STOCHASTIC ANALYSIS AND ADAPTIVE CONTROL STUDIES IN LEGGED SYSTEMS

A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES
OF
MIDDLE EAST TECHNICAL UNIVERSITY

BY

GÜNER DİLŞAD ER

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR
THE DEGREE OF MASTER OF SCIENCE
IN
ELECTRICAL AND ELECTRONICS ENGINEERING

JUNE 2022

Approval of the thesis:

STOCHASTIC ANALYSIS AND ADAPTIVE CONTROL STUDIES IN LEGGED SYSTEMS

submitted by **GÜNER DILŞAD ER** in partial fulfillment of the requirements for the degree of **Master of Science in Electrical and Electronics Engineering Department, Middle East Technical University** by,

Prof. Dr. Halil KALIPÇILAR
Dean, Graduate School of **Natural and Applied Sciences** _____

Prof. Dr. İlkay ULUSOY
Head of Department, **Electrical and Electronics Engineering** _____

Assoc. Prof. Dr. Mustafa Mert ANKARALI
Supervisor, **Electrical and Electronics Engineering, METU** _____

Prof. Dr. Uluç SARANLI
Co-supervisor, **Computer Engineering, METU** _____

Examining Committee Members:

Prof. Dr. Kemal LEBLEBİCİOĞLU
Electrical and Electronics Engineering, METU _____

Assoc. Prof. Dr. Mustafa Mert ANKARALI
Electrical and Electronics Engineering, METU _____

Prof. Dr. Umut ORGUNER
Electrical and Electronics Engineering, METU _____

Assoc. Prof. Dr. Emre ÖZKAN
Electrical and Electronics Engineering, METU _____

Assist. Prof. Dr. İsmail UYANIK
Electrical and Electronics Engineering, Hacettepe University _____

Date: 07.06.2022

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name, Surname: Güner Dilşad ER

Signature :

ABSTRACT

STOCHASTIC ANALYSIS AND ADAPTIVE CONTROL STUDIES IN LEGGED SYSTEMS

ER, Güner Dilşad
M.S., Department of Electrical and Electronics Engineering
Supervisor: Assoc. Prof. Dr. Mustafa Mert ANKARALI
Co-Supervisor: Prof. Dr. Uluç SARANLI

June 2022, 106 pages

Underactuated legged robots depict highly nonlinear and complex dynamical behaviors that create significant challenges in accurately modeling system dynamics using both first principles and system identification approaches. Hence, the design of stabilizing controllers becomes more challenging due to inaccurate modeling. Suppose physical parameters on mathematical models have miscalibrations due to uncertainty in identifying and modeling processes. In that case, designed controllers could perform poorly or even result in unstable responses. Moreover, these parameters can change over time due to operation and environmental conditions. In that respect, analogous to a living organism modifying its behavior in response to novel conditions, adapting/updating system parameters, such as spring constant to compensate for modeling errors, could provide the advantage of constructing a stable gait level controller without needing "exact" dynamical parameter values. The first part of this thesis presents an online, model-based adaptive control approach for an underactuated planar hexapod robot's pronking behavior adopted from antelope species. Systematic simulation studies show that the adaptive control policy is robust to high levels of pa-

rameter uncertainties compared to a non-adaptive model-based dead-beat controller.

In the second part of the study, an efficient estimation method based on unscented transformation is proposed to quantify the stochastic stability characteristics of metastable legged systems. Unlike previous methods requiring high-dimensional state space discretization for a broad set of initial conditions to estimate the stability characteristics, this study aims to assess controller performances and analyze parametric dependencies with fewer experiments. In the proposed approach, the unscented transformation is employed because it utilizes prior knowledge of the noise statistics, and provides informed choices of initial conditions for the experiments, thus, reducing the computational complexity significantly. Additionally, it allows dealing with multiple sources of uncertainties and high-dimensional system dynamics. Finally, the capability of the proposed method is shown via analyzing a one-dimensional hopper and an underactuated bipedal walking simulation with a hybrid zero dynamics controller.

Keywords: legged robots, pronking, adaptive control, unscented transformation, stochastic stability, metastability

ÖZ

BACAKLI SİSTEMLERDE STOKASTİK ANALİZ VE ADAPTİF KONTROL ÇALIŞMALARI

ER, Güner Dilşad
Yüksek Lisans, Elektrik ve Elektronik Mühendisliği Bölümü
Tez Yöneticisi: Doç. Dr. Mustafa Mert ANKARALI
Ortak Tez Yöneticisi: Prof. Dr. Uluç SARANLI

Haziran 2022 , 106 sayfa

Eksik tahrikli bacaklı robotlar, hem temel prensipleri hem de sistem tanımlama yaklaşımlarını kullanarak sistem dinamiklerini doğru bir şekilde modellemede önemli zorluklar yaratan oldukça doğrusal olmayan ve karmaşık dinamik davranışları tasvir eder. Bu nedenle, hatalı modelleme nedeniyle stabilize edici kontrolcülerin tasarımı daha zor hale gelir. Matematiksel modellerdeki fiziksel parametrelerin, tanımlama ve modelleme süreçlerindeki belirsizlik nedeniyle yanlış kalibrasyonlara sahip olduğunu varsayalım. Bu durumda, tasarlanan kontrolcüler kötü performans gösterebilir ve hatta kararsız tepkilere neden olabilir. Ayrıca bu parametreler çalışma ve çevre koşulları nedeniyle zaman içinde değişebilir. Bu bağlamda, yeni koşullara tepki olarak davranışını değiştiren canlı bir organizmaya benzer şekilde, modelleme hatalarını telafi etmek için yay sabiti gibi sistem parametrelerini uyarlamak/güncellemek, "kesin" dinamik parametre değerlerine ihtiyaç duymadan kararlı bir yürüyüş seviyesi kontrolcüsü oluşturma avantajı sağlayabilir. Bu tezin ilk bölümü, antilop türlerinden uyarlanan, eksik tahrikli bir düzlemsel hexapod robotun pronking davranışı için çevrimiçi,

model tabanlı bir adaptif kontrol yaklaşımı sunar. Sistematik simülasyon çalışmaları, bu adaptif kontrol politikasının, adaptif olmayan model tabanlı ölü vuruşlu denetleyiciye kıyasla, yüksek düzeyde parametre belirsizliklerine karşı dayanıklı olduğunu göstermektedir.

Çalışmanın ikinci bölümünde, yarı kararlı bacaklı sistemlerin stokastik kararlılık özelliklerini ölçmek için sigma bazlı dönüşüme dayalı etkin bir kestirim yöntemi önerilmiştir. Kararlılık özelliklerini tahmin etmek için geniş bir başlangıç koşulları kümesi için yüksek boyutlu durum uzayı kesiklileştirilmesini gerektiren önceki yöntemlerin aksine, bu çalışma, daha az deneyle kontrolcü performanslarını değerlendirmeyi ve parametrik bağımlılıkları analiz etmeyi amaçlamaktadır. Önerilen yaklaşımda, gürültü istatistiklerine ilişkin ön bilgileri kullandığı ve deneyler için başlangıç koşullarının bilinçli seçimlerini sağladığı ve bu sayede hesaplama karmaşıklığını önemli ölçüde azalttığı için sigma bazlı dönüşüm kullanılmıştır. Ek olarak, birden fazla belirsizlik kaynağı ve yüksek boyutlu sistem dinamikleri ile başa çıkmayı sağlar Son olarak, önerilen yöntemin kabiliyeti, tek boyutlu bir zıplama mekanizmasının ve bir eksik tahrikli iki ayaklı yürüme simülasyonu analiz edilerek gösterilmiştir.

Anahtar Kelimeler: bacaklı robotlar, pronklama, adaptif kontrol, sigma bazlı transform, stokastik kararlılık, yarı kararlılık

ACKNOWLEDGMENTS

I would like to thank my supervisor, Assoc. Prof. Dr. Mustafa Mert Ankaralı for his guidance and support not only for this thesis but also for the next steps of my academic journey. I also thank my co-supervisor, Prof. Dr. Uluç Saranlı, for introducing me to the world of legged locomotion, all thesis committee members for providing valuable feedback and support during my thesis defense, Prof. Dr. Çağatay Candan together with all faculty in the METU EEE Department, for their support throughout my BSc and MSc, and finally, Dr. Orhan Can Görür for driving me into the world of robotics and becoming one of my supporters for my next steps.

This study is supported by Turk Telekom Inc. within the framework of the 5G and Beyond Joint Graduate Support Programme coordinated by the Information and Communication Technologies Authority and the Scientific and Technological Research Council of Turkey (TÜBİTAK). I especially thank my supervisor in Turk Telekom Inc. R&D Operations, Selami Çiftçi, for his sincere support.

I am grateful to my friends from ATLAS laboratory: Furkan Davulcu, Emre Tanfener, and Sait Sovukluk, who have provided a great work environment and helped me through invaluable discussions. Especially with Sait's contribution to the bipedal walker, I could elevate the scientific contribution of this thesis.

I would like to thank my parents, my little sister, Nilgün, Ümit, and Dilara Er, and my amazing extended family for their undying support from the beginning of my academic journey. I want to thank my cousin Emrecan for bringing joy to our home as my roommate. I always felt appreciated and surrounded by lovely and fun people. I would like to especially mention my late grandparents, Döne and Ali Nazmi Er, for setting great examples for me. I would not be the same person without anyone of you.

Last but not least, I would like to express my gratitude to Semih Kafkas, who was always by my side over the last decade. I feel extremely lucky to have met someone full of love and joy. I am excited to launch our next adventure together.

TABLE OF CONTENTS

LIST OF TABLES

TABLES

LIST OF FIGURES

FIGURES

LIST OF ABBREVIATIONS

2D	2 Dimensional
3D	3 Dimensional
SLIP	Spring Loaded Inverted Pendulum
SLIP-T	Spring Loaded Inverted Pendulum with torque actuation at the hip
F-SLIP	Spring Loaded Inverted Pendulum with constant forcing and damping
DOF	Degree of Freedom
MFPT	Mean First Passage Time
EKF	Extended Kalman Filter
UKF	Unscented Kalman Filter
UT	Unscented Transformation
HZD	Hybrid Zero Dynamics
PCA	Principal Component Analysis

CHAPTER 1

INTRODUCTION

When the real systems operating outdoors are employed with the controllers working perfectly in the lab environment, the main obstacle encountered is nature itself. The cumulative effect of different internal and external discrepancies should be accounted for in the systems' analysis, design, and control. This thesis compiles different perspectives on legged locomotion in this manner. Hereby, the objectives of the studies compiled in this thesis are twofold. The first one is whether it is possible to compensate for the errors and maintain the locomotion by adding a layer of adaptation just like in animals' adaptation traits if an existing controller structure is vulnerable to discrepancies in the modeling step. Second, if there are inevitable uncertainties in the environment, the way to conduct the stability and return map analyses is investigated. The methodology for the stochastic analysis of legged systems is improved and presented with the underlying literature.

1.1 The Outline of the Thesis

This thesis consists of two separate research subjects related to different discrepancies that can be observed in actual physical systems.

First, in Chapter 2, an adaptive controller is designed as a middle layer for the hexapedal pronking platform. In Section 2.2, the existing dead-beat controller and the hexapedal system are presented. In Section 2.3, miscalibrations and modeling errors are discussed, and an indirect adaptive control scheme is presented. The proposed method is applied to the nonlinear simulation of planar hexapod platform RHex. An adaptive law is implemented inside the existing dead-beat controller to handle the

miscalibration problem. Later in Section 2.4, the performance of the middle layer controller is investigated, and in Section 2.5 stability of the closed-loop system with the adaptive controller is discussed with supporting results from the numerical stability analysis.

Next, in Chapter 3, an estimation method is proposed to extract the stochastic properties of the metastable dynamical systems and analyze the stochastic return maps. In Section 3.1, the existing approach to conducting stochastic analysis and the fundamental knowledge of borrowed tools are presented. Then, Section 3.2 introduces the stochastic analysis methodology. The proposed method is based on unscented transformation borrowed from Kalman Filters and compared with existing methods such as Monte Carlo simulations and a linearization-based estimation. After the introduction of literature and novel methodology, two different systems with different properties are investigated in Sections 3.3 and 3.5.

CHAPTER 2

ADAPTIVE CONTROL OF UNDERACTUATED PLANAR PRONKING HEXAPOD

2.1 Introduction

Underactuated legged platforms have various motion capabilities than the wheeled robots that preceded them with added mobility such as running, pronking [4], flipping [5], and self-righting [6]. They use their natural dynamics to reach high speed, efficiency, and robustness performance. In return for this mobility, complex hardware structure and controller design arise as challenges. Marc Raibert decreased this complexity using dynamic modes of locomotion [7] in his runners first. Since 1970, Raibert's work has inspired researchers to establish a large new field of study on legged robots [8, 9, 10, 11].

If second-order dynamics are appropriately designed and tuned, the model can achieve a wide range of behaviors despite the underactuated nature of many of the legged robotic platforms. However, as the systems become more agile and faster, substantial challenges and problems arise when controlling robot dynamics. Template-based control is an approach to isolate and "independently" control the degrees of freedom relevant to the task [12, 13, 14, 15]. In literature, researchers developed template-based control methods for a variety of robots and motions. Saranli and Koditschek applied template-based control for a running hexapod [14]. Oehlke et al. used template-based hopping control in their research on a bio-inspired segmented robotic leg [13]. Peekema introduced the template-based control of the bipedal robot ATRIAS in his work [12]. Furthermore, in recent studies, Kurtz et al. proposed a template-based whole-body controller and simulated it on the 30-DOF Valkyrie humanoid model

[15].

This study focuses on the pronking behavior on a planar dynamic model of the RHex platform [16]. RHex is a structurally simple and yet highly mobile hexapod robot with a single rotary actuator on each hip and capable of executing various gaits [16]. There are contralateral legs that can be used in synchrony for some behaviors. Figure 2.1 shows some examples of previously built RHex robots having different capabilities such as pronking, stair climbing, navigation on rough terrains, and swimming.

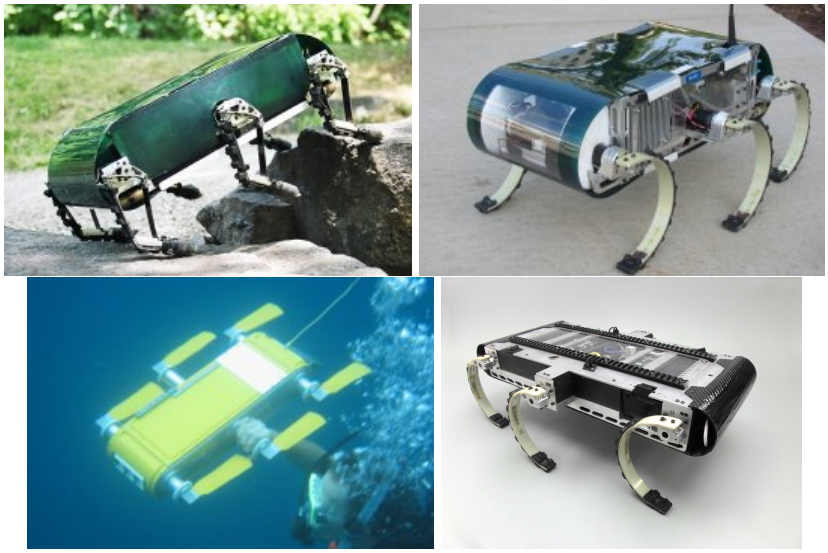

Figure 2.1: Variety of hexapedal RHex platforms such as RHex 0.8, RHex 1.1 (with camera), Aqua [1] and X-RHex [2]

Pronking behavior represents leaping in the air with an arched back and stiff legs, as shown in Figure 2.2, and it is a gait adopted by legged animals such as springbok or other antelope to show their strength to their predators. It is an example of honest signaling in zoology [17]. While pronking, animals use their legs in synchrony, and a flight phase follows the stance phase, as depicted in Figure 2.3. Pronking gives robots the advantage of considerable jumping heights requiring little ground contact during locomotion. Jumping higher could help them overcome different obstacles and traverse rough terrain. This advantage appealed to scientists to apply controllers to analyze and perform pronking [18]. Moreover, this is still a living research topic for robots having different types and numbers of legs [4, 19, 20, 21]. In this thesis, a new method is proposed to improve the pronking motion of the hexapedal RHex platform.

