UNIVERSITY OF CALIFORNIA

Los Angeles

Locomotion Control of Legged Robots
using Data-Driven Techniques:
Application to a Buoyancy Assisted Biped

A dissertation submitted in partial satisfaction
of the requirements for the degree
Doctor of Philosophy in Mechanical Engineering

by

Hosik Chae

2023

ABSTRACT OF THE DISSERTATION

Locomotion Control of Legged Robots using Data-Driven Techniques: Application to a Buoyancy Assisted Biped

by

Hosik Chae
Doctor of Philosophy in Mechanical Engineering
University of California, Los Angeles, 2023
Professor Dennis W. Hong, Chair

The field of legged robotics has made significant advancements and shown potential practicality in various applications. Although these robots are becoming more popular, they are still not widely used due to the inherent danger when malfunctioning as well as their high cost.

BALLU, Buoyancy Assisted Lightweight Legged Unit, is a robot that never falls down due to the buoyancy provided by a set of helium balloons attached to its lightweight body. This platform solves many issues that hinder current robots from operating close to humans while also providing affordability. However, the advantages gained also lead to the platform's distinct difficulties caused by severe underactuation and nonlinearities due to external forces such as buoyancy and drag.

This dissertation presents a motion planning approach using data-driven techniques motivated by these challenges and its application to BALLU. The paper describes the concept of the platform, the hardware design of different generations of BALLUs, the software architecture, the nonconventional characteristics of BALLU as a legged robot, and an analysis of its unique behavior. Based on the analysis, a data-driven approach is proposed to achieve non-teleoperated walking: a statistical process is proposed to form low-dimensional state vectors from the simulation data, and a deep neural network-based controller is trained. The controller is tested on both simulation and real-world hardware. Its performance is assessed by observing the robot's limit cycles and trajectories in Cartesian coordinates. The controller generates periodic walking sequences in simulation as well as on the real-world robot, even without additional transfer learning. It is also shown that the controller can deal with unseen conditions during the training phase. The resulting behavior not only shows the robustness of the controller but also implies that the proposed statistical process effectively extracts a state vector that is low-dimensional yet contains the essential information of the high-dimensional dynamics of BALLU's walking.

The dissertation of Hosik Chae is approved.

Tsu-Chin Tsao

Veronica Santos

Lin Yang

Dennis W. Hong, Committee Chair

University of California, Los Angeles

2023

To my dear family and loved ones.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGMENTS

There are so many people to whom I would like to express my gratitude, as I could not make this achievement alone:

I would like to thank Dr.Dennis Hong. You have always encouraged me to take on new challenges and created a research lab where we can realize whatever robots we dreamed of. You have always listened to my concerns with a positive attitude, celebrated my achievements as if they were your own, and never turned away when I faced difficulties, but instead helped me to find solutions. Thanks to your guidance and attention, I was able to successfully complete my Ph.D. Not only as a researcher but also as a person, I have learned from you what kind of person I want to be and work with. Additionally, I am grateful for the unforgettable experiences and broadened perspectives gained while working on demos and traveling together.

To Professors Dr.Tsu-Chin Taso, Dr.Veronica Santos, and Dr.Lin Yang, who served on my committee, I extend my gratitude as well. From different courses of yours, you all provided detailed feedback and guidance on projects, and I was able to learn not only academic knowledge but also learned the meticulousness, thoroughness, and proper attitudes required to be a good researcher. I am grateful for the sincere guidance provided when I visited later, despite not being able to converse frequently during the COVID-19 pandemic. I am grateful for your sincere guidance when I visited you afterward, even though we couldn't talk often during the COVID-19 pandemic.

I would like to thank RoMeLa members. The achievements during my Ph.D. journey were not something I could have accomplished alone. I grew a lot by learning from the strengths of my talented colleagues. I remember the contributions left by each of RoMeLa's alumni and the footsteps we've created together with the current members. I cherish all the memories and experiences we've shared. I remember the countless demos and trips we prepared for together, and the all-nighters we pulled for robot projects and competitions. As an international student, the lab was my home, and my lab members were my family.

Without them, I would not have discovered the joy of working together, and my life in the United States would have been much more difficult. I confess that I could make it this far because I had members who generously provided emotional and mental support like brothers during hardships, making me feel that I was not alone.

I want to express highest my gratitude to my loving family and friends: my father, who always put the family first and took responsibility in any situation; my mother, who always prayed for me and the family; and my sister, who is my best friend. Also, to my dear Wonbin, who has been waiting and supporting me from both near and far throughout the lengthy doctoral period, loving even the smallest parts of me. I believe that I could persevere because of your unwavering love, support, and encouragement. My beloved Cookie and Maru, who have taught me what unconditional love is and whom I dearly miss, have always been my strength. To my friends who, despite not being able to meet often, have always been by my side, cheering me on – I express my endless gratitude and thanks to each and every single one of you.

Lastly, above all else, I give thanks to God, who has loved me unconditionally and guided me this point in His grand plan.

VITA

2008–2015	B.S. in Mechanical Engineering, Yonsei University, Seoul, Republic of Korea
2015–2018	M.S. in Mechanical and Aerospace Engineering, University of California, Los Angeles (UCLA), United States
2018–2023	Ph.D. Candidate in Mechanical and Aerospace Engineering, University of California, Los Angeles (UCLA), United States
2015–2023	Graduate Student Researcher, University of California, Los Angeles (UCLA), United States

PUBLICATIONS

Hosik Chae, Min Sung Ahn, Donghun Noh, Hyunwoo Nam, and Dennis Hong. ”BALLU2: A Safe and Affordable Buoyancy Assisted Biped,” *Frontiers in Robotics and AI*, page 290, 2021.

Nitish Sontakke, **Hosik Chae**, Sangjoon Lee, Tianle Huang, Dennis W. Hong, Sehoon Ha, “Residual physics learning and system identification for sim-to-real transfer of policies on buoyancy assisted legged robots,” 2023. *ArXiv.* /abs/2303.09597

Min Sung Ahn, **Hosik Chae**, Colin Togashi, Dennis Hong, Joohyung Kim, and Sungjoon Choi. ”Learning-based Motion Stabilizer Leveraging Offline Temporal Optimization,” In *2022 19th International Conference on Ubiquitous Robots (UR)*, pages 129–136. IEEE, 2022.

June-sup Yi, Tuan Anh Luong, **Hosik Chae**, Min Sung Ahn, Donghun Noh, Huy Nguyen

Tran, Myeongyun Doh, Eugene Auh, Nabih Pico, Francisco Yumbla, Dennis Hong, and Hyungpil Moon. ”An Online Task-Planning Framework using Mixed Integer Programming for Multiple Cooking Tasks using a Dual-Arm Robot,” *Applied Sciences*, 12(8):4018, 2022.

Donghun Noh, Hyunwoo Nam, Min Sung Ahn, **Hosik Chae**, Sangjoon Lee, Kyle Gillespie, and Dennis Hong. ”Surface Material Dataset for Robotics Applications (SMDRA): A Fataset with Friction Coefficient and RGB-D for Surface Segmentation,” In *2020 25th International Conference on Pattern Recognition (ICPR)*, pages 6275–6281. IEEE, 2021.

June-sup Yi, Min Sung Ahn, **Hosik Chae**, Hyunwoo Nam, Donghun Noh, Dennis Hong, and Hyungpil Moon. ”Task Planning with Mixed-Integer Programming for Multiple Cooking Task using Dual-Arm Robot,” In *2020 17th International Conference on Ubiquitous Robots (UR)*, pages 29–35. IEEE, 2020.

Joshua Hooks, Min Sung Ahn, Jeffrey Yu, Xiaoguang Zhang, Taoyuanmin Zhu, **Hosik Chae**, and Dennis Hong. ”ALPHRED: A Multi-Modal Operations Quadruped Robot for Package Delivery Applications,” *IEEE Robotics and Automation Letters*, 5(4):5409–5416, 2020.

Min Sung Ahn, **Hosik Chae**, Donghun Noh, Hyunwoo Nam, and Dennis Hong. ”Analysis and Noise Modeling of the Intel Realsense D435 for Mobile Robots,” In *2019 16th International Conference on Ubiquitous Robots (UR)*, pages 707–711. IEEE, 2019.

Min Sung Ahn, Daniel Sun, **Hosik Chae**, Itsui Yamayoshi, and Dennis Hong. ”Lessons Learned from the Development and Deployment of a Hotel Concierge Robot to be Operated in a Real World Environment,” In *2019 16th International Conference on Ubiquitous Robots (UR)*, pages 55–60. IEEE, 2019.

Min Sung Ahn, **Hosik Chae**, and Dennis W Hong. ”Stable, Autonomous, Unknown Terrain Locomotion for Quadrupeds based on Visual Feedback and Mixed-Integer Convex Optimization,” In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 3791–3798. IEEE, 2018.

CHAPTER 1

Introduction

1.1 Motivation

One class of robots most commonly used in our daily lives is service robots. In particular, a central application is social robots that interact with humans and provide information. For instance, LG Electronics' CLOi at an airport [31] leads passengers to find a route inside the airport and informs them of their flight schedule. SoftBank's Pepper at a library [47] helps visitors find books. As a similar example, LinkedIn makes use of Double Robotics' telepresence robot to telecommute [20]. These robots are more economical than human employees in providing an intuitive and easy interface to information. As a result, more service providers are seeking to introduce robots to their business [8, 24]

Meanwhile, all these service robots are wheeled robots, and there are only a few legged ones such as LARA [5] and Connie [30]. However, though these legged service robots can provide information standing at a reception desk, they are small so limited in taking advantage of mobility as legged robots. On the other hand, most of the legged robots near human size pursue to be strong and powerful, and Boston Dynamics' Spot [12] and Atlas [11], Agility Robotics' Digit [2], and ANYbotics' ANYmal [6] would fall into this case. Taking advantage of the characteristics, they are preferred by some specialized industries such as last-mile delivery or construction site inspection.

While these platforms have shown remarkable progress in technology, essential yet often overlooked aspects that are contributing to their full deployment in close proximity to humans are safety and cost. When they malfunction, the heavyweight and powerful actuation methods could act as a potential cause of serious damage to its surrounding environments and even threaten human lives. In the context of service robots, such robots' capacity might be redundant for common needs in everyday life but also dangerous.

Additionally, these machines are still expensive for households to adopt. Since many conventional robots use powerful and strong components, they are much higher priced than your average home appliances, and, among the well-known platforms, one of the most affordable and small-scaled quadrupeds is 2700 USD at the time of this dissertation [57].

Made with helium balloons and lightweight body parts, BALLU (Buoyancy Assisted Lightweight Legged Unit) has the possibility to overcome the aforementioned issues concerning existing robots. The first is safety and inherent stability. Because of the buoyancy provided by the balloons, BALLU is a robot that literally cannot fall down. More importantly, its light parts and soft balloon body can only generate so much momentum and force, allowing it to be operated without worry when there is physical interaction or even collision with young children. This allows BALLU to potentially act in the future as a safe, interactive service robot in the vicinity of people. The second is its cost. BALLU is merely as affordable as many low-cost home appliances. In the long run, this even opens up opportunities for such platforms to act as disposable robots, where a number of them can be easily built and explore unknown environments with less economic burden.

1.2 Buoyancy Assisted Robots

Since the concept of BALLU was first unveiled by [26], there have been further studies of robot platforms adopting helium balloons and leveraging their buoyancy force. One noticeable work is the Giacometti Arm designed by [53], a 20-meter helium balloon supported robot arm designed for inspection tasks. This manipulator has 20 joints driven by pneumatic and thin artificial muscles. Among mobile platforms, GerWalk by [59] is one that is very resemblant to BALLU. Because its body is a helium balloon, it is able to easily traverse stairs and other obstacles with stability. [38] of JPL also proposed a balloon based walking robot for Mars exploration. Beyond these platforms, there are many other applications with balloons because of their affordability and intrinsic stability: including manipulation [9], disaster investigation [54, 60], security [39], social interactions [39], and root cleaning [21].

Unlike the previously mentioned platforms that rely on passively acting forces (e.g., buoyancy from helium balloons), there are also works that have directly integrated active thrusters. The concept of a bipedal robot supported by a propulsion system is first proposed by [61]. Aerial-biped by [37] is a bipedal robot attached to a quadrotor, and the robot walks using a gait sequence generated by a policy learned by reinforcement learning. Though it is not published, LEONARDO from [1] is another bipedal robot with drone-like propellers. On a more extreme note, the feasibility of a combination of propellers, buoyancy force, and active rappelling to lift rigid bodies has also been studied by [35]. Unlike the previously mentioned platforms that rely on passively acting forces (e.g. buoyancy generated from helium balloons), there are also works that have directly integrated active thrusters. The concept of a bipedal robot supported by a propulsion system is first proposed by [61]. Aerial-biped by [37] is a bipedal robot attached to a quadrotor, and the robot walks using a gait sequence generated by a policy learned by reinforcement learning. Though it is not published, LEONARDO from [1] is another bipedal robot with drone-like propellers. On a more extreme note, the feasibility of a combination of propellers, buoyancy force, and active rappelling to lift rigid bodies has also been studied by [35].

1.3 Background

1.3.1 Spearman Correlation

Spearman correlation coefficient, also known as Spearman's rank correlation, is a nonparametric measure of the strength of a monotonic relationship between two variables. The coefficient can range from -1 to $+1$, each of which indicates a perfect negative and positive monotonic relationship, and zero indicates no relationship.

The calculation of the Spearman coefficient is based on the computation of Pearson correlation. The Pearson correlation coefficient [41] indicates the strength of the linear relationship between two random variables:

$$\rho_p(X, Y) = \frac{\mathrm{cov}(X, Y)}{\sigma_X \sigma_Y}$$

where, ρ_p represents Pearson correlation coefficient, $\mathrm{cov}(\cdot, \cdot)$ the covariance, and $\sigma_{(\cdot)}$ the standard deviations of the variables. The Spearman correlation coefficient is calculated by converting the values of the two variables into ranks, and then computing the Pearson correlation coefficient on the ranked data.

The Spearman correlation coefficient [51] is calculated as follows:

$$\rho_s(X, Y) = \rho_p(\mathrm{R}(X), \mathrm{R}(Y)) = \frac{\mathrm{cov}\left(\mathrm{R}(X), \mathrm{R}(Y)\right)}{\sigma_{\mathrm{R}(X)} \sigma_{\mathrm{R}(Y)}}$$

where, ρ_s represents the Spearman correlation coefficient, $\mathrm{R}(\cdot)$ is the rank of the given random variable.

While the Pearson correlation coefficient is being used, the Spearman correlation coefficient can capture a more nonlinear relationship in the sense of monotonic function. This makes it particularly useful for analyzing data that may not have a linear relationship.

Despite its limited representation in monotonic functions, Spearman correlation is useful because of its lightweight computation by a simple extension of Pearson coefficients.

1.3.2 Feature Selection

Feature selection is a critical topic both in Statistics and Machine Learning. It refers to the process of selecting a subset of relevant features from a larger set of input features. Finding an optimal set of features from a large set of candidates is a problem that occurs in many contexts, including control theory [58], bioinformatics, and finance.

The concept of feature selection can be traced back to the 1960s and 1970s when researchers started to explore the application of statistical techniques for variable selection in regression analysis. With the rise of machine learning in the late 20th and early 21st century, feature selection gained increasing attention, as it proved crucial in handling high-dimensional data and improving the generalization capabilities of learning algorithms.

By selecting the most representative features, data-driven models can be improved. First, the performance of the models can be improved. By eliminating irrelevant or redundant features, feature selection can enhance the performance of models, reduce the risk of overfitting, and improve generalization to unseen data. In addition, by reducing the size of models, the models would require fewer data and can become computationally efficient. Furthermore, models with fewer features are easier to understand, which is particularly important when providing insights or explanations during decision-making processes in real-world applications.

Feature selection methods are designed to reduce the number of input variables to those that are considered most useful for predicting the target variable, primarily focusing on the removal of non-informative or redundant predictors. Feature selection methods can be broadly categorized into three groups: filter, wrapper, and embedded methods.

The first category is filter methods: these techniques evaluate the relevance of features based on their individual characteristics or relationships with the target variable. Popular filter methods include correlation coefficients [41, 51], ANOVA [23], mutual information, and chi-square tests [42]. Filter methods are generally computationally efficient but may not account for the interactions between features. The second category is wrapper methods [14]:

the methods in this category evaluate subsets of features by training a specific learning algorithm and assessing its performance on a validation set. Examples of wrapper methods include forward selection, backward elimination, and recursive feature elimination. Although these methods can capture feature interactions, they can be computationally expensive due to the need to train the model multiple times.

Lastly, embedded methods: these methods incorporate feature selection within the learning algorithm itself. Examples include LASSO (Least Absolute Shrinkage and Selection Operator) [49, 55] for linear regression and feature importance measures in decision trees or ensemble models like Random Forests. Embedded methods can balance the trade-off between computational efficiency and accounting for feature interactions.

Feature selection is a vital aspect of both Statistics and Machine Learning that facilitates the selection of relevant features from high-dimensional data sets. It plays a crucial role in improving model performance, reducing computational complexity, and enhancing interpretability. By employing filter, wrapper, or embedded methods, researchers can tailor their feature selection approach to the specific requirements of their problem domain.

1.4 Organization

The remainder of the paper is organized as follows:

Chapter 2 introduces the BALLU, Buoyancy Assisted Lightweight Legged Unit, with a description of the overall differences between BALLU platforms over generations. The hardware design of BALLU is then detailed in Section 2.2, followed by an explanation of the software framework in Section 2.3.

Chapter 3 analyzes the challenges and the behavior of BALLU. The design decisions made in Chapter 2 solved many problems of existing robots but created unique issues at the same time. These difficulties made BALLU a highly unconventional robot. This chapter describes an analysis of the robot's behavior based on the author's experience of teleoperation BALLU as well as mathematical derivation.

Chapter 4 represents a walking controller motivated by the challenges discussed in Chapter 3. A set of data-driven approaches are taken to define a state vector and train a deep learning-based controller based on the state definition. The performance and robustness of the controller are evaluated on both simulated and real-world hardware.

Chapter 5 provides a summary of the dissertation and discusses possible future works. Appendices A and B offer mathematical derivations to support claims made in the text and detailed explanations for those interested in reproducing the BALLU platform.

CHAPTER 2

System Design

2.1 Generations of BALLU

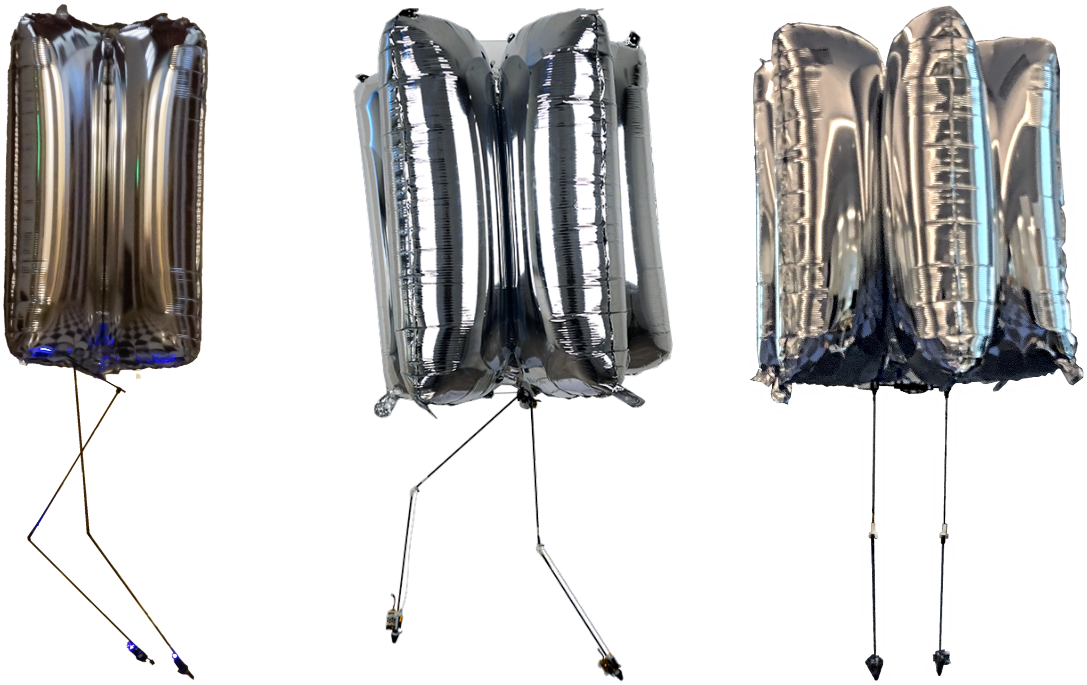


Figure 2.1: BALLU1, BALLU2, and BALLU3

The initial prototype, BALLU1 [26], was designed by Ghassemi and Hong. It had a set of O-rings keeping the knee at the desired equilibrium, allowing the tibia to move about the equilibrium. It also demonstrated teleoperated locomotion via radio control signals.

BALLU2 was designed with several changes to enhance functionality, most notably the addition of computation capabilities. BALLU1 was a prototype for the feasibility test and implemented the idea of BALLU. A minimal actuation system was constructed by exploiting off-the-shelf remote-controlled appliances to allow teleoperation only. In BALLU2, a single board computer was added to calculate control commands, and all mechanical and electronic system was redesigned to implement the same mobility. Also, a set of torsional springs is added to each knee joint to assist faster response rate of the knee joint and support the weight of BALLU2, which got heavier than that of BALLU1. To accommodate the electronics, custom parts were designed and 3d-printed with ABS at the lowest fill-in rate.

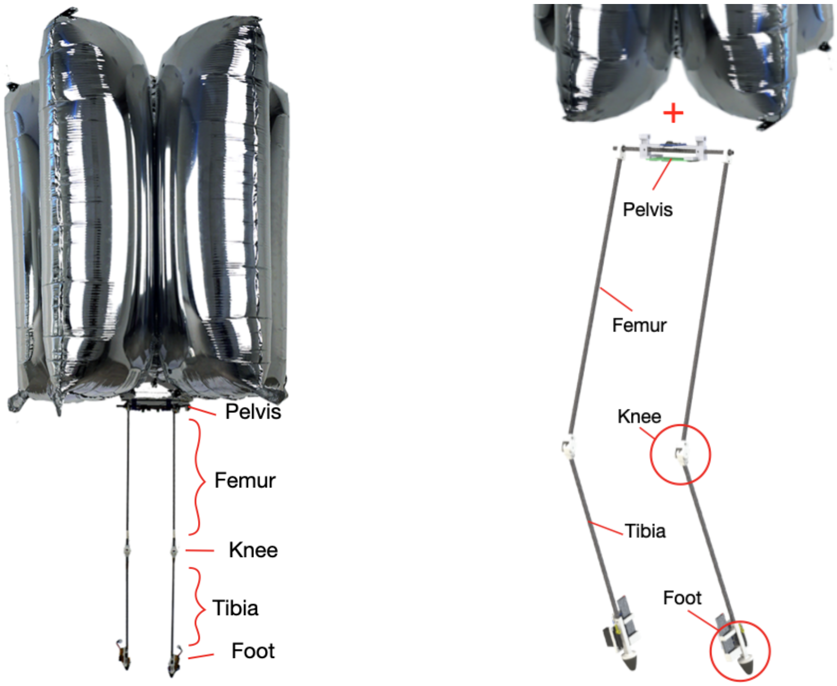
BALLU3 embraced various sensors and an onboard state estimator, and the corresponding mechanical design is updated. One inertial measurement unit (IMU) and time-of-flight (ToF) sensors were added to the pelvis link of BALLU, and a magnetic encoder is added to each joint. The housing was printed with nylon carbon fiber, reducing weight while increasing strength. As shown in Table 2.1, despite the added sensors, the weight of each part was kept at a similar level or even reduced for some parts, thanks to optimized design and the use of lighter printing materials.

BALLU3 further improved the mechanical design of BALLU, incorporating various sensors and an onboard state estimator. One inertial measurement unit (IMU) and time-of-flight (ToF) sensors were added to the pelvis link, and a magnetic encoder was added to each joint. The housing was printed with nylon carbon fiber, reducing weight while maintaining strength. Despite the added sensors, the weight of each part was kept at a similar level or even reduced for some parts, thanks to optimized design and the use of lighter printing materials.

In this paper, the term "BALLU" refers to both BALLU2 and BALLU3 as a whole, and it will be used to describe shared properties between the two. The following sections will describe the design based on BALLU2, with additional explanations for the updates in BALLU3.

