Stabilization of a Quintuple Inverted Pendulum System in Isaac Sim

Denys Kotelovych[1][0009−0007−7031−2500], Yurii Borzov[2][0000−0002−0604−0498], and Ihor Malets[3][0000−0001−8263−8136]

[1] Lviv State University of Life Safety, Lviv, Ukraine
d.kotelovych@ldubgd.edu.ua
[2] Lviv State University of Life Safety, Lviv, Ukraine
uob1968@gmail.com
[3] Lviv State University of Life Safety, Lviv, Ukraine
igor.malets@gmail.com

Abstract. This paper presents an overview of a quintuple inverted pendulum system within the Isaac Sim simulation environment. Inverted pendulum systems, with their inherently unstable nature, are common benchmarks in robotics and control engineering for testing new control strategies. While controlling pendulum configurations has seen significant advancements, managing the increased complexity and unpredictability of systems with multiple pendulums remains a technical challenge. This study extends the scope of existing research by introducing a more complex system with five pendulums in a stacked configuration. By utilizing the Isaac Sim environment, we demonstrate the feasibility of simulating and controlling such system, achieving stable upright positions. Additionally, this research contributes to bridging the gap between hardware and simulation by integrating with ROS2, a widely-used robotics middleware framework. Our results underscore great capabilities of Isaac Sim for modeling and testing sophisticated control algorithms and open new avenues for the development of scalable control solutions for complex multibody systems.

Keywords: inverted pendulum, linear quadratic regulator, NVIDIA Isaac Sim, system architecture, ROS2, equations of motion

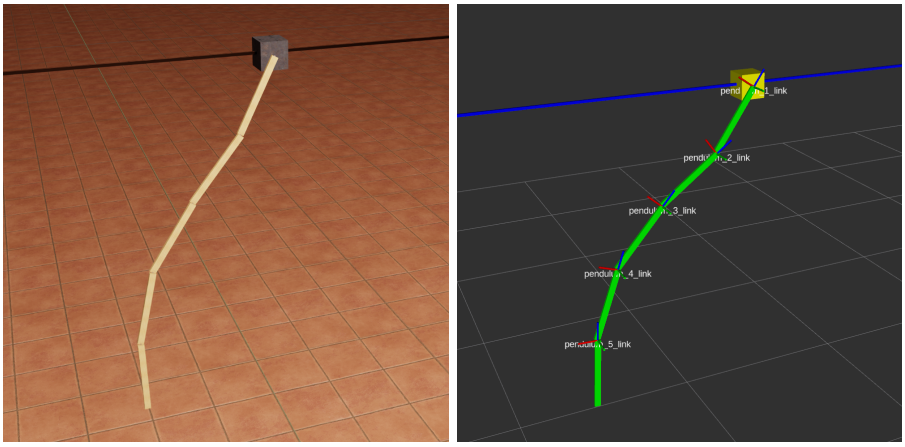
1 Introduction and Literature Review

Inverted pendulum systems have been widely studied due to their practical applications in robotics, automation, and control engineering [7]. It is a simple yet highly dynamic system that provides an ideal testbed for control and stability analysis. A classic example of an inverted pendulum system is a pendulum attached to a cart, capable of moving along a horizontal track. The challenge is to maintain the pendulum in an upright position using a control input that moves the cart back and forth along the track. This problem has been extensively studied in the literature and has led to the development of a wide range of control algorithms.

Recent research has focused on developing control algorithms for single and double inverted pendulum systems to maintain stability and perform dynamic maneuvers. One approach typically involve deriving mathematical models of the system dynamics and designing controllers based on these models. Other approaches, on the other hand, rely on machine learning and reinforcement learning techniques to learn control policies directly from data. These approaches have been successful in achieving stable upright positions and performing swing-up maneuvers for single and double pendulum systems.

Simulation environments provide a safe and cost-effective way to test and evaluate complex control algorithms before deploying them in the real world. Isaac Sim is a powerful and flexible platform for building and simulating robots and other complex systems. It allows for high-fidelity simulations of complex physical systems and provides a range of tools for designing and testing algorithms.

This paper aims to extend the scope of research on inverted pendulum control and stability by implementing a system in Isaac Sim and demonstrating the effectiveness of a hybrid control algorithm for stabilizing the system in upright positions and performing swing-up maneuvers. The results showcase the capabilities of the platform for modeling and testing for multi-pendulum systems in a virtual environment.



(a) Isaac Sim view **(b)** RViz2 view

Fig. 1: Swing-up maneuver

In the context of controlling the inherently unstable inverted pendulum (IP) system, one work explores the application of linear techniques, specifically the Proportional-integral-derivative (PID) and Linear quadratic regulator (LQR) control methods [11]. Despite their conventional use in linear systems, the study adapts these methods to effectively control the highly nonlinear IP system. The primary goal is to achieve precise cart positioning while maintaining the inverted pendulum in its upright orientation. The research highlights the practicality and robustness of LQR and PID control through extensive simulations conducted in Matlab-Simulink. Both scenarios, with and without disturbance input, are considered, revealing the advantages of the LQR control approach in optimizing system performance. Amidst the array of advanced control techniques, this investigation underscores the simplicity and effectiveness of LQR and PID control when applied to nonlinear systems, offering valuable insights for the control of complex systems like the inverted pendulum-cart dynamic system.