Figure 2.2: Pronking of a springbok [3]

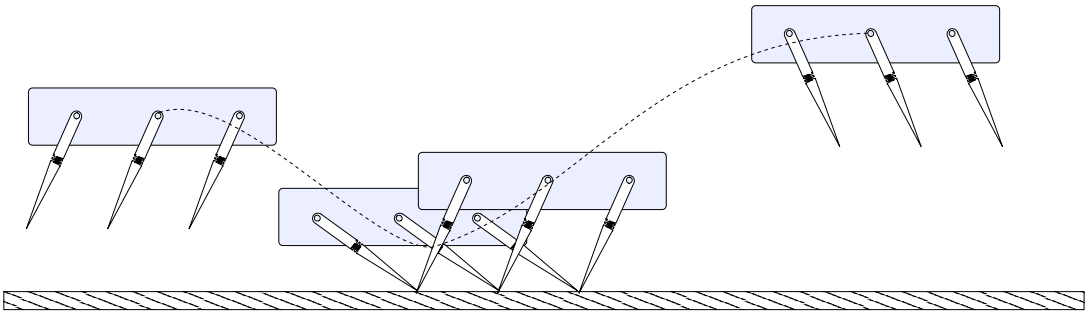

Figure 2.3: Pronking behavior of planar hexapedal robot

Although legged robots are advantageous in agility, complex nonlinear dynamics of legged locomotion create substantial challenges to the control policy design. Indeed, the complexity of the problem is much deeper than just dealing with complex nonlinear computations to generate control actions. The challenge in designing legged locomotion behavior starts with the designing control target itself and developing associated performance metrics, which are relatively trivial in non-legged robotic platforms, such as wheeled systems and aerial platforms. This fact pushed the researchers to inspire from nature and adopt bio-inspired design and control strategies. One of the most famous examples of bio-inspiration in legged locomotion is the adoption of the SLIP model in the design of legged platforms and control policies. Numerous studies [22] reported that the SLIP model (and its variants) could accurately capture

the center of dynamics and the rhythmic interchange between kinetic and potential energy in running animals by studying biomechanical data [23]. This discovery led the roboticists to develop a number of robotic systems based on the core mechanical principles of the SLIP model [24, 25, 26]. In connection with these developments, several researchers utilized the SLIP model as a high-level control interface for designing control policies for more complex legged platforms [27, 28, 29].

The spring-loaded inverted pendulum (SLIP) model [30], which consists of a point mass attached to a free rotating massless leg equipped with a linear spring-damper pair, is a fundamental and straightforward model to understand fundamental principles in legged locomotion. It is widely used to describe many kinds of robots' leg behavior and brings the advantage of using existing control methods through analytical return maps [31, 32]. This thesis' proposed scheme is based on analytical approximations to SLIP dynamics [32], which will be briefly discussed later. Following the introduction of the extended SLIP model with torque actuation at the hip (SLIP-T), the model named Slimpod is developed as a simpler model of hexapedal pronking robot and utilized with an embedded dead-beat controller to realize the pronking motion [4].

The aforementioned embedded controller is sensitive to the changes inside the approximate analytical map. Parameters inside the approximate map are fixed but might have deviations from their measured/estimated values on some variables, such as the position of the center of mass, stiffness of the spring, damping, etc. These discrepancies may result from many reasons, such as measurement errors, corrosion, dirt, and fatigue. In this context, model parameters could adapt to internal or external changes to sustain a stable and robust pronking motion, which is the fundamental starting point of this study.

Adaptation to external influence is a broad and essential subject for robotics studies. Especially in physical systems, many sources lead the identification of inner system parameters' to be inaccurate. Even if the measurement accuracy is nearly perfect during the initial calibration phase, updating the parameters automatically as time goes by or the control task changes can still be necessary. In literature, there are many examples where the adaptive control framework is utilized as an adaptive locomotion

control [33, 34] and for interlimb coordination [35]. Besides, some precedents combining with different approaches borrowed from other engineering applications exist, such as adaptive control of a legged robot using an artificial neural network [36], or a combination of evolutionary and adaptive control strategies for a quadruped robot [37].

Adaptive control strategies centers around different approaches on legged robots [35, 33, 36, 37, 38]. In this study, unlike the previous work in literature [38], the intention is not an accurate system identification or an estimation of the environmental effects [39]. The main goal is to enhance the controller performance for varied circumstances by adding an extra layer to the structure. This strategy evocates the term adaptability for biological organisms [40]. Robots are desired to inherit the organisms' ability to modify their behavior in response to novel conditions. This ability may lead to a possible change in neural control circuits while meaning a change in inner controller parameters in the robotic systems. Also, prior adaptive controller studies [39, 38] had focused on simple template models. Unlike the previous literature work, a more complex anchor multi-legged model is adopted in this thesis.

An unknown plant's adaptive control can be carried out by directly adjusting control parameters in a feedback loop based on the error between plan and model outputs, known as direct control. An alternative method is to estimate the plant parameters and to adjust the control parameters based on such estimates described as indirect control [41]. Proposed adaptation indirectly affects the dead-beat controller output by amending the chosen parameter in the approximate map used in the embedded dead-beat controller.

Motivated by the previous work in literature [4], this study presents an adaptive control method for a pronking hexapedal robot with a spring-loaded inverted pendulum template-based controller. The controller with the proposed adaptation scheme is expected to work better than classical dead-beat controllers thanks to reducing the modeling error. In some sense, adaptive controllers correct approximation errors and, consequentially, provide better tracking of desired height and desired velocity.

The organization of this chapter will be as follows. The model dynamics and control section presents the basis model SLIP and its variations. Adaptive control of

Slimpod section introduces the structure of the indirect adaptive control utilized on the objective system. The performance analysis part addresses the simulation results, comparisons, and stability analysis. Finally, the last section closes the work with a conclusion.

2.2 Model Dynamics and Control

As previously stated in the introduction section, the proposed method is implemented on a validated planar hexapod model, namely, Slimpod [4]. This section presents an overview of underlying models for the application.

2.2.1 Dynamics and Control of SLIP Template

This section refers to the prior SLIP model in order to build a template for the controller. Spring Loaded Inverted Pendulum Model is composed of a point mass m, a massless leg with length r, and angle θ. The leg is equipped with a passive linear spring-damper structure of stiffness k and viscous damping d. Figure 2.4 depicts the SLIP model.

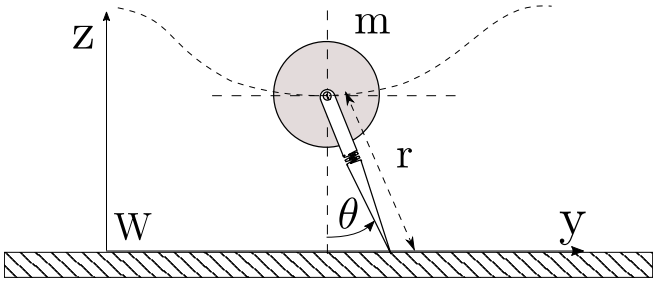

Figure 2.4: The Spring-Loaded Inverted Pendulum template

Throughout locomotion, the SLIP model has successive phases named stance and flight. The discrete transitions between those phases are called events, and four important events are touchdown, bottom, lift-off, and apex, as shown in Figure 2.5. Touchdown and lift-off represent the events on the transition from flight to stance,

stance to flight, respectively, and apex represents the event at the point where vertical speed equals zero. Apex state includes the height and horizontal velocity (${}^{a}X = [z, \dot{y}]$).

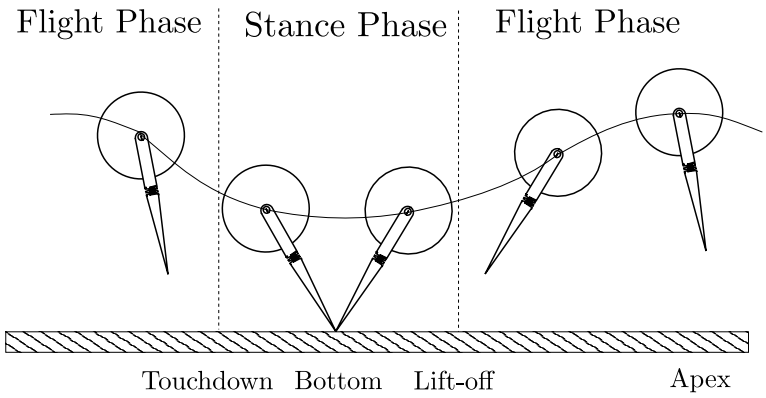

Figure 2.5: Locomotion of SLIP template with phases and transition events

Throughout this study, the dimensionless formulation is adopted to parametrize the problem and eliminate redundant parameters. Nondimensionalization is conducted according to Table 2.1 retrieved from Ankarali's study [42].

Table 2.1: Physical state variables and parameters with the definitions of their dimensionless counterparts. Variables with bars represent the physical quantities with dimension.

Dimensionless Variables	Definition	Physical Quantity Description
t	$:= \bar{t}/\lambda$	Time (where $\lambda := \sqrt{l_0/g}$)
$[y, z]$	$:= [\bar{y}/l_0, \bar{z}/l_0]$	Body position
$[r, \theta]$	$:= \left[\bar{r}/l_0, \bar{\theta}\right]$	SLIP leg length and leg angle
$[z_a, \dot{y}_a]$	$:= [\bar{z}_a/l_0, \dot{\bar{y}}_a\,(\lambda/l_0)]$	Apex height and velocity
k_s	$:= k_s\,(l_0/(mg))$	SLIP leg spring stiffness

Flight and stance dynamics of SLIP model [43, 4, 44] obtained from Euler-Lagrange formulation can be written in dimensionless coordinates as follows.

Flight:

$$\begin{aligned} \ddot{y} &= 0 \\ \ddot{z} &= -1 \end{aligned} \tag{2.1}$$

Stance:

$$\begin{aligned} \ddot{r} &= r\dot{\theta}^2 - k_s(r-1) - cos\theta - d\dot{r} \\ \ddot{\theta} &= -2\dot{r}\dot{\theta}/r + sin\theta/r \end{aligned} \tag{2.2}$$

where y and z are the horizontal and vertical positions of the point mass in flight, and r and θ are the leg length and leg angle in the stance phase. k_s and d indicate the leg spring stiffness and leg damping in the formulation, respectively. As can be deduced from the system equations, the system has a simple projectile trajectory during flight. At the same time, in the stance phase, the dynamics become more complex and, more importantly, non-integrable. This non-integrability issue is critical and will appear further in later sections.

The control inputs chosen for the SLIP template are leg touchdown angle θ together with leg lengths at touchdown r_{td} and lift-off r_{lo}. These control inputs make stance dynamics passive and allow the stance dynamics to be embedded with the placement of the virtual toe at touchdown into the planar hexapedal model.

The leg lengths at touchdown r_{td} and lift off r_{lo} can be computed using the energy difference between the current ($[\dot{y}_a, z_a]$) and desired apex state ($[\dot{y^*}_a, z^*_a]$).

$$\Delta E = (z^*_a - z_a) + \frac{1}{2}(\dot{y^*}^2_a - \dot{y}^2_a) \tag{2.3}$$

where $\dot{y}_a$ and z_a are the current apex height and horizontal velocity, whereas $\dot{y^*}_a$ and z^*_a are the desired apex height and horizontal velocity. Disregarding the damping in the SLIP model, if the energy difference is positive ($\Delta E > 0$), the leg lengths become as follows;

$$\begin{aligned} r_{td} &= r_0 - \sqrt{\frac{2\Delta E}{k}} \\ r_{lo} &= r_0 \end{aligned} \tag{2.4}$$

But if the energy difference is negative ($\Delta E > 0$), the leg lengths become as follows;

$$\begin{aligned} r_{lo} &= r_0 - \sqrt{\frac{-2\Delta E}{k}} \\ r_{td} &= r_0 \end{aligned} \tag{2.5}$$

Analytical stance map introduced by Geyer et al. [31] and modified by Ankarali et al. [32] with associated dead-beat controller proposed by Ankarali et al. [43, 4] underlie this study. Thanks to the derivation of an accurate stance map, the leg angle at touchdown can be calculated through the approximate expressions. In this modified approximate analytical map, the leg is assumed to stay close to vertical so that the effect of gravity during stance can be linearized. Linearization of gravity effect leads to the assumption of constant angular momentum p_θ and constant total mechanical energy E. The stance trajectories can be expressed as in (2.6).

$$\begin{aligned} r(t) &= 1 + a + bsin(\hat{\omega}_0 t) \\ \theta(t) &= \theta_{td} + p_\theta(1-2a)(t-t_{td}) + \frac{2bp_\theta}{\hat{\omega}_0}(cos(\hat{\omega}_0 t) - cos(\hat{\omega}_0 t_{td})) \end{aligned} \tag{2.6}$$

where p_θ, $\hat{\omega}_0$,a and b are defined as following

$$\begin{aligned} p_\theta &= r_{td}^2 \dot{\theta}_{td} \\ \hat{\omega}_0 &= \sqrt{k + 3p_\theta^2} \\ a &= \frac{p_\theta^2 - 1}{\hat{\omega}_0^2} \\ b &= \sqrt{a^2 + \frac{2E - p_\theta^2 - 2}{\hat{\omega}_0^2}} \end{aligned} \tag{2.7}$$

This stance map calculation is combined with the descent and ascent equations, and the combination yields the following analytical return map.

$$\begin{aligned} \hat{f}([\dot{y}_a, z_a]_k, [\theta, r_{lo}, r_{td}]) &= [\dot{y}_a, z_a]_{k+1} \\ \hat{f}(X_k, u_n) &= X_{k+1} \end{aligned} \tag{2.8}$$

This return map is not invertible in closed form but admits a numerical minimization problem to find touchdown leg angle θ.

$$\theta = \underset{\frac{-\pi}{2}<\theta<\frac{-\pi}{2}}{argmin} \left(\dot{y}_a^* - \left(\pi_{\dot{y}_a} o \hat{f}\right)([\dot{y}_a, z_a]_k, [\theta, r_{lo}, r_{td}])\right)^2 \tag{2.9}$$

where $\pi_{\dot{y}_a}$ is an operator to retrieve the horizontal velocity output of the analytical map.

The dead-beat stride controller for the SLIP template grounds on this analytical return map for high-level control of hexapedal pronking.

2.2.2 Dynamics of Slimpod Model

The hexapedal model consists of a body and six legs in contralateral pairs; each has a rotary actuator on its hips. Ankarali utilized a saggital planar model [4] using the leg pairs synchronically for pronking behavior. The planar Slimpod model [45, 46], illustrated in Figure 2.6, allows us to design a feedback controller.

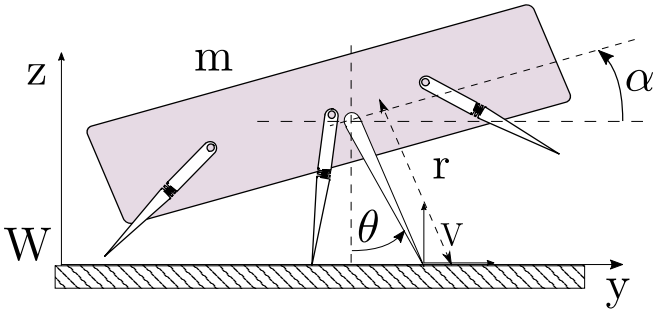

Figure 2.6: Slimpod, a planar dynamic model underlying hexapedal RHex robot. Unlike the generic drawing in this figure, legs are used synchronously during objective pronking motion.

The Slimpod model consists of a rigid body with inertia I and mass m and three legs representing contralateral pairs of RHex, each with a controllable torque. The legs are considered massless during the stance phase, and each leg has a spring with stiffness k_i and viscous damping coefficient d_i. Also, a virtual leg is defined (illustrated in Figure 2.6), extending from the body's center of mass to a stationary point on the ground.

A flight controller on a lower level drives all legs to their required positions. This drive is based on SLIP control decisions by solving kinematic equations for all legs. This inner controller realizes the desired control inputs by the placement of the virtual leg. After the placement of this virtual toe, the stance controller mimics ideal SLIP dynamics by choosing proper hip torque inputs for each leg of the Slimpod model. Stance dynamics in virtual toe coordinates in 2.10 will be the same as SLIP dynamics

(2.2).

$$\begin{aligned} \ddot{r} &= r\dot{\theta}^2 - \cos\theta + K_r \\ \ddot{\theta} &= \frac{-2\dot{r}\dot{\theta} + \sin\theta}{r} + K_\theta \\ \ddot{\alpha} &= \frac{K_\alpha}{j} \end{aligned} \tag{2.10}$$

where the forcing vector K values capture the effect of radial forces $F_{r,s}$ and external hip torques τ_s and can be represented as

$$\begin{aligned} K &= [K_r, K_\theta, K_\alpha]^T \\ &= J\tau_s + B \end{aligned} \tag{2.11}$$

where $J = D_c\phi$ and $B = (D_c\rho)F_{r,s}$. $D_c\phi$ and $D_c\rho$ denotes the Jacobian matrices of leg angles and leg lengths with respect to virtual toe coordinates.

The primary goal of the embedding controller is to find appropriate hip controls to force the dynamics of Slimpod on (2.10) to the dynamics of simple SLIP on (2.2). Investigating all four equations for dynamics, one can see that this match can only be accomplished if K is determined as

$$K^* = [U^*(r), 0, M_\alpha^*] \tag{2.12}$$

where $U(r)$ is the desired radial potential law for the SLIP template. In order to stabilize the pitch angle, M_α^* is chosen as

$$M_\alpha^* = -K_\alpha\alpha - K_{\dot{\alpha}}\dot{\alpha} \tag{2.13}$$

If all legs are parallel, defined Jacobians will be noninvertible. In order to handle the singularities, premature lift-off, and enforcing control input limits, Ankaralı et al. [4] first assumed that the three legs were touching the ground and later included the partial touchdown and lift-off event situations. With the all-in-contact assumption, the radial component is excluded from the inversion, and consecutively, the inverse dynamics controller attempts to simultaneously satisfy angular template dynamics and pitch stabilization.

$$\boldsymbol{\tau}_{\psi,\alpha}(v) := \mathbf{J}_{\psi,\alpha}^T \left(\mathbf{J}_{\psi,\alpha}\mathbf{J}_{\psi,\alpha}^T\right)^{-1} \times \left([0 M_\alpha^*]^T - \mathbf{B}_{\psi,\alpha}\right) + \mathbf{J}_{\psi,\alpha}^{\perp} v \tag{2.14}$$

where $\mathbf{J}_{\psi,\alpha}^{\perp}$ spans the nullspace of $\mathbf{J}_{\psi,\alpha}$ and v covers the remaining freedom.