2.2 Hardware Design

At its core, BALLU is a bipedal robot attached to a set of helium balloons, which provide sufficient buoyancy force to prevent the robot from falling down. Figure 2.2 illustrates the overall design of BALLU. The body consists of a pelvis link, which has two identical legs attached to its ends, with each leg having a joint at the hip and the knee. To reduce the weight that the buoyancy has to support, the majority of the components are made or chosen to be light, with unavoidably heavy components being placed at the foot. The detailed design parameters are listed in Table 2.1.



(a) (b)

Figure 2.2: Overall Design of BALLU: (A) BALLU platform and (B) the name of each part of the lower body.

Table 2.1: Design Parameters

	Symbol	BALLU2	BALLU3	Description
Mass [g]	m_p	31.2	21.8	Pelvis link with electronics
	m_{fm}	9.0	6.0	Femur link
	m_{tb}	9.2	6.0	Tibia link
	m_{hp}	2.2	4.9	Hip joint
	m_k	6.1	5.8	Knee joint
	m_{ft}	23.5	30.4	Foot part with electronics
	$m_{balloon}$	19.8	19.8	Single balloon
Length [mm]	l_{pelvis}	163.0	163.0	Pelvis
	l_{femur}	370.0	370.0	Femur
	l_{tibia}	385.0	385.0	Tibia
	l_{fm}	185.0	185.0	From Knee to Femur's center
	l_{tb}	192.5	192.5	From Knee to Tibia's center
	l_{ft}	370.0	370.0	From Knee to Foot's center
Force [gf]	F_B	195.0	195.0	Buoyanct force due to helium gas
	F_b	76.2	76.2	Net buoyancy

2.2.1 Design Considerations

BALLU's unique nature requires specific design considerations that differ from conventional robots. One of the most significant design considerations is mass distribution. While a large buoyancy is required to support many heavy body parts, the use of too many balloons creates new problems, such as increased drag force, which necessitates stronger actuators and heavier body parts. To maintain a reasonable number of balloons, it is important to adopt smaller and lighter parts and materials while staying within a certain amount of total mass, or the *mass budget*.

Mass distribution means simply more than the mass budget. In BALLU's walking, the balance between various forces, such as buoyancy, weight, ground reaction forces, and friction, is critical. Therefore, the center of buoyancy and the center of mass of each part significantly affect the robot's movement. The detailed description is provided in Section 3.2. Mass distribution on the foot is beneficial in utilizing the mass budget efficiently. Unlike existing legged robots that aim to minimize the weight of the leg end to minimize the effect of leg motion on the body's momentum, BALLU concentrates the mass distribution on the foot to allow more body weight with the same buoyancy.

Mass distribution on the foot is beneficial in utilizing the mass budget efficiently. While buoyancy primarily supports the weight above the knee joint, the weight below the knee can be supported by the ground reaction force. Unlike existing legged robots that aim to minimize the weight of the leg end to minimize the effect of leg motion on the body's momentum [7, 19, 29, 52, 62], BALLU concentrates the mass distribution on the foot to allow more body weight with the same buoyancy.

In addition, lightweight and small hardware primarily has the disadvantage of low fidelity and high uncertainty. Furthermore, the fact that the foot is heavy distinguishes BALLU from existing robots and makes it unconventional. Such characteristics can lead to control problems, which are discussed in Section 3.1.

2.2.2 Balloons

Because the buoyancy force plays an important role for BALLU, such a consistent and reliable external force is achieved through the use of off-the-shelf mylar balloons filled with helium. These balloons are low-cost and easy to purchase, while they also result in minimal deflation per the authors' experience. Any number of balloons (in our case, we have experimented with 2 to 6) with various shapes could be used, and they are held together by threading lightweight wires through holes located at the balloon's corners.

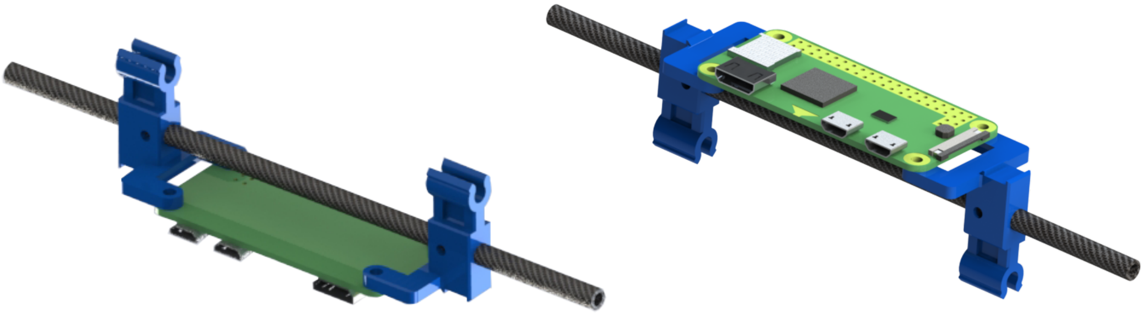
The magnitude of the balloons' net buoyancy, which is the difference between the buoyancy of the helium and the weight of the balloons, must be smaller than the total body weight to prevent the robot from floating in the air. This net buoyancy is controlled to support most of the body weight, with the normal forces at each foot supporting the rest.

In practice, as injecting the same amount of helium every time to keep the body afloat is difficult, a generous amount of helium is initially injected, and counterweights are attached to the body to adjust the net upward force. This is done by calibrating the robot's normal weight on a scale, which is empirically chosen to be 55 gf in the presented version of BALLU. Such a choice allows the robot to stay vertically upright when in a double support phase, but sink when in a single support phase, which will be important in the subsequent locomotion approach.

2.2.3 Pelvis Link

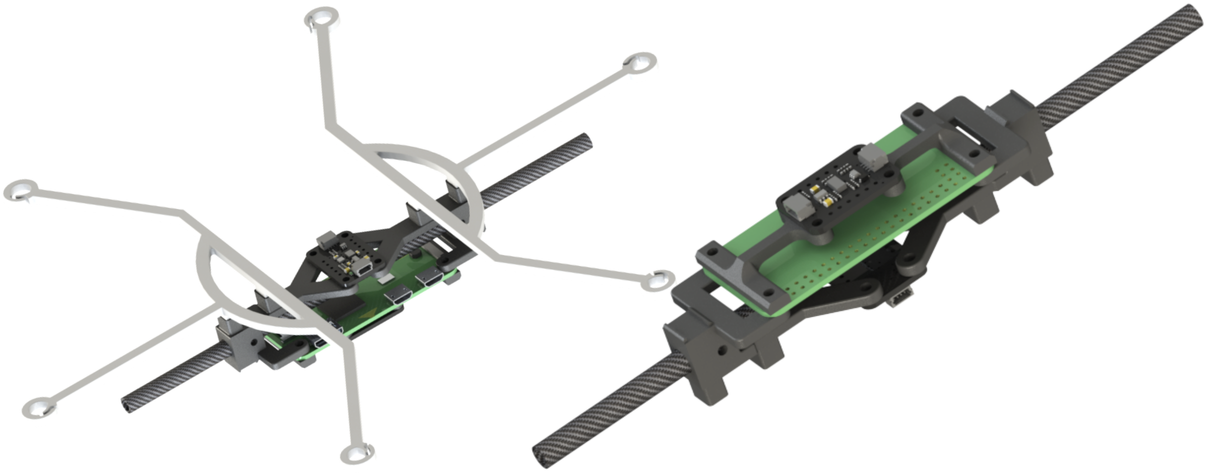
The pelvis link is what holds the legs and the balloons together, and is also the mount for BALLU's onboard controller. A Raspberry Pi Zero W is used for its low-cost, lightweight, potential onboard computing, and flexible communication (e.g. WiFi and Bluetooth) capabilities. These updates are a distinct difference from the first version [26], which was limited to teleoperation via radio controllers, and allows BALLU2 to walk based on algorithms.

On BALLU3, an upgraded Raspberry Pi Zero 2 W is used as a computing module, whose computation is $1.8 \sim 10.3$ times faster [25, 40] compared to the previous version. a small



(a) Top (b) Bottom

Figure 2.3: Pelvis design of BALLU2



(a) Top (b) Bottom

Figure 2.4: Pelvis design of BALLU3

neodymium magnet is inserted at each end of the pelvis carbon fiber tube. The corresponding encoder board is embedded in the hip joint part and attached to the pelvis when the leg is assembled with the pelvis link.

In the case of BALLU2, the bottom of the balloon set is attached to a thin carbon fiber tube, and the tube slides into the slots of the 3d printed pelvis parts as illustrated in Figure 2.3. This design allows easy maintenance due to its detachability but also forms a pitch joint between the balloon set and the pelvis link, called the Neck joint. The redundant pitch rotation around the neck joint makes it harder to estimate the pose of the balloon set and the analysis of the locomotion. BALLU3 uses the balloons using a pair of balloon holders, as depicted in Section 2.2.3 to connect them to the pelvis, which eliminates the redundant joint and makes a more rigid connection. The housing

In control aspects, it is a convenient choice to take the center of the pelvis link as the origin of the floating body since the pelvis is where all the forces from the legs and the balloons are congregated at.

2.2.4 Legs

Each leg is a modular component that can be attached to the pelvis link, comprised of a hip joint, a femur and tibia links, a knee joint holding the two links, and a foot. The links are hollow carbon fiber parts that the wires go through. Square carbon fiber tubes (are chosen so it is easier to align the hip joint, knee joint, and foot.

2.2.5 Hip Joints

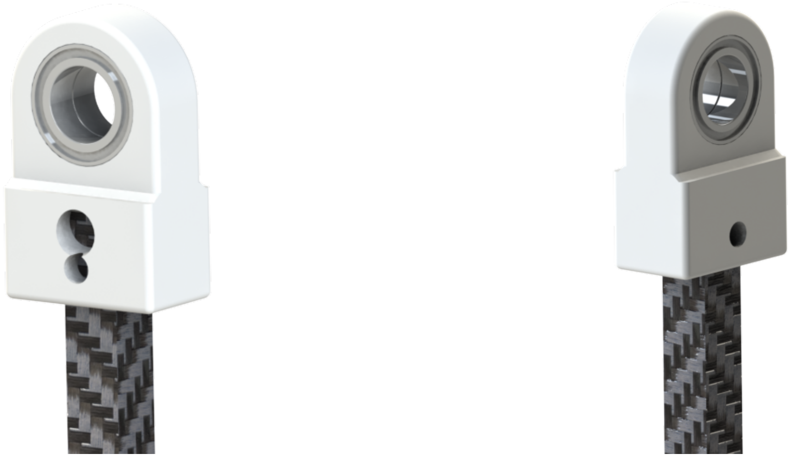


Figure 2.5: Hip Design of BALLU2

The hip joint is simply a 3D-printed part with a bearing. in it that slots around the pelvis link so that they freely swing without actuation.

As shown in Section 2.2.7, BALLU3's hip joint is redesigned to have two bearings at each hip joint, and there is an offset between them. This design improves robustness and prevents the leg's movement in the lateral direction more securely.

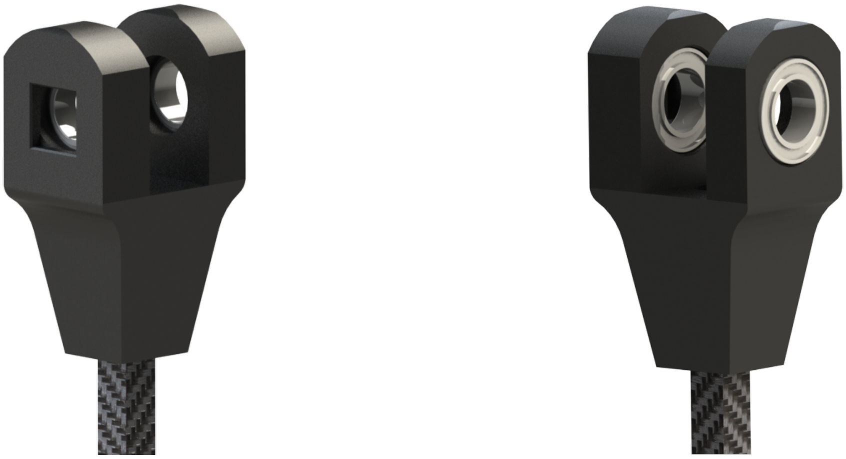
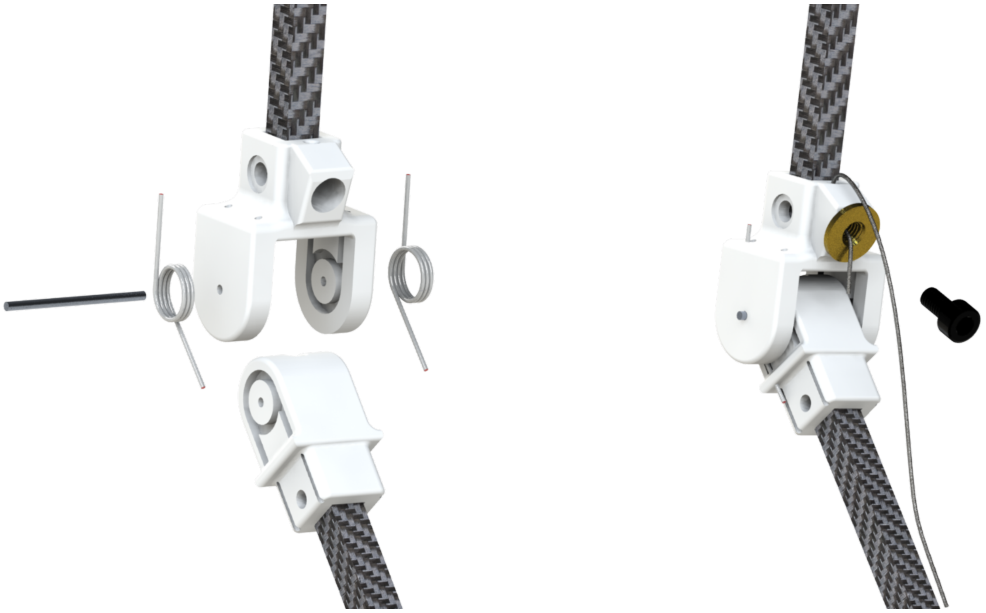


Figure 2.6: Hip Design of BALLU3

2.2.6 Knee Joints



(a) Exploded view

(b) Tendon bolt and wire

Figure 2.8: Knee Design of BALLU2

The knee joint design is illustrated in Figure 2.8: It consists of upper and lower parts, two symmetric torsion springs attached to each side, a metal pin, a tendon wire, and a tendon bolt. As shown in Section 2.2.5, the motor arm, tendon adjustment module, the lower femur, and tibia form a four-bar mechanism. The knee springs are preloaded, and it allows for the

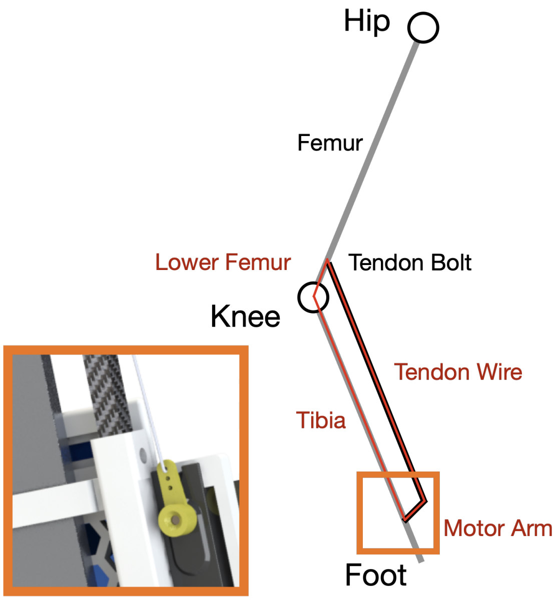


Figure 2.7: Fourbar linkage diagram of a leg

knee to be quickly unbent when the wire is not in tension. The tendon wire starting from the servo motor arm goes into the socket embedded behind the upper joint part, and the initial knee joint angle and the length of the tendon wire can be adjusted with a bolt.

The spring constant is determined so that the legs extend in a double support phase and bend in a single support phase, which assists with controlling BALLU, as will be described in Sections 3.2.4 and 3.2.5. Considering Figure 2.9, these conditions can be represented first in terms of the vertical force acting on the pelvis

$$F_{p,z,ds} = 2F - m_p g + F_b > 0 \tag{2.1}$$

$$F_{p,z,ss} = F - (m_p + m_{leg})\, g + F_b < 0 \tag{2.2}$$

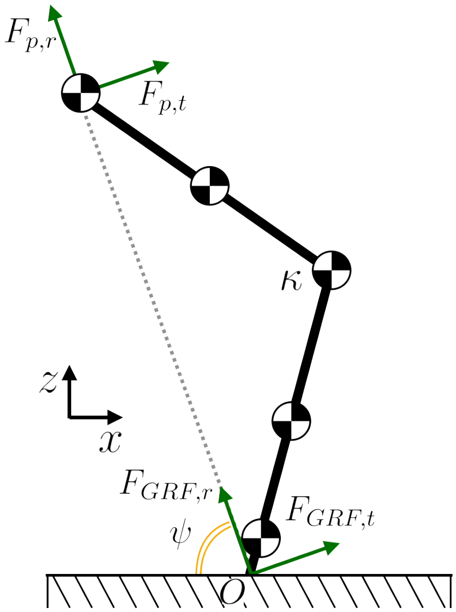


Figure 2.9: Leg diagram seen from the side

, where

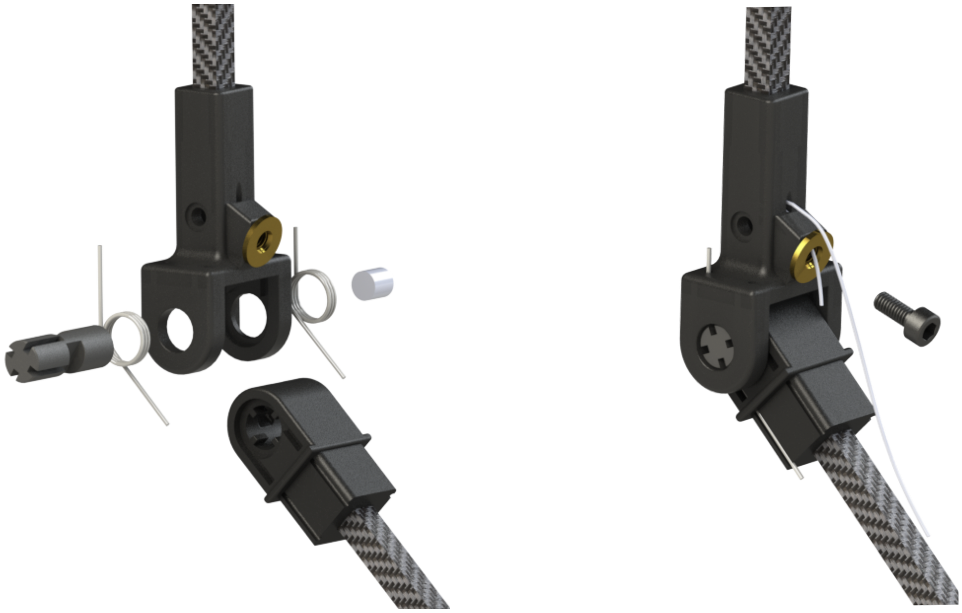
$$
\begin{aligned}
m_{leg} &= m_{hp} + m_{fr} + m_k + m_{tb} + m_{ft} \\
F_z &= F_{p,r} \sin\psi + F_{p,t} \cos\psi \\
F_{p,r} &= -A_r(\phi(q_k), l)\ \tau_k - f_{hip,r}(\phi(q_k), l, m) \qquad (2.3) \\
F_{p,t} &= -\underbrace{A_t(\phi(q_k), l)}_{0}\ \tau_k - f_{hip,t}(\phi(q_k), l, m) \qquad (2.4)
\end{aligned}
$$

F_p is the force from the leg at the hip in the radial direction connecting the hip and the foot, and $F_{p,t}$ is in the tangential direction.

The inequalities Equations (2.1) and (2.2) can be rewritten in terms of the knee spring torque. The torsion spring is preloaded, and the torque should be positive since we only want to consider the torque in straightening the knee joint.

$$
\begin{aligned}
\tau_k &= \kappa\,(\theta + \theta_0) \\
\tau_{k,ds} <& \tau_k < \tau_{k,ss} \qquad (2.5) \\
\tau_k &> 0
\end{aligned}
$$

For the design parameters and the knee joint displacement θ within the joint limits, a range of spring constant κ can be obtained from Equation (2.5). Among satisfying κ, we prefer to choose a relatively low stiffness. It is because, when the spring constant takes a lower value, the dynamics change more drastically between a single support phase and a double support phase, and the controller would have more options to control its motion by adjusting each phase length. Through a handful of empirical tests, a torsion spring with 0.1409 N mm/deg with 135° was chosen.



(a) Exploded view

(b) Tendon bolt and wire

Figure 2.10: Knee Design of BALLU3

As shown in Figure 2.10, BALLU3's knee has a different design. It has a custom knee shaft unlike BALLU2's uses a thin metal pin as a shaft. The new custom knee shaft is 3d-printed and a neodymium magnet is embedded at its core, which works with an encoder attached to the outside of the upper knee part.

2.2.7 Feet

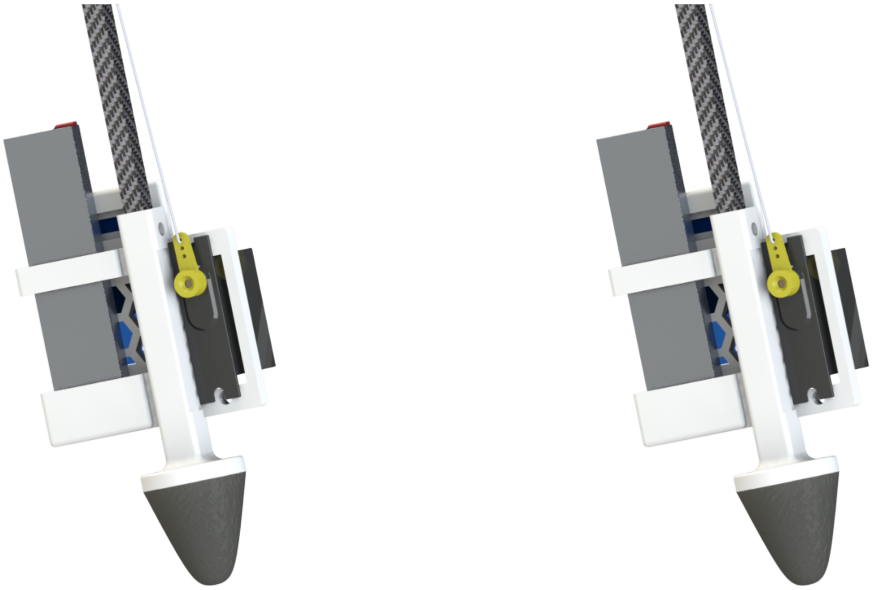


Figure 2.11: Foot Design of BALLU

Unlike the relatively lightweight legs, the feet hold heavier components, which include a power board, a 3.7 V Lithium Polymer battery, and a servo motor; BALLU2 uses Dymond D47 while BALLU3 adopted KST X08+, which has stronger motor torque but in a similar form factor. It allows the knee to have a wider range of motion. The power board (Adafruit PowerBoost 1000 Basic) converts 3.7 V to 5 V for the servos and the computing board at the pelvis. The computing board commands the servo, which effectively actuates the knee through a wire-driven four-bar mechanism as shown in Section 2.2.5. For high friction point-like contact, a cone-shaped rubber is attached at the end.

While the foot design of BALLU2 (Section 2.2.7) can effectively hold all the electronic parts, due to its rather flat shape, the bottom of the battery pocket touches the ground when BALLU2 walks aggressively. This issue was addressed in the foot design of BALLU3 (Section 2.2.7) by moving the battery pocket backward.