One interesting work introduces a concept of a 2-DOF inverted pendulum incorporated within an Omnidirectional mobile platform (OMP) [17]. The OMP utilizes three omnidirectional wheels arranged in a triangular configuration, granting it the capability to move in all directions on the floor while maintaining the balance of the 2-DOF IP. Unlike previous studies, this approach decouples the system into two subsystems: the 2-DOF IP and the OMP. It presents dynamic modeling of both components and proposes an adaptive backstepping control method, particularly addressing situations where the distance between the center of gravity of the rod and the rotary

point on the OMP is unknown. The stability and convergence of the control method are rigorously established through Lyapunov function analysis offering both simulation and experimental results that provide compelling evidence of the controller's effectiveness.

Another paper delves into the balance control challenges of an inverted pendulum system which is uniquely combined with an omni-directional wheeled mobile robot [10]. This system can be configured either as a rotary inverted pendulum or as a spherical one. Notably, the omni-directional wheeled mobile robot is designed to deliver translational force for the balancing of the spherical pendulum and provide a moment for the rotary counterpart. The authors put forth detailed dynamic models for these configurations and design stabilizing controllers leveraging second-order sliding mode control. A salient feature of this work is its emphasis on robustness to uncertainties and disturbances, wherein the second-order sliding mode control showcases superiority over conventional sliding mode control and LQR control in various scenarios, such as managing significant initial deviations from the pendulum's upright position. This work sheds light on the intricacies of integrating complex systems and offers a comprehensive approach to balancing control in multifaceted environments.

Another method utilizes a pole-independent, single-input, multi-output explicit linear Model predictive control (MPC) strategy to achieve stability around desired equilibrium points [13]. By relying on a generalized prediction model and a quadratic cost function with just two adjusted control parameters, the controller ensures a priori prescribed system gain margin and rapid pendulum response, all while simplifying the tuning process. Furthermore, the approach provides practical insights into the impact of velocity cost function weight factors on system transient performance, offering valuable guidelines for real-world implementation.

In the context of hardware implementation, recent research presents a contribution where a triple inverted pendulum system is successfully implemented using state-of-the-art technology and a custom ARM/FPGA-based control platform [15]. This work addresses the challenges of hardware precision, low-latency wireless data communication for feedback, and dynamic control requirements. It highlights the importance of both control theory and hardware design for creating a well-functioning physical model of complex multi-link inverted pendulum systems.

Zhang, Wang, and Li's study [18] introduces stabilization method for a quadruple inverted pendulum. Utilizing a combination of adaptive fuzzy control and variable gain H ∞ regulator, this work addresses the complex dynamics of high-degree-of-freedom systems.

Chen's dissertation [8] investigates the dynamics of multiple-degree-of-freedom inverted pendulums, particularly focusing on stabilization under periodic vertical forces. This work extends beyond single-degree systems, combining numerical simulations and experiments to analyze stabilization effects through amplitude and frequency variations. Bondada and Nair's 2023 study [6] examines the dynamics of multiple pendulum systems with a moving pivot. Their research provides insights into the complex behavior of these systems when subjected to both translational and tilting movements.

Inverted pendulum systems present a challenging control problem, with a wide variety of approaches and techniques proposed to stabilize the system. Linear and nonlinear control algorithms, as well as machine learning-based approaches, have all been used to control inverted pendulum systems. While each approach has its strengths and weaknesses, there is no one-size-fits-all solution to controlling inverted pendulums. Instead, the choice of control algorithm will depend on the specific requirements of the application, such as the desired performance metrics, computational resources available, and system parameters.

This paper discusses our efforts in simulating a pendulum system using the Isaac Sim environment. Our primary focus has been on developing a robust Robot Operating System 2 (ROS2) infrastructure to facilitate control and simulation, acknowledging that while our work is an initial step towards understanding complex systems, it remains primarily explorative. By leveraging

Isaac Sim's simulation capabilities, we aim to provide insights into the dynamics and potential control strategies for multi-pendulum systems. This research contributes to the field by offering a foundation for future studies, rather than definitive solutions, and highlights the challenges and potential of simulation in advancing control theory.

2 Materials and Methods

2.1 System Overview

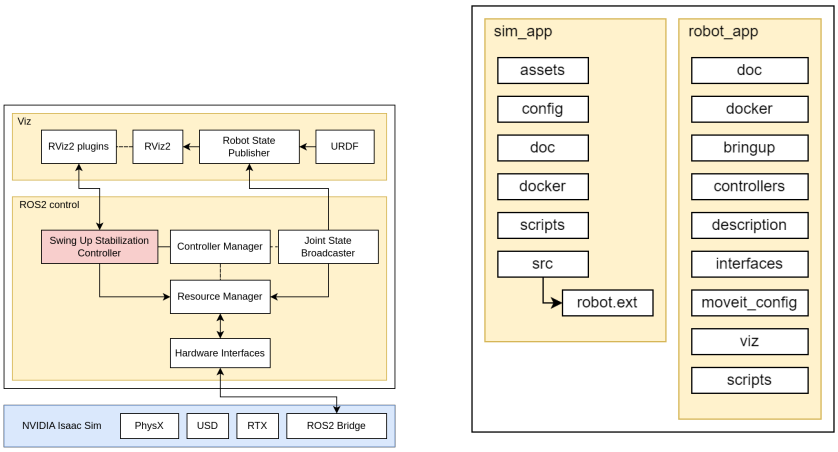
To facilitate the modeling of the control and stabilization of a quintuple inverted pendulum system and to study the stabilization process, a system architecture has been developed utilizing the Isaac Sim platform [4]. This architecture comprises components and packages structure highlighted in Figure 2a, 2b respectively:

- **Isaac Sim simulator**: a powerful tool for robotic simulations which uses the Universal Scene Description (USD) format to represent the custom robot model, including the physical characteristics and dynamics of the quintuple inverted pendulum.
- **ROS2 Bridge Extension**: serves as a bridge between the Isaac Sim and the ROS2 ecosystem allowing to control and monitor the simulated robot using ROS2.
- **Stabilization controller**: primary role is to control the linear module of the quintuple inverted pendulum stabilizing in an upright position. It receives sensor data, processes it, and sends control commands to the specific hardware interface.
- **ROS2 control**: provide a framework for controlling and managing robot hardware. They offer standardized interfaces for robot hardware and simulation components.
- **Hardware interfaces**: This component represents the custom hardware interface abstractions specific to the robot. It acts as a bridge between the ROS2 and the actual hardware components of the robot. It translates high-level control commands from ROS2 into low-level commands that the hardware can understand.
- **RViz2**: a powerful visualization tool in the ROS 2 ecosystem. It allows users to visualize robot data, sensor data, and other relevant information in a graphical interface. It's commonly used for debugging, monitoring, and understanding the behavior of robots [3].
- **RViz2 plugins**: an extendable modules for Rviz2. They enhance the system's capabilities by enabling the addition of custom displays and interactive elements specific to the robotic application's requirements.

2.2 NVIDIA Isaac Sim

Isaac Sim offers several critical features that make it an ideal choice for simulating multi-pendulum systems:

- **Physics simulation**: Isaac Sim incorporates a physics engine that provides accurate and realistic simulations of rigid body dynamics, including collision detection and response.
- **Sensor simulation**: The environment supports a variety of sensors, including cameras, lidar, and depth sensors. These sensors can be integrated into the simulated robotic system, allowing for sensor-based control and perception experiments.
- **Customizable environments**: Isaac Sim enables the creation of customized environments, including both physical layouts and virtual terrains. This flexibility is valuable for designing scenarios that mimic real-world conditions.



(a) Components **(b)** Apps structure

Fig. 2: Architecture overview

- **Integration with ROS2**: The simulation environment seamlessly integrates with the ROS2, a widely used robotics middleware. This integration simplifies the development and testing of control algorithms, as it provides a framework for managing hardware interfaces and communication between system components.
- **Support for GPU Acceleration**: NVIDIA's GPU technology accelerates simulations, making it possible to simulate complex systems efficiently. This is particularly advantageous for real-time control experiments and rapid prototyping.
- **Isaac Gym**: Extends simulator capabilities to support Reinforcement learning (RL) experiments allowing for the training of control policies for robotic systems in a simulated environment [12].

2.3 ROS2

The integration of ROS2 into the simulation framework provides several advantages [16]:

- **Modular framework**: offers a modular and flexible framework for managing robot hardware interfaces, communication between nodes, and high-level robot behaviors.
- **Hardware interfaces**: allows easy integration with various hardware interfaces, facilitating the connection of virtual sensors and actuators to the simulated robotic system.
- **Communication**: provides a communication infrastructure that facilitates data exchange between different components of the system. This enables the seamless flow of information, enhancing system control and coordination.
- **Real-time capabilities**: ROS2 includes real-time communication mechanisms that are crucial for applications requiring precise timing and control. It allows for the integration of real-time controllers and systems.
- **Cross-Platform compatibility**: ROS2 supports various operating systems, making it versatile for deployment on different hardware platforms. This cross-platform compatibility simplifies the deployment of robotic systems across a range of environments.

– **Community and ecosystem**: ROS2 benefits from a thriving community and an extensive ecosystem of libraries, tools, and packages. This ecosystem accelerates development by providing access to pre-built components and solutions.

2.4 ROS2 Control Framework

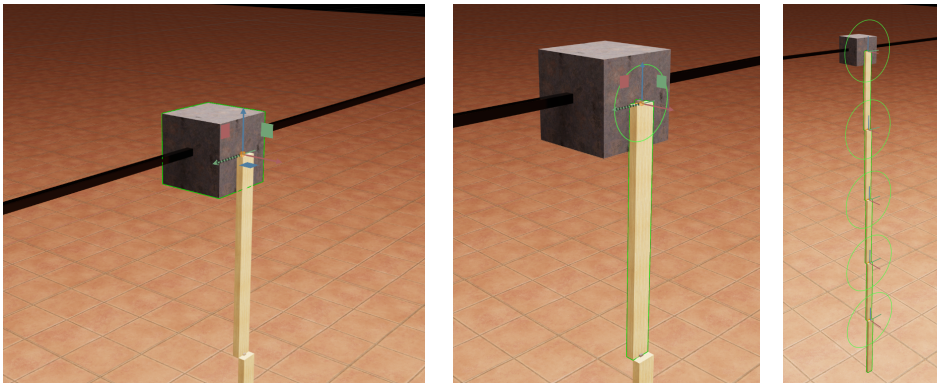
The ROS2 Control framework is a pivotal component of our system architecture, offering a comprehensive set of tools and capabilities for controlling robotic systems [5]. Key features of the framework include:

– **Hardware abstraction**: ROS2 Control provides a standardized and modular interface for hardware abstraction. It allows for the seamless integration of various robot components, including sensors and actuators, while maintaining hardware independence.
– **Controller management**: The framework offers a controller management system that simplifies the deployment and execution of control algorithms. It supports both position and velocity controllers, making it versatile for a wide range of robot types.
– **Real-time control**: ROS2 Control is designed to meet real-time requirements, enabling precise control of robotic systems. It includes features for managing control loops with low-latency and high-frequency updates.
– **Configuration flexibility**: The framework allows for dynamic reconfiguration of control parameters, which is crucial for adapting to different tasks and environments. This flexibility enhances the adaptability of the robot's control system.
– **Simulation integration**: ROS2 Control seamlessly integrates with simulation environments like the Isaac Sim. This integration enables the development, testing, and validation of control algorithms in a simulated environment before deploying them to physical robots.