Real system applications are different from simulation studies. In a simulation, one can apply infinite torques to ensure stability; however, motors on the actual robots have limited torque ranges. That's why enforcing control input constraints increases the practical applicability of the designed controllers. Hip torques are constrained based on RHex's actuator torque-speed characteristics in this study. These constraints are defined for each leg as a torque space.

$$\mathcal{T}_{\text{lim}} := \{\boldsymbol{\tau} \mid \tau_{i,\min} \leq \tau_i \leq \tau_{i,\max}, 1 \leq i \leq 3\} \tag{2.15}$$

The intersection between (2.14) and (2.15) is utilized by the controller to compute hip torques.

$$\boldsymbol{\tau}_s = \underset{\boldsymbol{\tau}_{\psi,\alpha}(v) \in \mathcal{T}_{\text{lim}}}{\arg\min} \left\| \boldsymbol{\tau}_{\psi,\alpha}(v) \right\|. \tag{2.16}$$

For the particular situations in which those spaces in (2.14) and (2.15) do not intersect, preservation of angular momentum is prioritized by the alternative solution.

$$\boldsymbol{\tau}_\psi(\omega) := \mathbf{J}_\psi^T \left(\mathbf{J}_\psi \mathbf{J}_\psi^T \right)^{-1} \left(\begin{bmatrix} 0 & M_\alpha^* \end{bmatrix}^T - \mathbf{B}_\psi \right) + \mathbf{J}_\psi^\perp \omega \tag{2.17}$$

where $\mathbf{J}_\psi^\perp$ spans the nullspace of $\mathbf{J}_\psi$ and w covers the remaining freedom. Then the overall solution in (2.16) becomes

$$\boldsymbol{\tau}_s = \underset{\boldsymbol{\tau}_\psi(w) \in \mathcal{T}_{\text{lim}}}{\arg\min} \left\| \boldsymbol{\tau}_\psi(\omega) \right\|. \tag{2.18}$$

In summary, the assumption is that if the passive dynamics of the robot are appropriately chosen, they will approximately yield the desired result for the remaining coordinate in the virtual leg coordinates. Further details of the embedding can be found in Ankarali's studies [4, 42, 43].

2.2.3 Gait-Level Template-Based Dead-beat Control

In this study, the template-based approximate dead-beat control strategy introduced by Ankarali [4] is chosen as the baseline (non-adaptive) control strategy. The stiffness and damping parameters are chosen as all legs in parallel.

$$\begin{aligned} d &= \sum_{i=1}^{3} d_i \\ k &= \sum_{i=1}^{3} k_i \end{aligned} \tag{2.19}$$

According to control inputs from embedded SLIP $[\theta^*, r_{lo}^*, r_{td}^*]$, positions of toes (p_i) and target leg angles (ϕ_i^*) are given as

$$\begin{aligned} p_i &= [p_{iy} \;\; p_{iz}]^T = r^*[sin\theta^* \;\; cos\theta^*]^T + R(\alpha_t)a_i \\ \phi_i^* &= arccos(p_{iz}) - \alpha_t \end{aligned} \tag{2.20}$$

where p_i are hip positions with respect to virtual toe, a_i is position of the hips with respect to body.

Slimpod has a gait level, embedded spring-mass hopper template dead-beat controller to reach the desired apex state. The gait level behavior is summarized through the Poincaré section of its trajectories at each apex point since running is a nonlinear rhythmic motion. The touchdown angle, leg lengths at touchdown, and lift-off are adjusted to achieve the desired apex state. The hyperplane, called the Poincaré section, can be considered passing through apex points where vertical velocity is zero ($\dot{z} = 0$). The time-independent relation between two successive intersections can be defined as a Poincaré map interpreted as

$$f({}^aX_n) = {}^aX_{n+1} \tag{2.21}$$

Poincaré map and analytical approximations proposed by Ankarali et al. [32] allow us to define a discrete return map f, and approximate return map indicated as $\hat{f}$, respectively. Note that the discrete return map f has all the information about the physical system parameters accurately while $\hat{f}$ knows these parameters provided in the modeling step initially.

$$\hat{f}({}^aX_n, u_n, \hat{p}_n) = {}^a\hat{X}_{n+1} \tag{2.22}$$

The dead-beat controller relies on this approximate return map. In simple terms, the map takes the current state and the next touchdown angle and outputs the next state. Dead-beat controller seeks for touchdown angle and leg lengths at touchdown and liftoff ($u = [\theta, r_{td}, r_{lo}]$) with given current apex state (${}^aX_n = [\dot{y}_a, z_a]$) in order to achieve desired apex height and horizontal velocity (${}^aX^* = [\dot{y}_a{}^* z_a^*]$) in one step, through optimization on the map. Indeed, this analytical map includes the system parameters that should be regulated. A simple diagram for dead-beat stride control is given in Figure 2.7.

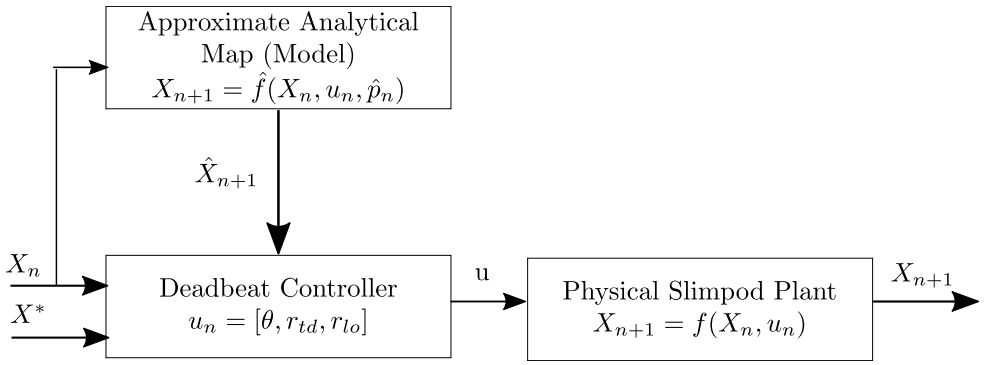

Figure 2.7: Gait-level open-loop dead-beat control scheme. The baseline controller that is vulnerable to the parameter discrepancies inside Approximate Analytical Map block

Previously designed gait-level dead-beat controllers do not consider the miscalibration or time-varying physical parameter case in the literature. On the contrary, they assume a perfect match for the physical quantities in the existing system and the approximate map, making their analytical calculations for control input accordingly. However, the physical parameters of the approximate analytical map inside the baseline controller may be uncertain and time-varying due to various reasons, such as variation in the environment or given task, effects of corrosion, and dust. These distortions and deformations cause extra difficulty in controlling the autonomous system. Also, the measurement error itself arises as an issue, especially for stiffness and damping values. Since the controller performance is closely related to the accuracy of the approximate predictive map, there can be many sources for discrepancies that cause the previous controllers to fail. Therefore, the controller should be adjusted according to the error in objective states in order to bring the system to a more practical side. So, an adaptive control scheme is introduced to deal with the miscalibration problem.

The adopted approximate dead-beat controller utilizes an approximate analytical map based on known system parameters to optimize the control input. If those parameters are miscalibrated, the dead-beat controller eventually will make inaccurate predictions and decisions. Moreover, even if there is a perfect match between all the parameters on the map and the actual system parameters, there will always be some discrepancies between the predictions of the map and actual system outputs due to the approximate nature of the analytical map.

Dynamics of pronking behavior of planar hexapod robot during stance phase (i.e., toe contact) [32] are non-integrable under the effect of gravity [47]. SLIP dynamics, and also slimpod dynamics, during the stance phase, are related to the restricted three-body problem [44]. Restricted three-body problems do not admit to a closed-form solution [47]. Therefore, unlike many other adaptive controller applications, [48], constructing a Lyapunov function and deriving adaptive laws for dynamics for pronking motion with a hexapod robot is not reasonably possible.

2.3 Adaptive Control of Slimpod

This section addresses the proposed indirect adaptive dead-beat controller embedded on the planar hexapedal system model Slimpod. Initially, the control scheme is proposed. Then, the work is followed by the error definition and the parameter update strategy.

Indirect adaptive control is chosen to implement because the existing dead-beat controller is affected by the objective system parameter and calculates the plant input. Figure 2.8 depicts the proposed controller scheme.

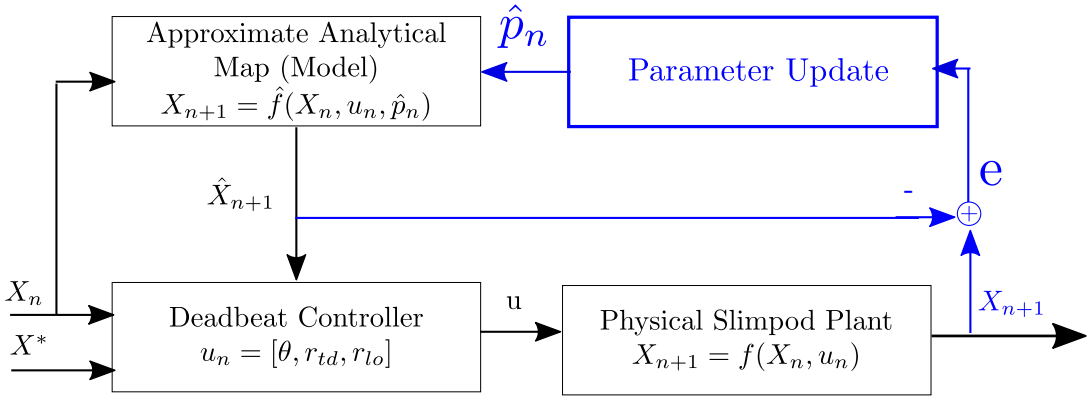

Figure 2.8: Gait-level adaptive dead-beat control scheme. The baseline controller that is vulnerable to the parameter discrepancies inside Approximate Analytical Map block. Blue parts are added according to the adaptive control strategy

As explained under the dynamics of the slimpod model subsection, the Poincaré map and analytical approximations proposed by Ankarali [32] allow us to define a discrete return map f, and approximate return map indicated as $\hat{f}$, respectively. Noting that the

approximate return map is constructed based on the inaccurate parameter estimates for the spring's stiffness, the prediction error is also given as

$$\begin{aligned} e &:= {}^{a}X_{n+1} - {}^{a}\hat{X}_{n+1} \\ &= f({}^{a}X_{n}, u_{n}) - \hat{f}({}^{a}X_{n}, u_{n}, \hat{p}_{n}) \end{aligned} \tag{2.23}$$

The adaptive controller should ensure this error approaches zero. That means the parameter update strategy should satisfy the condition that the model output and the actual output would be equal without disturbing the system's stability.

$$lim_{n\to\infty}({}^{a}X_{n+1} - {}^{a}\hat{X}_{n+1}) = 0 \tag{2.24}$$

To decide how to update the parameter, the impact of miscalibration on the system should be investigated. In this context, experiments are conducted with three different parameters. Figure 2.9 shows the resulting errors with respect to percentage deviations from "true" values. Red marked points represent the experiments when the system can reach a fixed point. The target scenario is to drive both the state error values e_z and their derivatives, $e_{\dot{y}}$, to zero.

As a result of these experiments, chosen parameter and corresponding apex state error need to have a linear relation to proving the system's stability through linearized system matrices [49, 50]. However, deviations in damping (Figure 2.9) have more of a quadratic behavior, and horizontal velocity state error $e_{\dot{y}}$ is negative for all deviation amounts regardless of its sign. Therefore, damping is not a proper candidate to adapt in order to reduce absolute error.

Pursuing a consistent notation with the previous study of Ankarali [4], a dimensionless formulation is adopted for the SLIP model and its variations. These dimensionless expressions eliminate inessential parameters and advance powerful ways of conducting the simulations as well as adapting the controllers to the real physical systems. The SLIP model's dimensionless flight and stance dynamics obtained from the Euler-Lagrange formulation are applied to the dynamics. Representation in dimensionless coordinates converts the physical stiffness $\bar{k}_s$ to the dimensionless version k_s with $k_s = \bar{k}_s(l_0/(mg))$ statement [4], l_0 being the leg length in rest. Therefore, physical individual leg stiffness $\bar{k}_s$ and physical body mass m affects the leg stiffness in dimensionless coordinates. Due to dimensionless coordinates inside the controller, stiffness and mass have similar but symmetric effects on state errors, as observed in

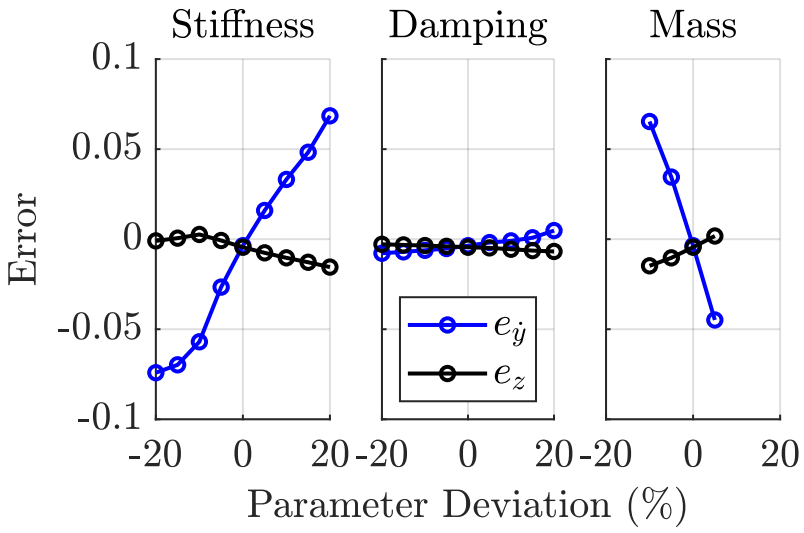

Figure 2.9: State errors with respect to percentage deformation

Figure 2.9. Suppose only the points regarding the ability to reach a fixed point are studied. In that case, it can be inferred that deviations in stiffness and mass values inside the approximate analytical map have a linear relation with the state's error value, while deviated damping's effect is nonlinear yet stable.

Considering that stiffness and mass have both linear effects, there are two candidates. Measurement of the mass value does not require disassembling all the legs and some specific equipment. Mass calibration is more accessible than the stiffness constant calibration, even if the robot has a payload. Hence, stiffness is chosen over mass to update. This analysis is the starting point for defining the parameter update strategy.

As mentioned before in the model dynamics and control section, because the restricted three-body problem does not admit to a closed-form solution, constructing a Lyapunov function and deriving adaptive laws for dynamics of pronking motion with a hexapod robot is not reasonably possible.

MIT rule is the key to remedying this obstacle of nonexistent closed-form solutions [51]. The state error is defined as in (2.23). The objective parameter k must be updated so that the loss function $V(k)$ in (2.25) is minimized.

$$V(k) = \frac{1}{2}e^2 \tag{2.25}$$

The MIT Rule makes $V(k)$ small by changing k in the direction of the negative gradient of $V(k)$.

$$\frac{\partial V(k)}{\partial k} = e\frac{\partial e}{\partial k}$$
$$\dot{k} = -\gamma\frac{\partial V(k)}{\partial k} = -\gamma e\frac{\partial e}{\partial k} \tag{2.26}$$

γ is a positive constant representing the adaptive gain in further steps. Also, the sensitivity derivative $\frac{\partial e}{\partial k}$ can be extracted from Figure 2.9.

In the literature, the corrective parameter adjustment strategy adopted from the MRAC method [52, 38] is very similar to how estimation methods such as Kalman filters use innovation on sensory measurement to perform state updates. Based on all these relations and inspiration from previous work on control related to the spring-mass hopper [4], which is the basis of the embedded template base controller, a parameter update strategy is proposed as

$$\hat{p}_{n+1} = \hat{p}_n - K_e * X_n * e \tag{2.27}$$

where K_e is a gain coefficient used to tune convergence of parameter values and regulate the oscillations; note that convergence behavior is strongly related to the adaptive gain K_e. The implementations will be based on this strategy. Stating the adaptive law leaves us questioning the system's stability with the adaptive controller. The stability issue will be discussed in the stability analysis subsection later.

2.4 Performance Analysis

This section presents simulation results for reference apex state tracking introduced in Section 2.3 using different percentage error conditions on two physical parameters, stiffness and damping, and the approximate map itself. Comparisons of adaptive and non-adaptive controller structures are provided to show that the proposed adaptive controller is more capable of producing stable and controllable pronking motion in the case of parametric miscalibration.

All simulations were run on MATLAB utilizing a hybrid dynamical simulation toolkit based on SimSect [45] previously verified on RHex [46], including necessary additions about slimpod model [4]. A variable-step, variable-order solver based on the numerical differentiation formulas of orders 1 to 5 is chosen as the solver.

Basically, two types of experiments are conducted according to two control schemes in Figure 2.7 and 2.8, and the simulation results are compared in terms of achieving the desired apex state and steady-state error for the given desired state. The simulations were run with various apex goals ${}^{a}X^{*}$ and percentage of miscalibration errors. Since the pronking is a periodic motion, any point can be tracked during the gait, i.e., the one-step prediction with the map could be made with any point. By choice, an apex return map is used, and the desired apex state is tracked.