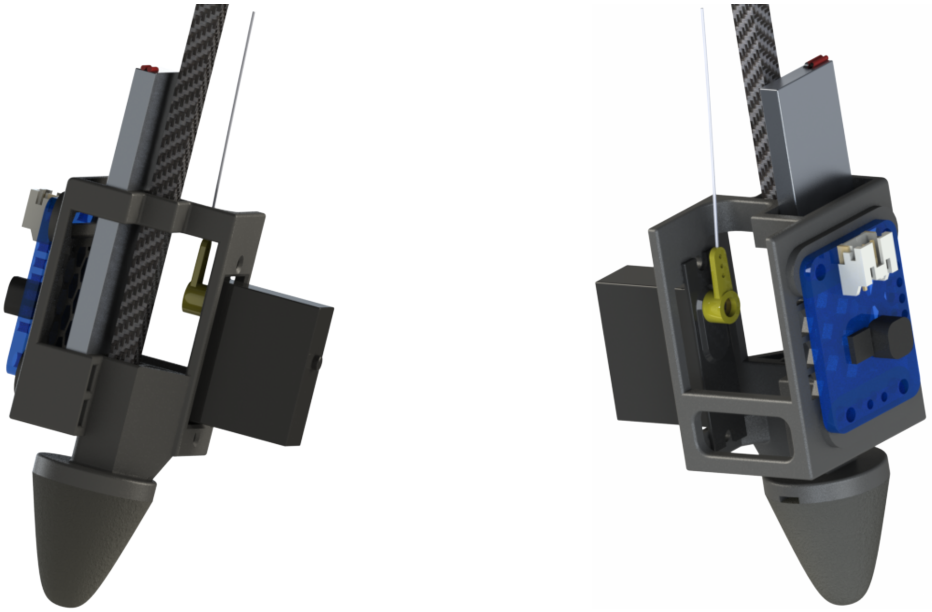
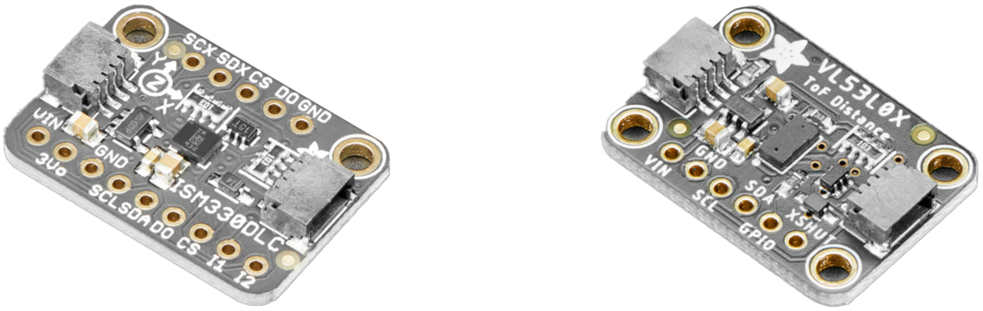


Figure 2.12: Foot Design of BALLU



(a) ISM330DHCX

(b) VL53L0X

Figure 2.13: Sensors: (a) IMU, (b) ToF

2.2.8 Sensors

For the inertial measurement unit (IMU) at the center of the pelvis, Adafruits' IMU based on ST's 6-DoF ISM330DHCX is chosen (Figure 2.13a). For the time of flight (ToF) sensor, Adafruits' ToF sensor based on ST VL53L0X is selected (Figure 2.13b), which can measure about up to 1000 mm. A custom encoder board based on RLS' AM4096 12-bit rotary encoder Figure 2.14. On each Knee and Hip joint. For the communication protocol between the sensors and the computer, I^{2}C is adopted because it supports daisy chaining, which can reduce the number and the length of wires significantly.

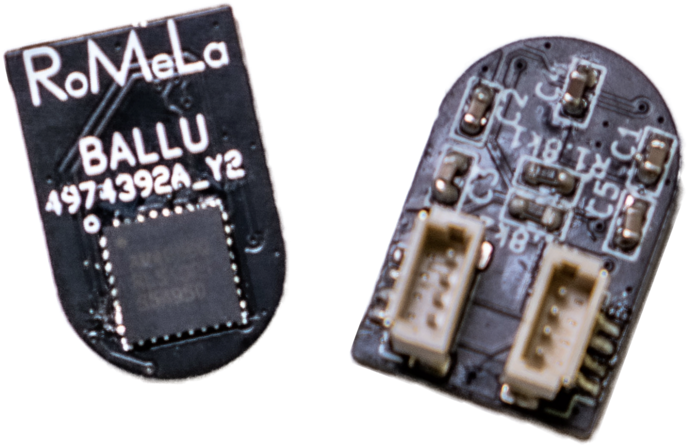


Figure 2.14: Encoder boards for BALLU3

2.3 Software Design

The current system is set up for easy and modular development, as various processes are concurrently running on board, including the motor controller and the communication and messaging module. Shared memory is leveraged to share data between processes. Python is primarily used for simplicity, while C++ is used for low-level modules.

2.3.1 Design Constraints

Due to the mass budget described in Section 2.2.1, BALLU is required to equip a lightweight single-board computer, which has limited computing power compared to the computers on the other legged robots [18, 27, 29, 33, 46]. In this case, it is restricted from running an immature controller in the earlier development phase or training a data-driven controller on-board. One possible solution is to develop the controller with the assistance of external computing resources and put it back on the onboard computer once it is optimized. This idea led BALLU to have both off-board computing and on-board computing architectures.

On the other hand, BALLU is part of a collaborative project funded by the National Science Foundation[1], and researchers from different disciplines are collaborating, ranging from implementing the hardware platform to devising locomotion algorithms based on machine learning. To expedite research and satisfy different technical needs, the BALLU framework provides interfaces in different levels of abstraction: a low-level interface that provides details of the robot and a high-level interface, which is an OpenAI Gym [13] style wrapper around the low-level interface. Also, various simulators are supported: CoppeliaSim (formerly V-Rep) [48], PyBullet [16], MuJoCo [56]. To assist research based on reinforcement learning, the framework supports the vectorized environment of stable_baseline3 [43] library.

In addition, several modules need to be implemented based on mathematical models to increase numerical stability. This chapter covers the mathematical model behind the implementations.

[1] Grant# 2024940

2.3.2 Control Arthitecture

2.3.2.1 Off-board Computing Architecture

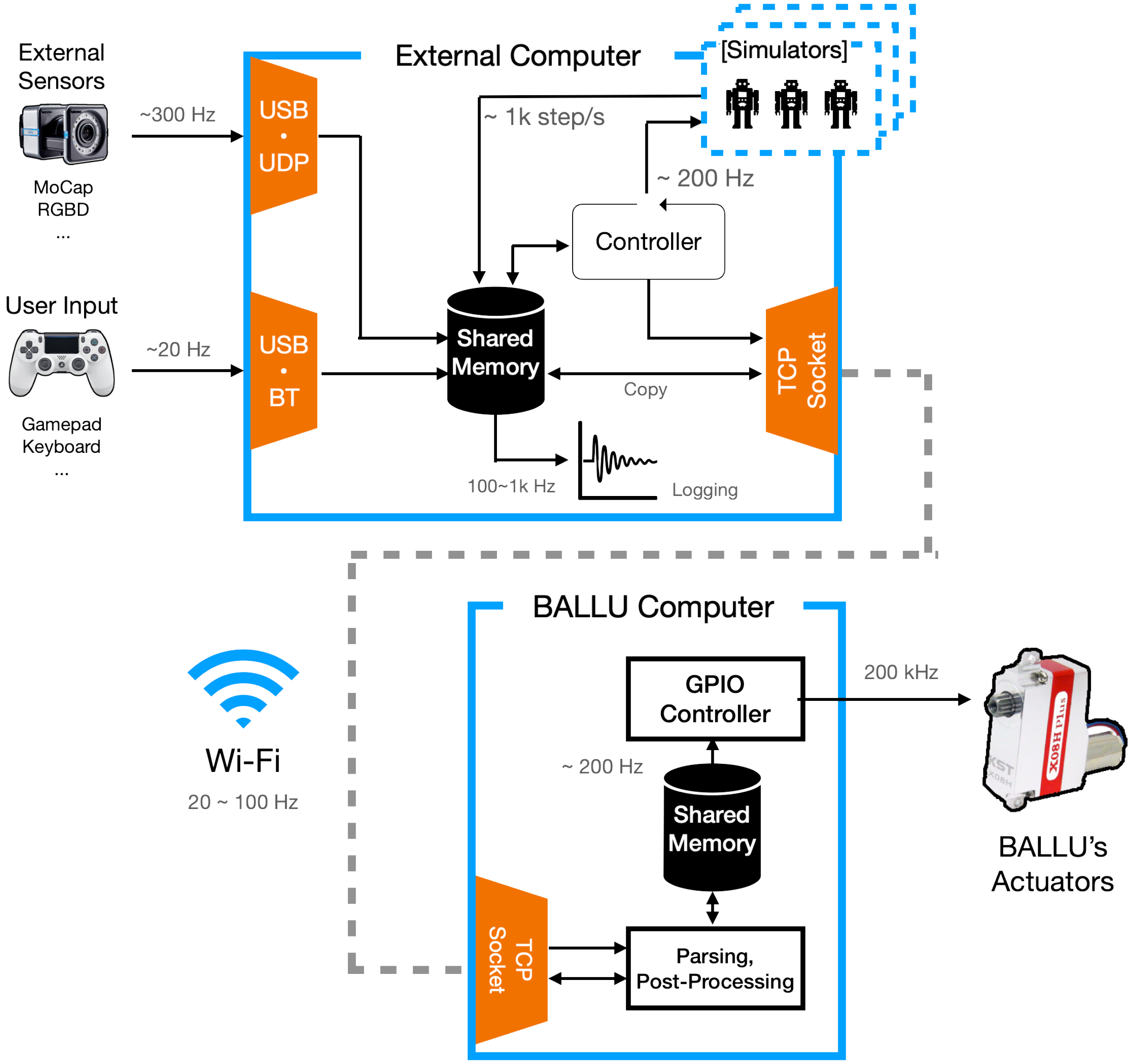


Figure 2.15: Off-board Computing Architecture

Figure 2.15 represents a diagram of the off-board computing architecture. Most of the computing takes place on an external computer, with various modules running in parallel. The architecture, which consists of multiple processes running simultaneously, shares data across the architecture through shared memory. This allows for a modular and flexible structure, as multiple modules are loosely connected via shared memory.

The main computer can receive data from user input and external sensors. In the case of BALLU2, which lacks an on-board state estimator, state feedback is provided by an RGBD camera or a motion capture system.

The command output calculated by the controller is sent to a set of simulation environments via shared memory or sent to the BALLU computer via TCP wireless communication. Similarly, data from BALLU or simulations is collected and stored in shared memory.

BALLU receives data from the main computer, which requires a few steps of post-processing before being stored in shared memory. Then the command actuates the actuators. Shared memory also exists in the BALLU computer. The communication modules between the external computer and the BALLU computer synchronize each other's shared memory. As a result, modules behind the shared memory can run the command fetched from the shared memory without knowing whether they are in the off-board or on-board architecture.

Furthermore, logging is performed on the main computer.

2.3.2.2 On-board Computing Architecture

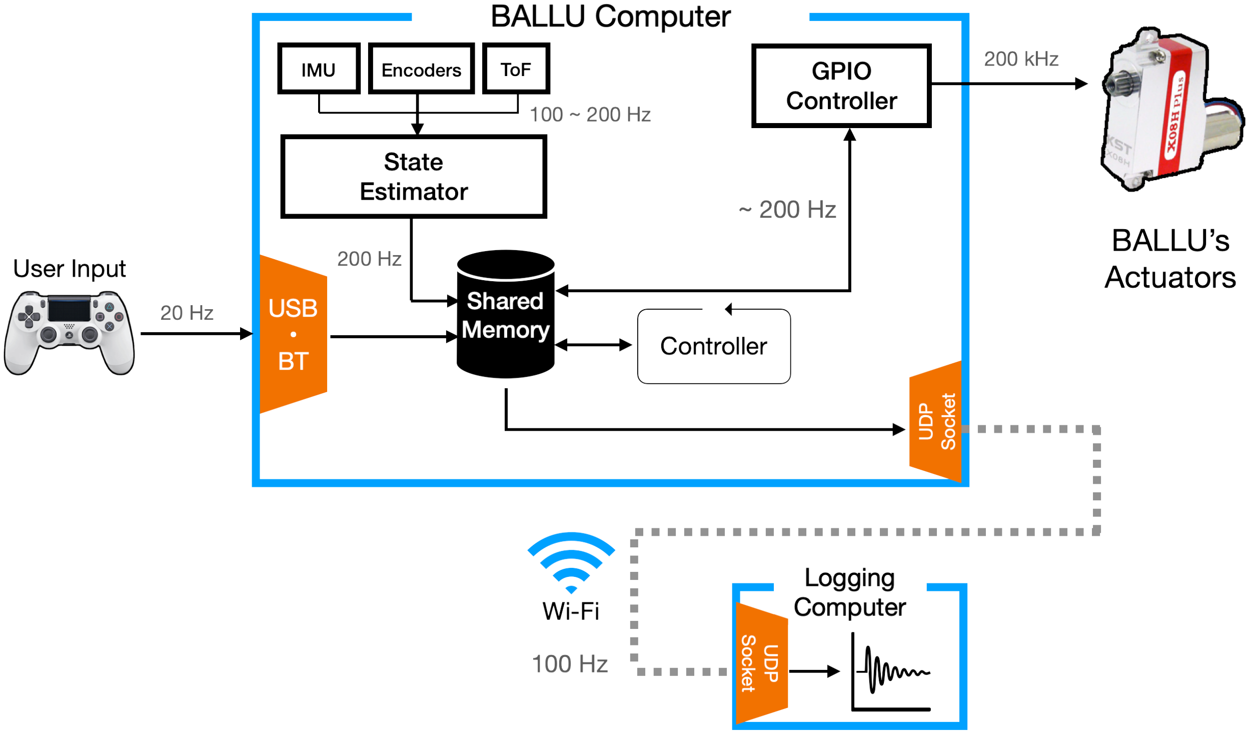


Figure 2.16: On-board Computing Architecture

Once the controller has matured as sufficiently optimized as it could run on-board, it can reduce the overhead of off-baord computing by taking the on-board architecture.

Figure 2.16 shows a diagram of the on-board computing scheme. Like the off-board architecture, shared memory is at the core of the architecture, which facilitates data sharing across multiple modules. BALLU3 obtains data from IMU, Encoder, and ToF sensors to perform onboard state estimation as well as get user input. For more information on state estimation, refer to Section 2.3.4. The controller computes the control command based on the estimated state and the user input, and the command is sent to the actuator.

Meanwhile, logging is still performed on an external computer through wireless communication as it is a relatively heavy task.

2.3.3 Mathematical Modeling for Simulation Environments

Aerodynamics simulation is crucial for BALLU simulation. However, not all of the aforementioned simulators support buoyancy and drag force simulation. To support the aerodynamics simulation in a common interface, these forces are mathematically modeled and programmatically implemented in the framework.

Furthermore, the tendon-driven four-bar linkage mechanism is implemented mathematically. Although there are several common techniques to approximate the behavior of flexible elements such as chains or fabrics, the difference in the mass of the tendon string and the other robot parts causes severe numerical instability in the case of BALLU simulation. To secure numerical stability and faster simulation, BALLU's kinematics including the tendon-driven four-bar linkage system is modeled.

2.3.3.1 Buoyancy

Buoyancy is an external force that always acts on the pelvis link in the opposite direction of gravity. The magnitude of the net buoyant force, $\mathbf{F_b}$, taking into account the weight of the balloons themselves, can be modeled as follows:

$$\mathbf{F_b} = \tilde{F_b}\hat{\mathbf{z}}$$

, and the magnitude, $\tilde{F_b}$ can be measured by a scale. The magnitude of this net buoyant force is calibrated as on the physical platform to maintain a normal force of 55 gf.

2.3.3.2 Drag

What is just as important as buoyancy for BALLU is a drag. Because the robot is lightweight yet the body takes a large portion, drag force plays a nontrivial role in BALLU's orientation. Consequently, the drag forces for the transitional and rotational directions with the robot's X, Y, and Z axes were calculated using computational fluid dynamics software. Because

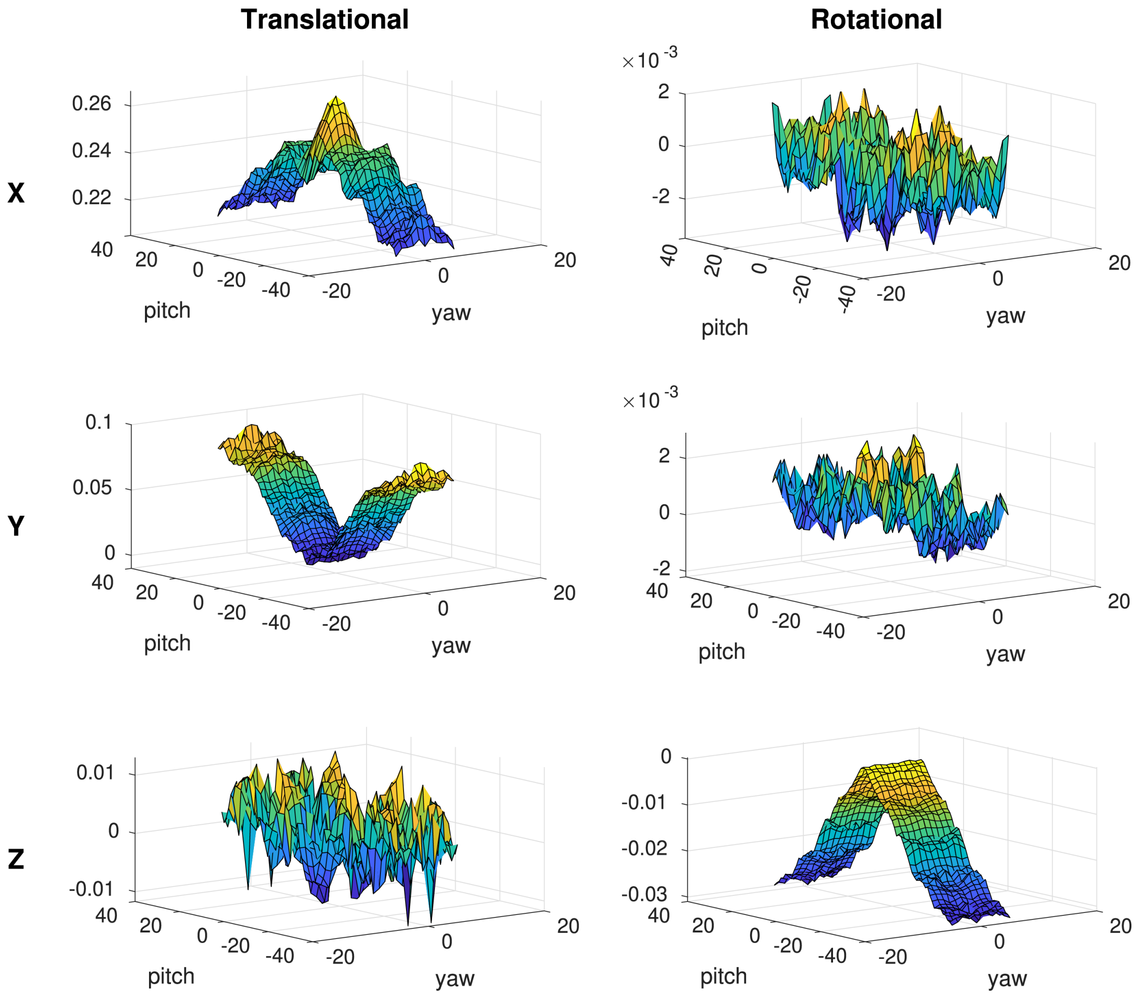


Figure 2.17: Drag force computed using computational fluid dynamics software. Since they are based on the airflow with unit velocity, the drag coefficient and drag force can be calculated using them. The X-axis directs the heading direction, the Y-axis lateral, and the Z-axis vertical.

the lateral distance between two feet keeps the robot from rotating in the roll direction, only the rotations in the pitch and yaw direction are taken as the domain variables for the computation. Considering the rate of change in the pitch and yaw directions that BALLU normally takes, the drag force was computed over ± 40 deg for the pitch and ± 5 deg for the yaw angle with the unit speed.

The results represented in Figure 2.17 show that the translational force in the X (heading) axis was dominant ($7.0 \sim 1000.0$ times larger) among the remaining five directions. Although the value in the Y-direction might be comparable, again, the robot barely moves in the lateral direction due to the distance between the two feet, and the expected lateral drag force is very small. It suggests that it would be sufficient to model the drag force as a single force acting in the X-direction.

We adopt the following commonly used model for representing transitional drag force, which is a quadratic function of the relative flow speed of the object to the fluid:

$$F_{drag} = -\mathbf{C_d v} \approx -C_{d,x}\ v_x \hat{\mathbf{x}}$$

where C_d is the drag coefficient, A is the reference area which is a function of the heading direction, and v_{com} is the speed of the robot.

The simulation result was taken as the drag coefficient at a pitch and yaw pair since it is computed with unit speed. In the dynamics simulation, the drag force is computed using the model and applied as an opposing force in the simulation.

2.3.3.3 Kinematics

In order for faster computation, numerical stability, and a consistent development environment no matter if a robot simulator supports a kinematics module or not, a mathematical kinematics model is derived and implemented in the BALLU framework. A mathematical model for the four-bar linkage system that relates the knee joint and the servo motor at the ankle is derived in Appendix A.1. The leg kinematics in terms of the hip and knee joint is developed in Appendix A.

2.3.4 State Estimator

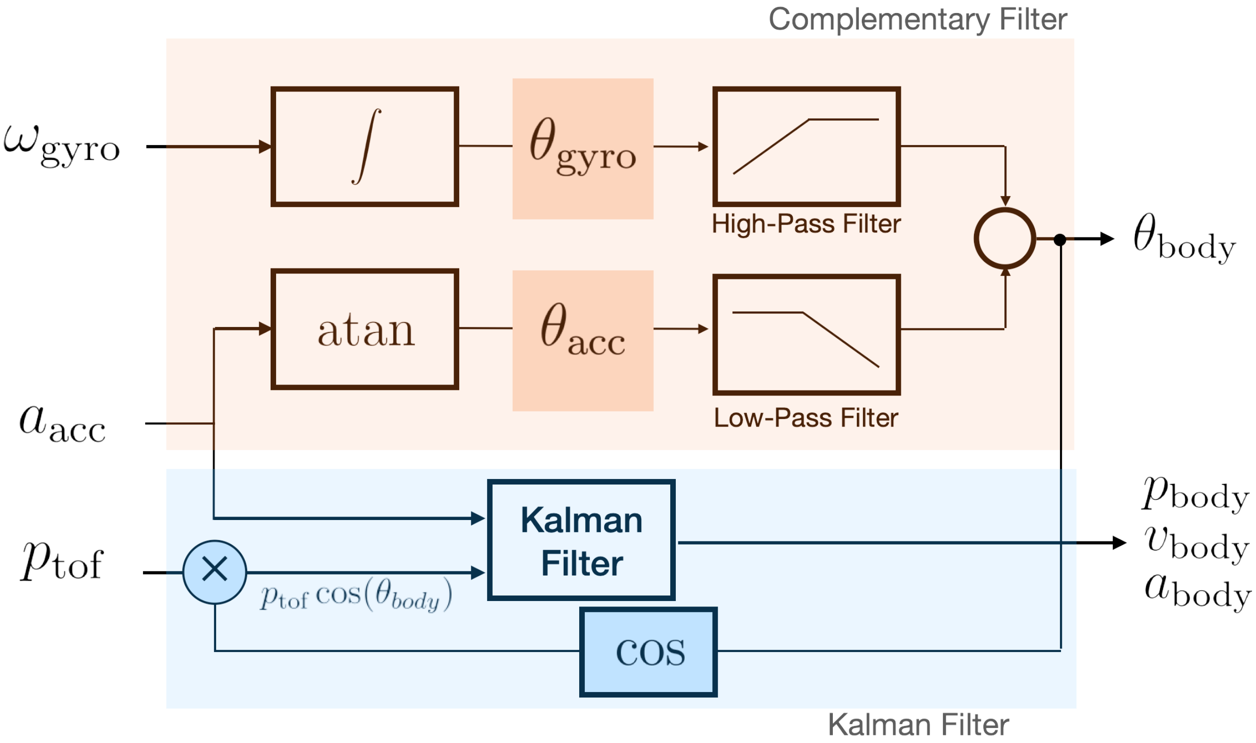


Figure 2.18: State estimator architecture

Figure 2.18 shows the configuration of BALLU's state estimator. The state estimator estimates the position and the orientation of the center of the pelvis.

In order to decrease the complexity of the computation, a complementary filter [28] is adopted for the orientation. It calculates orientation from two sources: integration of the angular velocity and comparison of the linear acceleration components. Then the filter fuses the two orientations with the filter gain.

On the other hand, the position of the body is calculated by a Kalman Filter [32]. Based on the accelerometer and ToF sensor measurements, it calculates filtered acceleration, velocity, and position of the body. When the ToF sensor measures the body height, the rotation of the sensor has to be addressed because it is attached on the bottom of the pelvis link. The orientation from the complementary filter is used in this step.

