Fig. 5.12 Lateral tip-over self-launching process

实验结果表明，机器人能够成功地从侧翻状态恢复到站立姿势。在恢复过程中，机器人通过精确的控制调整俯仰角和横滚角，逐步恢复平衡并回到正常的站立状态。通过分析这些数据，我们可以进一步优化机器人的恢复控制策略，以提高其在复杂环境中的自起能力和稳定性。

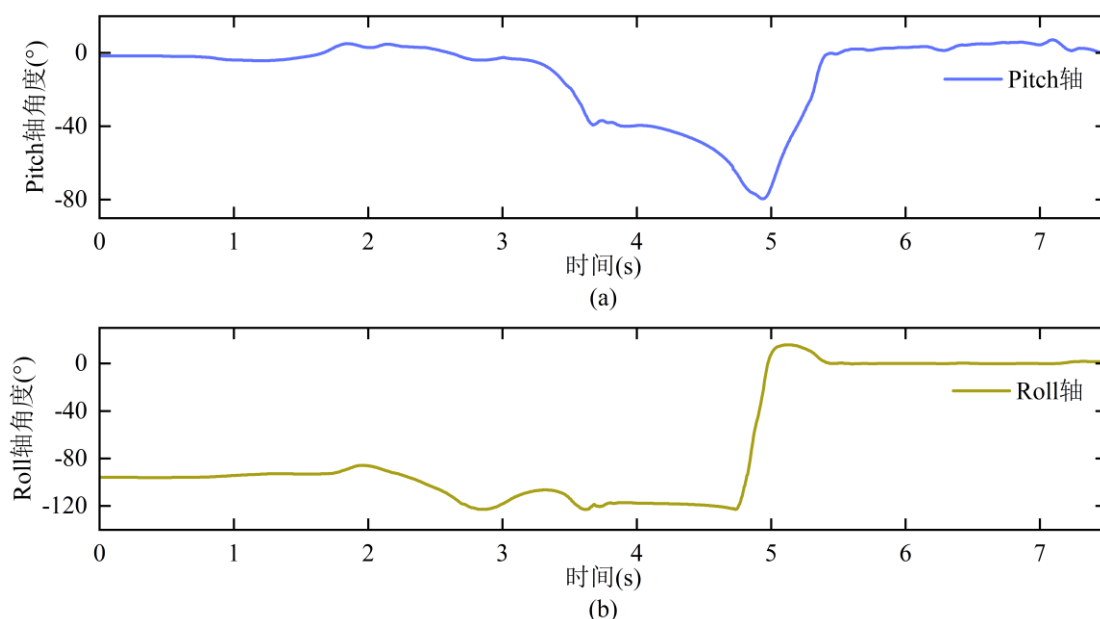


图 5.13 机器人侧向翻倒恢复过程 Pitch 轴俯仰角和 Roll 轴横滚角角度

Fig. 5.13 Pitch axis pitch angle and Roll axis roll angle during robot lateral tip-over recovery.

第 6 章 总结与展望

参考文献

- Wardana A A, Takaki T, Aoyama T, et al. Development of a control system for a stair-climbing inverted pendulum robot[C]//2016 International Conference on Instrumentation, Control and Automation (ICA). IEEE, 2016: 1-5.
- Klemm V, Morra A, Salzmann C, et al. Ascento: A two-wheeled jumping robot[C]//2019 International conference on robotics and automation (ICRA). IEEE, 2019: 7515-7521.
- Hutter M, Gehring C, Jud D, et al. Anymal-a highly mobile and dynamic quadrupedal robot[C]//2016 IEEE/RSJ international conference on intelligent robots and systems (IROS). IEEE, 2016: 38-44.
- 官渐. 轮腿式移动机器人主/被动变形机构设计及研究[D]. 重庆大学, 2019. DOI:10.27670/d.cnki.gcqdu.2019.000192.