The ranges for the dynamic parameters for the Slimpod model, which apply to a wide range of parameter combinations due to the dimensionless formulation inside the code, detailed in [42], were chosen to closely match the physical SensoRHex robot to ensure future applicability of the results to an experimental implementation and given as in Table 2.2. The states z and $\dot{y}$ are plotted from the dimensionless group in the following simulation results.

Table 2.2: Parameters of the Slimpod Model

Quantity	Symbol	Value	Unit
Body mass	m	9	kg
Leg stiffness	k	2000	N/m
Leg damping	d	12	Nm/s
Rest leg length	l	0.175	m
Desired height	z^{*}	0.195	m
Desired velocity	y^{*}	1.6	m/s

Primarily, the simulations are performed, as shown in Figure 2.10, when the system parameters fully match the controller parameters. This experiment shows that even if there is no miscalibration, the adaptive controller reduces the error between model output and actual plant output by updating the k value. This oscillation on the k value leads to fading oscillations on horizontal velocity tracking performance. However, system response seems to become more aggressive, caused by the high parameter update gain.

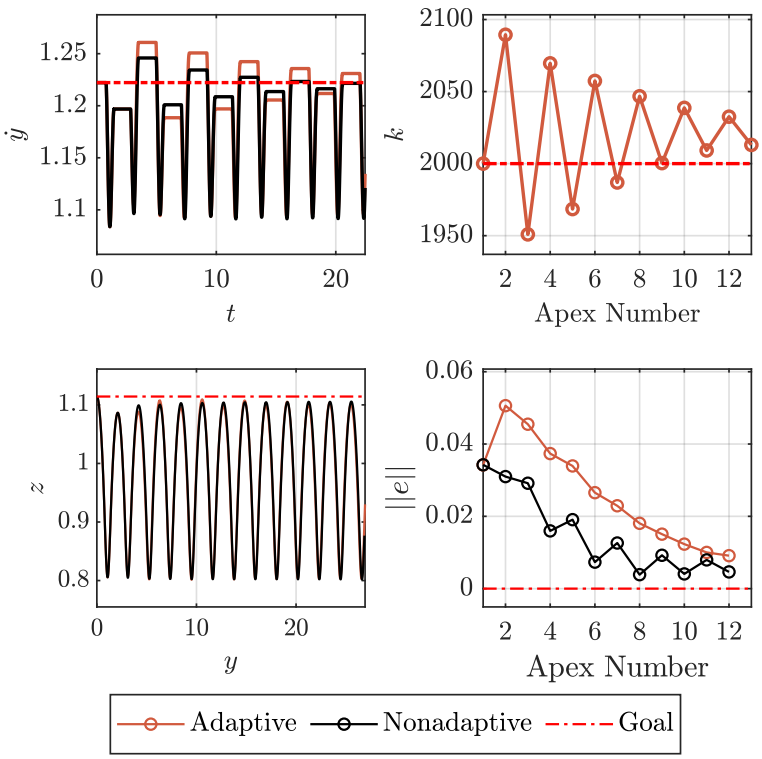

Figure 2.10: System response if there is no miscalibration. The plot on the top left corner represents the horizontal velocity with respect to time. Oh the top right corner, the evolution of the adapted parameter at each apex is depicted. The curve on the bottom left shows the trajectory of the hexapedal robot. And the magnitude of the apex state error is demonstrated for both adaptive and non adaptive cases.

To signify the effect of the proposed adaptation scheme, another experiment is conducted for the case of a miscalibrated stiffness value and depicted response in Figure 2.11. As expected, using the non-adaptive controller with miscalibrated parameters results in significant steady-state errors. In addition, after some time, unstable behavior causes the robot to turn upside down. On the other hand, the proposed adaptation scheme clearly reduces errors and promises improved performance.

The simulation in Figure 2.12 is repeated for the miscalibrated damping value and notes that stiffness adaptation leads the robot to recover the tracking properly. Just updating stiffness helps to compensate for the error caused by the miscalibrated damping.

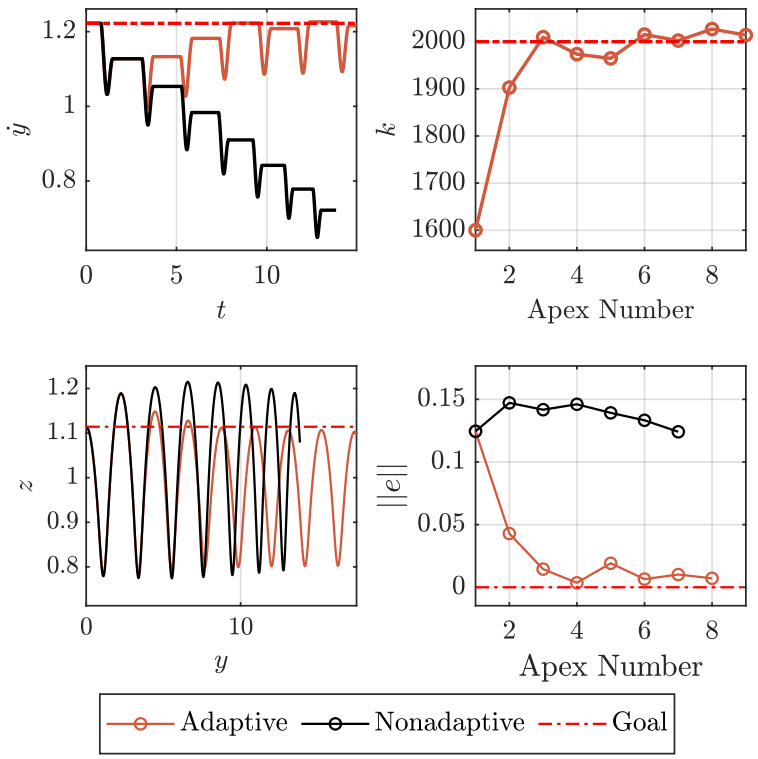

Figure 2.11: System response if stiffness on the map is 20% less than the system value

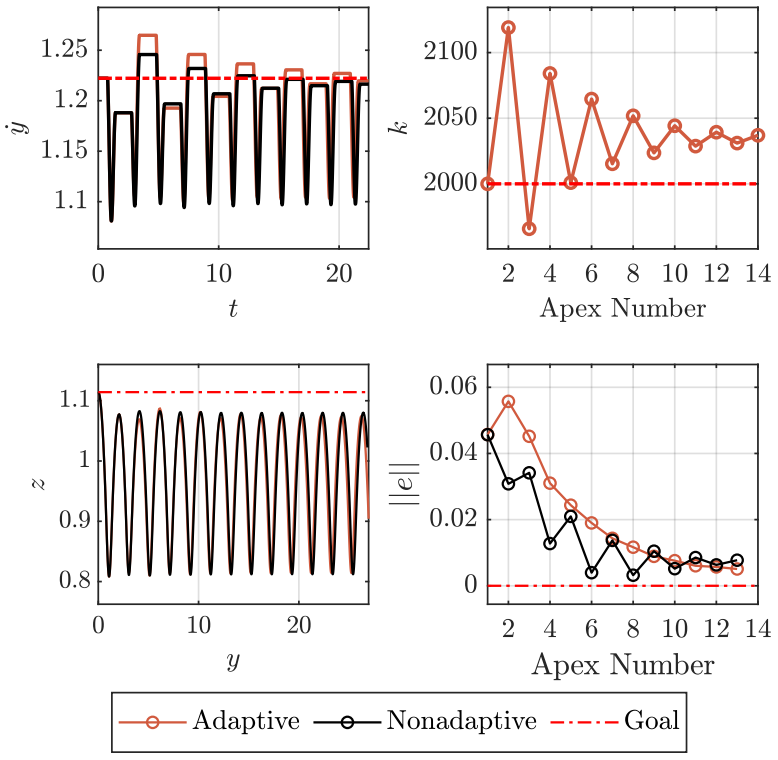

Figure 2.12: System response if viscous damping on the map is 10% less than the system value

In the following simulation, the performance for a multiple miscalibration problem is shown in Figure 2.13. According to these simulation results, when both stiffness and damping parameters are miscalibrated, updating stiffness helps decrease estimation error, and consequentially tracking performance is improved.

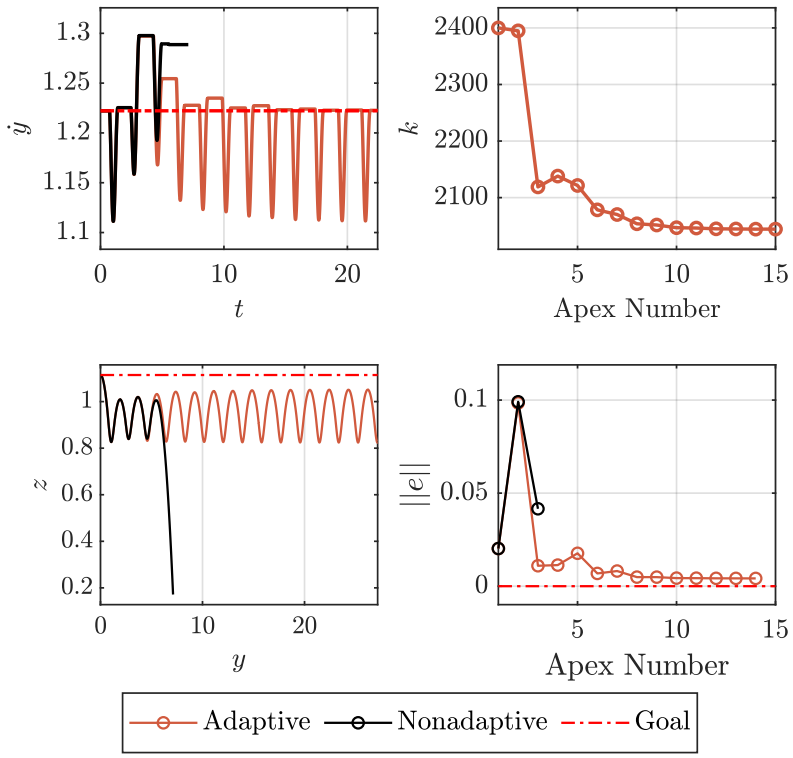

Figure 2.13: System response if stiffness on the map is 20% and viscous damping is -20% deviate from the system value

The final two simulations are devoted to assessing the controller's performance in the case of directly disturbing the output of the approximate map with a constant amount corresponding to 5% of the desired state values. Resulting responses will be as in Figure 2.14 and 2.15. According to these figures, if the approximate map's output deviates from its value by 5% higher or lower, states are drastically affected, making the system unstable. The addition of the adaptation compensates for the error and ensures the system's stability to some extent.

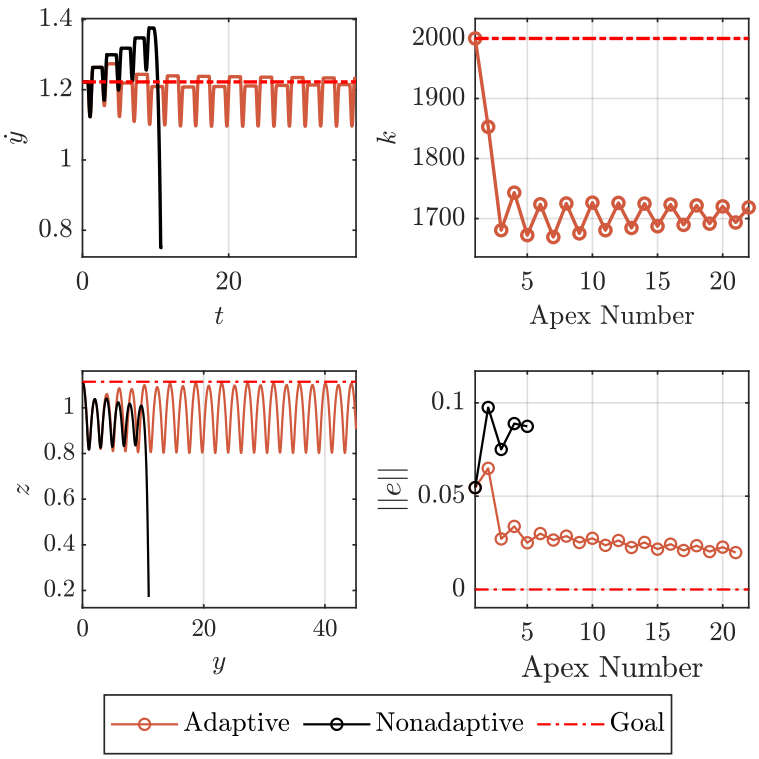

Figure 2.14: System response if the result of approximate map deviates -5%

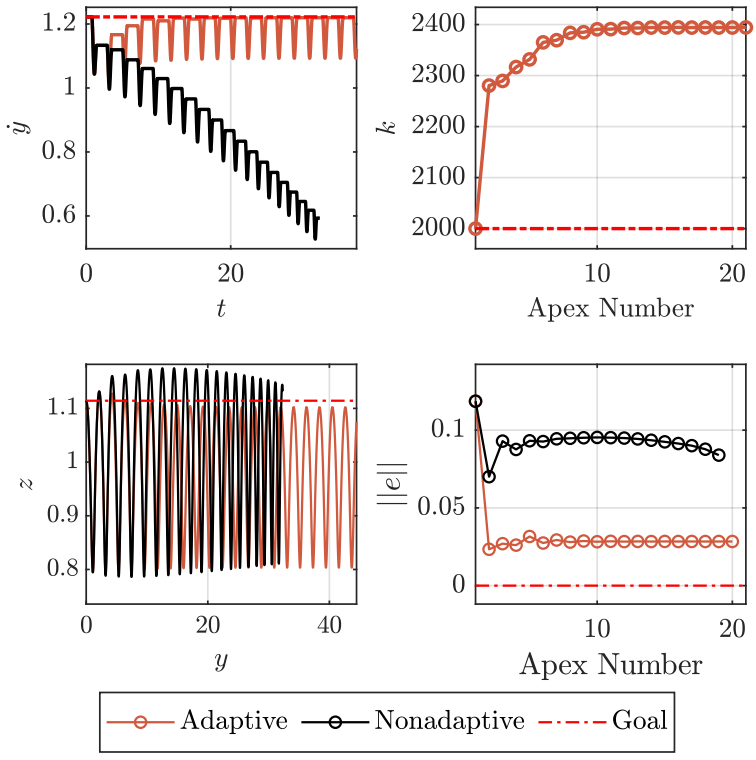

Figure 2.15: System response if the result of approximate map deviates +5%

In each case, the adaptive controller updates the stiffness value utilized on the approximate analytical map inside the dead-beat controller and improves tracking performance. Briefly, it may be concluded that the proposed adaptive control scheme to adjust stiffness value could confront the miscalibration problem in the planar hexapedal system.

In Figure 2.16, the parameter adaptation for stiffness is run together with the dead-beat controller when stiffness, mass, and damping are miscalibrated. In other words, the experiments are repeated in Figure 2.9 with adaptive stiffness. Comparing Figure 2.9 and 2.16, the systems with parameter adaptation are able to regulate a broader range of percentage errors, driving state errors closer to zero levels.

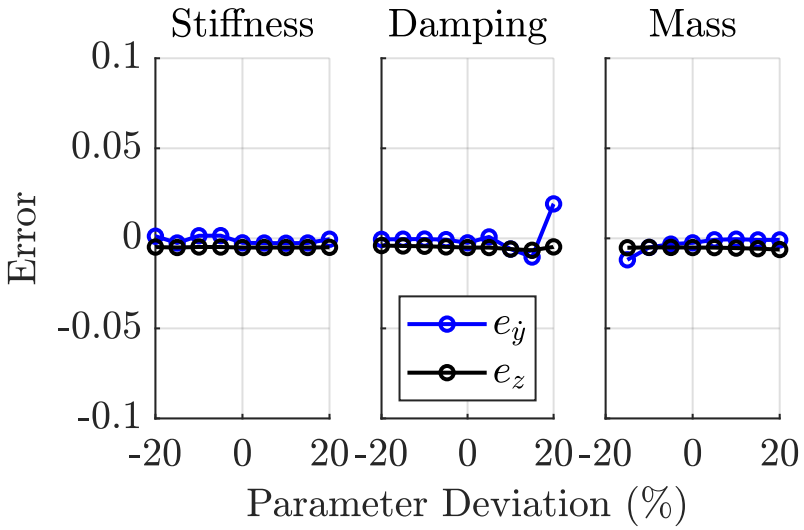

Figure 2.16: State errors with respect to percentage deformation after the involvement of parameter adaptation scheme for stiffness

2.5 Stability Analysis

A need arises for developing a way to show the system's stability with an adaptive controller since an adaptive law cannot be developed using standard design procedures in adaptive control [41]. Because running is a nonlinear, complex periodic motion, the Poincaré Map method is used in the analysis. This method intersects a

hyperplane with the periodic trajectory of a system with n-dimensional state space. This hyper-plane is called the Poincaré section, as illustrated in Figure 2.17. If the intersection is called X_k, Poincaré Map is defined as $f(X_k) = X_{k+1}$. Therefore, the relationship between two consecutive intersections can be defined as independent of time. Suppose the map is $f(X^*) = X^*$, i. e., trajectories intersect with Poincaré section on the same point; that point is called a fixed point. It is possible to comment on the system's stability by looking at the local stability at its fixed points [53, 49].