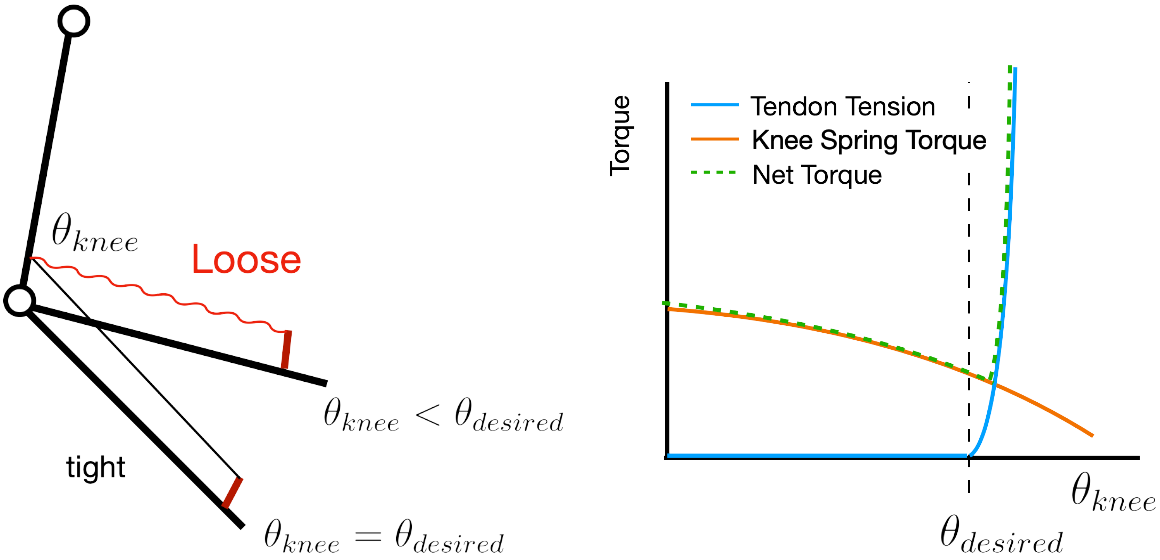
CHAPTER 3

System Analysis

3.1 Challenge

BALLU is a distinctive bipedal robot that, due to the external buoyancy force and its simple configuration, demonstrates a unique locomotion behavior, unlike conventional bipeds. When a first-time operator is tasked to trigger each leg individually and make the robot walk, the robot can exhibit non-intuitive behaviors. The most prominent of these behaviors is the turning in the yaw direction, where because of the underactuation and the mass of the legs relative to the entire body, the body's yaw orientation can change significantly depending on the duration that the robot is in the single stance phase. Furthermore, as the body oscillates up and down during locomotion and since buoyancy and drag distort the speed at which the robot moves, BALLU looks as if it is slowly striding in space. Such non-intuitive and unconventional behavior inherent in the design calls for a detailed analysis of the platform's motions to potentially leverage them for control.

3.1.1 Underactuation



(a) (b)

Figure 3.1: Underactuation due to tendon-driven mechanism

BALLU has only 2 DoF for each leg and only 1 active DoF on the knee because the hip joint is passive and freely rotates. Even the single active actuation is driven by the tendon mechanism, which is directionally underactuated. As shown in Figure 3.1a, depending on the configuration, the tendon wire gets loose and lost the actuation. This issue is mitigated (Figure 3.1b) by adding torsional springs at the knee, and the knee spring's effect is discussed in Section 3.2.3.

Unlike the majority of robots that can follow the desired trajectory generated by a controller, passive dynamics govern BALLU's hip joints and the controller has to realize the desired motion considering that the hip joint can only swing freely.

3.1.2 Nonlinear Dynamics

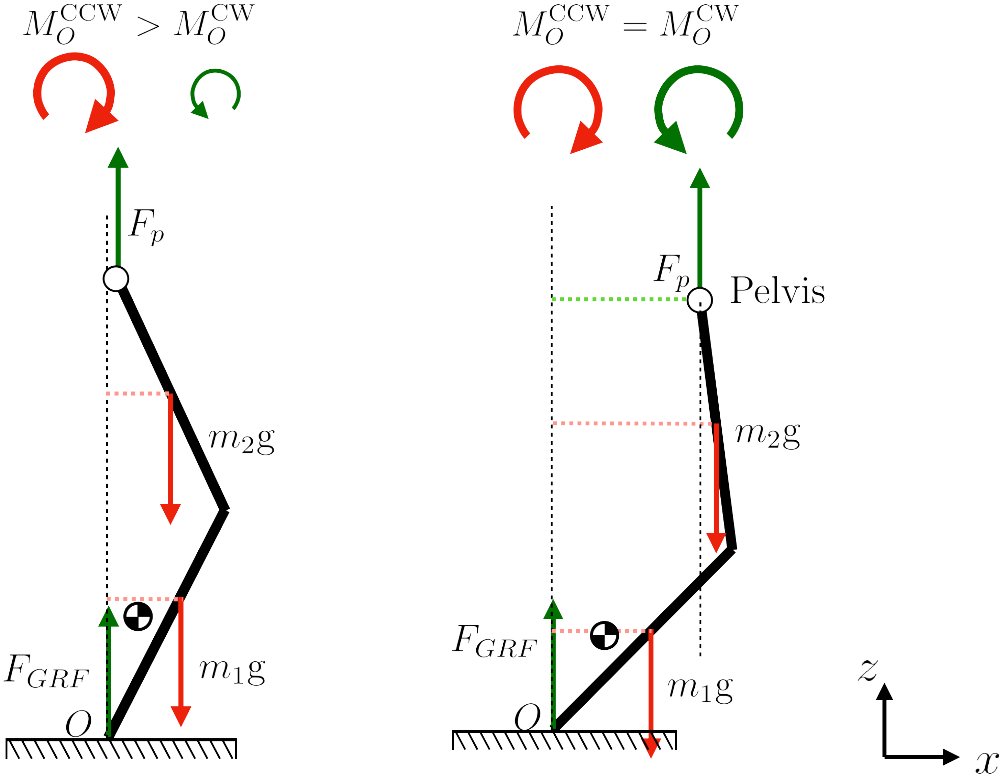
The drag force disrupts BALLU's walking motion because of the balloon's large cross-section. This adds additional complexity to the already nonlinear dynamics of the rigid body. The large fluctuation makes it even harder to analyze. The magnitude is about 5.5% compared to its body weight and can peak up to about 12.0%. Moreover, there are multiple sources of uncertainty on the platform. For example, while it is necessary to adopt lightweight and affordable parts, smaller and low-cost sensors and actuators tend to have lower fidelity. In addition, light materials are prone to wear, and helium balloons lose their buoyancy over time, which makes the system time-varying and the identified parameters unknown.

Developing a motion planner for BALLU is a nontrivial task. From the authors' remote control experiences, BALLU is able to walk, climb stairs, jump, and turn with proper actuation timing. However, because of the complex interaction between the balloons, which are affected by aerodynamics, and the underactuated rigid body, it is difficult and counterintuitive to imagine how BALLU should locomote. To find insights from observing behaviors from successful teleoperations, a substantial amount of remote control experiments in various environments were done. As a result, the authors were able to get a few insights that would be the cornerstones to develop a controller in Section 4.1 and the future.

3.2 Motion Analysis

BALLU's motion incorporates complex dynamic components, such as underactuation and aerodynamics, which differ from the rigid body dynamics typically found in conventional robots' dynamics. Consequently, understanding the reasons behind its movements is not intuitively straightforward. This section discusses the principles of BALLU's motion generation, which were identified through extensive teleoperation experience and observation.

3.2.1 Principle behind Leg's Swing



(a) Upright (b) Moments at equilibrium

Figure 3.2: Static moment balance: when the counter-clockwise momentum and the clockwise moment components make balance, the center of mass of the leg is located behind the vertical line from the pelvis. The gravitational forces and their resultant moment about point O are denoted by the red arrows. The net buoyancy, the ground reaction force, and their resultant momentum are denoted by the green arrows.

This section discusses how BALLU's legs swing. Consider the stationary leg of BALLU as shown in Figure 3.2. The mass distribution and gravitational forces are simplified for clarity, but the principle still applies the same in the general case. First, let us consider the situation where BALLU is standing upright, as depicted in Figure 3.2. In this case, calculating the moments acting on the leg around the contact point O reveals that the clockwise moment is larger than the counterclockwise moment, causing the leg to rotate forward and the body to lean forward, as shown in Figure 3.2b. The body will continue to lean forward until the moments reach equilibrium.

As mentioned in Section 2.2.1, most of the mass is concentrated at the end of the leg. Thus, the center of mass of the leg is close to the foot, as shown in the diagram. When the moments are in equilibrium, the center of mass of the leg must be located behind the vertical line drawn from the hip joint. At this moment, The reason the leg does not swing is that the foot is in contact with the ground, and the ground reaction force impedes the leg's swing.

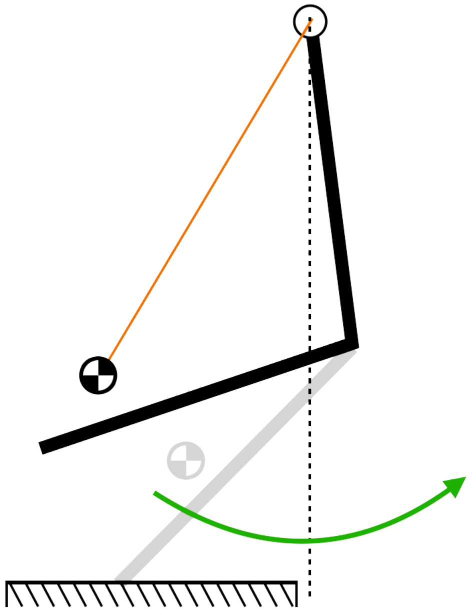


Figure 3.3: Take off: the leg swings because the center of mass of the leg is located behind the vertical line from the pelvis when the knee is actuated.

As depicted in Figure 3.3, when the leg is actuated in this state, it momentarily folds, breaking contact between the foot and the ground, and the center of mass of the leg rises slightly. Now, BALLU's leg can be thought of as a pendulum swinging around the hip joint.

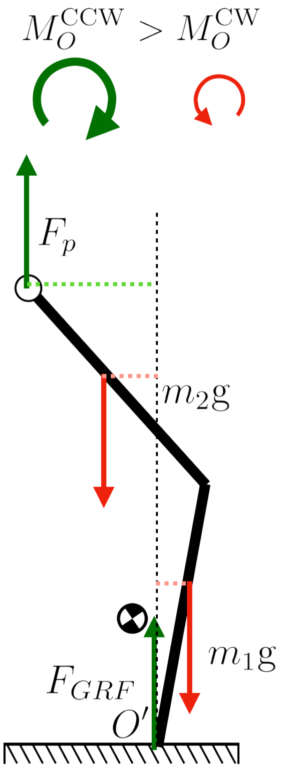


Figure 3.4: Touch down: again, the leg will lean forward since the counterclockwise component of the momentum around the new contact point O' is greater than the clockwise component.

Figure 3.4 illustrates the situation after the leg has swung and the foot has landed on the ground. At this point, similar to the first step, calculating the moments based on the new contact point O' shows that the leg rotates forward, creating a situation where the leg is leaning forward. Consequently, the body and balloon would move forward naturally.

Despite the directions of the moments created by each force pair being opposite to those in Figure 3.2a, the leg moves forward in the same manner. In other words, in most cases, the BALLU configuration inherently creates an environment that allows for swinging whenever the body leans forward and lifts the leg.

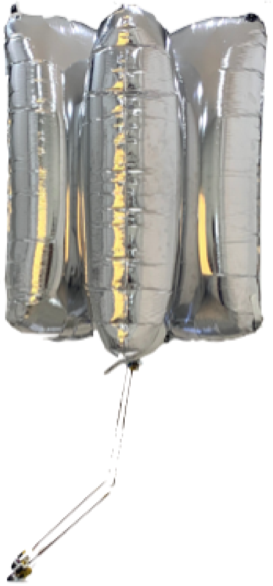


Figure 3.5: *Sink Down* state

3.2.2 Sink-Down State

BALLU is a robot that never falls down. It is a fact that BALLU, by design, cannot damage its surrounding environment or itself, unlike other heavier robots. However, interestingly, BALLU has its own counterpart to a conventional biped's *fall state*, which for BALLU, is when the robot has *sunk down* as shown in Figure 3.5. BALLU's body is slightly heavier than the net buoyancy that the balloons can exert. Hence, without any control, it sinks down until reaching equilibrium between the buoyancy, the GRFs, and the straightening force from the compressed knee joints. In this state, it is difficult to conduct any meaningful motion. In the *sink down* state, the knee joints are close to the joint limits and it is hard to make the leg swing without dragging its foot, despite most of the body parts still floating. Therefore, it is necessary for BALLU to manage its state with properly coordinated walking motions and avoid sinking down.

3.2.3 Spring Force Direction on Pelvis

Even without the knee springs, BALLU can balance and walk because of the force balance between the weights and the buoyancy. However, the torsional springs at the knee help the recovery of the knee when the wire is not in tension, which mitigates the underactuation issue described in fig. 3.1 and increase responsiveness. It is also helpful to keep the body not to fall into the sink down state. The following explains the effect of the springs.

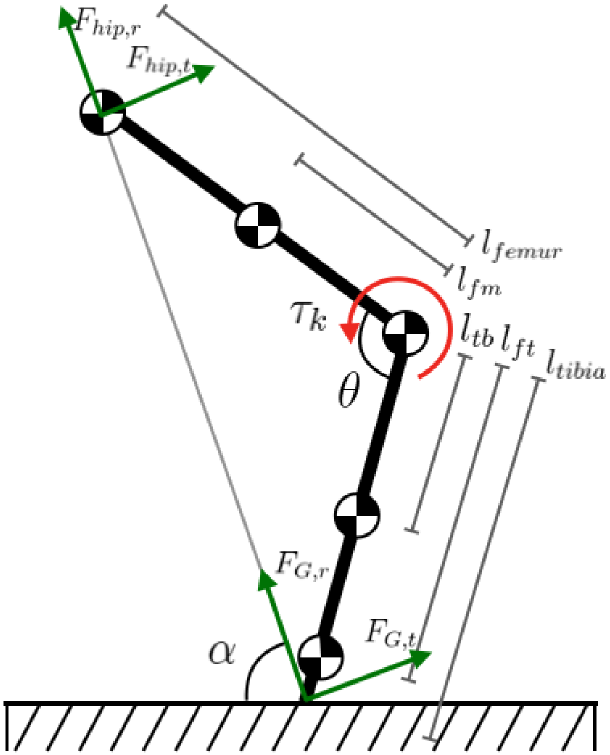
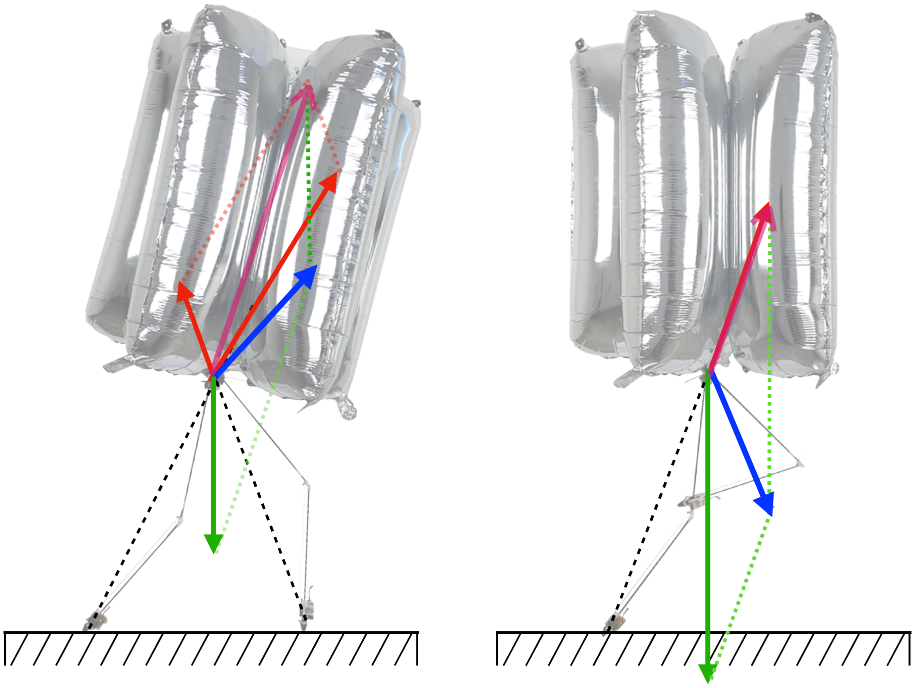


Figure 3.6: Forces acting on a leg

$$F_{p,r} = -A_r(\phi(q_k), l)\ \tau_k - f_{hip,r}(\phi(q_k), l, m) \tag{2.3 revisited}$$

$$F_{p,t} = -\underbrace{A_t(\phi(q_k), l)}_{0}\ \tau_k - f_{hip,t}(\phi(q_k), l, m) \tag{2.4 revisited}$$

Figure 3.6 depicts the side view of a leg. Decomposed into the radial direction connecting the hip and the foot and the tangential direction, the force applied to the pelvis link through



(a) Double support phase

(b) Single support phase

Figure 3.7: The forces acting on the pelvis: A red arrow represents a force due to the spring force from the knee, acting along the dashed line connecting the foot and the pelvis. The purple arrow is the vector sum of the red arrows. The green arrow is the sum of all the weights and buoyancy. The blue arrow represents the resultant force.

the hip joint is represented as Equations (2.3) and (2.4). The full derivation of the equations can be found in Appendix B. Here, $f_{hip,r}$ and $f_{hip,t}$ are the rest of the terms not related to the spring torque of the knee joint, and they are mostly gravitational effects. The spring torque from the knee joint always contributes along the radial direction.

This means the leg pushes off the pelvis along the line connecting the foot and the pelvis and allows to look at the entire body dynamics as interactions between the pelvis link and the two force vectors from each leg.

3.2.4 Height Control Strategy

Continuing the discussion in the previous section, we can analyze the relationship between the foot positions and the height of the body. In particular, the height shows a different pattern in single support phases and double support phases.

3.2.4.1 Double Support

As shown in Figure 3.7a, the pelvis is supported by two forces, and the resultant force acting on the pelvis link is more likely to be upward. In particular, when the footsteps are close, the forces are more focused in the vertical direction and strongly push the pelvis link. When the two footsteps are far, the front leg's knee is almost relaxed and the contribution from the hind leg is greater. The hind leg pushes the pelvis so that it moves towards the front and upward direction.

3.2.4.2 Single Support

When the robot is in a single support phase (Figure 3.7b), there is only one supporting force upward, and additionally, the force needs to support the weight of the swing leg, resulting in the height decreasing, and the body sinking down.

This behavior suggests that some feedback controllers could potentially regulate the height of the pelvis by looking at the current height and allocate appropriate single support and double support phases.

3.2.5 Speed Regulation Strategy

During remote control, there are a few ways to control the forward speed. One way is to control the swing time. Since a leg takes a great part of BALLU's total weight, after passing the nadir of its swing, the whole body gains velocity from the moment of the swing leg. The closer the leg swings to the apex, the greater forward speed the body gains when it exits the single support phase. However, the length of a single support phase directly affects the following footstep position, which influences the states in the subsequent phases.

If BALLU's feet are behind the center of the pelvis during the double support phase, the horizontal components of the two legs are effectively aligned and the robot can induce a large acceleration. However, this has a risk, as the robot can fall into the aforementioned sink-down state if it holds this state for too long.

When BALLU's speed drops considerably, it can recover it through a large footstep. As mentioned in Section 3.2.6, if it can position both feet forward, the further the feet are put, the larger acceleration the body gains when it reaches the apex. During this sequence, BALLU can recover both height and velocity, and move on to the next sequence of motions.

3.2.6 Footstep Position Selection

One widely used method [29] [4] [10] known as Raibert heuristics [44] is often used in legged robots to determine the next footstep position:

$$x_f = \frac{1}{2}\dot{x}T_s + k_{\dot{x}}(\dot{x} - \dot{x}_{des}) \tag{3.1}$$

where x_f is the next footstep position with respect to the center of mass, T_s is the duration of the phase, and $\dot{x}$ is the forward speed. The heuristic assumes the robot as a linear inverted pendulum and regulates the robot's CoM velocity by its foot placement. If the second term in Equation (3.1) is positive, the robot steps further than the nominal footstep position and accelerates in the following phase, and if the second term is negative, it steps closer and decelerates.

When BALLU is in a double support phase, the net force acting on the body is upward (unlike a convention robot, whose force would be downwards due to gravity), which results in behaviors that would match those exhibited by Equation (3.1), except with a negative on the velocity gain, as shown below:

$$x_f = \frac{1}{2}\dot{x}T_s - k_{\dot{x}}(\dot{x} - \dot{x}_{des}) \tag{3.2}$$

CHAPTER 4

Data-Driven Controller

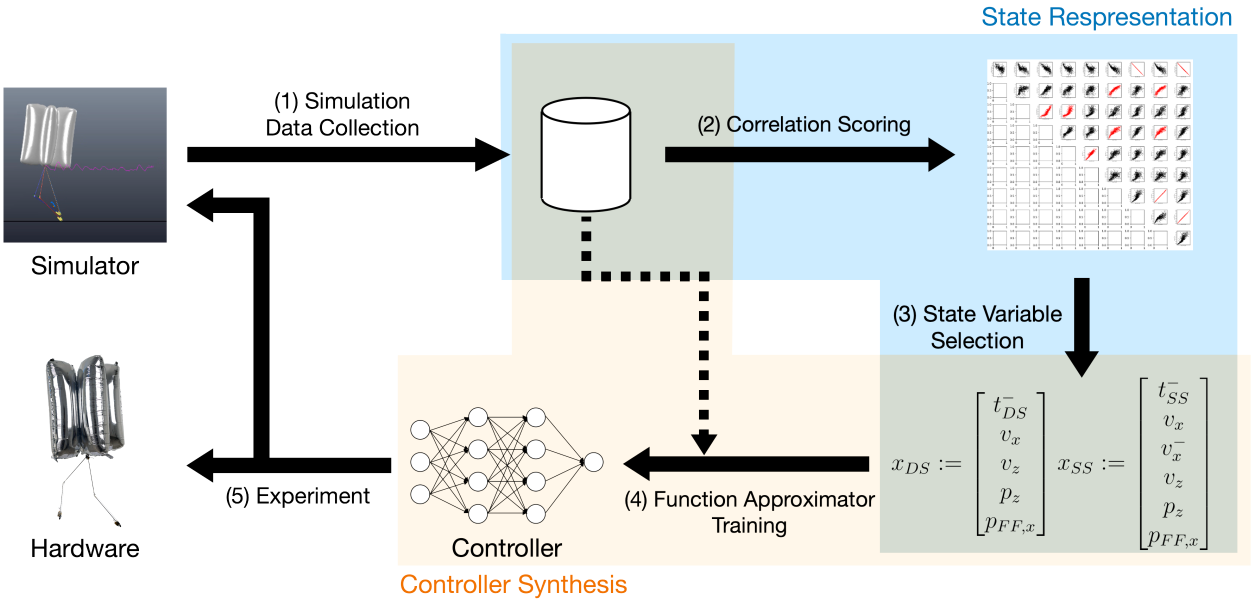

State Respresentation
(1) Simulation Data Collection
(2) Correlation Scoring
Simulator
(3) State Variable Selection
$x_{DS} := \begin{bmatrix} t^-_{DS} \\ v_x \\ v_z \\ p_z \\ p_{FF,x} \end{bmatrix}$
$x_{SS} := \begin{bmatrix} t^-_{SS} \\ v_x \\ v^-_x \\ v_z \\ p_z \\ p_{FF,x} \end{bmatrix}$
Hardware
(5) Experiment
Controller
(4) Function Approximator Training
Controller Synthesis

Figure 4.1: Approach summary

As discussed in Sections 2.2.1, 3.1 and 3.2, the walking principle of BALLU is fundamentally different from that of the conventional legged robots. BALLU walks differently from the majority of legged robots that push the ground to walk; instead, it places its footsteps by pulling the feet, and the body follows. While the light legs of the other robots support its body, BALLU's heavy legs pull down the balloons and the body so that they do not fly away.

In addition, as shown in Section 3.1, BALLU's severe underatuation and complex dynamics makes BALLU's walking very unique and challenging. The controller needs to compute control effort by taking advantage of the balloons' momentum and the legs' passive dynamics. To tackle this problem, a set of data-driven approaches are taken.