2.5 Unified Robot Description Format (URDF)

The robot consists of several interconnected components, each defined using `xacro` macros and includes for modularity and ease of customization:

1. **Base link**: The robot's base link, denoted as `base_link`, serves as the central reference point for the entire system.
2. **Rail**: The component represents a rail or track on which the robot moves. It's attached to the `base_link` and is designed to constrain the robot's motion along a linear path.
3. **Foots**: These components, denoted as `foot_1_base_link` and `foot_2_base_link`, are attached to the `base_link` and represent the supports or feet on which the robot rests. They also serve as the connection points for the robot's legs.
4. **Legs**: The `leg_1_base_link` and `leg_2_base_link` components are connected to the foot links and represent the robot's legs. These legs are responsible for supporting and controlling the pendulum segments.
5. **Caret**: The `caret_base_link` component is connected to the rail and represents a movable carriage-like structure. It can slide along the rail to change the robot's configuration, which is presented in Figure 3a.
6. **Connectors**: These components, denoted as `connector_1_link` through `connector_5_link`, serve as connectors between the pendulum segments and the caret. They facilitate the attachment of pendulum segments and enable their rotational movement.
7. **Pendulums**: The robot comprises five pendulum segments, each represented by components such as `pendulum_1_link` through `pendulum_5_link`. These pendulum segments are connected in series and can rotate freely, which is preseteted in Figure 3b, 3c).



(a) Caret joint **(b)** Pendulum joint **(c)** Pendulum joints

Fig. 3: Joints model details

2.6 Mathematical Modeling

The mathematical model includes kinematics, dynamics, and the derivation of equations of motion, addressing the complexity and nonlinear dynamics arising from the coupling of multiple pendulums.

The kinematic analysis involves examining the motion of each pendulum and the carriage. The horizontal and vertical displacements, along with the respective velocities of each pendulum's center of mass, are given by the following equations:

$$x_i = X_c + \sum_{k=1}^{i-1} l_k \sin(\theta_k) + \frac{1}{2} l_i \sin(\theta_i), \tag{1}$$

$$y_i = -\sum_{k=1}^{i-1} l_k \cos(\theta_k) - \frac{1}{2} l_i \cos(\theta_i), \tag{2}$$

$$\dot{x}_i = \dot{X}_c + \sum_{k=1}^{i-1} l_k \dot{\theta}_k \cos(\theta_k) + \frac{1}{2} l_i \dot{\theta}_i \cos(\theta_i), \tag{3}$$

$$\dot{y}_i = \sum_{k=1}^{i-1} l_k \dot{\theta}_k \sin(\theta_k) + \frac{1}{2} l_i \dot{\theta}_i \sin(\theta_i). \tag{4}$$

Various parameters and variables are defined in Table 1.

The dynamics of the system are characterized by gravitational forces, inertial forces, external forces, and the effects of friction and damping. Specifically:

- The external force $F(t)$ impacts the translational movement of the system.
- Each pendulum experiences gravitational forces and is subject to damping at its pivot.
- Friction at the carriage and damping at the pendulum pivots are modeled as viscous damping, proportional to the velocity and angular velocities.

The kinetic and potential energies, as well as the dissipative energy due to damping, are expressed as:

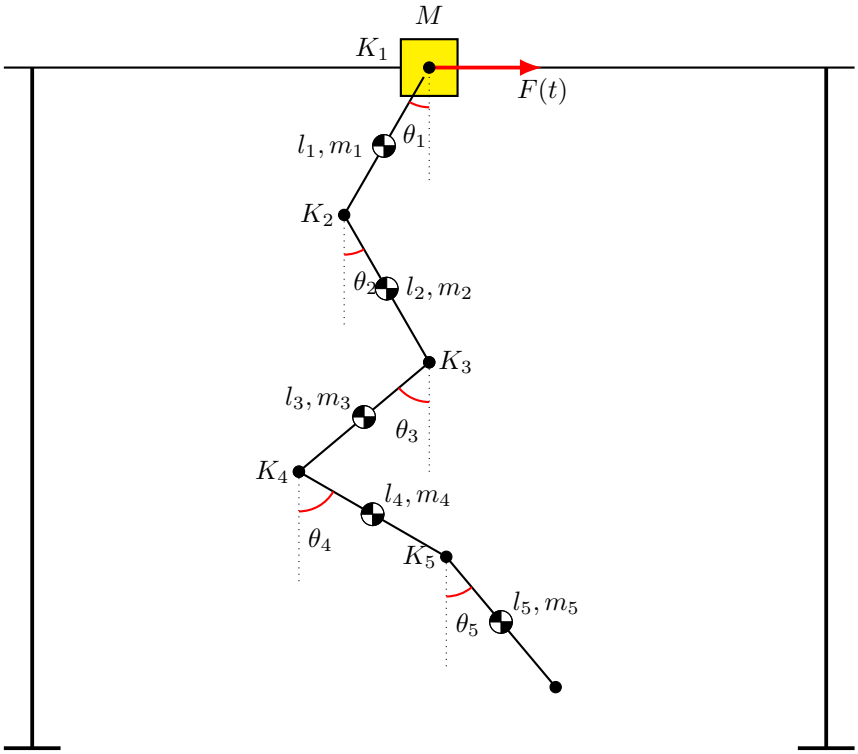


Fig. 4: Sketch diagram of the system.