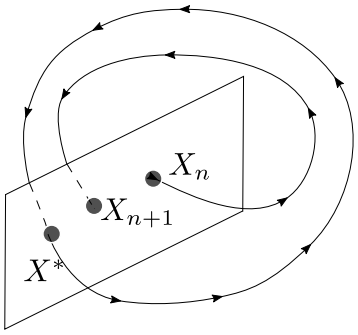

Figure 2.17: Example Poincaré Map

The embedded dead-beat controller assures the fixed points in a wide subspace. Fixed point subspace given in Table 2.3 is constructed by considering the gait level controllable region [4] in order to investigate stability. Fixed points are depicted in Figure 2.18.

Table 2.3: Chosen regions for fixed points

State	Physical Values
z^*	[0.1850,0.2750] m
y^*	[1.3096,1.9644] m/s

The system's stability resulting from the proposed parameter adjustment strategy based on its linearized system matrix is analyzed. Study in [49] supports that hybrid systems' behavior can be reduced to lower-dimensional subsystems near periodic orbits. Therefore, the stability of fixed points is decided by investigating eigenvalues of the numerically calculated linearized system matrix in (2.28) [54]. As known from all discrete systems, if all eigenvalues' magnitudes (i.e., $\|\lambda\|_{max}$) are smaller than 1,

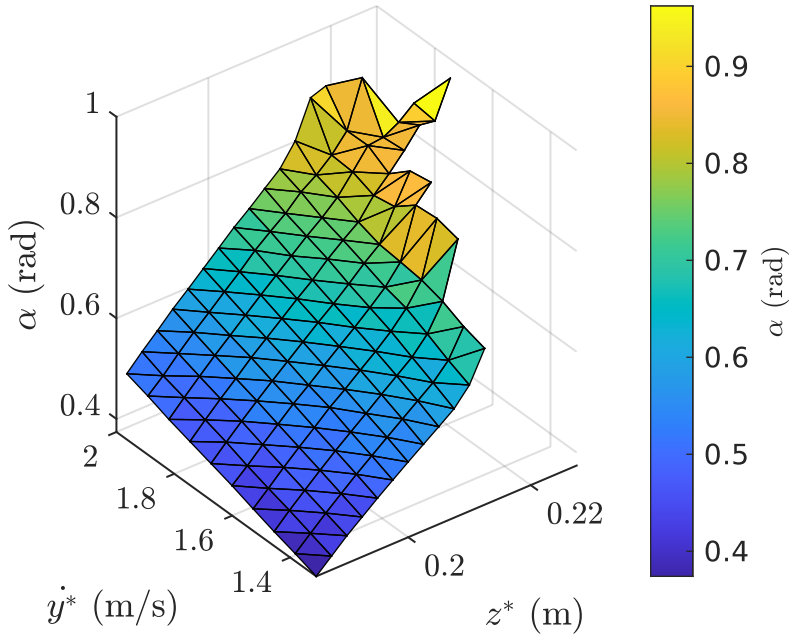

Figure 2.18: Variation of the fixed point subspace by height z, horizontal velocity $\dot{y}$ and the body angle α

the system is stable.

$$\hat{J} = \frac{\partial f}{\partial X_k}(X^*) \tag{2.28}$$

This linearized system matrix $\hat{J}$ relates the infinitesimal changes in apex state predictions $\partial\hat{X}$ to infinitesimal changes in states. The linearized system matrices are calculated for tracking different height and horizontal velocity pairs. Eigenvalues at those points can be observed in Figure 2.19. As depicted, all linearized system matrices have their $\|\lambda\|_{max}$ inside the unit circle, i.e., their magnitudes are smaller than one, which is the stability criterion for a discrete-time system.

Linearized numerical stability is also investigated for different leg stiffness and adaptive gain pairs. Figure 2.20 exhibits that the system with the adaptive controller remains stable in a wide range of stiffness and gain values.

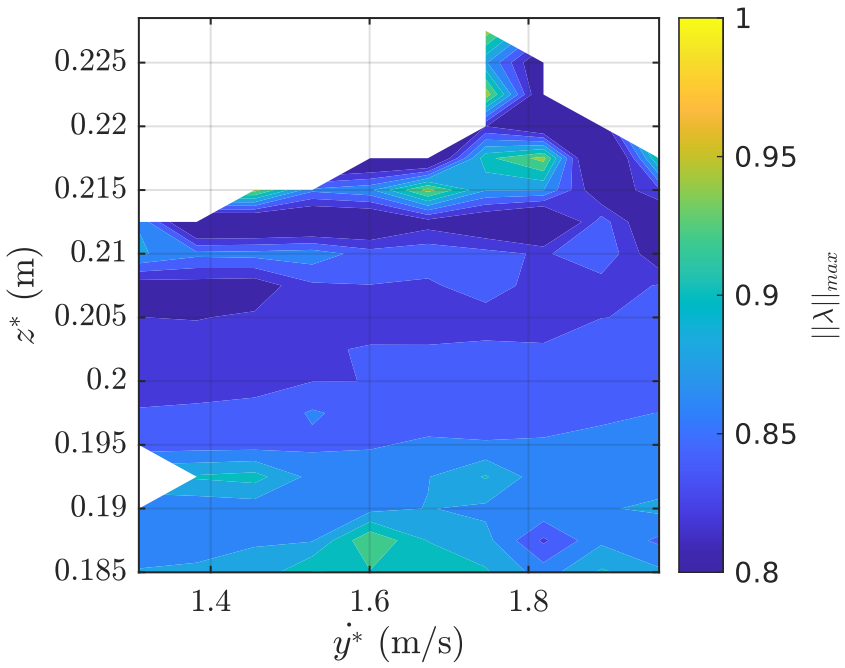

Figure 2.19: $\|\lambda\|_{max}$ of linearized system matrices calculated by using the fixed points given in Figure 2.18, $z^* \in [0.1850, 0.2750]m$ and $y^* \in [1.3096, 1.9644]m/s$ region

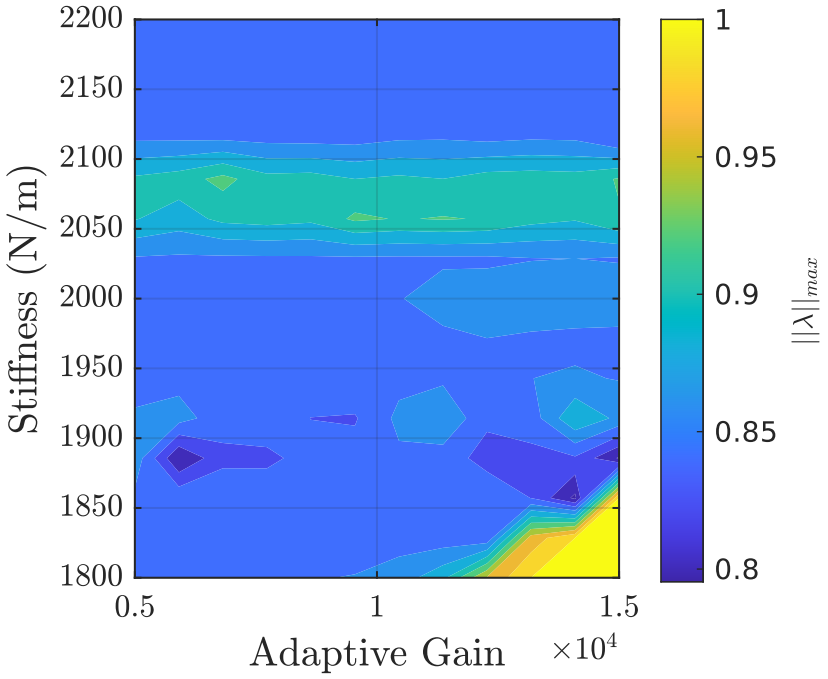

Figure 2.20: $\|\lambda\|_{max}$ of linearized system matrices calculated for the fixed point at $(0.195m, 1.6m/s)$ with different stiffness and gain pairs. White parts on the figures means that system cannot reach fixed points in this part of the region

2.6 Conclusion

This chapter proposes an online parameter adaptation scheme for an underactuated legged robot to improve the previously proposed control algorithms' tracking performance and robustness to parametric uncertainties. First, the apex state error-parameter relation is analyzed, and the parameter update rule is defined accordingly. Then the contribution of the additional level is demonstrated to the existing approximate dead-beat controller scheme. In succession, these steps admit to deploying a high-level layer to the current controller for the Slimpod model.

The formerly developed dead-beat controller aims to bring the apex states to desired values. Resultant control input entirely relies on the actual values of the parameters. Inevitably, miscalibrations in the measurements of physical instruments or parameter changes due to environmental effects will create a significant estimation error, so the dead-beat controller's decisions become inaccurate. A layer for adaptation is added to improve the tracking performance as much as possible. Simulation results for constructing a parameter adaptation structure show that adding the parameter update realizes this objective successfully. It reduces the error between model and actual plant outputs and substantially improves control performance relative to existing non-adaptive controllers.

This study extends the relationships between the apex state errors and the legs' physical parameters. The complexity of these relations mainly arises from the coupled effects of the parameters on gait behavior. How parameter calibration affects the apex states is inferred with an experimental investigation. Following the presented approach in this thesis, this experimental design procedure can now be generalized to different systems which are not suitable for analytical calculations of indirect adaptive control. In addition, local exponential stability of the closed-loop system under the adaptive rule is proven via computing linearized (numerical) Poincaré return map around the emergent fixed-points and checking the associated eigenvalues [49, 55].

Inspiration from the different animals' natural ability for varying gaits draws attention to legged systems. Employing controllers that mimic various motions allows robots to exploit different advantages, such as energy efficiency, considerable jumping heights,

or speeds. Through adaptation, robots can now take a further step to embark on an essential biotic characteristic to increase the feasibility of the existing theoretical controllers and practice in more complex places. In the future, it is intended to scale this process to different robotic platforms with unmeasurable (directly) inner or environmental parameters. Online adaptive parameter update rules can create a relatively simple yet powerful and practical approach for legged robotic platforms. As a result, the applicability of the theoretical controllers to the actual physical systems will be foreseen to rise significantly.

CHAPTER 3

STOCHASTIC ANALYSIS OF LEGGED LOCOMOTION BY UNSCENTED TRANSFORMATION

3.1 Introduction

Analytical models do not represent the real systems perfectly because some phenomena like impact and friction always bring discrepancies to the actual implementation. In exchange for their agility and capabilities, legged robots are much more vulnerable to the stochastic effects of unknown terrains than the other types of robotic systems. They eventually encounter different abnormalities, which might be slight differences in elevations (e.g., holes, rocks) or some contaminated surfaces that drastically affect friction constant and impact dynamics. Accordingly, controllers should take account of the external noises.

Legged locomotion involves, in general, periodic gaits such as walking, running, galloping, trotting, and pronking. This motion's stability analysis can be conducted based on the linearization of nonlinear dynamics around a periodic motion. Poincaré return map analysis is widely used to simplify the limit cycles of the periodic trajectories [56], as used and explained in Section 2.5 of the previous project in this thesis. This simplification ignores the stochastic effects of the external disturbances, and its results are only local, meaning that the stable periodic gait corresponds to an asymptotically and locally periodic motion around the fixed point. In addition, Poincaré methods handle the stability characteristics deterministically. Deterministic limit cycle stability analyses cover the stability properties of legged systems. However, they are usually difficult to apply and often fail to capture the stability properties of the systems without true limit cycle dynamics. In addition, they are technically incorrect

in the existence of stochastic disturbance. For example, the MARLO robot was able to walk in a laboratory environment; however, it took only few steps before falling down in outside tests due to a slight inclination of the sidewalk [57]. That is why the characterization of stochastic dynamics of walking should be accounted for during system identification and controller design to improve stability. This is an important yet understudied approach in the robotics community and is mainly addressed as a robust control problem [58, 57] rather than an analysis of stochastic dynamics.

3.1.1 Metastable Walking

Legged locomotion is characterized by the dynamic interactions between the feet and the contact surface. The term dynamic locomotion usually stands for an unbalanced walking cycle leading to a stable gait behavior. Underactuated legged systems leverage the underactuation to achieve dynamic locomotion. As the trade-off between stability and agility in the control theory, there exists an essential relationship between stability and maneuverability for legged systems [59]. Under disturbance, as in many stochastic dynamical systems, legged robots exhibit long-living, locally stable behaviors up to some point that cannot handle the external effects anymore. Once the disturbed system's states go into a region with a different attractor, the system behavior irreversibly adjusts to the new local dynamics. In simpler words, a legged robot can run for quite some time, but it will definitely fall down due to the external stochastic effects. Since they eventually leave the locally stable gait behavior, they cannot be considered "stable." On the other hand, they obviously operate for long periods of time, making calling them "unstable" wrong. A need arises for a new type of classification in the control theory. Metastability is a well-defined candidate for defining this phenomenon.

Metastable systems exhibit long-living behaviors that are guaranteed to transition to another success or failure state. The term metastability is, in fact, broadly used by other disciplines such as physics [60], chemistry [61] and electronics [62].

The systems with metastable states have another equilibrium state with less energy. A minor disturbance will cause a system in a metastable state to fall to a lower energy level, i.e., the stable equilibrium. The basic toy example to explain metastable systems

is depicted in Figure 3.1. In this simple example, a particle is swinging in a valley, and a sufficient disturbance can cause the particle to roll into another valley that has another local minimum.

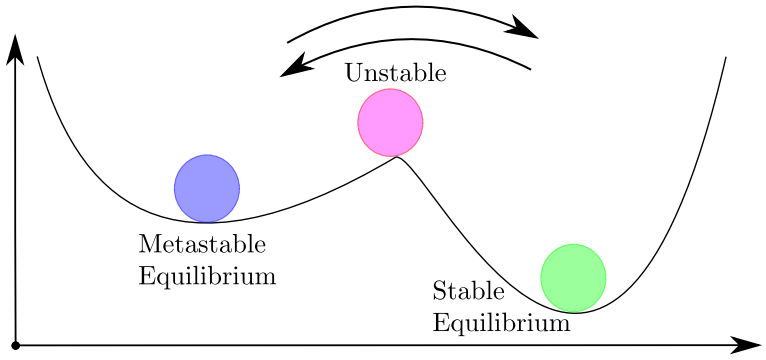

Figure 3.1: Toy example to explain metastable systems. The particle swings around the metastable equilibrium point unless an external disturbance leads the ball to fall into the other valley with an equilibrium point with less energy.

The legged counterpart of this toy example substitutes the metastable equilibrium with dynamic locomotion and the absolute minimum with falling to the ground. Hereby, walking is well-characterized as a metastable process. In the literature, Byl and Tedrake introduced the conceptual connection and utilized the metastability concept to quantify the stochastic stability of rimless wheel and compass-gait walking on rough terrain [63, 64]. They also utilized stochastic optimization to improve the overall stability of legged systems.

Byl and Tedrake's metastable limit cycle analysis methodology deals with walking systems by their closed-loop return map dynamics. For the first step, the return maps are represented as Markov chains. States of this Markov chain consist of mesh points of the state space for the particular legged system. State transition matrices of this Markov chain are numerically obtained from systematic experiments by integrating the system dynamics from each mesh point for different values of terrain slope. Finally, eigenvalues and eigenvectors of the state transition matrix are used to make deductions on stochastic stability.

Referring to Byl's studies [65], Benallegue and Laumond questioned the computational feasibility of the prior method for complex walking systems and proposed a

solution for the legged systems with high-dimensional states using the limit-cycle property of stable walking [66]. Actually, the biggest problem in the former approach is computational complexity. The complexity comes from two different aspects: simulation time and meshing methodology. Firstly, obtaining the state transition matrix requires numerous simulations in order to get the most accurate results. For instance, for a 5-link bipedal walking simulation, a one-step simulation takes up to 0.5 seconds in MATLAB. Running 10^6 simulations lasts more than five days, which can be reduced to less than one day by parallel programming, but it will still be too long to conduct enough experiments to build a smooth stochastic return map. In addition to the infeasibility of conducting thousands of experiments, if experimental setups are in the loop, another problem arises; physical damage. The more experiment is conducted, the more likely the system will fall down due to its metastable nature. Therefore, the system is more likely to get damaged. Secondly, for lower dimensional (1-DOF or 2-DOF) systems with one-dimensional noise, meshing the state space will be quite easy as slicing a range of values or meshing a surface. However, as the number of dimensions increases, meshing a cube or a 4D structure becomes more complex, even impossible. For example, for a 5-link bipedal robot, one must discretize all the state-space in ten dimensions along with the noise space. If the noise comes from only one source, noise space discretization is relatively trivial. Whereas, in the case of multiple noise sources, the former method fails to present an efficient way to analyze the dynamics. For 3D walking, the required degree of freedom increases quickly. Subsequently, Saglam and Byl introduced an improved meshing technique [67], but unfortunately, this improvement did not totally break the curse of dimensionality. On the other hand, Saglam and Byl's studies contributed to the previous breakthrough to handle legged systems as metastable systems by compilation of the methodology [68], new meshing technique [69] and optimal controller designs [70]. Now, this thesis proposes a methodology to further improve the metastable analysis by borrowing estimation methods from stochastic tools.