4.1 Controller Formulation

This section outlines a preliminary walking algorithm for BALLU. There are various atypical components in BALLU's dynamics, and as no other bipeds experience such a situation, it necessitates a different type of walking controller. In this work, we focus on planar walking

Table 4.1: Potential State Variables

Variable	Description
t_{phase}	the time duration of a phase, $phase \in \{DS, SS\}$
$t_{act,L}$, $t_{act,R}$	the duration of actuation (left, right)
$p_{c,x}, p_{c,z}$	the position of the center of mass of the pelvis
$v_{c,x}, v_{c,z}$	the velocity of center of mass of the pelvis
$p_{b,x}$	the position of balloon
$v_{b,x}$	the velocity of balloon
$\alpha_p, \beta_p, \gamma_p$	orientation of the pelvis about each axes
s	foot distance
$p_{FF,x}, p_{FF,z}$	coordinates of the position of the front foot
$p_{HF,x}, p_{HF,z}$	coordinates of the position of the hind foot
q_n, q_h, q_k, q_m	joint positions of neck, hip, knee, motor
$\omega_n, \omega_h, \omega_k, \omega_m$	joint velocities of neck, hip, knee, motor

in the sagittal plane. At a high level, the data-driven approach attempts to extract low-dimensional yet essential information that heavily affects a successful walking behavior, out of the numerous observable high-dimensional states. The summary of the proposed approach is demonstrated in Figure 4.1.

As discussed in the previous sections, it is advantageous to apply different strategies depending on whether the robot is in a single or a double support phase. Moreover, if the actuation profile is fixed, the phase time becomes the only parameter that determines the walking motion in each phase. This relationship is given by

$$\begin{aligned} x_{DS}^{out} &= f_{DS}(t_{DS}, x_{DS}^{in}) \\ x_{SS}^{out} &= f_{SS}(t_{SS}, x_{SS}^{in}) \end{aligned} \tag{4.1}$$

where f_{DS} and f_{SS} are transition functions for each phase, and the "state" $x \in X$, where X is the state space. For clarification, the term "state" is loosely used in this work to represent any potential variables pertaining to the robot during its locomotion. To constrain the actuation profile, it is assumed that the motors accept only the binary input and instantly move to each extreme position.

Rather than directly finding each transition function Equation (4.1), the proposed approach looks for an inverse relationship between x^{out} and t_*, g, and its approximation $\hat{g}$ from data as follows:

$$\begin{aligned} t_{DS} &\approx \hat{g}_{DS}(x^{out}, x^{in}) \\ t_{SS} &\approx \hat{g}_{SS}(x^{out}, x^{in}) \end{aligned} \tag{4.2}$$

4.2 State Definition

To determine which variables should be considered in the state vector, a statistical investigation was first conducted on the extensive potential relationships between the state variables.

4.2.1 Data Collection

The analysis in section 3.2 suggests the possibility that the BALLU's walking dynamics can be written in terms of kinematic quantities and relationships between them, and the potential state variables are listed based on this assumption. Table 4.1 is the full list of the potential states investigated.

All data for analysis were collected in simulation. Not only was nominal walking data from teleoperation collected, but walking sequences that include intentionally elongated and shortened double and single support phases as well as following recovery from such abnormal timings were also recorded.

From multiple simulation trials running at 100 Hz, 149291 raw data samples were collected, and 1585 phase changes (740 phase changes into the double support phase and 745 phase changes into the single support phase) were obtained. Each data point contains the variables listed in Table 4.1.

4.2.2 Data Correlation Investigation

Spearman Correlation Coefficient (SCC) [51] was used to account for the nonlinearities in the variables' relationships as well as the indicator's simplicity.

For each variable listed in Table 4.1, not only were the correlations between the different types of temporally adjacent phases (e.g. single support and double support) considered, but also between the same type of phases (e.g. double phase to double phase) to capture the relationship over the longer time horizon. Note that the state can drastically change within each phase, which could result in two very different states upon entering and exiting a phase. Consequently, the states' correlations are evaluated at both the beginning and end of a phase.

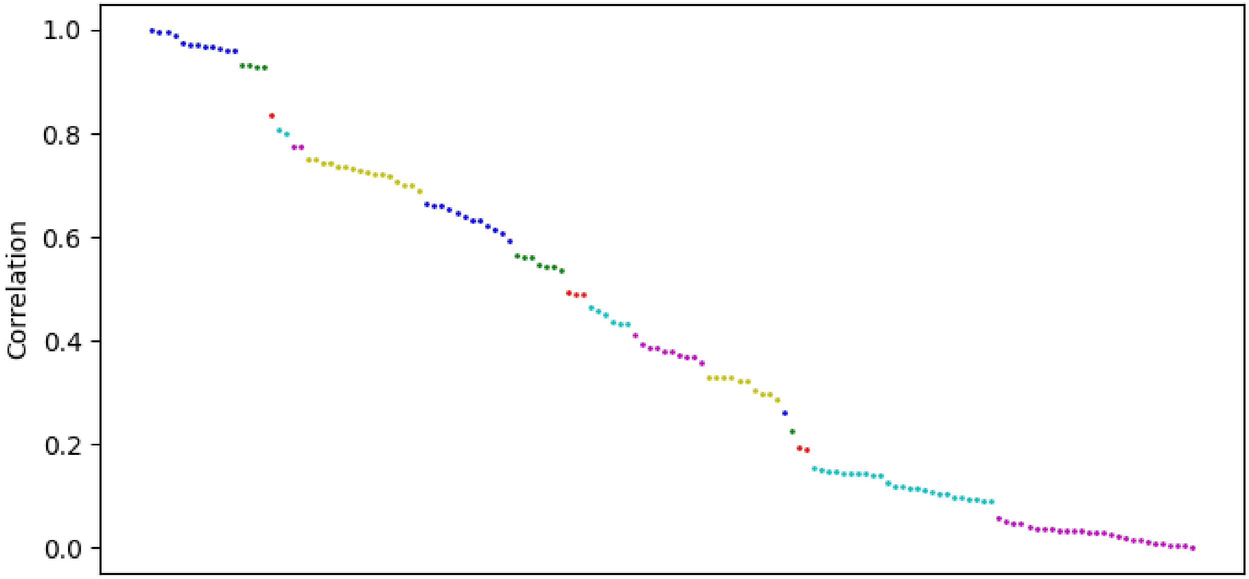


Figure 4.2: A result of scoring the correlation between the phase duration and kinematic variables. The variables are sorted in descending order, and the name of the variables are hidden for brevity.

4.2.3 Feature Selection

Now, the state vector is defined out of these collected data. Our approach is similar to those in the filter methods Section 1.3.2. As Equation (4.2) shows, the target is the phase duration of each walking phase. First, calculate the correlation between each kinematic variable and the phase duration to select the state variables that are highly correlated with the target. In this step, Spearman Correlation Coefficient [51], which is described in Section 1.3.1, is used as our measure, though any correlation measure can be taken in this step. This results in several groups of variables as shown in Figure 4.2.

Then, among those highly scored kinematic variables in the first step, we filter out the redundant variables that are highly correlated with another kinematic variable. To get rid of multicollinearity, we score the correlation again, but in this step, only between the kinematic variables chosen in the first step. Then, select those who have a low correlation with the others.

These steps can be automated by clustering algorithms. In our case, Density-based spatial clustering of applications with noise (DBSCAN) [22] is adopted because it does not require the number of clusters in advance, and the algorithm can be tuned with only a couple of parameters. Different colors in the Figure 4.2 represent each group clustered by this algorithm. For given clusters, by expanding the candidates from the most highly scored cluster, we can investigate all possibilities made in the first and the second scoring step.

Finally, the combination that makes the largest correlation with the target and the least correlation between the variables is taken as the final state vector. As a result, the state vectors x_{SS} and x_{DS} are $x_{SS} := [t^-_{SS}, v_{c,x}, v^-_{c,x}, v_{c,z}, p_{c,z}, p_{FF,x}]^T$ and $x_{DS} := [t^-_{DS}, v_{c,x}, v_{c,z}, p_{c,z}, p_{FF,x}]^T$, where the '-' superscript stands for the value of the previous phase. Considering the analysis in Section 3.2.3, the result is an acceptable choice and consistent with the author's experience.

4.3 Controller Training

A neural network is trained as a function approximator for each g. Both networks individually consist of a multilayer perceptron with ELU nonlinear activation functions [15] and ADAM optimizer [34] with MSE (Minimum Square Error) loss and a constant learning rate.

The hyperparameters are chosen via grid search to have the least test loss. Each network takes 3 hidden layers with 8 hidden units and a learning rate of 0.002. For more details, refer to Table 4.2. The network is trained for 100 epochs with early stopping. For each set of hyperparameters, they are mostly stopped in less than 50 epochs. The dataset was randomly split so that 70%of data is used for training, 15% for validation, 15% for testing,

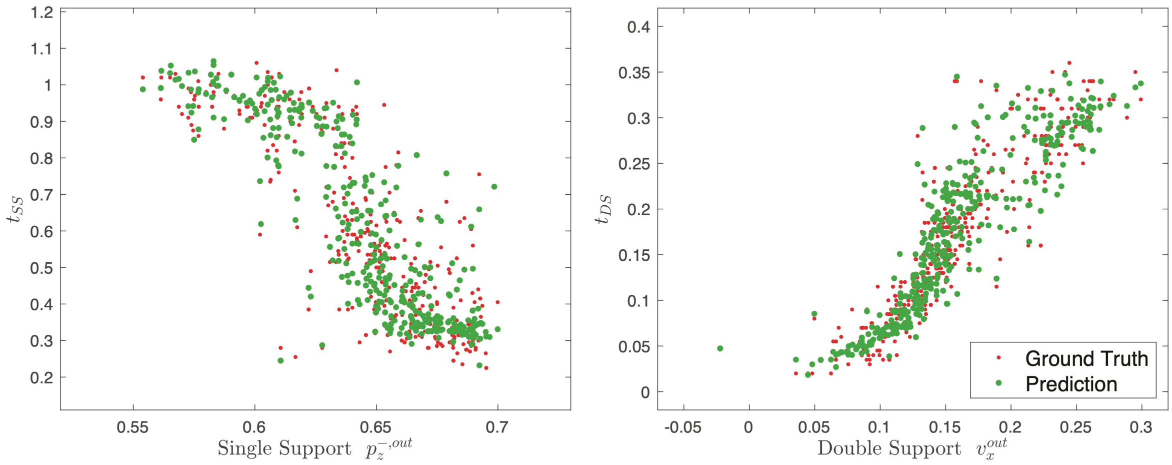


Figure 4.3: Examples of the training result. The left figure shows the trained relationship between the length of the phase time t_{SS} and the x coordinate of the front foot at the beginning of the phase $p_z^{-,out}$ in the case of single support phase. The right shows between the phase time t_{DS} and the velocity of CoM in Z-direction when it leaves the phase v_x^{out} in the case of double support.

and the training and validation set is shuffled at every epoch.

Table 4.2: Hyperparameters

Parameter	Value	Candidates	Description
h_l	8	$\{1, 2, 3, 4\}$	the number of hidden layers in each network
h_u	3	$\{2^0, 2^1, \cdots, 2^7\}$	the number of hidden units in each layer
α	2e−3	$\{1\text{e}{-4}, 2\text{e}{-4}, \cdots, 1\text{e}{-2}\}$	the learning rate of optimizer

The trained models with the hyperparameters above showed 0.7222 and 0.2177 test errors for the single support model and the double support model. The training result was also qualitatively evaluated, by looking at how well the true data points are covered by the predictions for each variable. For example, Figure 4.3 shows the relationship between the single support time and the height of the pelvis in the previous single support phase ($p_z^{-,out}$), and the double support time and the desired pelvis velocity in X-direction at the end of the phase (v_x^{out}). The prediction by the trained model is widely covering most of the given data.

4.4 Controller Composition

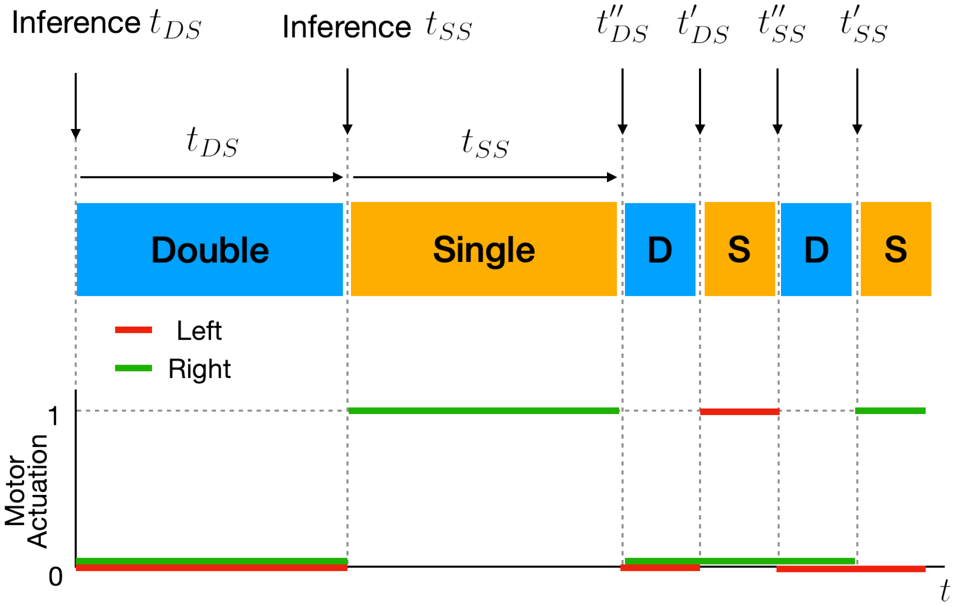


Figure 4.4: Controller Composition

Figure 4.4 illustrates how the final controller is composed using the trained controller for each walking phase. It is assumed that the double support phase and the single support phase alternate. At the beginning of each phase, the neural network corresponding to the current walking phase inferences the phase duration at this step. During the runtime, the x^{in} in Equation (4.2) works as a current state when the controller enters into the phase and takes the values from the state estimator(Section 2.3.4). Similarly, the x^{out} serves as a desired state when the controller leaves the phase. At each single support phase, the left and right legs are actuated alternatively.

4.5 Experiment

4.5.1 Simulation

To evaluate how well the proposed approach identifies from data the core states responsible for BALLU's walking, a set of simulations is conducted not only for walking at a nominal velocity, 0.18 m/s, but also in other varying conditions. By looking at how the controller deals with these variations, not only the robustness of the controller but also whether the statistical approach described in Section 4.1 has well-extracted variables that are closely related to walking are accessed.

The first set of simulations makes changes in mass properties. With heavier feet or pelvis, the body would be easier to sink down. First, the pelvis mass is increased by 6.4% (from 31.2 g to 33.2 g), and secondly, the feet masses are increased by 16.5% (from 24.2 g to 28.2 g), respectively. The second set of modifications is the change in commanded velocity, and two simulations are conducted at a slower velocity and a higher velocity.

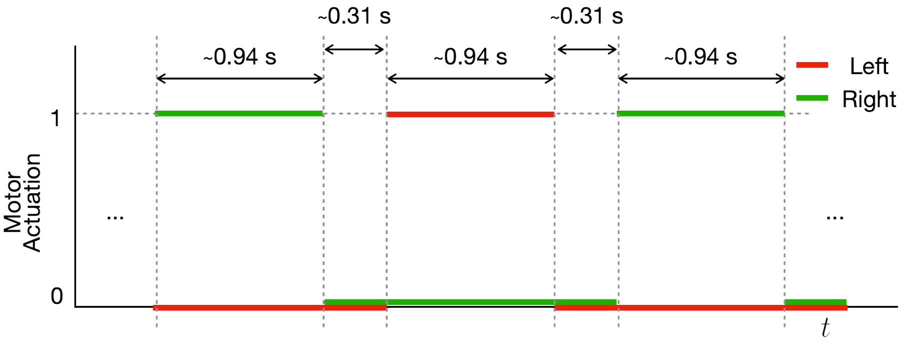


Figure 4.5: [
The regular actuation pattern of the data-driven controller in a steady-state]The regular actuation pattern of the data-driven controller (Figure 4.6) in a steady state (after 12 s)

Furthermore, in order to evaluate the effectiveness of the proposed controller, it is compared to a simple pattern generator. The motivation behind this experiment is based on the observation that the stabilized walking in Figure 4.6 shows a periodic motion with a certain frequency. In fact, after about 12 s, a regular alternating pattern of 0.94 s actuation and 0.31 s resting was found as shown in Figure 4.5, and a simple controller that generates the same pattern was created. First, the pattern generator is applied to a simulation model constrained to the sagittal plane so that the robot makes a planar walking. Then, the comparison is also made in 3D walking.

4.5.2 Hardware Verification

For verification of the proposed locomotion approach, straight walking of about 1.4 m is tested with a desired walking speed of 0.18 m/s on the actual hardware.

This experiment is conducted with BALLU2 with an external state estimator. The inputs to the neural networks are obtained via color tracking using an off-the-shelf RGBD camera. Using OpenCV and the Intel RealSense D435i, three differently colored LEDs attached to each foot and pelvis are detected. Cartesian coordinates of the colored positions relative to the camera are obtained from RealSense's internal algorithm, which does include significant noise [3]. To mitigate this issue, the coordinates are filtered using a Kalman filter using a constant acceleration model. The data capture and filtering are run at 60 Hz and 100 Hz, respectively, and the positions and velocities obtained from the filter are fed into the trained controller.

4.6 Results and Discussion

4.6.1 Simulation in the nominal condition

Because of the complexity and uncertainty of the system, rather than conducting formal nonlinear system analysis, BALLU's walking performance is first qualitatively assessed by analyzing the pelvis' trajectories and phase plots.

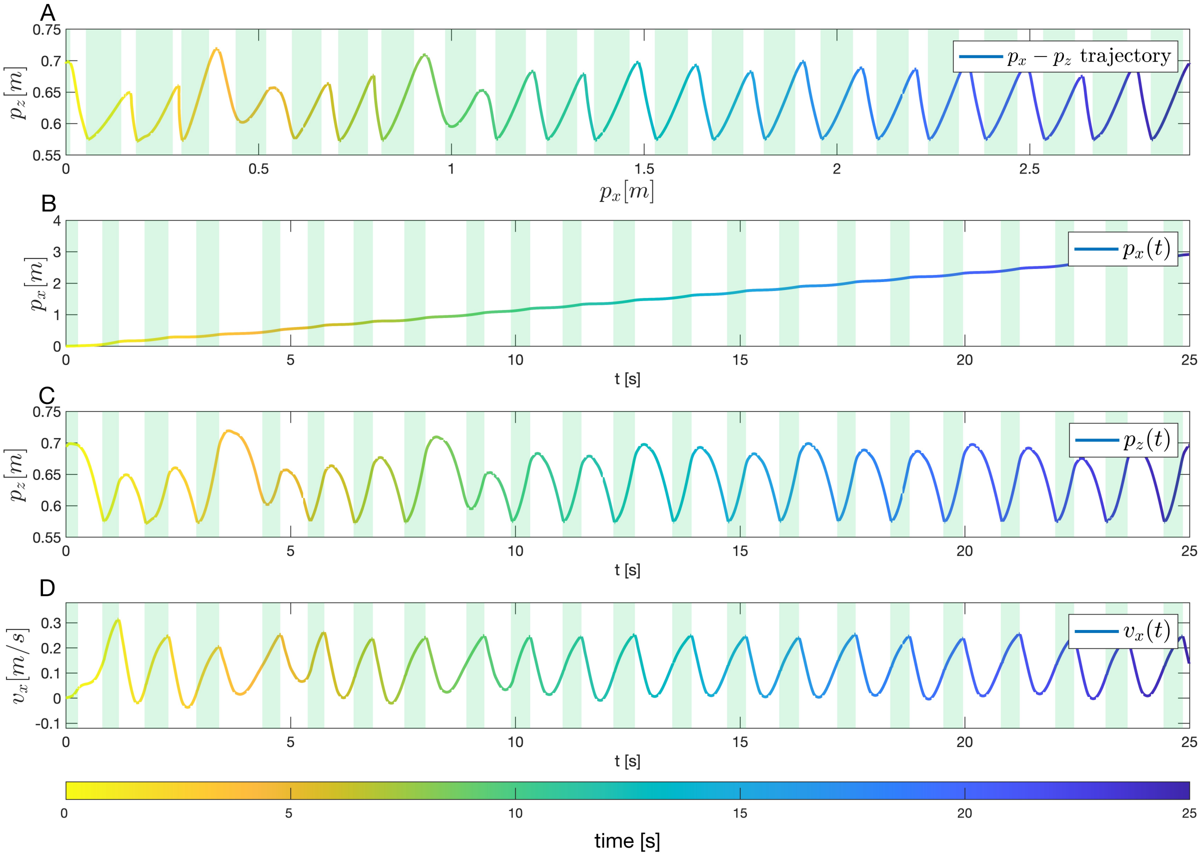


Figure 4.6: The simulation result in the nominal condition. (A) shows the trajectory of the CoM in the sagittal plane. Plot (B), (C), and (D) show the p_x, p_z, v_x of the CoM over time.

As a baseline, the result in the nominal walking condition is shown in Figure 4.6. The controller is generating a stable walking sequence. Although the system shows a transient response until around 12 s, a stable and periodic pattern appears since 12 s. As analyzed in Section 3.2, when it takes single support phases the body height falls and the forward velocity increases, and when the controller takes double support phases (shaded area), the

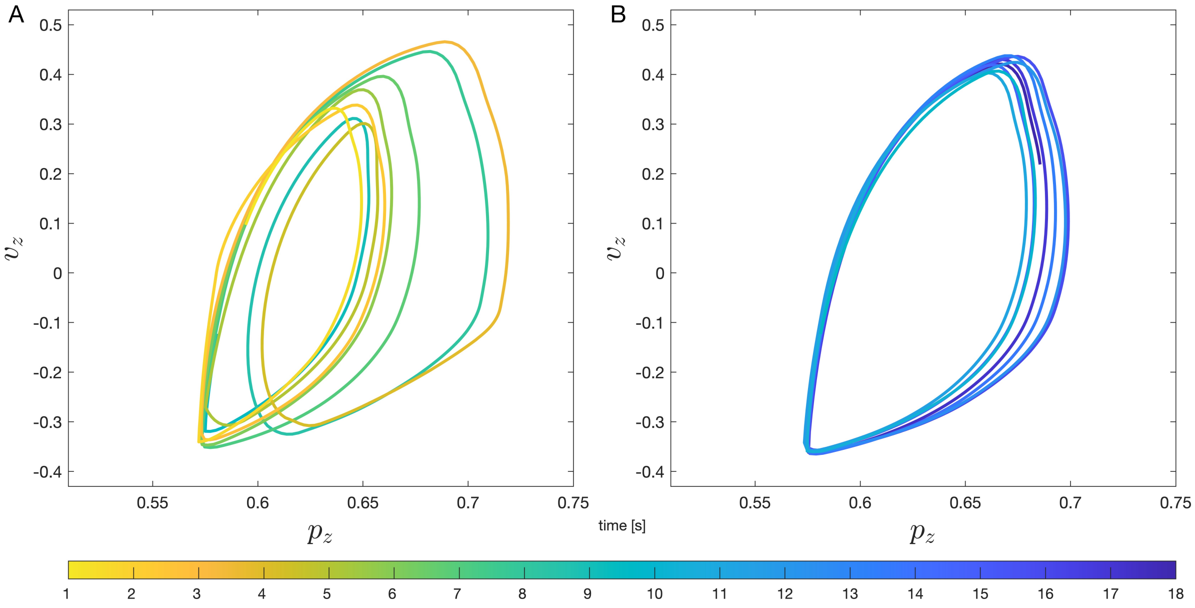


Figure 4.7: Phase plot of the pelvis' height Z. (A) in an earlier phase ($0\,\text{s} \leq t \leq 10\,\text{s}$), (B) in a latter phase ($12\,\text{s} \leq t \leq 25\,\text{s}$)

body height rises.

The first noticeable behavior that is important for successful walking is the ability to regulate the body's height within an interval that BALLU can successfully conduct subsequent motions. In the simulation, we can see that the body height is maintained between $0.58 \sim 0.70$. In the case that such an interval is not preserved, BALLU will exhibit the aforementioned sink-down behavior, leading to an inability to continue walking. Aside from the Z height oscillating within an interval, we can also notice that BALLU does indeed stride forward in the X-direction and its velocity trajectory shows a gradual increase from rest.

Another interesting point is that the slopes are not symmetric when the pelvis moves up and down which implies BALLU's unique walking dynamics. It comes from the fact that the velocity in X-direction periodically goes up and down. This behavior becomes more obvious later in the hardware test, and the velocity even goes down to the negative. The difference is due to the reality gap including the calibration of the knee joint's initial position, errors in the mass distribution, and the approximation error of drag force. The corresponding phase plot is shown in Figure 4.7. Similarly, as BALLU starts from rest, we can observe that the general circular shape of the limit cycle starts small in the earlier state (Figure 4.7A) but gradually expands until it converges in the latter stage (Figure 4.7B). This behavior is in parallel with that seen in previous works [4].