Table 1: Definitions of variables and parameters.

Variable	Description
θ_i	Angular position of the i^{th} pendulum relative to the vertical.
l_i	Length of the i^{th} pendulum.
m_i	Mass of the i^{th} pendulum.
M	Mass of the carriage.
K_i	Stiffness of the hinge connecting the i^{th} pendulum to the previous one or to the carriage.
$F(t)$	External force applied to the carriage as a function of time.
x_i	Horizontal displacement of the center of mass of the i^{th} pendulum.
y_i	Vertical displacement of the center of mass of the i^{th} pendulum.
$\dot{x}_i$	Horizontal velocity of the center of mass of the i^{th} pendulum.
$\dot{y}_i$	Vertical velocity of the center of mass of the i^{th} pendulum.
X_c	Horizontal displacement of the carriage's center of mass.
I_i	moment of inertia of the i^{th} pendulum.
g	Acceleration due to gravity.
b	Coefficient of viscous friction for the carriage.
d_i	Damping coefficients for each pendulum joint.
H	Height of the stand.

$$T_{total} = \frac{1}{2}M\dot{X}_c^2 + \frac{1}{2}\sum_{i=1}^{5} m_i\left(\dot{x}_i^2 + \dot{y}_i^2\right) + \frac{1}{2}\sum_{i=1}^{5} I_i\dot{\theta}_i^2, \tag{5}$$

$$V_{total} = \sum_{i=1}^{5} m_i g h_i, \tag{6}$$

$$\tag{7}$$

where $h_i = H - y_i$ the height of the center of mass of the i^{th} pendulum, $I_i = \frac{1}{12}l_i m_i$ moment of inertia of the the i^{th} pendulum center of mass (thin rod).

The Euler-Lagrange equations, extended to include non-conservative forces, govern the dynamics of the system. These forces are incorporated through the Rayleigh dissipation function:

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{q}_j}\right) - \frac{\partial L}{\partial q_j} + \frac{\partial R}{\partial \dot{q}_j} = Q_j, \tag{8}$$

where L is the Lagrangian, defined as

$$L = T_{total} - V_{total}, \tag{9}$$

and R is the Rayleigh dissipation function:

$$R = \frac{1}{2}b\dot{X}_c^2 + \frac{1}{2}\sum_i d_i\dot{\theta}_i^2, \tag{10}$$

where $\frac{\partial R}{\partial \dot{q}_j}$ corresponds to the dissipative forces acting on the generalized coordinates q_j.

The complete set of EoMs is derived by substituting kinematic equations (1, 2, 3, 4) into dynamic equations (5, 6), and then along with the damping terms (10), into the Euler-Lagrange

equation (8) and taking derivatives (see 15, 16). Initially, Kane's method [9] was considered for its great efficiency in handling nonholonomic systems. However, opted for Lagrange's method due to its more straightforward application to multi-body systems. This process yields a comprehensive description of the motion for each component of the system, encapsulating the intricate dynamics resulting from the coupling of multiple pendulums.

The state-space representation is a widely used mathematical model for control systems. It describes the system's state at any given time and how it evolves over time in response to inputs. For the quintuple inverted pendulum system, we can define the state-space representation as follows:

$$\dot{X} = AX + Bu, \tag{11}$$

$$Y = CX + Du, \tag{12}$$

where A is the system matrix, B is the input matrix, C is the output matrix, and D is the feedforward matrix. These matrices are determined based on the system's linearized equations of motion.

Here, A is the system matrix representing the dynamics of the pendulums and the cart, B is the input matrix relating the input forces to the state changes, C is the output matrix that maps the state to the output Y, and D is the feedforward matrix. These matrices will be determined based on the linearized equations of motion of the system.

Linearization simplifies the nonlinear dynamics of a system around a specific operating point, often an equilibrium. For the quintuple inverted pendulum system, we linearize around the equilibrium point where all pendulums are upright and stationary, and the cart is at rest. The linearized state-space model is given by:

$$\begin{aligned} A &= \text{Jacobian of system dynamics evaluated at equilibrium,} \\ B &= \text{Jacobian of input dynamics evaluated at equilibrium,} \\ C &= \text{Matrix relating state variables to measured outputs,} \\ D &= \text{Matrix relating inputs directly to outputs.} \end{aligned}$$

The equilibrium point of the quintuple inverted pendulum system is defined as the state where all pendulums are perfectly vertical and stationary, and the cart is not moving. This translates to all angular positions θ_i being set to π and all angular velocities $\dot{\theta}_i$ as well as the cart's velocity $\dot{x}$ being zero. This point represents a state of balanced forces and moments, crucial for analyzing the system's stability and designing appropriate control strategies.