3.1.2 Stochastic Estimation Tools

Kalman filter [71] is a well-known estimation method and widely used in robotic platforms. Theoretically, Kalman Filters are called the linear quadratic estimator since

they minimize a quadratic function of estimation error for a linear dynamical system. As the name signifies, the Kalman filters use a linear estimator and fail to handle nonlinearities. Later, extended Kalman Filtering (EKF) is introduced to include the nonlinear effects by linearizing the problem about the estimated trajectory [72, 73]. EKF is used widely, so it has become almost the standard estimation method for the aerospace industry. As a matter of fact, linearized Kalman Filters' high capability was enough to be applied to navigation for the Apollo Project [74]. One of the methods in this thesis to compare the proposed estimation methodology will be a formulation based on linearization. EKF formulation is modified to involve nonadditive noise cases and adapt to this study's estimation scheme. Following Section 3.1.2.1 will present the estimation method based on the linearized model and its limitation. The alternative method with unscented transformation to overcome this limitation is introduced in Section 3.2.2.

3.1.2.1 Estimation Over Linearized Model

The Extended Kalman Filter (EKF) extends the classic Kalman Filter for nonlinear systems where nonlinearity is approximated using the first or second-order derivative. It tries to capture nonlinearity using Taylor expansion around a local point. By the same principle with EKF, a covariance prediction can be formulated either using a linearized analytical model of the system or a numerically linearized version. For the legged systems that this study is based on, linearization should be conducted numerically, noting that it is not efficient. However, it is obligatory because there are no explicit linearized analytical models for these highly nonlinear hybrid dynamical legged systems.

Formulation steps of the linearization-based estimation method for a generalized case of nonadditive noise begin with the same nonlinear system description in Markov chains and unscented transformation later. The noise is applied at the impact, and its effect on the output state $\mathbf{x}_{k+1}$ is not explicitly known.

$$\mathbf{x}_{k+1} = \mathbf{f}(\mathbf{x}_k, \mathbf{w}_k), \quad \mathbf{w}_k \sim \mathcal{N}(0, \mathbf{Q}_k) \tag{3.1}$$

At the model forecast step (3.6), the forecast value of $\mathbf{x}_{k+1}$ (indicated by $\mathbf{x}^f_{k+1}$) is produced by propagating the initial optimal estimate $\mathbf{x}^a_k$ through the nonlinear system

and used to compute the mean and covariance of the forecast value of $\mathbf{x}_{k+1}$. In this thesis, the initial optimal estimate is defined as the initial condition, more clearly, the midpoint of the range defining each Markov state.

$$\mathbf{x}_k^a := \mathbf{x}_k \tag{3.2}$$

The predictable part of $\mathbf{x}_k$, i.e., the forecast value, is given by

$$\mathbf{x}_{k+1}^f = \mathbb{E}[\mathbf{x}_k] = \mathbb{E}[\mathbf{f}(\mathbf{x}_k, \mathbf{w}_k)] \tag{3.3}$$

The next step is expanding system description $\mathbf{f}(.)$ in Taylor Series about the optimal estimate $\mathbf{x}^a$ as follows,

$$\mathbf{f}(\mathbf{x}_k, \mathbf{w}_k) \approx \mathbf{f}(\mathbf{x}_k^a, 0) + \mathbf{f}_{\mathbf{x}_k}(\mathbf{x}_k^a, 0)(\mathbf{x}_k - \mathbf{x}_k^a) + \mathbf{f}_{\mathbf{w}_k}(\mathbf{x}_k^a, 0)\mathbf{w}_k + H.O.T. \tag{3.4}$$

where $\mathbf{f}_{\mathbf{x}_k} = \frac{\partial \mathbf{f}}{\partial \mathbf{x}_k}$, $\mathbf{f}_{\mathbf{w}_k} = \frac{\partial \mathbf{f}}{\partial \mathbf{w}_k}$. Higher order terms are ignored and the partial derivatives with respect to state and noise should be calculated by numerical methods. Then forecast value is calculated by substituting (3.4) into (3.3),

$$\begin{aligned}
\mathbf{x}_{k+1}^f &\approx \mathbb{E}\left[\mathbf{f}(\mathbf{x}_k^a, 0) + \mathbf{f}_{\mathbf{x}_k}(\mathbf{x}_k^a, 0)\underbrace{(\mathbf{x}_k - \mathbf{x}_k^a)}_{e_k} + \mathbf{f}_{\mathbf{w}_k}(\mathbf{x}_k^a, 0)\mathbf{w}_k\right] \\
\mathbf{x}_{k+1}^f &\approx \mathbf{f}(\mathbf{x}_k^a, 0) + \mathbf{f}_{\mathbf{x}_k}(\mathbf{x}_k^a, 0)\underbrace{\mathbb{E}[e_k]}_{0} + \mathbf{f}_{\mathbf{w}_k}(\mathbf{x}_k^a, 0)\underbrace{\mathbb{E}[\mathbf{w}_k]}_{0} \\
\mathbf{x}_{k+1}^f &\approx \mathbf{f}(\mathbf{x}_k^a, 0)
\end{aligned} \tag{3.5}$$

The forecast error equation becomes as follows:

$$\begin{aligned}
e_{k+1}^f &= \mathbf{x}_{k+1} - \mathbf{x}_{k+1}^f \\
&= \mathbf{f}(\mathbf{x}_k, \mathbf{w}_k) - \mathbf{f}(\mathbf{x}_k^a, 0) \\
&\approx \mathbf{f}_{\mathbf{x}_k}(\mathbf{x}_k^a, 0)e_k + \mathbf{f}_{\mathbf{w}_k}(\mathbf{x}_k^a, 0)\mathbf{w}_k
\end{aligned} \tag{3.6}$$

The forecast error covariance is calculated as

$$\begin{aligned}
\mathbf{P}_{k+1}^f &= \mathbb{E}[e_{k+1}^f (e_{k+1}^f)^T] \\
&= \mathbb{E}\left[(\mathbf{f}_{\mathbf{x}_k}(\mathbf{x}_k^a, 0)e_k + \mathbf{f}_{\mathbf{w}_k}(\mathbf{x}_k^a, 0)\mathbf{w}_k)(\mathbf{f}_{\mathbf{x}_k}(\mathbf{x}_k^a, 0)e_k + \mathbf{f}_{\mathbf{w}_k}(\mathbf{x}_k^a, 0)\mathbf{w}_k)^T\right] \\
&= \mathbf{f}_{\mathbf{x}_k}(\mathbf{x}_k^a, 0)\mathbb{E}[e_k(e_k)^T](\mathbf{f}_{\mathbf{x}_k}(\mathbf{x}_k^a, 0))^T + \mathbf{f}_{\mathbf{w}_k}(\mathbf{x}_k^a, 0)\mathbb{E}[\mathbf{w}_k(\mathbf{w}_k)^T](\mathbf{f}_{\mathbf{w}_k}(\mathbf{x}_k^a, 0))^T \\
&= \mathbf{f}_{\mathbf{x}_k}(\mathbf{x}_k^a, 0)\mathbf{P}_k(\mathbf{f}_{\mathbf{x}_k}(\mathbf{x}_k^a, 0))^T + \mathbf{f}_{\mathbf{w}_k}(\mathbf{x}_k^a, 0)\mathbf{Q}_k(\mathbf{f}_{\mathbf{w}_k}(\mathbf{x}_k^a, 0))^T
\end{aligned} \tag{3.7}$$

The estimation in this study considers a one-step calculation of successive states. One can reformulate the output mean and variance equations for the one-step calculation

instead of a recursive calculation of forecast and error values with time points such as k and $k+1$. Since the initial value for $\mathbf{x}_k$ $(= \mathbf{x}_0)$ is deterministically known, $\mathbf{P}_0$ will be 0, and estimated mean and variances for the next step can be computed as

$$\begin{aligned} \boldsymbol{\mu}_1 &= \mathbf{x}_1^f = \mathbf{f}(\mathbf{x}_0^a, 0), \\ \mathbf{P}_1^f &= \mathbf{f}_{\mathbf{w}}(\mathbf{x}_0^a, 0)\mathbf{Q}_0 \left(\mathbf{f}_{\mathbf{w}}(\mathbf{x}_0^a, 0)\right)^T \end{aligned} \tag{3.8}$$

Using the estimated output mean $\boldsymbol{\mu}_1$ and output variance $\mathbf{P}_1^f$, the estimated normal distribution can be constructed as $\mathbf{X}_1 \sim \mathcal{N}\left(\mu_1, \mathbf{P}_1^f\right)$.

3.1.2.2 Unscented Transformation

The nonlinear extensions of the Kalman Filter consist of nonlinear propagation of probability densities. The sample-and-propagate methods can be generalized as perturbation methods, using samples as initial values, which are perturbations from the mean trajectory. The continuous state domain is actually transformed into a discrete set of points by sampling. EKF fails to conduct nonlinear propagation because of its basis for linearization and partial derivatives instead of propagation [75].

Unscented Kalman Filter is a special case of sigma point filters introduced to improve filtering performance. Unscented transformation [76, 77] is a powerful tool to estimate the statistics of a random variable that undergoes a nonlinear transformation [78] and is used in many applications ranging from sensor fusion for state estimation [79] to an unscented Kalman observer [80]. Moreover, in recent studies, Sieberg et al. combined an artificial neural network with confidence level adjustment and presented a hybrid state estimation structure using unscented transformation [81]. This strong stochastic process tool is borrowed to make informed choices of initial conditions for the stochastic analysis experiments in this thesis. Eliminating computational complexity, the mean first passage time metric is able to be utilized to characterize the stochastic stability of high-dimensional underactuated nonlinear systems. Additionally, unlike the previous studies [64, 67], estimation with unscented transformation allows us to deal with multiple sources of uncertainties on higher dimensional systems. Even though unscented transformation helps estimate the output distribution for nonlinear systems, nonlinear transformation does not help with the higher-order moments. The higher-order moments of the estimation distributions are not tracked.

Infinitely many possible distributions share the same mean and variances having distinct higher-order moments. Sample-and-propagate methods can capture the exact and unique solution if the transformation results are linear. In the existence of nonlinearities, the exact solution cannot be reached. Because there is no unique solution, estimation performance assessments for this type of estimator are also tricky. Therefore, tuning the parameters of the filters to reach a better estimation can be done by comparing the estimation with the results of the Monte Carlo experiments. This tuning procedure gives the proper parameters only for this particular nonlinear system. Tuning the weights in the unscented transformation-based estimation affects the estimated results, so one should be careful when choosing those parameters. This thesis assumes that the output distribution is a Gaussian, so the output distribution is built as a Gaussian with estimated mean and variances. The estimated mean and variances are compared with the results from Monte Carlo experiments.

Monte Carlo analysis is a common method to estimate a probability distribution's progress over time. The method is based on selecting a sufficient number of representative random samples and simulating them with the dynamical system model. The result is the probability density as an estimate of output distribution. This experimental output distribution can be visualized as a histogram and used for assessing the estimation error. As the number of samples increases, the accuracy of the output distribution will increase.

This study proposes a more efficient estimation method for metastable system properties based on unscented transformation; therefore, there will be no need for conducting many experiments through Monte Carlo sampling. The proposed method is implemented to examine the stochastic stability of a one-dimensional hopper and an idealized 5-link biped simulation with a hybrid zero dynamics controller under disturbance. The one-dimensional hopper states an example of a simple legged system. After observing the satisfactory estimation results, the methodology is extended to a higher dimensional system. This 5-link walker model is inclusive for robot walkers due to its nonlinear, underactuated, and hybrid nature. In addition, the proposed method is compared with the numerical linearization-based method, formulated in Section 3.1.2.1 for both systems.

3.2 Methodology

This section presents the methodology behind the metastable analysis of legged locomotion. This stochastic analysis framework is mainly adopted from [64]. The storyline is as follows. First the Markov chain representation for the system is introduced. Secondly the proposed estimation scheme based on unscented transformation is explained. Lastly, the stochastic stability metric mean first passage time is defined to utilize stochastic analysis in controller comparison.

3.2.1 Markov Chain Representation

Markov chains are defined as stochastic models that describe a sequence of possible events whose probability only depends on the previous event [82]. Markov processes have many applications as building statistical models for the real systems in engineering, biology, economy, social sciences, etc. These applications vary from modeling airport queues to stock market predictions. There also exist Markovian representations for storage models of dams, animal populations, and employment structures of companies [83].

There are two conditions for a matrix to be a state transition matrix of the Markov chain. First, the state transition matrix to a Markov chain needs to be a stochastic matrix, i.e., rows of a stochastic matrix are probability vectors. That means, if the transition matrix is built as in (3.9), all rows should add up to 1.

$$\mathbf{T} = \begin{matrix} \text{TO} \\ \left[\begin{array}{c} \\ \\ \end{array}\right] \end{matrix} \begin{matrix} \\ \text{FROM} \end{matrix} \tag{3.9}$$

$$X_1 = X_0\mathbf{T}$$

where X_0 is the initial state distribution and X_1 is the state distribution at step 1. Secondly, the matrix has to be regular. That means $\mathbf{T}^n$ $(n > 1)$ has only nonzero positive entries. The second condition is trivial to meet, because a stochastic matrix is also a regular matrix. If $\mathbf{T}$ is a regular matrix, there will be a $\mathbf{T}^n$ where $X_0\mathbf{T}^n = \bar{X}$, where $\bar{X}$ is a stable distribution state. Then, $\mathbf{T}, \mathbf{T}^2, \mathbf{T}^3, ..$ becomes a regular Markov chain.

Absorbing Markov chains are defined as Markov chains with absorbing states to which it is possible to go from any transient states in a finite number of steps. The standard form of the transition matrix of an absorbing Markov chain is as in (3.10).

$$\begin{aligned} \mathbf{T} &= \begin{bmatrix} I & 0 \\ S & R \end{bmatrix} \\ X_1 &= X_0\mathbf{T} \end{aligned} \tag{3.10}$$

Stable distribution matrix of an absorbing Markov chain with only one absorbing state is expected to be the first unit vector $\bar{X} = \begin{bmatrix} 1 & 0 & 0 & ... \end{bmatrix}$. That means that regardless of the initial distribution, all of the population is guaranteed to go to the absorbing state.

$$\begin{aligned} \bar{X}\mathbf{T} &= \bar{X} \\ \bar{X}(I - \mathbf{T}) &= 0 \\ e_1(\lambda_1 I - \mathbf{T}) &= 0 \\ \lambda_1 = 1 \, , \, e_1 &= \bar{X} \end{aligned} \tag{3.11}$$

According to the calculations in (3.11), absorbing Markov chains has one eigenvalue at $\lambda_1 = 1$ and stable distribution matrix of that Markov chain will be the first left eigenvector e_1. e_1 is the first unit vector, which means this system will eventually stop at the first (absorbing) state. If there is more than one absorbing state, the state distribution vector will be a state distribution among those absorbing states depending on the initial condition. In this thesis, all configurations representing the failure of motion or leaving a predetermined region are compiled to one absorbing state.

The eigenvector associated with the largest megnitude eigenvalue (first eigenvector) of the transpose of an absorbing Markov chain transition matrix, $\mathbf{T}$, represents the stable distribution matrix of the chain, $\bar{X}$. The second largest magnitude eigenvalue of the matrix $\mathbf{T}$ corresponds to the first eigenvalue of the $\bar{\mathbf{T}}$.

$$\mathbf{T} = \begin{bmatrix} 1 & 0 \\ r & \bar{\mathbf{T}} \end{bmatrix} \tag{3.12}$$

The eigenvector associated with the first (largest) eigenvector on $\bar{\mathbf{T}}$ describes the long-living (metastable) distribution of the state.

3.2.1.1 Toy example: Simple Absorbing Markov Chain

A simple example is inserted to visualize the basics of absorbing Markov chains. This Markov chain represents a person's travel probabilities between 5 cities; A, B, C, and D. Figure 3.2 represents the visualization of the Markov chain, its state transition matrix in (3.13) as a colored surface and its eigenvalue plot. Throughout the thesis, state transition matrices are visualized as in Figure 3.2 to observe the stochastic return maps.