Nonetheless, small fluctuations can be observed in the limit cycle. There can be a couple of explanations for such inconsistency. The first cause is the distribution of the training data. The neural network is trained considering that the expert data is optimal. However, the expert data does not form a perfect limit cycle but, in fact, rather makes a qualitative periodic trajectory. Therefore, it can be expected for the neural net to generate a periodic motion overall as the expert data does but not to make a perfectly overlapped limit cycle.

Another possible explanation can be the neural network's approximation error. In Figure 4.3, the ground truth and the predicted values show very close distribution, but there are slight errors between the apparent corresponding pairs. While the neural network outputs the required phase times quite accurately but with a small prediction error, which could contribute to the limit cycle so much out of phase.

Conversely, these two error sources prove the proposed controller's robustness: the errors do not accumulate, but the controller corrects them and pushes the trajectory back to the limit cycle.

4.6.2 Simulation with variations

When unexpected changes are given to the normal condition, it was able to observe the controller trying to overcome in the same way that the experts teleoperated, which is analyzed in Section 3.2.

4.6.2.1 Increased Mass

Similar patterns are observed when the pelvis mass increased (Figure 4.8C, Figure 4.8D) and when the feet masses increased (Figure 4.8E, Figure 4.8F). In both cases, the average height is decreased, and the controller takes shorter single support phases and longer double

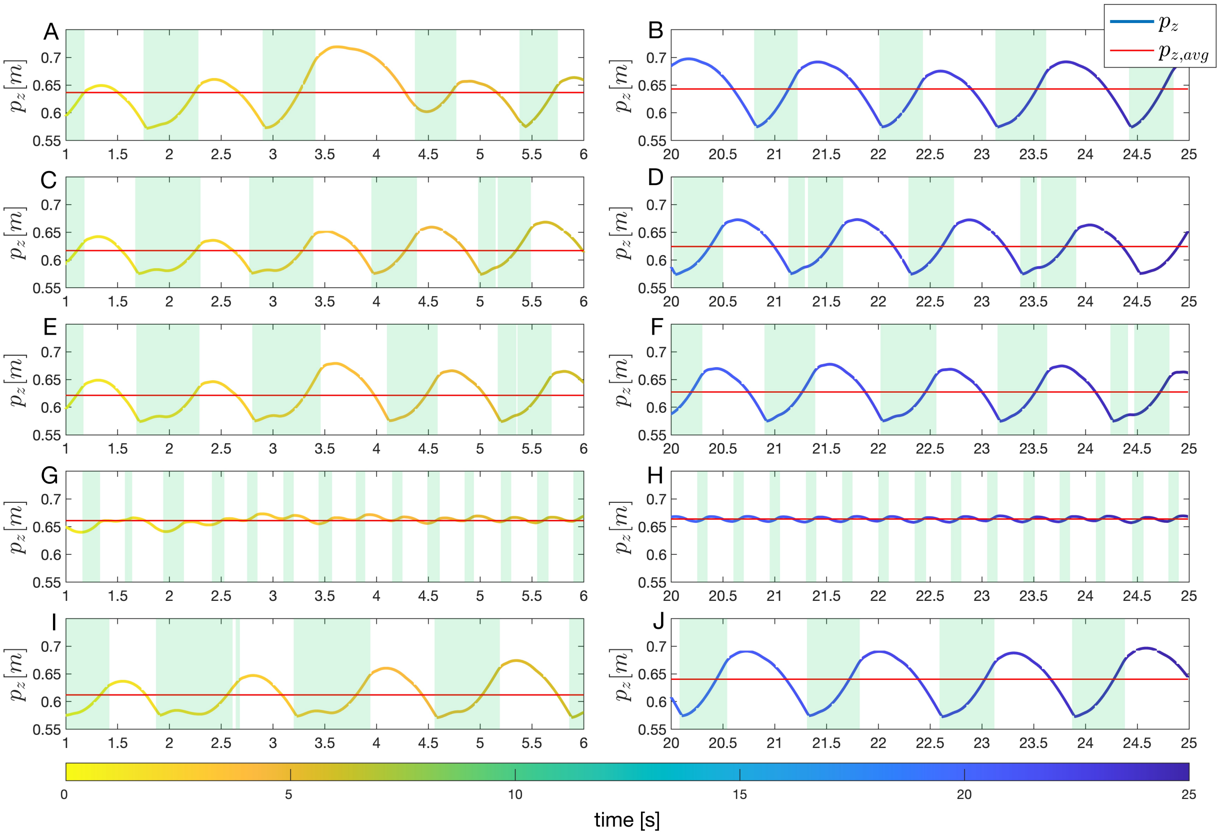


Figure 4.8: The height of the pelvis, p_z, over time when changes are applied to the nominal condition. (A, B) nominal condition, (C, D) increased pelvis mass (+6.4 %), (E, F) increased feet mass (+165 %, respectively), (G, H) decreased command speed (0.1 m/s), and (I, J): increased command speed (0.32 m/s). The red line is the average height in the given scope. The left column is when in an earlier phase ($1\,\mathrm{s} \leq t \leq 6\,\mathrm{s}$), the right column is when in a latter phase ($20\,\mathrm{s} \leq t \leq 25\,\mathrm{s}$)

support phases than them in the nominal condition. It can be interpreted that the height of the body falls easier in single support phases because of the increased mass, and the controller tries to regulate the body height not to sink down with a longer double support phase (Section 3.2.4.1) and minimal single support phase to track the commanded velocity. Since it is difficult to gain speed with the shortened single support phase, they were worse at tracking the desired speed.

In addition, the above results imply that concentrating weight on the feet is more advantageous than putting weight on the pelvis. While those two results show similar pelvis trajectories, the increment on feet is four times larger than that on the pelvis, and the controller fails if the weights are increased further in both cases. This proves once again our design approach to allocate most of the parts on the feet (Section 2.2).

4.6.2.2 Extreme Command Velocities

As a second modification, two unachievable velocities were commanded. When the commanded velocity is too low (Figure 4.8G, Figure 4.8H), it can be observed that the controller takes tiny steps. As a result, the two feet become closer and the lines connecting each foot and the pelvis gets towards the vertical so to minimize the forward force. It can be interpreted that the controller tries to take the minimum length of single support phases not to increase speed (Section 3.2.4.2). As the controller drives the body rather upward, the average height is higher than the nominal condition. Contrarily, the controller takes big steps when the commanded velocity is too high (Figure 4.8I, Figure 4.8J). It is to take single support phases to catch up with the high commanded velocity. Since the controller takes single support phases as much as it can, the average body height is lower than that under the nominal condition.

4.6.3 Comparison to Periodic Pattern Generator

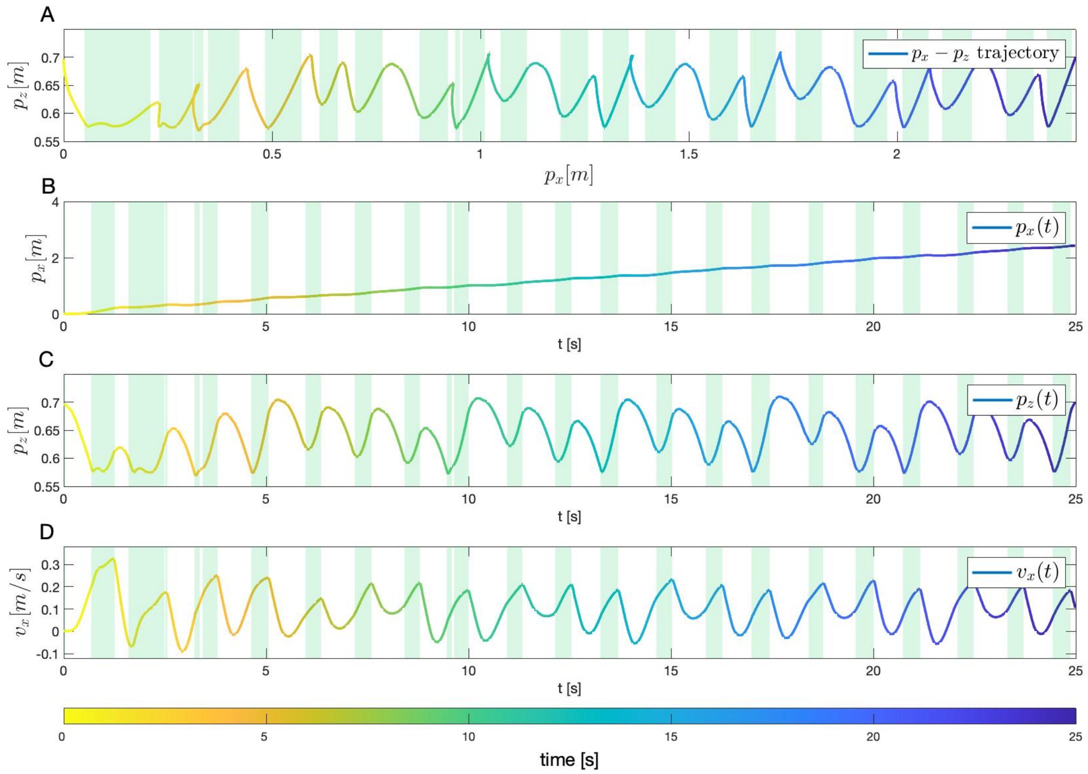


Figure 4.9: A 2D walking trajectory with the pattern generator. The body is constrained in the XZ plane.

The pattern generator is applied to a simulation model constrained to the XZ plane. Several interesting observations can be made from the results shown in Figure 4.9. Even with the simple pattern generator, BALLU was able to produce a cyclic motion.

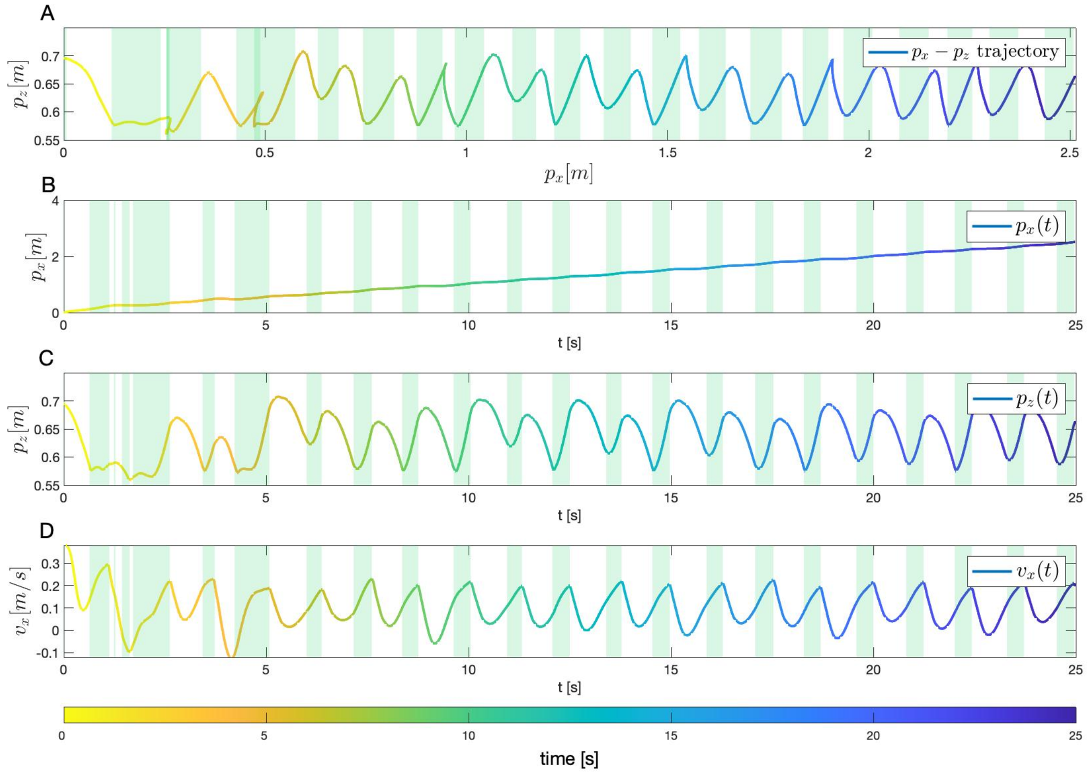


Figure 4.10: A 2D walking trajectory with the pattern generator, starting from a non-zero initial state ($v_{\text{balloon},x} = v_{c,x} = 0.5m/s$). The body is constrained in the XZ plane.

Surprisingly, when the model is disturbed, the simple pattern generator stabilizes the system. This suggests that BALLU is an inherently stable robot and that there exists a *natural frequency* within the BALLU system.

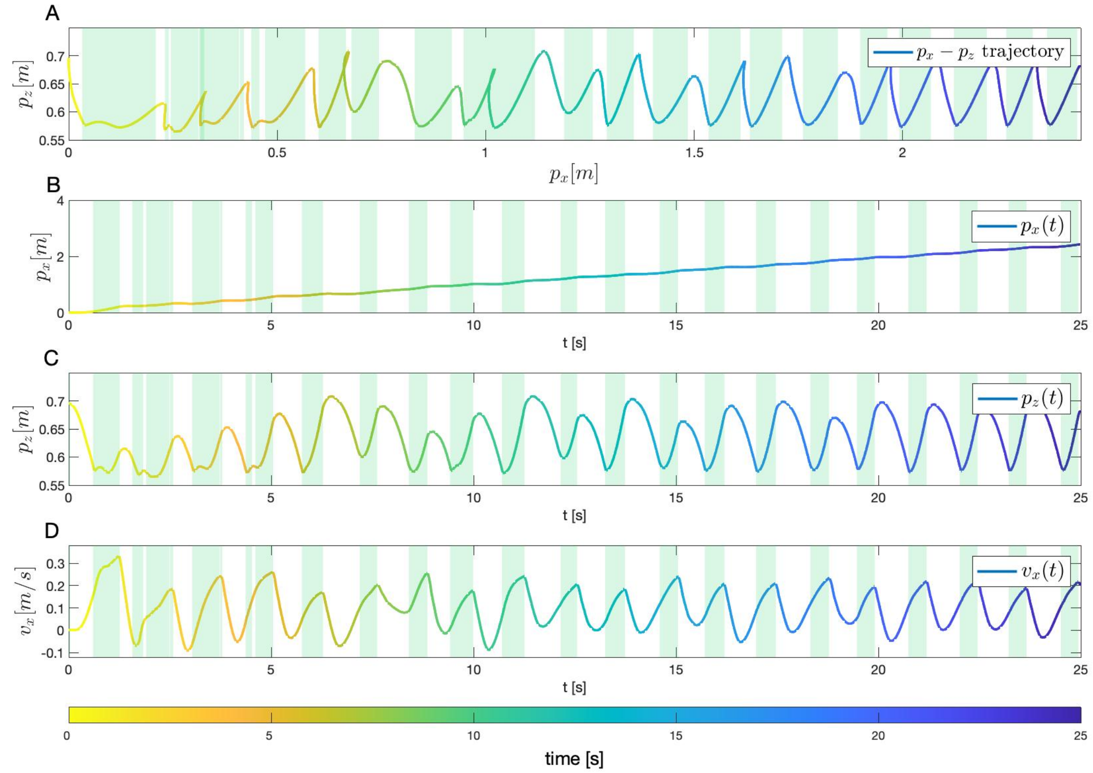


Figure 4.11: A 2D walking trajectory with the pattern generator with the unseen model condition (increased feet mass). The body is constrained in the XZ plane.

However, unlike the results from the data-driven controller, the pattern is repetitive yet irregular. Interestingly, the two legs exhibit different behavior when each of them is in the same walking phase. Clearly, some components of the error are not eliminated with the simple generator and persist in the system over time, as shown in Figure 4.5. This demonstrates the need for a controller that can stabilize the model under a broader range of disturbances.

To emphasize the differences more effectively, the results of 3D walking are investigated. As shown in Figure 4.12, while the data-driven controller and the pattern generator appeared to have similar performance in 2D walking, the pattern generator failed to stabilize the system in 3D. Figure 4.13 represents the top view of the robot's trajectory with the pattern generator.

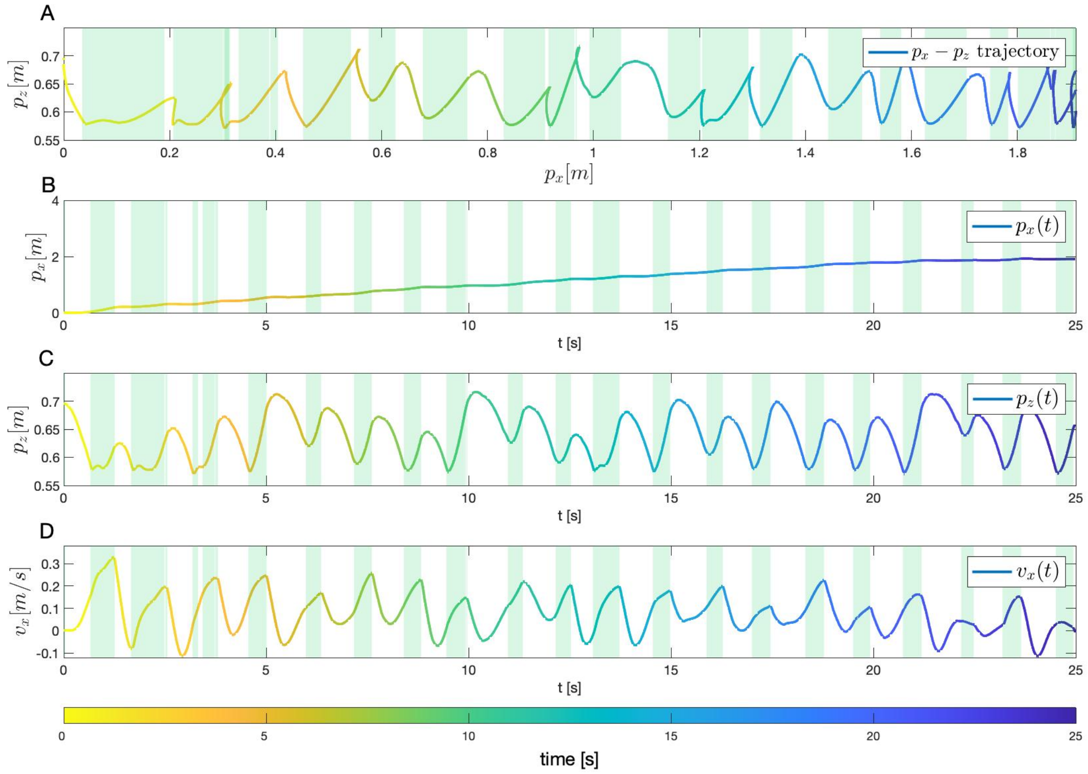


Figure 4.12: A 3D walking trajectory with the pattern generator.

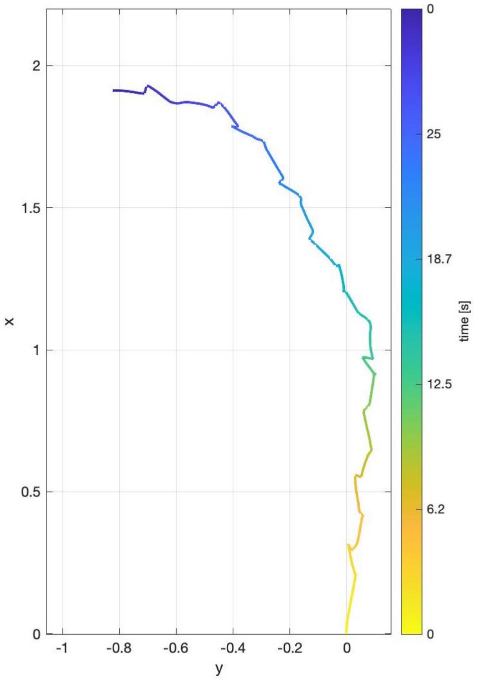


Figure 4.13: A 3D walking trajectory in XY-plane with the pattern generator.

When the 2D constraint is removed, the model with the pattern generator cannot walk straight. This clearly highlights the difference between the pattern generator and the data-driven controller, as the data-driven controller can regulate motion in the Y-direction. Interestingly, the data-driven controller is also trained solely on the expert's 2D walking data. Nevertheless, the data-driven controller effectively controls the legs' swing and prevents the robot from rotating about the Z-axis or deviating from the straight line. This explains why the data-driven controller worked in 3D space and why the 2D data-driven controller could operate on real world hardware, overcoming the reality gap.

4.6.4 Hardware Test

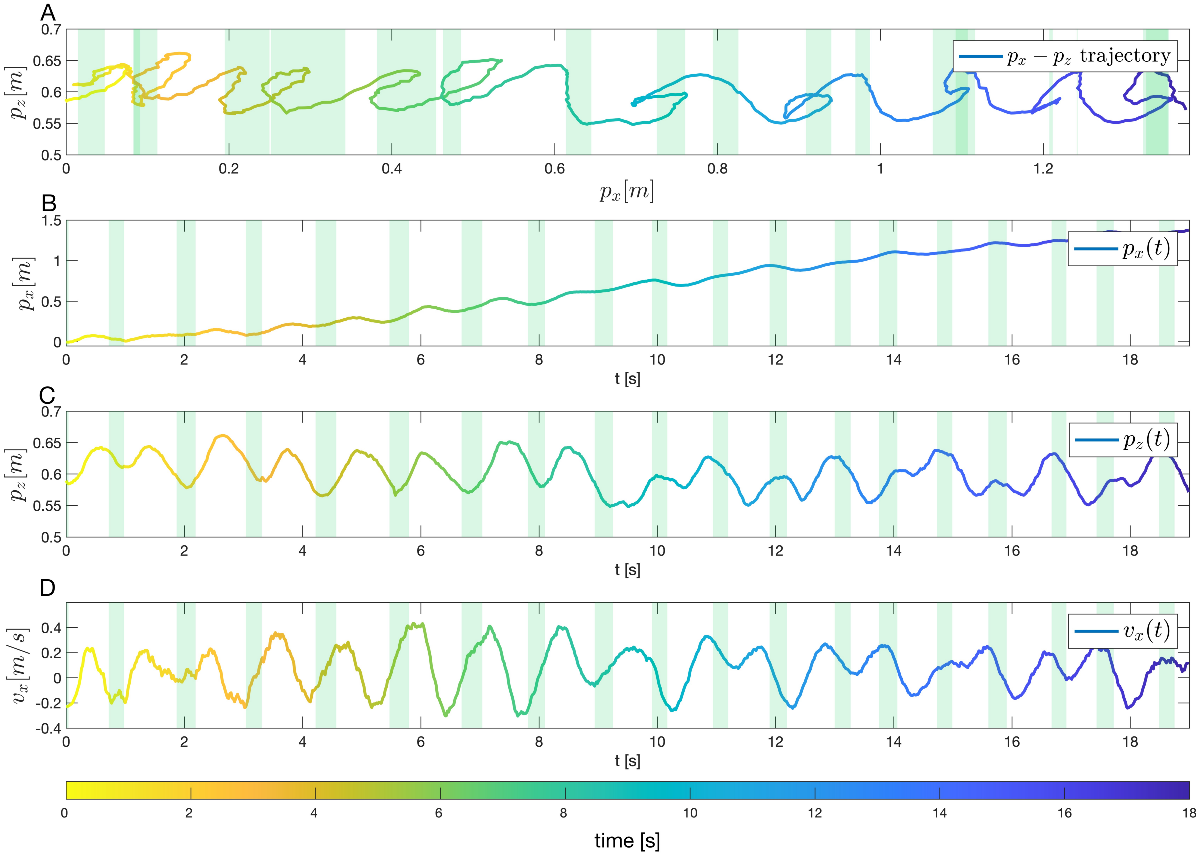


Figure 4.14: The result on the hardware. (A) shows the trajectory of the CoM in the sagittal plane. Plot (B), (C), and (D) show the p_x, p_z, and v_x of the CoM over time.