To achieve upright stabilization of the quintuple inverted pendulum system, we implement a LQR controller. The LQR controller is designed to minimize the following quadratic cost function:

$$J = \int_0^\infty (X^T Q X + u^T R u)\, dt \tag{13}$$

Here, J is the cost function, Q and R are weight matrices for the states and control inputs, respectively. This function balances the system's performance in terms of state deviation and control effort. The SymPy library was employed for deriving the system's equations of motion and verifying the controllability condition.

Given the linearized state-space model with matrices A and B, the optimal control law is obtained as:

$$u = -KX \tag{14}$$

where K is the gain matrix calculated to achieve the desired system performance. The controller's implementation includes solving for K using the Algebraic Riccati Equation and applying the control input u to maintain each pendulum in its upright position.

3 Experiment, Results and Discussions

(a) Whole stand **(b)** Pendulums **(c)** Caret

Fig. 5: USD model overview

The experiment was set up in the Isaac Sim environment, utilizing its advanced simulation capabilities to model a system. The key focus was on integrating an LQR controller within this simulation to explore its potential for stabilizing. Given the exploratory nature of this research, the setup was primarily aimed at understanding the dynamics of the system and the feasibility of the control approach.

The planned procedure involves conducting a series of simulations under various initial conditions to assess the pendulum system's response. These trials are designed to incrementally challenge the LQR controller, starting from minor perturbations and gradually increasing to more significant deviations. This step-by-step approach is intended to provide insights into the controller's effectiveness across a spectrum of scenarios.

Although default URDF[4] importer works to get USD model, some physical adjustments had to be done in order to start experimenting with physical engine (although final values are to be discovered), which are composed into Table 2:

To ensure that simulations closely approximate real-world conditions, the SimReady specification [14] is leveraged. By using SimReady assets in simulations, a virtual environment closely mirroring the physical world is created. This level of realism is essential for validating the performance of control algorithms and assessing their suitability for real-world applications. The model and project structure can be seen in Figure 6, 5.

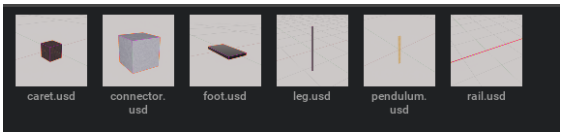
This research opens several avenues for future work in controlling and stabilizing the complex control systems:

[4] http://wiki.ros.org/urdf/XML/model

Table 2: Physics adjustments.

Prim	Property	Value
pendulum_*_link	linear damping	0.5
pendulum_*_link	angular damping	1.5
caret_joint	damping	10000.0
caret_joint	stiffness	0.0

(a) Model



(b) Props

Fig. 6: USD model structure
kuro is robot's codename

- **Adoption of Advanced Control Techniques**: The exploration of advanced control methods, notably RL and DRL, holds great promise. With the advent of NVIDIA's GR00T and Isaac Lab [1,2], there are now more opportunities to integrate these sophisticated approaches, potentially leading to more adaptive and robust control strategies for complex systems.
- **Real-world Validation and Experiments**: To bridge the gap between virtual simulations and real-world applications, aim is to conduct physical experiments. These experiments will not only validate virtual results but also test the effectiveness and reliability of the control algorithms under real-world conditions.

4 Conclusion

This study explored the application of Isaac Sim and ROS2 to simulate and manage a quintuple inverted pendulum system. Through this investigation, we demonstrated the practical utility of integrating an LQR controller within this simulation environment, aimed primarily at understanding the system's dynamics and testing the feasibility of control strategies under simulated conditions.

While the findings suggest that Isaac Sim is a powerful tool for detailed and realistic simulations, they also underscore the challenges that remain in translating these simulations to real-world applications. In conclusion, this research serves as a preliminary exploration into the possibilities and challenges of simulating complex control systems. It highlights the importance of continued development and testing within simulated environments to advance our understanding and capabilities in controlling highly dynamic systems.

Future work will seek to refine control strategies and explore the potential of advanced control techniques such as RL and DRL, potentially utilizing platforms like NVIDIA's GR00T and Isaac Lab.