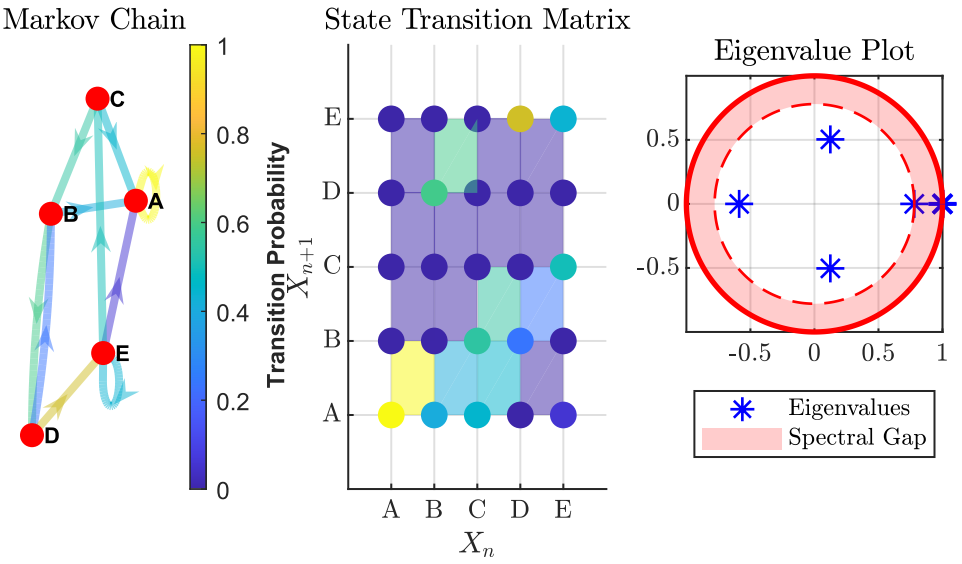

Figure 3.2: Toy example to explain absorbing Markov chain representing the transition between five states; A, B, C, D

$$\mathbf{T} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0.41 & 0 & 0 & 0.59 & 0 \\ 0.45 & 0.55 & 0 & 0 & 0 \\ 0 & 0.24 & 0 & 0 & 0.76 \\ 0.06 & 0 & 0.5 & 0 & 0.44 \end{bmatrix} \tag{3.13}$$

To find out the probability distribution between possible next cities, state transition matrix $\mathbf{T}'$ needs to be multiplied by the state vector X_n as in $X_{n+1} = X_n\mathbf{T}$. For example, if the person is known to be in city B, the state vector will be $X_0 = [0, 1, 0, 0, 0]$. Therefore, at the next time step, the state probability vector will be

$X_1 = [0.41, 0, 0, 0.59, 0]$. The person will be in city A with a probability of 0.41 and city D with a probability of 0.59. If the person keeps moving between states, the next state vectors will be as follows,

$$
\begin{aligned}
X_2 &= \begin{bmatrix} 0.4100 & 0.1416 & 0.0000 & 0.0000 & 0.4484 \end{bmatrix} \\
X_3 &= \begin{bmatrix} 0.4950 & 0.0000 & 0.2242 & 0.0835 & 0.1973 \end{bmatrix} \\
X_4 &= \begin{bmatrix} 0.6077 & 0.1434 & 0.0986 & 0.0000 & 0.1503 \end{bmatrix} \\
&\ldots \\
X_{10} &= \begin{bmatrix} 0.9172 & 0.0204 & 0.0182 & 0.0128 & 0.0315 \end{bmatrix} \\
&\ldots \\
X_{30} &= \begin{bmatrix} 0.9994 & 0.0001 & 0.0001 & 0.0001 & 0.0002 \end{bmatrix} \\
&\ldots \\
X_{40} &= \begin{bmatrix} 1.0000 & 0.0000 & 0.0000 & 0.0000 & 0.0000 \end{bmatrix}
\end{aligned}
\tag{3.14}
$$

At the 30^{th} time step, the person is in city A by more than 99.99% probability. As the time goes to infinity, the probability of traveling to city A becomes 1, i.e., state A represents the absorbing state. The person cannot avoid arriving at city A no matter what his/her initial spot is.

3.2.1.2 System as an Absorbing Markov Chain

In metastable limit cycle analysis, the discrete-time system dynamics are represented as a Markov process. Walking dynamics of legged systems demonstrate a hybrid behavior having discrete impact events that interrupt the continuous flight dynamics. The Poincare map resulting from this natural time-discretization is the basis for the return map used to build the corresponding Markov chain.

The apex to apex dynamics should be represented as a discrete system for the stochastic stability analysis.

$$\mathbf{x}_{k+1} = \mathbf{f}(\mathbf{x}_k, \mathbf{w}_k) \tag{3.15}$$

where $\mathbf{x}_k$ and $\mathbf{w}_k$ represent the states and noises at time step k respectively. The noise values are drawn from a Gaussian distribution with zero mean and covariance of R_ω.

The state transition dynamics can be approximately represented as a finite-state Markov process if the state space is discretized into a finite set of states. State discretization is required for the computation of a finite-state Markov chain. The state space is divided into N pieces and assigned to Markov states. The state transition matrix $\mathbf{T}_{N\times N}$ of this Markov chain collects the transition probabilities between the N predefined states. The probability of transition from state i to state j is,

$$\mathbf{T}_{ij} = \mathbb{P}(\mathbf{x}_{k+1} = s_j | \mathbf{x}_k = s_i). \tag{3.16}$$

An absorbing Markov chain is a Markov chain with at least one state that is impossible to leave. In legged locomotion, the absorbing state can be considered as collecting all configurations where the robot falls down [64]. Besides, a particular region desired to operate can be specified, and the other configurations can be taken as belonging to the absorbing state. Assuming s_1 is the absorbing state, the following can be stated.

$$\mathbf{T}_{11} = 1 \quad \text{and} \quad \mathbf{T}_{1j} = 0 \quad \text{for} \quad j \neq 1. \tag{3.17}$$

Absorbing Markov chains has one eigenvalue at $\lambda_1 = 1$ and the stable distribution matrix of that Markov chain will be the first left eigenvector of $\mathbf{T}$ in (3.18) which is the first unit vector, which means this system will eventually stop at the first (absorbing) state. The second-largest magnitude eigenvalue of the matrix $\mathbf{T}$ corresponds to the largest magnitude eigenvalue of the $\bar{\mathbf{T}}$ and is related to the metastable characteristic of the system. The eigenvector associated with the largest magnitude eigenvalue of $\bar{\mathbf{T}}$ describes the long-living (metastable) distribution of the state.

$$\mathbf{T} = \begin{bmatrix} \mathbf{1}_{1\times 1} & 0_{1\times N-1} \\ \mathbf{T}_{j1_{N-1\times 1}} & \bar{\mathbf{T}}_{N-1\times N-1} \end{bmatrix} \tag{3.18}$$

The state transition matrix of the Markov chain can be built via Monte Carlo experiments. According to the law of large numbers, the average of the results obtained from a large number of trials should be close to the expected value and tends to become closer to the expected value as more trials are performed [84]. However, building the state transition matrix with Monte Carlo experiments requires too many trials, which is highly inconvenient for the complex legged systems. Thus, this study prefers to choose sigma points using prior knowledge of noise characteristics and simulate the system accordingly. As a result, the unscented transform concept results in a more efficient estimation of state transition matrices.

3.2.2 Unscented Transformation

This section explains the formulation of the unscented transformation method. The fundamental motivation behind using unscented transform is that approximating a probability distribution is easier than approximating an arbitrary nonlinear function [76]. Instead of approximating the system equations by linearization, sigma points are calculated and used in the unscented transformation to directly approximate the output probability density functions. The assumption under the probability distribution estimation is to have a Gaussian noise and expect the output distributions to be Gaussian. The central limit theorem states that the sampling distribution approaches a normal distribution as the sample size increases [85]. That is why it can be assumed that, under the exposure of multiple noise sources, the output distributions will coincide with a Gaussian distribution. It is also observed in later sections. Even if the output distribution is not Gaussian, the estimation can approximately capture its mean and variance.

The formulation steps are similar to the Unscented Kalman Filters [76, 77]. First of all, formulation for the generalized case of nonadditive noise requires an augmented state definition $\mathbf{x}_k^a$ with system states $\mathbf{x}_k$ and zero mean noises $\mathbf{w}_k$,

$$\mathbf{x}_k^a = \begin{bmatrix} \mathbf{x}_k^T & \mathbf{w}_k^T \end{bmatrix}^T . \tag{3.19}$$

Previously known nonlinear system dynamics $\mathbf{f}$ and the noise variance characteristics $\mathbf{P}_k$ are,

$$\begin{aligned} \mathbf{x}_{k+1} &= \mathbf{f}(\mathbf{x}_k^a), \\ \mathbf{P}_k &= \begin{bmatrix} \varepsilon & 0 \\ 0 & R_w \end{bmatrix}, \end{aligned} \tag{3.20}$$

where $\mathbf{P}_k$ contains the known variances as diagonal entries, and R_w represents the noise variances. Since the initial states $\mathbf{x}_k$ are deterministic in this study, their variance will be zero; however, for computational purposes, their variance is specified as a very small value ε. If the state variance is taken as zero, that will cause a problem in the matrix square root step.

Sigma points represent the chosen initial conditions so that the output of the nonlinear system to these initial conditions will provide the information related to output

distribution. The sigma point set $\mathbf{X}_k$ (3.21) contains $2n+1$ sigma points $\mathbf{x}_k^j$ and their associated weights $\mathbf{W}^j$ so that their mean will be $\mathbf{x}_k^a$ and variance $\mathbf{P}_k$, where n is the dimension of augmented state.

$$\begin{aligned}
\mathbf{X}_k &= \{(\mathbf{x}_k^j, \mathbf{W}^j) | j = 0 \dots 2n\} \\
\mathbf{x}_k^0 &= \mathbf{x}_k^a, \quad -1 < \mathbf{W}^0 < 1 \\
\mathbf{x}_k^j &= \mathbf{x}_k^a + A_j, \quad j = 1 \dots n \\
\mathbf{x}_k^j &= \mathbf{x}_k^a - A_j, \quad j = n+1 \dots 2n \\
\mathbf{W}^j &= \frac{1 - \mathbf{W}^0}{2n}, \quad j = 0 \dots 2n \\
A_i &= \left(\sqrt{\frac{n}{1 - \mathbf{W}^0} \mathbf{P}_k} \right)_i
\end{aligned} \tag{3.21}$$

The weight of the first sigma point $\mathbf{W}^0$, in (3.21), controls the proximity of sigma points to their mean. If $\mathbf{W}^0 \leq 0$ or $\mathbf{W}^0 > 0$, the sigma points tend to be closer or further from the origin. It is not trivial to tune $\mathbf{W}^0$ which affects the estimation results. Tuning process is also explained in [76].

At the model forecast step (3.22), the transformed points ($\mathbf{x}_{k+1}^{f,j}$) are produced by propagating each sigma point through the nonlinear system and used to compute the mean and covariance of the forecast value of $\mathbf{x}_{k+1}$ in (3.22).

$$\mathbf{x}_{k+1}^{f,j} = \mathbf{f}(\mathbf{x}_k^{f,j}) \quad j = 0 \dots 2n \tag{3.22}$$

After forecasting, estimated mean and variances are computed as

$$\begin{aligned}
\boldsymbol{\mu}_k &= \sum_{j=0}^{2n} \mathbf{W}^i \mathbf{x}_{k+1}^{f,j}, \\
\mathbf{P}_{k+1}^f &= \sum_{j=0}^{2n} \mathbf{W}^i \{\mathbf{x}_{k+1}^{f,j} - \boldsymbol{\mu}_k\} \{\mathbf{x}_{k+1}^{f,j} - \boldsymbol{\mu}_k\}^T.
\end{aligned} \tag{3.23}$$

Using the estimated output mean $\boldsymbol{\mu}_0$ and output variance $\mathbf{P}_1^f$, the estimated normal distribution can be constructed as $\mathbf{X}_1 \sim \mathcal{N}(\mu_k, \mathbf{P}_1^f)$.

3.2.3 Analysis and Controller Performance Metric

One of the core purposes of stochastic analysis is to assess the system's stability in the existence of noise. The methodology adopted in this thesis uses Markov chains to model system behavior. The aim is to improve the procedure for building the state transition matrix using the mean and variance estimations from the unscented transformation. The stochastic characteristics can be inferred by investigating the state transition matrix, and the effect of noise for each configuration can be commented on. To complete the state transition matrix, the transition probabilities of nonabsorbing states in (3.16) are calculated as the following,

$$\begin{aligned}\mathbf{T}_{ij} =& \mathbb{P}(\mathbf{x}_{k+1} = s_j | \mathbf{x}_k = s_i) \\ =& \mathbf{F}_{\mathbf{X}_{k+1}}(\frac{s_{j+1} + s_j}{2}) - \mathbf{F}_{\mathbf{X}_{k+1}}(\frac{s_j + s_{j-1}}{2})\end{aligned} \tag{3.24}$$

where $\mathbf{F}_{\mathbf{X}_{k+1}}$ represents the cumulative distribution function of output distribution.

Transition probabilities to the absorbing state are equal to the total probability of not going into nonabsorbing states.

$$\begin{aligned}\mathbf{T}_{i1} &= \mathbb{P}(\mathbf{x}_{k+1} = s_1 | \mathbf{x}_k = s_i) \\ &= 1 - \sum_{i=0}^{N} \mathbb{P}(\mathbf{x}_{k+1} \neq s_1 | \mathbf{x}_k = s_i)\end{aligned} \tag{3.25}$$

Finally, setting transition probabilities from the absorbing state to zero completes the state transition matrix structure in (3.18).

The state transition matrix of the Markov Chain includes information about the stability of the system under disturbance. Mean first passage time is a stability metric for metastable systems and extracted from the second-largest magnitude eigenvalue of the state transition matrix $\mathbf{T}$. Definition of system-wide mean first passage time value M is the following;

$$M = \frac{1}{1 - \lambda_2}. \tag{3.26}$$

The closed-loop system's estimated mean first passage time values can be used to assess different controllers' performances. In addition, the closed-loop system behaviors can be compared with respect to different levels of noise variances.

3.3 Stochastic Analysis of One Dimensional Hopper

This section states an example of the stochastic analysis by the proposed methodology in Section 3.2. This hopper with one-dimensional vertical motion is expected to demonstrate the applicability of the proposed stochastic analysis methodology. Its quasi-linear nature prohibits generalizing the deductions on more complex dynamical systems; however, promising results are introduced in the following sections.

3.3.1 Model and Dynamics

Chosen one-dimensional hopper model is a variation of the Spring Loaded Inverted Pendulum (SLIP) template with constant forcing and damping called F-SLIP studied by [86, 87, 88]. Using SLIP model variations brings the advantage of simplicity for implementation and analysis together with its applicability to many legged systems as a template [14]. Figure 3.3 depicts the dynamical model of the F-SLIP template model with one-dimensional vertical motion.

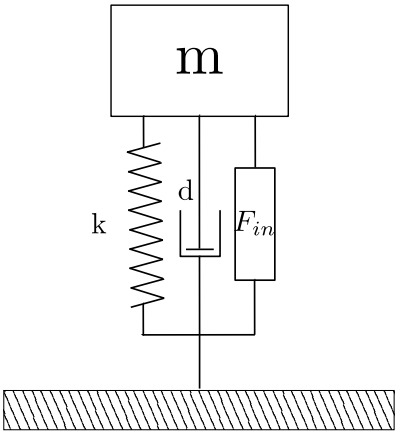

Figure 3.3: Illustration of the hopper with one-dimensional vertical motion

Detailed solutions of differential equations used in apex-to-apex simulations are given in Appendix A. The MATLAB script for one-step calculation is built according to those steps and parameters.

3.3.2 Stochastic Analysis

In this section, the behavior of the one-dimensional hopper is examined in case there is a disturbance at the impact velocity. This stochastic analysis will lead to deductions on the applicability of the proposed method and gives a layout to study the effects of internal parameters on the system's stability under noise.

Running the system for each initial condition in the state space when no noise is involved in the calculation, one can build the deterministic return map for the system. Return map for the one-dimensional hopper is depicted in Figure 3.4 together with the line with unity slope. The intersection of these two curves gives the fixed point location. A fixed point is a point from which the hopper starts its motion and arrives at the same point at the next step. From this Figure, the fixed point for the system can be found as $h = 0.6146m$. Also, the slope of the deterministic return map shows that the system is stable.

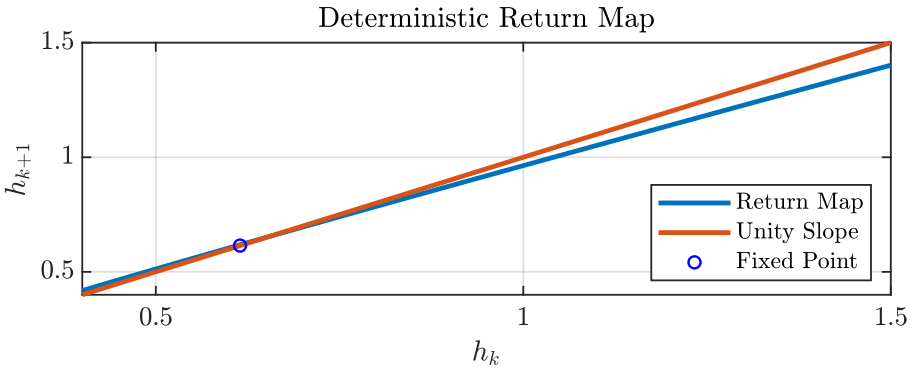

Figure 3.4: Return map and fixed point for a one leg hopper without external disturbance. h_k is defined as the apex height at time step k where h_{k+1} represents the apex height at the next step (Fixed point $\approx$ 0.6146 m)

Impact disturbance is assumed to be an additive noise drawn from the zero-mean Gaussian distribution with a variance of 0.05. This noise is added to the velocity value at the impact so that the noise value also goes through nonlinear transformations. That is why augmented state representation was required during the formulation of estimation. In order to conduct systematic experiments to build output distributions, noise values are sliced as in Figure 3.5.