The trajectories of the pelvis are presented in Figure 4.14, and the corresponding phase

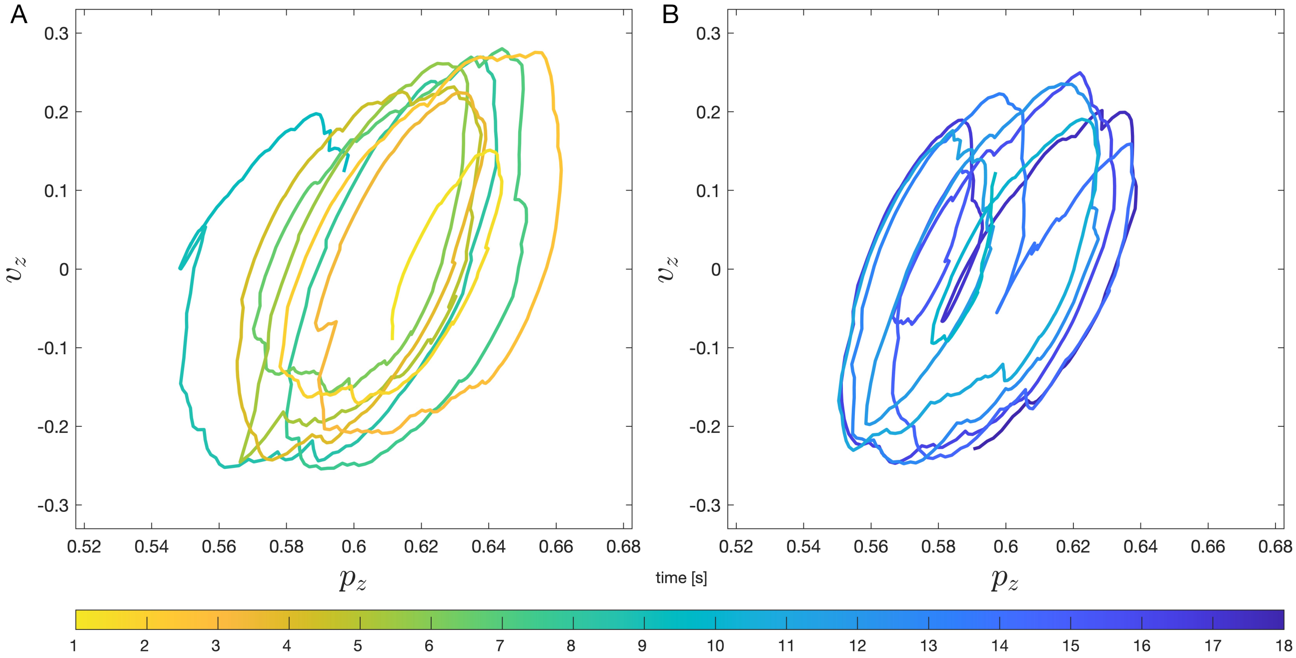


Figure 4.15: Phase plot of the pelvis' height Z. (A) in an earlier phase ($1\,\mathrm{s} \leq t \leq 10\,\mathrm{s}$), (B) in a latter phase ($10\,\mathrm{s} \leq t \leq 19\,\mathrm{s}$)

plots are given in Figure 4.15. Although the response is less uniform and much noisier, the body gradually walks forward in X-direction and shows relatively more periodic behavior after 12 *sec.* Considering the significant noise that exists from state estimation, the body height oscillates between $0.54 \sim 0.66$, with an average of approximately 0.6. While the mean may be different, this aligns well with the collected data from the simulation. Specifically, as shown in Figure 4.14A and Figure 4.14B, when the height drops (for example, at p_x =0.19 m (t =4.2 s), p_x =0.38 m (t = 6.65 s), and p_x =0.61 m (t =8.95 s)) the controller takes longer double support phases to recover the pelvis height. The function approximator, despite being trained based on simulation data, worked well on the physical platform without any additional tuning despite the unavoidable model differences. This suggests that BALLU is capable of walking using the proposed data-driven approach.

In addition, what is more amusing from Figure 4.14 is BALLU's negative velocity in the X-direction. This behavior is uncommon for bipeds walking forward as the two legs are able to crossover, unlike walking sideways where oscillation is common because the legs cannot crossover. This is a distinct feature of BALLU as walking forward is only achievable by a combination of the support leg's spring injecting energy into the system and the body and the swing leg's momentum. Hence, at intervals where such a force and momentum are not sufficient, which includes the period after the swing leg moves past the pelvis (moments after the double support phase in Figure 4.14), the body temporarily gets pushed backward because of the swing leg moving forward. This behavior is an artifact of the system's passive dynamics. This reinforces the belief that conventional locomotion controllers may not be suitable for such a system and possibly why the proposed data-driven approach is the first successful non-teleoperated walking for BALLU.

4.6.5 Collision

Touching on safety, it is also evident that BALLU can only produce so much force and momentum in any given direction. In its walking direction, the maximum acceleration and velocity are approximately 0.3 m/s^2 and 0.4 m/s, where the mass of the entire robot is only about 170 g. Even then, because of the balloon body's radius, the first point of collision in a human environment will likely be the balloons and not the legs. This shows that even if a system like BALLU were to malfunction and collide, it will cause no harm to its surrounding environment or humans.

CHAPTER 5

Conclusion and Future Works

5.1 Conclusion

Although significant advancements have been made in the field of legged robots and their potential has expanded, there remain challenges to address. To be used in proximity to humans, safety has to be guaranteed, and cost concerns must also be addressed. Balloon-based robots may offer an innovative and practical solution to these concerns, and in such cases, data-driven control based on statistics and machine learning can offer a novel approach to robot control.

This dissertation thoroughly discusses the characteristics, necessities, challenges, implementation, and control techniques of BALLU, Buoyancy Assisted Lightweight Legged Unit. Chapter 1 highlights the areas where robots need to improve to be used in human daily life and proposes the use of helium balloons and buoyancy as a creative alternative. By adopting helium balloons and simplifying bipedal locomotion, BALLU is designed to be affordable and lightweight. These characteristics allow BALLU to be deployed in the real world and provide service in close proximity to humans. Chapter 2 discusses the design considerations for implementing the BALLU robot, and delves into the hardware and software implementation. The aerodynamics and severe underatuation introduce another challenge. This new approach not only offers solutions to existing problems but also raises new issues. In Chapter 3, the BALLU platform's unique challenges are discussed, and the robot's walking principles are analyzed in depth. These distinctive challenges call for a nonconventional walking controller: Chapter 4 introduces the walking algorithm developed based on prior analysis. Statistical methods are employed to define a compact yet essential state vector, followed by a description of a deep learning-based controller trained based on this state definition. The controller's performance and robustness are evaluated through extensive simulations and hardware experiments.

BALLU represents an innovative approach to assistive robotics, offering an extremely high level of safety, cost-effectiveness, and various scaleability such as a disposable robot. Both the platform and the control method developed for walking hold the potential to provide alternative solutions for motion planning problems, complementing the advancements in conventional legged robots. The subsequent section introduces further research on BALLU.

5.2 Future Works

In light of the advances made in this work, future research can be conducted to enhance buoyancy assisted robots' capabilities in many areas:

Extension of the work The proposed process of defining the state from data and using it to train the controller can be further expanded. For instance, although this study specifically employed the Spearman correlation coefficient, alternative feature selection methods introduced in the Section 1.3.2 or other measures addressing a wider range of nonlinear relationships could be adopted. Moreover, instead of using supervised learning for training, reinforcement learning agents that have the defined state definition as the observation space could be trained. Furthermore, a fully automated pipeline could be developed by combining the aforementioned approaches instead of gathering data from expert users. I believe that, even with the same scheme, a controller can be produced exhibiting significantly enhanced performance.

Various Locomotion Through teleoperation, BALLU has been verified to be capable of not only walking but also jumping, climbing up and down stairs, walking sideways, rotating, and performing various other motions. It requires a deeper understanding of the BALLU's dynamics, especially the motion of the balloon set, under different modes and situations. Developing controllers for these various motions is an immediate next research topic.

Navigation and High-level Planning As a primary application of BALLU, we have considered an indoor mobile kiosk. Once BALLU becomes capable of performing various types of locomotion, research on diverse high-level planning and decision-making will be required to enable it to interact with people, navigate through environments, and provide useful information.

Minimizing the Reality Gap Reducing the discrepancy between simulation models and real-world hardware is an important challenge in various fields such as robotics, machine learning, and reinforcement learning. There are various research efforts to reduce the difference between simulation and real hardware regarding physical properties such as friction, inertia, and damping. Under the reality gap, controllers developed solely through simulation are less likely to perform well on real hardware. In the case of BALLU, the use of lightweight materials and small, low-fidelity sensors are great sources of unmodeled uncertainties. To overcome this, collaborative research has been conducted to train residual dynamics between simulation and hardware using reinforcement learning [50] so that the agent trained with the improved simulation environment feels less difference. However, much more research is needed in the future to address this challenge.

Adaptive Controller The dynamics of BALLU change over time due to various factors such as the wear of 3D printed parts and the balloons which leak as they are repeatedly used. Estimating these changing dynamics and developing a controller that adapts to the changed system to compute optimal control commands is an interesting research topic.

Disposible Robotics and Multi-Agent Collaboration One of the unique advantages of BALLU is its affordability. This not only implies that the robot can become widely accessible like home appliances but also includes the possibility of considering the robot disposable so that can be sent to hazardous environments and discarded if needed without concern about the cost. In this case, BALLU made out of bio-biodegradable material becomes another interesting research topic. Furthermore, controlling a swarm of BALLUs, especially considering that BALLU has a similar size as an adult, is a promising research topic.

Enhancing Software Framework Based on its affordability and accessibility, we have plans to make BALLU an open-source project to contribute to the society and robotics field. I am planning to improve the software stack to make it easier for individuals and researchers from different backgrounds to make use of BALLU. Various possibilities are being considered, including supporting ROS [36]. By making BALLU open source, we hope to encourage more researchers and enthusiasts to explore its potential and contribute to the development of the project.

APPENDIX A

Mathematical Models

A.1 Four-bar Linkage Analysis

BALLU's kinematics has a two-level structure. The lower level kinematics is of the mechanism that relates an actuation of an actuator at its ankle to the motion of a knee. The higher level is the kinematics of the whole structure incorporating knee joints and hip joints.

As shown in Figure A.1, BALLU's single leg forms a typical four-bar linkage mechanism between the arm of the servo motor at the ankle, the string attached to the motor, the lower part of the femur, and the tibia. Given the angular position, velocity, and acceleration of the actuator, the resulting angular position, velocity, and acceleration of the knee joint can be determined from the four-bar linkage analysis.

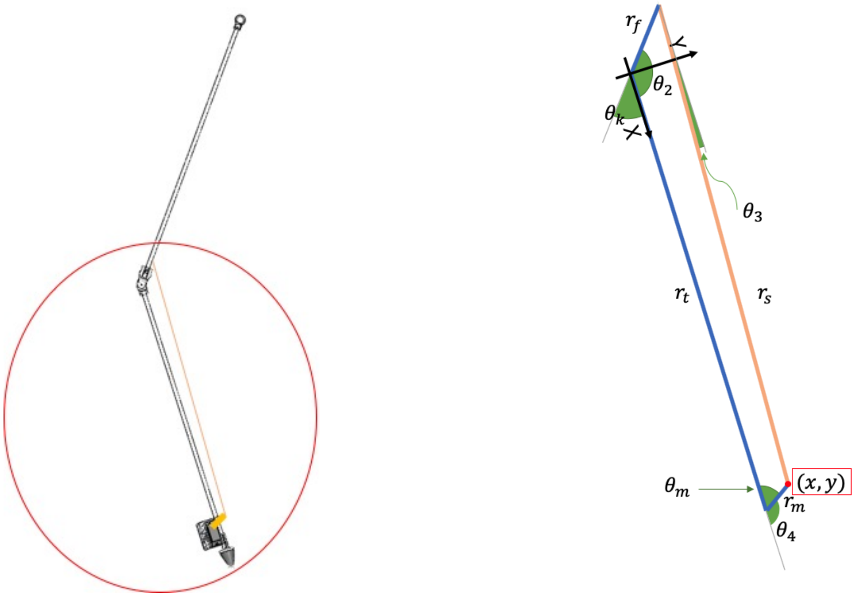


Figure A.1: Four-bar linkage mechanism in a leg of BALLU

A.1.1 Position Analysis

$$\vec{r_t} + \vec{r_m} = \vec{r_f} + \vec{r_s} \tag{A.1}$$

where

$$\vec{r_t} = r_t \qquad \vec{r_m} = r_m e^{j\theta_4} \qquad \vec{r_f} = r_f e^{j\theta_2} \qquad \vec{r_s} = r_s e^{j\theta_3}$$

$$\theta_k = \pi - \theta_2 \qquad \theta_m = \pi - \theta_4$$

$$\cos\theta_2 = \left(x - r_s\cos\theta_3\right)/r_f$$

$$\sin\theta_2 = \left(y - r_s\sin\theta_3\right)/r_f$$

$$\theta_2 = \operatorname{atan2}\left(\frac{x - r_s\cos\theta_3}{r_f}, \frac{y - r_s\sin\theta_3}{r_f}\right) \tag{A.2}$$

from $\sin\theta^2 + \cos\theta^2 = 1$

$$\left(x - r_s\cos\theta_3\right)^2 + \left(y - r_s\sin\theta_3\right)^2 = r_f^2$$

$$\theta_3 = \operatorname{atan2}\left(y, x\right) + \pm\arccos\left(\frac{x^2 + y^2 + r_s^2 - r_f^2}{2r_s\sqrt{x^2 + y^2}}\right) \tag{A.3}$$

A.1.2 Velocity Analysis

$$\cancel{\dot{\vec{r_t}}} + \dot{\vec{r_m}} = \dot{\vec{r_f}} + \dot{\vec{r_s}}$$

$$j\dot{\theta}_4 r_m e^{j\theta_4} = j\dot{\theta}_2 r_f e^{j\theta_2} + j\dot{\theta}_3 r_s e^{j\theta_3}$$

$$\begin{aligned}\dot{\theta}_4 r_m e^{j(\theta_4-\theta_3)} &= \dot{\theta}_2 r_f e^{j(\theta_2-\theta_3)} + \dot{\theta}_3 r_s \\ \dot{\theta}_4 r_m \sin\left(\theta_4-\theta_3\right) &= \dot{\theta}_2 r_f \left(\theta_2-\theta_3\right)\end{aligned}$$

$$\dot{\theta}_2 = \dot{\theta}_4 \frac{r_m \sin\theta_4 - \theta_3}{r_f \sin\theta_2 - \theta_3} \tag{A.4}$$

A.1.3 Acceleration Analysis

$$\cancel{\ddot{\vec{r_t}}} + \ddot{\vec{r_m}} = \ddot{\vec{r_f}} + \ddot{\vec{r_s}}$$

$$\begin{aligned}\left(j\alpha_4 - \omega_4^2\right) r_4 e^{j\theta_4} &= \left(j\alpha_3 - \omega_3^2\right) r_3 e^{j\theta_3} + \left(j\alpha_2 - \omega_2^2\right) r_2 e^{j\theta_2} \\ \left(j\alpha_4 - \omega_4^2\right) r_4 e^{j\theta_4-\theta_2} &= \left(j\alpha_3 - \omega_3^2\right) r_3 + \left(j\alpha_2 - \omega_2^2\right) r_2 e^{j\theta_2-\theta_3}\end{aligned}$$

$$\ddot{\theta}_4 = \ddot{\theta}_2 \frac{r_2 \sin\left(\theta_2-\theta_3\right)}{r_4 \sin\left(\theta_4-\theta_3\right)} + \frac{\dot{\theta}_2^{\,2} r_2 \cos\left(\theta_2-\theta_3\right) + \dot{\theta}_3^{\,2} r_3 - \dot{\theta}_4^{\,2} r_4 \cos\left(\theta_4-\theta_3\right)}{r_4 \sin\left(\theta_4-\theta_3\right)} \tag{A.5}$$

A.2 Leg Kinematics

On the top of the abstraction discussed in Appendix A.1, a leg of BALLU can be seen as a double pendulum in the 2D plane. The forward kinematic from the hip joint to the foot is given by:

$$
\begin{aligned}
x_{knee} &= \quad l_{\text{femur}} \cos\left(-\frac{1}{2}\pi + q_h\right) \\
y_{knee} &= \quad \pm\frac{1}{2} l_{\text{pelvis}} \\
z_{knee} &= -l_{\text{femur}} \sin\left(-\frac{1}{2}\pi + q_h\right)
\end{aligned}
$$

$$
\begin{aligned}
x_{foot} &= \quad l_{\text{femur}} \cos\left(-\frac{1}{2}\pi + q_h\right) + l_{\text{tibia}} \cos\left(-\frac{1}{2}\pi + q_h + q_k\right) \\
y_{foot} &= \quad \pm\frac{1}{2} l_{\text{pelvis}} \\
z_{foot} &= -l_{\text{femur}} \sin\left(-\frac{1}{2}\pi + q_h\right) + l_{\text{tibia}} \sin\left(-\frac{1}{2}\pi + q_h + q_k\right)
\end{aligned}
$$

where q_h and q_k are the angles of the hip and the knee joints, and l_{femur}, l_{tibia} and l_{pelvis} are the lengths of the femur, tibia, and pelvis links, respectively. The sign of the y coordinates are positive for the left leg and negative for the right leg.

By taking derivative, Jacobian matrices are obtained:

$$
J_{knee}(q) = l_{\text{femur}} \begin{bmatrix} \sin(-\frac{1}{2}\pi + q_h) & 0 \\ 0 & 0 \\ \cos(-\frac{1}{2}\pi + q_h) & 0 \end{bmatrix}
$$

$$
J_{foot}(q) = J_{knee} + l_{\text{tibia}} \begin{bmatrix} \sin(-\frac{1}{2}\pi + q_h + q_k) & \sin(-\frac{1}{2}\pi + q_h + q_k) \\ 0 & 0 \\ \cos(-\frac{1}{2}\pi + q_h + q_k) & 0 \end{bmatrix}
$$

where $q = [q_h, q_k]^{\text{T}}$ and $\dot{q} = [\dot{q_h}, \dot{q_k}]^{\text{T}}$.

APPENDIX B

Full equation of foces acting on the pelvis

In Section 3.2.3, it is claimed that the force acting on the pelvis due to the torsional springs on the knee is only along the line connecting the foot and the hip joint. This principle was highly useful in analyzing the motion of BALLU in subsequent sections. This chapter describes the mathematical derivation behind this founding, which was simplified as Equation (2.5) earlier.

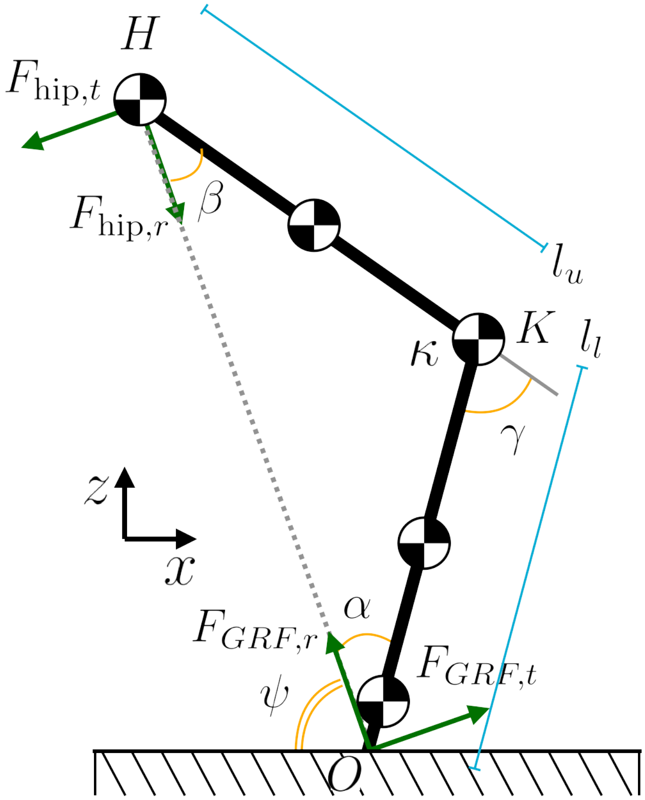


Figure B.1: Free body diagram of each part of a leg

This derivation extends Rao's formulation [45] in Chapter 3. The formulation assumed massless leg segments having equal lengths. However, in the following formulation, the femur (the upper leg segment) and the tibia (the lower leg segment) have different lengths, l_u and l_l respectively, and the leg segments, the hip and knee joints, and the foot are assumed to be a point mass. Although the list of mass points is introduced corresponding to BALLU's configuration, the number of the point masses does not degrade the generality of the formulation as far as point masses are located along the leg segments. This implies that the same result will be deduced with continuum bodies that have the second moments of inertia.

Figure B.1 and Table B.1 show the variables and the definition of the symbols used in this chapter. Here, the forces $F_{hip,r}$ and $F_{hip,t}$ are reaction forces of $F_{p,r}$ and F_p, t used in Equation (2.5), i.e. $F_{p,r} = -F_{hip,r}$ and $F_{p,t} = -F_{hip,t}$.

Table B.1: Variables used in the force analysis around the knee

	Symbol	Description
Mass	m_h	Hip joint parts and Pelvis link with electronics
	m_{fm}	Femur link
	m_{tb}	Tibia link
	m_{kn}	Upper knee joint parts
	m_{kn}	Lower knee joint parts
	m_{ft}	Foot part with electronics
	M	Sum of $m_{(\cdot)}$ above
Length	l_u	Femur
	l_l	Tibia
	l_{fm}	From Knee to Femur's center
	l_{tb}	From Knee to Tibia's center
	l_{ft}	From Knee to Foot's center
Force	$F_{hip,(\cdot)}$	Force acting on Hip joint
	$F_{GRF,(\cdot)}$	Ground reaction force acting on the contact point
Angle	ψ	Angle between Ground and $\overline{OH}$
	α	Angle between Tibia link ($\overline{OK}$) and $\overline{OH}$
	β	Angle between Femur link ($\overline{HK}$) and $\overline{OH}$
	γ	Knee joint i.e. q_k
	$\phi(q_k)$	Vector of angles α, β, γ
Point	H	Hip joint
	K	Knee joint
	O	Contacting point between Foot and Ground

B.1 Equations of Motion

Let's set up a set of equations of the moment balance about the contacting point O and force balance along the radial direction and the tangential direction of the line $\overline{OH}$ connecting the end of the foot and the hip joint.

First, the angles can be obtained from the following relationships. Equation (B.1) is according to the Law of Sines [17]:

$$\begin{gathered} \gamma = \alpha + \beta \\ \frac{l_l}{\sin\beta} = \frac{l_u}{\sin\alpha} \end{gathered} \tag{B.1}$$

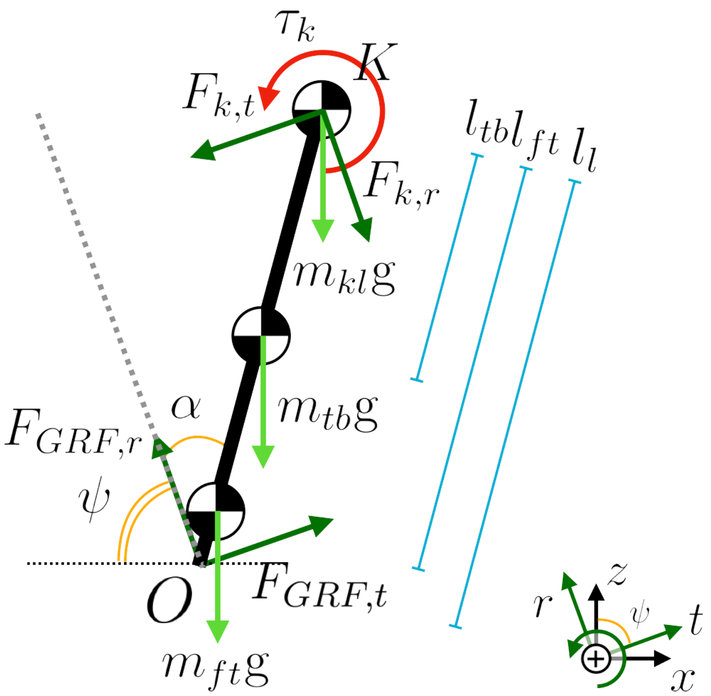


Figure B.2: Free body diagram of the lower leg (tibia) segment

Figure B.2 illustrates a free body diagram of the lower leg segment. For the member

$\overline{OK}$,

$$\sum F_t = 0 : F_{G,t} - F_{k,t} - (m_{kl} + m_{tb} + m_{ft}) g \cos\psi = 0 \tag{B.2}$$

$$\sum F_r = 0 : F_{G,r} - F_{k,r} - (m_{kl} + m_{tb} + m_{ft}) g \sin\psi = 0 \tag{B.3}$$

$$\sum_K M = 0 : T_{\text{knee}} - l_l (F_{G,r} \sin\alpha - F_{G,t} \cos\alpha) - (m_{tb} l_{tb} + m_{ft} l_{ft}) g \cos(\alpha + \psi) = 0 \tag{B.4}$$

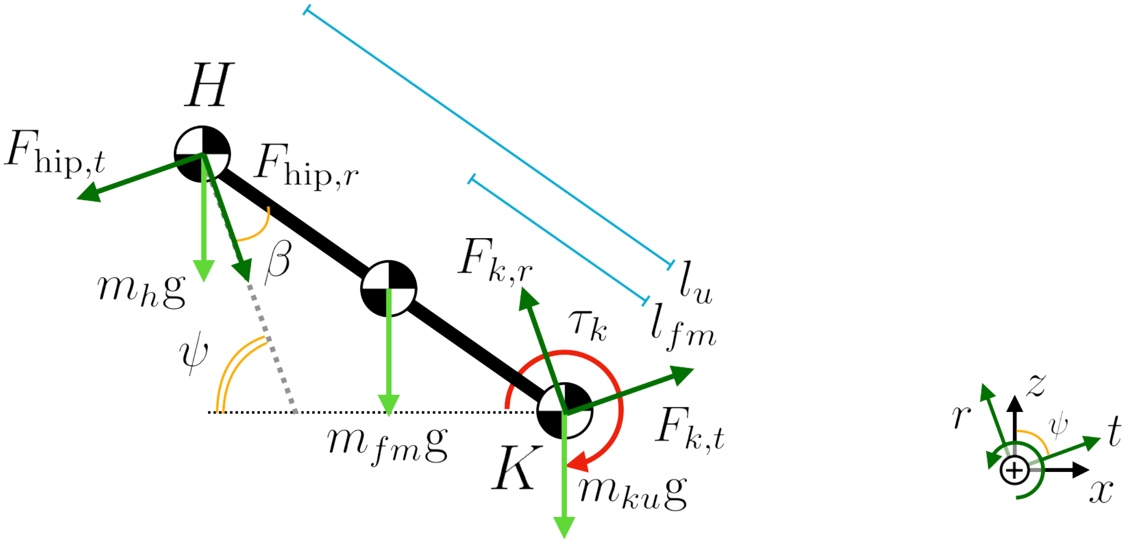


Figure B.3: Free body diagram of the upper leg (femur) segment

Figure B.3 illustrates a free body diagram of the upper leg segment. For the member $\overline{HK}$, assuming $T_{hip} = 0$,

$$\sum F_t = 0 : F_{k,t} - F_{hip,t} - (m_h + m_{fm} + m_{ku}) g \cos\psi = 0 \tag{B.5}$$

$$\sum F_r = 0 : F_{k,r} - F_{hip,r} - (m_h + m_{fm} + m_{ku}) g \sin\psi = 0 \tag{B.6}$$

$$\sum_K M = 0 : -T_{\text{knee}} + l_u (F_{hip,r} \sin\beta + F_{hip,t} \cos\beta) + (m_h l_u + m_{fm} l_{fm}) g \cos(\psi - \beta) = 0 \tag{B.7}$$

From Equations (B.5) and (B.6),

$$F_{k,t} = F_{hip,t} + (m_h + m_{fm} + m_{ku})g\cos\psi \tag{B.8}$$

$$F_{k,r} = F_{hip,r} + (m_h + m_{fm} + m_{ku})g\sin\psi \tag{B.9}$$

Let $M = m_h + m_{fm} + m_{ku} + m_{kl} + m_{tb} + m_{ft}$.