References

1. NVIDIA GR00T project, https://developer.nvidia.com/project-gr00t
2. NVIDIA Isaac Lab, https://developer.nvidia.com/isaac-sim#isaac-lab
3. 3D visualization tool for ROS (2023), https://github.com/ros2/rviz
4. Isaac sim extension templates (2023), https://docs.omniverse.nvidia.com/isaacsim/latest/advanced_tutorials/tutorial_extension_templates.html
5. The ros2_control is a framework for (real-time) control of robots using (ROS 2) (2023), https://control.ros.org/master/index.html
6. Bondada, A., Nair, V.G.: Dynamics of multiple pendulum system under a translating and tilting pivot. Archive of Applied Mechanics **93**, 3699–3740 (2023). https://doi.org/10.1007/s00419-023-02473-6
7. Boubaker, O.: The inverted pendulum benchmark in nonlinear control theory: A survey. International Journal of Advanced Robotic Systems **10**(5), 233 (Jan 2013). https://doi.org/10.5772/55058
8. Chen, C.Y.J.: Multiple Degree of Freedom Inverted Pendulum Dynamics: Modeling, Computation, and Experimentation. PhD dissertation, University of Southern California (May 2009), copyright 2009 Cheng-Yuan Jerry Chen
9. Kane, T.R., Levinson, D.A.: Dynamics: Theory and Application. McGraw Hill (1985), http://hdl.handle.net/1813/638
10. Kao, Sho-Tsung, e.a.: Balance control of a configurable inverted pendulum on an omni-directional wheeled mobile robot (Oct 2022). https://doi.org/10.3390/app122010307
11. Lal Bahadur Prasad, e.a.: Optimal control of nonlinear inverted pendulum system using PID controller and LQR: Performance analysis without and with disturbance input. International Journal of Automation and Computing **11**(6), 661–670 (Dec 2014). https://doi.org/10.1007/s11633-014-0818-1
12. Makoviychuk, Viktor, e.a.: Isaac gym: High performance gpu-based physics simulation for robot learning (2021). https://doi.org/10.48550/ARXIV.2108.10470
13. Messikh, Lotfi, e.a.: Stabilization of the cart-inverted-pendulum system using state-feedback pole-independent mpc controllers. Sensors **22**(1) (2022). https://doi.org/10.3390/s22010243
14. NVIDIA: The SimReady specification (2023), https://docs.omniverse.nvidia.com/simready/latest/overview/simready-spec.html
15. Setka, V., Cecil, R., Schlegel, M.: Triple inverted pendulum system implementation using a new arm/fpga control platform. pp. 321–326 (05 2017). https://doi.org/10.1109/CarpathianCC.2017.7970419
16. Steven Macenski, e.a.: Robot operating system 2: Design, architecture, and uses in the wild. Science Robotics **7**(66) (may 2022). https://doi.org/10.1126/scirobotics.abm6074
17. Viet, Tuan, e.a.: Control of a 2-dof omnidirectional mobile inverted pendulum. Journal of Mechanical Science and Technology **26** (09 2012). https://doi.org/10.1007/s12206-012-0710-2
18. Yongli Zhang, e.a.: Stabilization of the quadruple inverted pendulum by variable universe adaptive fuzzy controller based on variable gain h ∞ regulator. Journal of Systems Science and Complexity **25**(5), 856–872 (Jun 2012). https://doi.org/10.1007/s11424-012-0011-y

A Mathematical Modeling

$$
\begin{aligned}
L =& \frac{1}{2}M\dot{x}^2 + \frac{9}{2}glm\cos(\theta_1) + \frac{7}{2}glm\cos(\theta_2) + \frac{5}{2}glm\cos(\theta_3) + \frac{3}{2}glm\cos(\theta_4) + \frac{1}{2}glm\cos(\theta_5) \\
&+ \frac{7}{2}l^2m\cos(\theta_1-\theta_2)\dot{\theta}_1\dot{\theta}_2 + \frac{5}{2}l^2m\cos(\theta_1-\theta_3)\dot{\theta}_1\dot{\theta}_3 + \frac{3}{2}l^2m\cos(\theta_1-\theta_4)\dot{\theta}_1\dot{\theta}_4 \\
&+ \frac{1}{2}l^2m\cos(\theta_1-\theta_5)\dot{\theta}_1\dot{\theta}_5 + \frac{5}{2}l^2m\cos(\theta_2-\theta_3)\dot{\theta}_2\dot{\theta}_3 + \frac{3}{2}l^2m\cos(\theta_2-\theta_4)\dot{\theta}_2\dot{\theta}_4 \\
&+ \frac{1}{2}l^2m\cos(\theta_2-\theta_5)\dot{\theta}_2\dot{\theta}_5 + \frac{3}{2}l^2m\cos(\theta_3-\theta_4)\dot{\theta}_3\dot{\theta}_4 + \frac{1}{2}l^2m\cos(\theta_3-\theta_5)\dot{\theta}_3\dot{\theta}_5 \\
&+ \frac{1}{2}l^2m\cos(\theta_4-\theta_5)\dot{\theta}_4\dot{\theta}_5 + \frac{13}{6}l^2m\dot{\theta}_1^2 + \frac{5}{3}l^2m\dot{\theta}_2^2 + \frac{7}{6}l^2m\dot{\theta}_3^2 + \frac{2}{3}l^2m\dot{\theta}_4^2 + \frac{1}{6}l^2m\dot{\theta}_5^2 \\
&- \frac{9}{2}lm\cos(\theta_1)\dot{\theta}_1\dot{x} - \frac{7}{2}lm\cos(\theta_2)\dot{\theta}_2\dot{x} - \frac{5}{2}lm\cos(\theta_3)\dot{\theta}_3\dot{x} - \frac{3}{2}lm\cos(\theta_4)\dot{\theta}_4\dot{x} \\
&- \frac{1}{2}lm\cos(\theta_5)\dot{\theta}_5\dot{x} + \frac{5}{2}m\dot{x}^2
\end{aligned}
\tag{15}
$$