From Equations (B.2), (B.3), (B.8) and (B.9),

$$\begin{aligned} F_{G,t} &= F_{k,t} + (m_{kl} + m_{tb} + m_{ft})g\cos\psi \\ &= F_{hip,t} + (m_h + m_{fm} + m_{ku})g\cos\psi + (m_{kl} + m_{tb} + m_{ft})g\cos\psi \\ &= F_{hip,t} + Mg\cos\psi \qquad \text{(B.10)} \\ F_{G,r} &= F_{k,r} + (m_{kl} + m_{tb} + m_{ft})g\sin\psi \\ &= F_{hip,r} + (m_h + m_{fm} + m_{ku})g\sin\psi + (m_{kl} + m_{tb} + m_{ft})g\sin\psi \\ &= F_{hip,r} + Mg\sin\psi \qquad \text{(B.11)} \end{aligned}$$

Let $M_u := -\frac{1}{l_u}\left(m_h l_u - m_{fm} l_{fm}\right)$, and rewriting Equation (B.4),

$$\begin{aligned} F_{hip,r}\sin\beta + F_{hip,t}\cos\beta &= \frac{1}{l_u}\left\{T_{\text{knee}} - (m_h l_u + m_{fm} l_{fm})g\cos(\alpha + \psi - \gamma)\right\} \\ &= \frac{1}{l_u}T_{\text{knee}} + M_u g\cos(\alpha + \psi - \gamma) \qquad \text{(B.12)} \end{aligned}$$

Likewise, let $M_l := \frac{1}{l_l}\left(Ml_l - m_{tb} l_{tb} - m_{ft} l_{ft}\right)$, and rewriting Equation (B.7),

$$\begin{aligned} F_{hip,r}\sin\alpha - F_{hip,t}\cos\alpha &= \frac{1}{l_l}\left\{T_{\text{knee}} + (Ml_l - m_{tb} l_{tb} - m_{ft} l_{ft})g\cos(\alpha + \psi)\right\} \\ &= \frac{1}{l_l}T_{\text{knee}} + \frac{1}{l_l}M_l g\cos(\alpha + \psi) \qquad \text{(B.13)} \end{aligned}$$

B.2 Radial Force

Let a vector of angles $\phi(q_k) = (\alpha, \beta, \gamma)$.

Now, $\big(\cos\alpha \cdot$Equation (B.12) $+\cos\beta \cdot$ Equation (B.13) $\big)$ yields an expression for $F_{hip,r}$:

$$\begin{aligned}
F_{hip,r}\sin(\alpha+\beta) &= F_{hip,r}\sin\gamma \\
&= \left(\frac{\cos\alpha}{l_u} + \frac{\cos\beta}{l_l}\right) T_{\text{knee}} + g\left\{M_u\cos\alpha\cos(\alpha+\psi-\gamma) + M_l\cos\beta\cos(\alpha+\psi)\right\} \\
F_{hip,r} &= \left(\frac{\cos\alpha}{l_u} + \frac{\cos\beta}{l_l}\right)\csc\gamma\; T_{\text{knee}} \\
&\quad + g\csc\gamma\left\{M_u\cos\alpha\cos(\alpha+\psi-\gamma) + M_l\cos\beta\cos(\alpha+\psi)\right\} \qquad \text{(B.14)}
\end{aligned}$$

Or, rewriting Equation (B.14) shortly gives

$$F_{hip,r} = A_r(\phi(q_k), l)T_{\text{knee}} + f_{hip,r}(\phi(q_k), l, m) \qquad \text{(B.15)}$$

$$A_r(\phi(q_k), l) = \left(\frac{\cos\alpha}{l_u} + \frac{\cos(\gamma-\alpha)}{l_l}\right)\csc\gamma \qquad \text{(B.16)}$$

$$f_{hip,r}(\phi(q_k), l, m) = g\csc\gamma\left\{M_u\cos\alpha\cos(\alpha+\psi-\gamma) + M_l\cos(\gamma-\alpha)\cos(\alpha+\psi)\right\} \qquad \text{(B.17)}$$

B.3 Tangential Force

Likewise, $\big(\sin\alpha \cdot$Equation (B.12) $-\sin\beta \cdot$ Equation (B.13)$\big)$ yields an expression for $F_{hip,t}$:

$$\begin{aligned}
F_{hip,t}\sin(\alpha+\beta) &= F_{hip,t}\sin\gamma \\
&= \left(\frac{\sin\alpha}{l_u} - \frac{\sin\beta}{l_l}\right) T_{\text{knee}} \\
&\quad + g\left\{M_u\sin\alpha\cos(\alpha+\psi-\gamma) - M_l\sin\beta\cos(\alpha+\psi)\right\} \\
F_{hip,t} &= \left(\frac{\sin\alpha}{l_u} - \frac{\sin\beta}{l_l}\right)\csc\gamma\; T_{\text{knee}} \\
&\quad + g\csc\gamma\left\{M_u\sin\alpha\cos(\alpha+\psi-\gamma) - M_l\sin\beta\cos(\alpha+\psi)\right\} \qquad \text{(B.18)}
\end{aligned}$$

Or, rewriting Equation (B.18) shortly gives

$$\begin{aligned} F_{hip,t} &= A_t(\phi(q_k), l)T_{\text{knee}} + f_{hip,t}(\phi(q_k), l, m) && \text{(B.19)} \\ A_t(\phi(q_k), l) &= \left(\frac{\sin\alpha}{l_u} - \frac{\sin\beta}{l_l}\right)\csc\gamma && \text{(B.20)} \\ f_{hip,t}(\phi(q_k), l, m) &= g\csc\gamma\left\{M_u \sin\alpha\cos(\alpha+\psi-\gamma) - M_l\sin\beta\cos(\alpha+\psi)\right\} \end{aligned}$$

B.4 Summary

Since T_{knee} refers to the torque generated by the torsional spring at the knee, the coefficients A_r and A_t tell how much T_{knee} makes contribute in each direction. Let's limit the scope of investigation of the angle γ to $0 \geq \gamma \geq \pi$, and $\alpha, \beta > 0$ in this scope. This will successfully cover most cases of the legged robots in the configuration unless the knee rotates over the straight knee. One can easily extend the investigation for $-\pi \geq \gamma \geq 0$. The knee joint of BALLU also falls into our scope of the investigation.

In this range of motion, while T_{knee} always contributes in the radial direction as $A_r > 0$, it is not the case in the tangential direction. From Equation (B.1), Equation (B.20) immediately becomes

$$A_t(\phi(q_k), l) = \left(\frac{\sin\alpha}{l_u} - \frac{\sin\beta}{l_l}\right)\csc\gamma = \left(\frac{\sin\alpha}{l_u} - \frac{\sin\alpha}{l_u}\right)\csc\gamma = 0$$

, and B.19 becomes

$$\begin{aligned} F_{hip,t} &= \cancelto{0}{A_t(\phi(q_k), l)}\, T_{\text{knee}} + f_{hip,t}(\phi(q_k), l, m) \\ &= f_{hip,t}(\phi(q_k), l, m) \end{aligned}$$

In other words, the torsional springs do not act in the direction perpendicular to the line connecting the hip and the foot but only in the direction along the line. □

Lastly, for completeness, negating the direction gives the expressions in Equations (2.3) and (2.4) since the force direction of the $F_{p,\cdot}$ and $F_{hip,\cdot}$ is in opposite, and aliasing T_{knee} as τ_k:

$$F_{p,r} = -F_{hip,r} = -A_r(\phi(q_k), l)\ \tau_k - f_{hip,r}(\phi(q_k), l, m) \tag{2.3}$$

$$F_{p,t} = -F_{hip,t} = \underbrace{-A_t(\phi(q_k), l)}_{0}\ \tau_k - f_{hip,t}(\phi(q_k), l, m) \tag{2.4}$$

BIBLIOGRAPHY

[1] Aerospace Robotics and Control at Caltech, "Leg on aerial robotic drone (leonardo)," [YouTube video], Jan. 10 2019, [Accessed Sep. 25, 2020]. [Online]. Available: https://www.youtube.com/watch?v=CFM2dRJkkzk

[2] Agility Robotics, "Robots," https://agilityrobotics.com/robots, 2023, [Accessed: 2023-03-20].

[3] M. S. Ahn, H. Chae, D. Noh, H. Nam, and D. Hong, "Analysis and noise modeling of the intel realsense d435 for mobile robots," in *2019 16th International Conference on Ubiquitous Robots (UR)*. IEEE, 2019, pp. 707–711.

[4] M. S. Ahn and D. Hong, "Dynamic, robust locomotion for a non-anthropomorphic biped," in *2020 17th International Conference on Ubiquitous Robots (UR)*. IEEE, 2020, pp. 185–191.

[5] M. S. Ahn, D. Sun, H. Chae, I. Yamayoshi, and D. Hong, "Lessons learned from the development and deployment of a hotel concierge robot to be operated in a real world environment," in *2019 16th International Conference on Ubiquitous Robots (UR)*, 2019, pp. 55–60.

[6] ANYbotics, "ANYmal," http://www.anybotics.com/anymal-autonomous-legged-robot/, 2023, [Accessed: 2023-03-20].

[7] T. Apgar, P. Clary, K. Green, A. Fern, and J. W. Hurst, "Fast online trajectory optimization for the bipedal robot cassie." in *Robotics: Science and Systems*, vol. 101. Pittsburgh, Pennsylvania, USA, 2018, p. 14.

[8] L. Aymerich-Franch and I. Ferrer, "Social robots as a brand strategy," pp. 86–102, 2020.

[9] C. M. Best, J. P. Wilson, and M. D. Killpack, "Control of a pneumatically actuated, fully inflatable, fabric-based, humanoid robot," in *2015 IEEE-RAS 15th*

International Conference on Humanoid Robots (Humanoids), 2015, pp. 1133–1140. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/7363495

[10] G. Bledt, M. J. Powell, B. Katz, J. Di Carlo, P. M. Wensing, and S. Kim, "Mit cheetah 3: Design and control of a robust, dynamic quadruped robot," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Oct 2018, pp. 2245–2252.

[11] Boston Dynamics, "Atlas™ — Boston Dynamics," https://www.bostondynamics.com/atlas, 2023, [Accessed: 2023-03-20].

[12] ——, "Spot® — Boston Dynamics," https://www.bostondynamics.com/spot, 2023, [Accessed: 2023-03-20].

[13] G. Brockman, V. Cheung, L. Pettersson, J. Schneider, J. Schulman, J. Tang, and W. Zaremba, "Openai gym," 2016.

[14] B. Butcher and B. J. Smith, "Feature engineering and selection: A practical approach for predictive models: by max kuhn and kjell johnson. boca raton, fl: Chapman & hall/crc press, 2019, xv+ 297 pp., 79.95 (h), isbn: 978-1-13-807922-9." 2020.

[15] D.-A. Clevert, T. Unterthiner, and S. Hochreiter, "Fast and accurate deep network learning by exponential linear units (elus)," *arXiv preprint arXiv:1511.07289*, 2015.

[16] E. Coumans and Y. Bai, "Pybullet, a python module for physics simulation for games, robotics and machine learning," http://pybullet.org, 2016–2021.

[17] H. Coxeter and S. Greitzer, "Geometry revisited," *Math. Assoc. America, Washington, DC*, 1967.

[18] T. Dang, M. Tranzatto, S. Khattak, F. Mascarich, K. Alexis, and M. Hutter, "Graph-based subterranean exploration path planning using aerial and legged robots," *Journal of Field Robotics*, vol. 37, no. 8, pp. 1363–1388, 2020.

[19] J. Di Carlo, P. M. Wensing, B. Katz, G. Bledt, and S. Kim, “Dynamic locomotion in the mit cheetah 3 through convex model-predictive control,” in *2018 IEEE/RSJ international conference on intelligent robots and systems (IROS).* IEEE, 2018, pp. 1–9.

[20] Double Robotics, “Hybrid Office,” May. 20 2015, [Accessed Jun. 19, 2021]. [Online]. Available: https://www.doublerobotics.com/stories/?story=true&id=35

[21] N. Elkmann, T. Felsch, M. Sack, J. Saenz, and J. Hortig, “Innovative service robot systems for facade cleaning of difficult-to-access areas,” in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, vol. 1. IEEE/RSJ, 2002, pp. 756–762 vol.1. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/1041481

[22] M. Ester, H.-P. Kriegel, X. Xu *et al.*, “A density-based algorithm for discovering clusters in large spatial databases with noise.” AAAI Press, 1996, p. 226–231.

[23] R. A. Fisher, *Statistical methods for research workers.* Springer, 1992.

[24] N. Fukawa, “Robots on blockchain: Emergence of robotic service organizations,” *SMR-Journal of Service Management Research*, vol. 4, no. 1, pp. 9–20, 2020.

[25] Gareth Halfacree, “Benchmarking the Raspberry Pi Zero 2 W,” Oct. 27 2021, [Accessed Mar. 19, 2023]. [Online]. Available: https://medium.com/@ghalfacree/benchmarking-the-raspberry-pi-zero-2-w-b868accdbb50

[26] S. Ghassemi and D. Hong, “Feasibility study of a novel robotic system ballu: Buoyancy assisted lightweight legged unit,” in *2016 IEEE-RAS 16th International Conference on Humanoid Robots (Humanoids)*, 2016, pp. 144–144.

[27] R. Grandia, F. Farshidian, R. Ranftl, and M. Hutter, “Feedback mpc for torque-controlled legged robots,” in *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS).* IEEE, 2019, pp. 4730–4737.

[28] W. T. Higgins, “A comparison of complementary and kalman filtering,” *IEEE Transactions on Aerospace and Electronic Systems*, vol. AES-11, no. 3, pp. 321–325, May 1975.

[29] J. Hooks, M. S. Ahn, J. Yu, X. Zhang, T. Zhu, H. Chae, and D. Hong, “Alphred: A multi-modal operations quadruped robot for package delivery applications,” *IEEE Robotics and Automation Letters*, vol. 5, no. 4, pp. 5409–5416, 2020.

[30] IBM, “Hilton and ibm pilot ”connie,” the world’s first watson-enabled hotel concierge,” Mar. 6 2016. [Online]. Available: https://newsroom.ibm.com/2016-03-09-Hilton-and-IBM-Pilot-Connie-The-Worlds-First-Watson-Enabled-Hotel-Concierge

[31] Incheon Airport, “[Incheon Airport] 2019 Incheon International Airport Promotional Video,” [YouTube video], Mar. 15 2019, [Accessed Jun. 19, 2021]. [Online]. Available: https://youtu.be/6rdTHm7_J6s?t=80

[32] R. E. Kalman *et al.*, “A new approach to linear filtering and prediction problems [j],” *Journal of basic Engineering*, vol. 82, no. 1, pp. 35–45, 1960.

[33] B. Katz, J. Di Carlo, and S. Kim, “Mini cheetah: A platform for pushing the limits of dynamic quadruped control,” in *2019 international conference on robotics and automation (ICRA)*. IEEE, 2019, pp. 6295–6301.

[34] D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” *arXiv preprint arXiv:1412.6980*, 2014.

[35] X. Lin, G. Fernandez, S. Ghassemi, and D. W. Hong, “Feasibility Study of an Aerial Lifting Device Using Aerodynamic Drag for Ascent,” in *Feasibility Study of an Aerial Lifting Device Using Aerodynamic Drag for Ascent*, ser. International Design Engineering Technical Conferences and Computers and Information in Engineering Conference, vol. Volume 5B: 43rd Mechanisms and Robotics Conference, 08 2019, v05BT07A016. [Online]. Available: https://doi.org/10.1115/DETC2019-98138

[36] S. Macenski, T. Foote, B. Gerkey, C. Lalancette, and W. Woodall, “Robot operating system 2: Design, architecture, and uses in the wild,” *Science Robotics*, vol. 7, no. 66, p. eabm6074, 2022. [Online]. Available: https://www.science.org/doi/abs/10.1126/scirobotics.abm6074

[37] A. Maekawa, R. Niiyama, and S. Yamanaka, “Aerial-biped: A new physical expression by the biped robot using a quadrotor,” in *ACM SIGGRAPH 2018 Emerging Technologies*, ser. SIGGRAPH ’18. New York, NY, USA: Association for Computing Machinery, 2018. [Online]. Available: https://doi.org/10.1145/3214907.3214928

[38] H. D. Nayar, M. T. Pauken, M. L. Cable, and M. A. Hans, “Balloon-based concept vehicle for extreme terrain mobility,” in *2019 IEEE Aerospace Conference*, 2019, pp. 1–12.

[39] D. Nowacka, N. Y. Hammerla, C. Elsden, T. Plötz, and D. Kirk, “Diri - the actuated helium balloon: A study of autonomous behaviour in interfaces,” in *Proceedings of the 2015 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, ser. UbiComp ’15. New York, NY, USA: Association for Computing Machinery, 2015, p. 349–360. [Online]. Available: https://doi.org/10.1145/2750858.2805825

[40] Patrick Fromaget, “Raspberry Pi Zero vs Zero 2: Should you buy it? (Benchmark),” [Accessed Mar. 19, 2023]. [Online]. Available: https://raspberrytips.com/raspberry-pi-zero-vs-zero-2/

[41] K. Pearson, “Vii. note on regression and inheritance in the case of two parents,” *proceedings of the royal society of London*, vol. 58, no. 347-352, pp. 240–242, 1895.

[42] ——, “X. on the criterion that a given system of deviations from the probable in the case of a correlated system of variables is such that it can be reasonably supposed to have arisen from random sampling,” *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, vol. 50, no. 302, pp. 157–175, 1900.

[43] A. Raffin, A. Hill, A. Gleave, A. Kanervisto, M. Ernestus, and N. Dormann, “Stable-baselines3: Reliable reinforcement learning implementations,” *Journal of Machine Learning Research*, vol. 22, no. 268, pp. 1–8, 2021. [Online]. Available: http://jmlr.org/papers/v22/20-1364.html

[44] M. H. Raibert, *Legged robots that balance.* MIT press, 1986.

[45] N. V. Rao, “Analysis of an actuated two segment leg model of locomotion,” Ph.D. dissertation, Purdue University, 2013. [Online]. Available: https://www.proquest.com/dissertations-theses/analysis-actuated-two-segment-leg-model/docview/1518520987/se-2

[46] J. Reher and A. D. Ames, “Inverse dynamics control of compliant hybrid zero dynamic walking,” in *2021 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2021, pp. 2040–2047.

[47] Roanoke County Public Library, “Pepper, the Humanoid Robot,” 2018, [Accessed Jun. 19, 2021]. [Online]. Available: https://www.roanokecountyva.gov/2046/Pepper

[48] E. Rohmer, S. P. N. Singh, and M. Freese, “Coppeliasim (formerly v-rep): a versatile and scalable robot simulation framework,” in *Proc. of The International Conference on Intelligent Robots and Systems (IROS)*, 2013, www.coppeliarobotics.com.

[49] F. Santosa and W. W. Symes, “Linear inversion of band-limited reflection seismograms,” *SIAM journal on scientific and statistical computing*, vol. 7, no. 4, pp. 1307–1330, 1986.

[50] N. Sontakke, H. Chae, S. Lee, T. Huang, D. W. Hong, and S. Ha, “Residual physics learning and system identification for sim-to-real transfer of policies on buoyancy assisted legged robots,” 2023.

[51] C. Spearman, “The proof and measurement of association between two things,” *The American Journal of Psychology*, vol. 15, no. 1, pp. 72–101, 1904.

[52] M. J. Spenko, G. C. Haynes, J. Saunders, M. R. Cutkosky, A. A. Rizzi, R. J. Full, and D. E. Koditschek, "Biologically inspired climbing with a hexapedal robot," *Journal of field robotics*, vol. 25, no. 4-5, pp. 223–242, 2008.

[53] M. Takeichi, K. Suzumori, G. Endo, and H. Nabae, "Development of giacometti arm with balloon body," *IEEE Robotics and Automation Letters*, vol. 2, no. 2, pp. 951–957, 2017.

[54] F. Takemura, K. Maeda, and S. Tadokoro, "Attitude stability of a cable driven balloon robot," in *2006 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2006, pp. 3504–3509. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/4058944

[55] R. Tibshirani, "Regression shrinkage and selection via the lasso," *Journal of the Royal Statistical Society: Series B (Methodological)*, vol. 58, no. 1, pp. 267–288, 1996.

[56] E. Todorov, T. Erez, and Y. Tassa, "Mujoco: A physics engine for model-based control," in *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems.* IEEE, 2012, pp. 5026–5033.

[57] Unitree, "Go1," https://www.unitree.com/products/go1, 2023, [Accessed: 2023-03-20].

[58] H. Vafaie and K. De Jong, "Robust feature selection algorithms," in *Proceedings of 1993 ieee conference on tools with al (tai-93).* IEEE, 1993, pp. 356–363.

[59] Y. Yamada and T. Nakamura, "Gerwalk: Lightweight mobile robot with buoyant balloon body and bamboo rimless wheel," in *2018 IEEE International Conference on Robotics and Biomimetics (ROBIO)*, 2018, pp. 1814–1821.

[60] H. Zhang, Y. Zhou, M. Dai, and Z. Zhang, "A novel flying robot system driven by dielectric elastomer balloon actuators," *Journal of Intelligent Material Systems and Structures. 2018*, pp. 2522–2527, 2018. [Online]. Available: https://journals.sagepub.com/doi/full/10.1177/1045389X18770879

[61] X. Zhang, Y. Zeng, and D. Hong, “A novel and efficient compact propulsion mechanism with high response speed for robotic applications,” in *2016 13th International Conference on Ubiquitous Robots and Ambient Intelligence (URAI)*, 2016, pp. 314–320.

[62] Y. Zhong, R. Wang, H. Feng, and Y. Chen, “Analysis and research of quadruped robot’s legs: A comprehensive review,” *International Journal of Advanced Robotic Systems*, vol. 16, no. 3, p. 1729881419844148, 2019.