$$
\begin{cases}
M\ddot{x}(t) + kx(t) + \frac{9lm\sin(\theta_1(t))\dot{\theta}_1(t)^2}{2} + \frac{7lm\sin(\theta_2(t))\dot{\theta}_2(t)^2}{2} + \frac{5lm\sin(\theta_3(t))\dot{\theta}_3(t)^2}{2} \\
+ \frac{3lm\sin(\theta_4(t))\dot{\theta}_4(t)^2}{2} + \frac{lm\sin(\theta_5(t))\dot{\theta}_5(t)^2}{2} \\
- \frac{9lm\cos(\theta_1(t))\ddot{\theta}_1(t)}{2} - \frac{7lm\cos(\theta_2(t))\ddot{\theta}_2(t)}{2} - \frac{5lm\cos(\theta_3(t))\ddot{\theta}_3(t)}{2} \\
- \frac{3lm\cos(\theta_4(t))\ddot{\theta}_4(t)}{2} - \frac{lm\cos(\theta_5(t))\ddot{\theta}_5(t)}{2} \\
+5m\ddot{x}(t) - F_{\text{ext}}(t) = 0, \\
c\dot{\theta}_1(t) + \frac{9glm\sin(\theta_1(t))}{2} + \frac{7l^2m\sin(\theta_1(t)-\theta_2(t))\dot{\theta}_2(t)^2}{2} + \frac{5l^2m\sin(\theta_1(t)-\theta_3(t))\dot{\theta}_3(t)^2}{2} \\
+ \frac{3l^2m\sin(\theta_1(t)-\theta_4(t))\dot{\theta}_4(t)^2}{2} + \frac{l^2m\sin(\theta_1(t)-\theta_5(t))\dot{\theta}_5(t)^2}{2} + \frac{7l^2m\cos(\theta_1(t)-\theta_2(t))\ddot{\theta}_2(t)}{2} \\
+ \frac{5l^2m\cos(\theta_1(t)-\theta_3(t))\ddot{\theta}_3(t)}{2} + \frac{3l^2m\cos(\theta_1(t)-\theta_4(t))\ddot{\theta}_4(t)}{2} + \frac{l^2m\cos(\theta_1(t)-\theta_5(t))\ddot{\theta}_5(t)}{2} \\
+ \frac{13l^2m\ddot{\theta}_1(t)}{3} - \frac{9lm\cos(\theta_1(t))\ddot{x}(t)}{2} = 0, \\
c\dot{\theta}_2(t) + \frac{7glm\sin(\theta_2(t))}{2} - \frac{7l^2m\sin(\theta_1(t)-\theta_2(t))\dot{\theta}_1(t)^2}{2} + \frac{5l^2m\sin(\theta_2(t)-\theta_3(t))\dot{\theta}_3(t)^2}{2} \\
+ \frac{3l^2m\sin(\theta_2(t)-\theta_4(t))\dot{\theta}_4(t)^2}{2} + \frac{l^2m\sin(\theta_2(t)-\theta_5(t))\dot{\theta}_5(t)^2}{2} + \frac{7l^2m\cos(\theta_1(t)-\theta_2(t))\ddot{\theta}_1(t)}{2} \\
+ \frac{5l^2m\cos(\theta_2(t)-\theta_3(t))\ddot{\theta}_3(t)}{2} + \frac{3l^2m\cos(\theta_2(t)-\theta_4(t))\ddot{\theta}_4(t)}{2} + \frac{l^2m\cos(\theta_2(t)-\theta_5(t))\ddot{\theta}_5(t)}{2} \\
+ \frac{10l^2m\ddot{\theta}_2(t)}{3} - \frac{7lm\cos(\theta_2(t))\ddot{x}(t)}{2} = 0, \\
c\dot{\theta}_4(t) + \frac{3glm\sin(\theta_4(t))}{2} - \frac{3l^2m\sin(\theta_1(t)-\theta_4(t))\dot{\theta}_1(t)^2}{2} - \frac{3l^2m\sin(\theta_2(t)-\theta_4(t))\dot{\theta}_2(t)^2}{2} \\
- \frac{3l^2m\sin(\theta_3(t)-\theta_4(t))\dot{\theta}_3(t)^2}{2} + \frac{l^2m\sin(\theta_4(t)-\theta_5(t))\dot{\theta}_5(t)^2}{2} + \frac{3l^2m\cos(\theta_1(t)-\theta_4(t))\ddot{\theta}_1(t)}{2} \\
+ \frac{3l^2m\cos(\theta_2(t)-\theta_4(t))\ddot{\theta}_2(t)}{2} + \frac{3l^2m\cos(\theta_3(t)-\theta_4(t))\ddot{\theta}_3(t)}{2} + \frac{l^2m\cos(\theta_4(t)-\theta_5(t))\ddot{\theta}_5(t)}{2} \\
+ \frac{4l^2m\ddot{\theta}_4(t)}{3} - \frac{3lm\cos(\theta_4(t))\ddot{x}(t)}{2} = 0, \\
c\dot{\theta}_5(t) + \frac{glm\sin(\theta_5(t))}{2} - \frac{l^2m\sin(\theta_1(t)-\theta_5(t))\dot{\theta}_1(t)^2}{2} - \frac{l^2m\sin(\theta_2(t)-\theta_5(t))\dot{\theta}_2(t)^2}{2} \\
- \frac{l^2m\sin(\theta_3(t)-\theta_5(t))\dot{\theta}_3(t)^2}{2} - \frac{l^2m\sin(\theta_4(t)-\theta_5(t))\dot{\theta}_4(t)^2}{2} + \frac{l^2m\cos(\theta_1(t)-\theta_5(t))\ddot{\theta}_1(t)}{2} \\
+ \frac{l^2m\cos(\theta_2(t)-\theta_5(t))\ddot{\theta}_2(t)}{2} + \frac{l^2m\cos(\theta_3(t)-\theta_5(t))\ddot{\theta}_3(t)}{2} + \frac{l^2m\cos(\theta_4(t)-\theta_5(t))\ddot{\theta}_4(t)}{2} \\
+ \frac{l^2m\ddot{\theta}_5(t)}{3} - \frac{lm\cos(\theta_5(t))\ddot{x}(t)}{2} = 0
\end{cases}
\tag{16}
$$