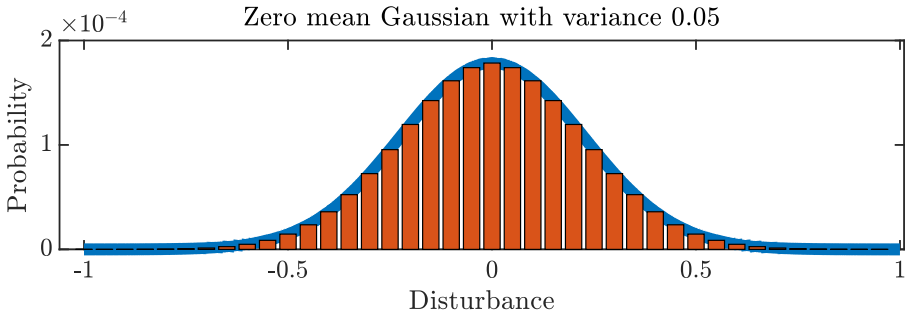

Figure 3.5: Noise values with their probabilities. Slices are drawn wider for visualization purposes.

For the stochastic analysis, the first step is representing the system as an absorbing Markov chain. Since this system has a one-dimensional state representation, determining the Markov states is pretty straightforward. States of the Markov chain are obtained by discretizing the state space, in this case, only the height values, using equally spaced 220 slices between 0.4 and 1.5 m and defining an absorbing state to represent the height values below 0.4 and higher than 1.5, this slicing is roughly illustrated in Figure 3.6.

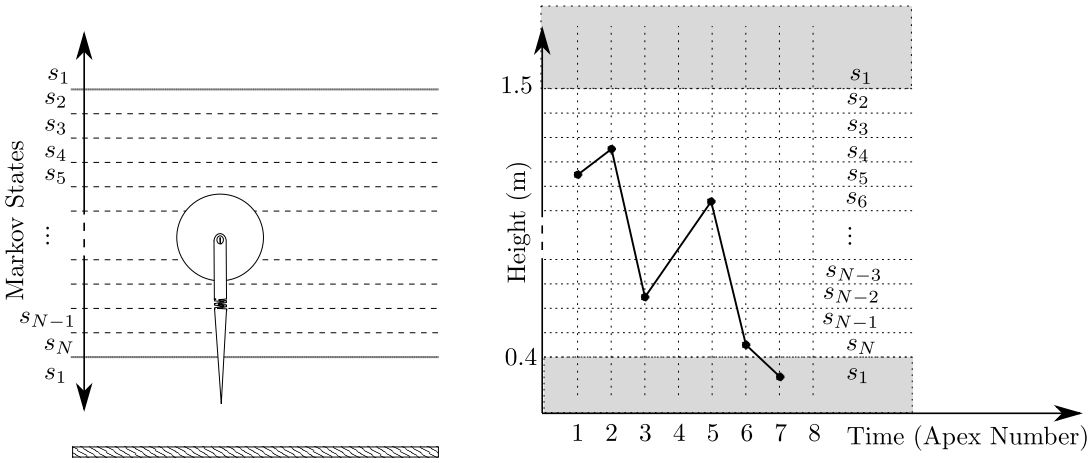

Figure 3.6: (on the left) Discretization of states for one dimensional hopper. The apex height of the top of of the leg is discretized into a finite set of slices. (on the right) An example passage time observation. Passage time is observed as 7, that means the robot leaves the predetermined region at the 7^{th} step.

During systematic experiments, all possible noise values are added to the system for all states of the Markov chain, and the results are weighted by their probabilities. A state transition matrix is calculated by the weighted sum of the state-to-state transition occurrences. Of course, it is impossible to span the entire one-dimensional noise space; however, as the points are getting further from the noise mean, probabilities, i.e., weights, are getting too small and do not significantly affect the result. Therefore, one can truncate the noise probability distribution after including enough range of values. On the other hand, it is crucial to span a range wide enough with thin enough slices to maximize the accuracy and smoothness of the output distribution. The methodology for systematic experiments is referred to as the "Exhaustive Method" throughout this chapter.

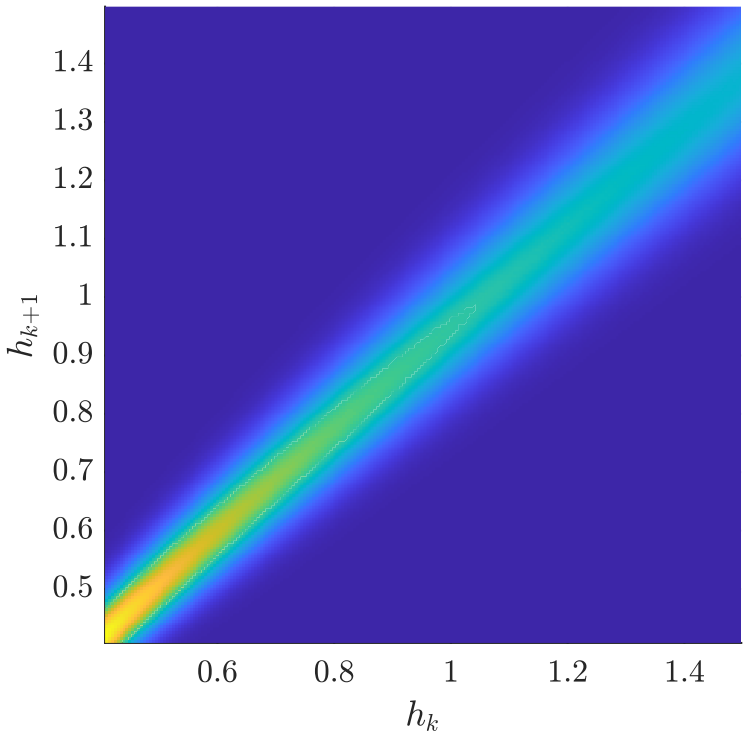

Figure 3.7: Return distribution for a one dimensional hopper with a impact disturbance with variance of 0.05 and zero mean. Rows of this matrix represent the transition probabilities from h_k to h_{k+1} and correspond to vertical lines on this surface plot.

Subsequently, under stochastic disturbance, the return map in Figure 3.4 is replaced by the probabilistic description of the transitions, as illustrated in Figure 3.7. This

figure is the surface plot of the 220×220 state transition matrix and represents the stochastic return map simultaneously. Some example rows of the state transition matrix in Figure 3.7 is depicted Figure 3.8. The vertical axis represents the probabilities of reaching the states on the horizontal axis. More clearly, the third subplot, for example, represents the probability distribution of the next jumping height when the current height is 0.8475m, and it is the 90^{th} row of the state transition matrix.

The largest eigenvalue of this state transition matrix is equal to 1, and the associated eigenvector is the first unit vector. That result is expected due to the nature of absorbing Markov chains and shows that the one-dimensional leg will escape from the predefined range sooner or later. The other eigenvectors associated with the other eigenvalues sum up to zero. The second-largest eigenvalue of this state transition matrix is equal to $\lambda_2 = 0.991707153083467$, which yields a system-wide mean first passage time of 120.58 steps, according to (3.26).

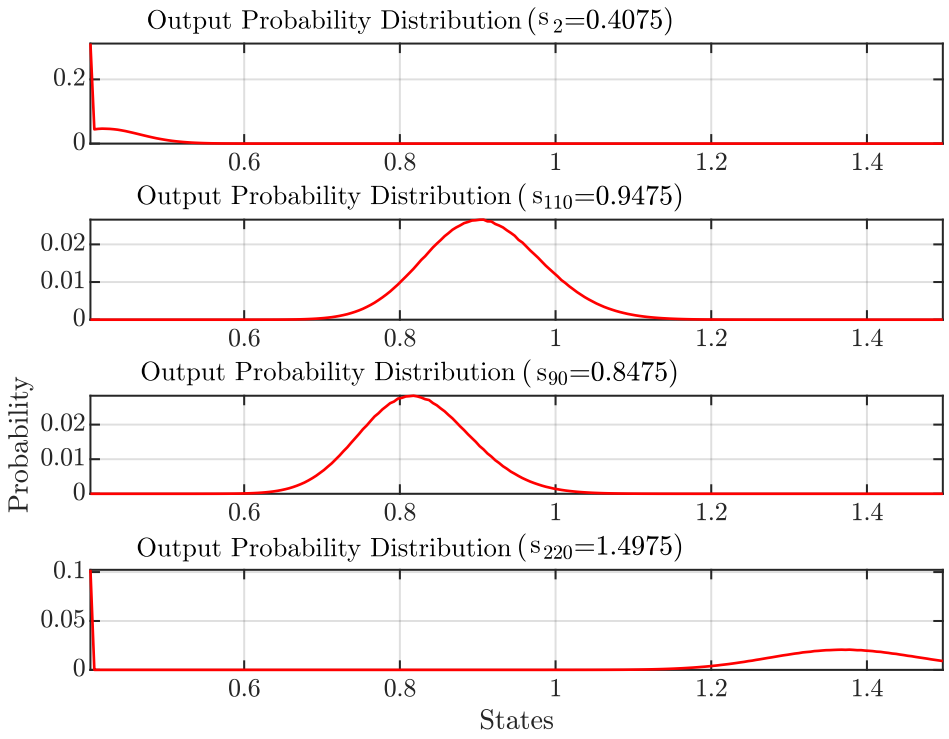

Figure 3.8: Return distribution for a one dimensional hopper with a impact disturbance with variance of 0.05 and zero mean, if the initial condition is the states on the subplot titles.

The red contour depicted in Figure 3.9 represents the metastable neighborhood of state-to-state transitions. Suppose the robot starts from some arbitrary initial condition and has not fallen after several steps. In that case, this contour map represents the joint probabilities of transitioning to the h_{k+1} while starting from state h_k. The contour lines drawn on Figure 3.9 are the contour plot of the metastable neighborhood. This metastable neighborhood represents that if the robot starts to jump from a random height, its height at the next apexes will most likely be around the fixed point unless it fails. In order to calculate the metastable neighborhood, first the metastable distribution should be defined. Metastable distribution is the stationary distribution of the Markov states unless the robot is not in the absorbing state and calculated by the replacing the first element of eigenvector associated with the second largest magnitude eigenvalue with zero and normalizing it. Then the metastable neighborhood, i.e. joint probability, can be calculated by multiplying the state transition matrix with the metastable distribution.

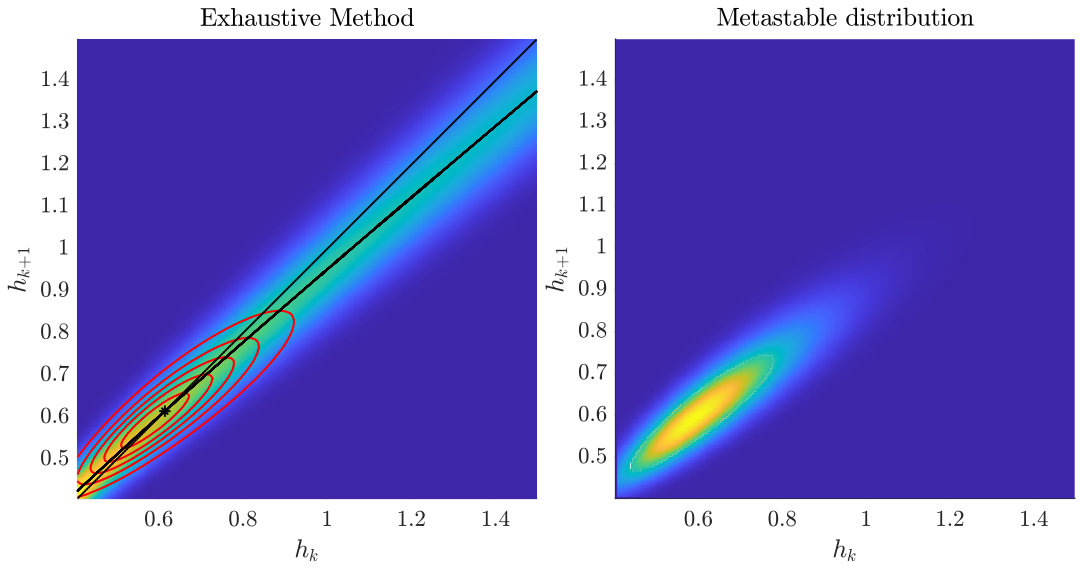

Figure 3.9: (on the left) Same return map as in Figure 3.7 with additions of metastable neighborhood contour, deterministic return map and unity slope, (on the right) The metastable neighborhood of state-to-state transitions.

There is much valuable information to be extracted from this state transition matrix representing a stochastic return map. Further deductions will be stated in Section 3.3.2.2. Due to its importance, the goal is to estimate this state transition matrix

accurately. This thesis covers three main approaches to estimating the state transition matrix. First of all, one can calculate the state transition matrix by running the aforementioned systematic experiments covering a wide range of noise values. This method can be interchangeably addressed as Monte Carlo simulations when the slicing of noise values is infeasible, so the initial conditions are randomly sampled in space. Monte Carlo simulation results are expected to be very close to systematic experiments if enough number of experiments are conducted. Secondly, one can use a linearized version of the system to calculate the mean and variance of the output distribution. This linearization can be conducted either, if available, using the analytically linearized version of the system or numerically calculated linearized system matrices (Jacobians) at respective points. This methodology is formulated in Section 3.1.2.1. Estimation results via linearization will be addressed in Section 3.3.2.1. Finally, another alternative is proposed based on the unscented transformation method, whose results will also be presented in Section 3.3.2.1. The third approach is proposed to handle nonlinear systems more efficiently without losing the information sourced from nonlinearity. This thesis hypothesized that using either linearization or unscented transformation will yield a satisfactory estimation of the output distribution depending on the system's nature. However, for the cases where nonlinear behavior is dominant, the estimation method based on unscented transformation is expected to outperform the linearization-based methods because unscented transformation is supposed to handle nonlinearities and takes the actual system dynamics into account for calculations.

3.3.2.1 Estimation with Linearization and Proposed Method

In this section, following the modified formulation in (3.8) in Section 3.1.2.1, state transition matrix is produced. In the numerical linearization step, the effect of noise values $\mathbf{w}$ to system output is calculated by the same method used in Section 2.5. This numerical Jacobian calculation method uses variable step-size for the perturbations to calculate the elements of the linearized matrix. Therefore, the linearization becomes much more accurate than a predefined step-size case because the most appropriate step size is used for the linearization radius. In the end, $\mathbf{f}_{\mathbf{w}_k} = \frac{\partial \mathbf{f}}{\partial \mathbf{w}_k}$ expression relates the infinitesimal changes in apex state to infinitesimal changes in noise.

Rows of the state transition matrix are calculated by the mean and variance values from (3.8). The resulting estimated state transition matrix is depicted in Figure 3.10. Due to its near-linear nature, output estimation for the one-dimensional leg is quite accurate with the numerical linearization of the system.

Following the proposed methodology in Section 3.2.2, for each state of the Markov Chain mean and variance of the output distribution is estimated. Note that each Markov state is actually a range of height values. For the estimations, the mid point of those ranges is picked and passed through the estimator. If the slices are thin enough, this sampling will not affect the result. Figure 3.10 shows the estimated state transition matrix.

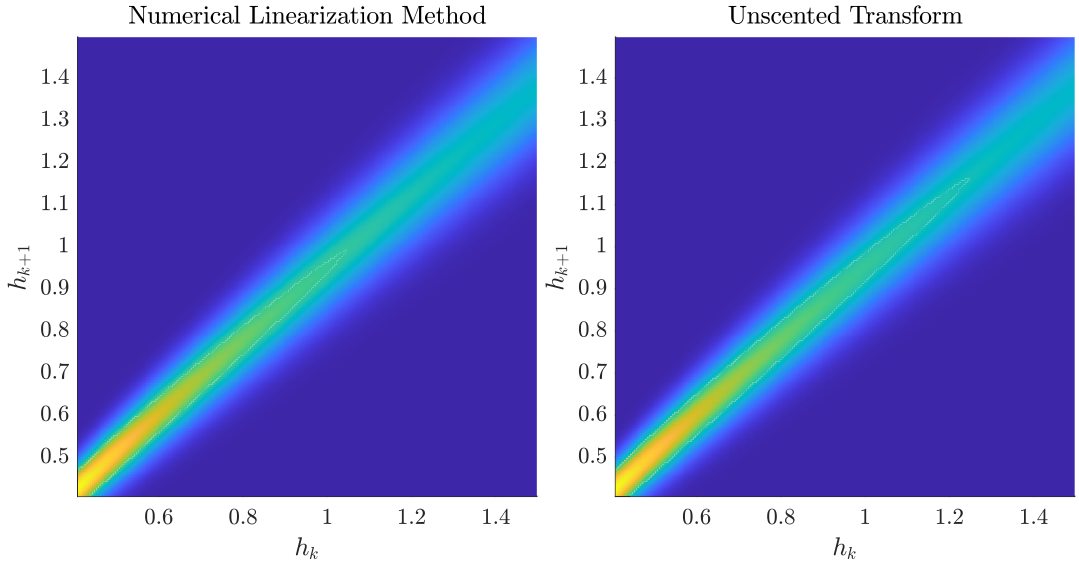

Figure 3.10: Resulting transition matrix from Numerical Linearization (on the left) and from proposed method based on unscented transform (on the right).

3.3.2.2 Comparison and Results

In this section, different estimation methods are compared and followed by the deductions from the state transition matrix properties. As stated before, there are mainly three different methods dealt with in this thesis. The systematic experiments (or Monte Carlo simulations), estimation based on linearization, and the proposed estimation method based on unscented transformation. Clearly, Figures 3.7, 3.10 are

almost indistinguishable. That means that the estimation methods have done quite a satisfactory job approximating the output distributions. Figure 3.11 shows some sample output distributions to visualize the rows of estimated state transition matrices.

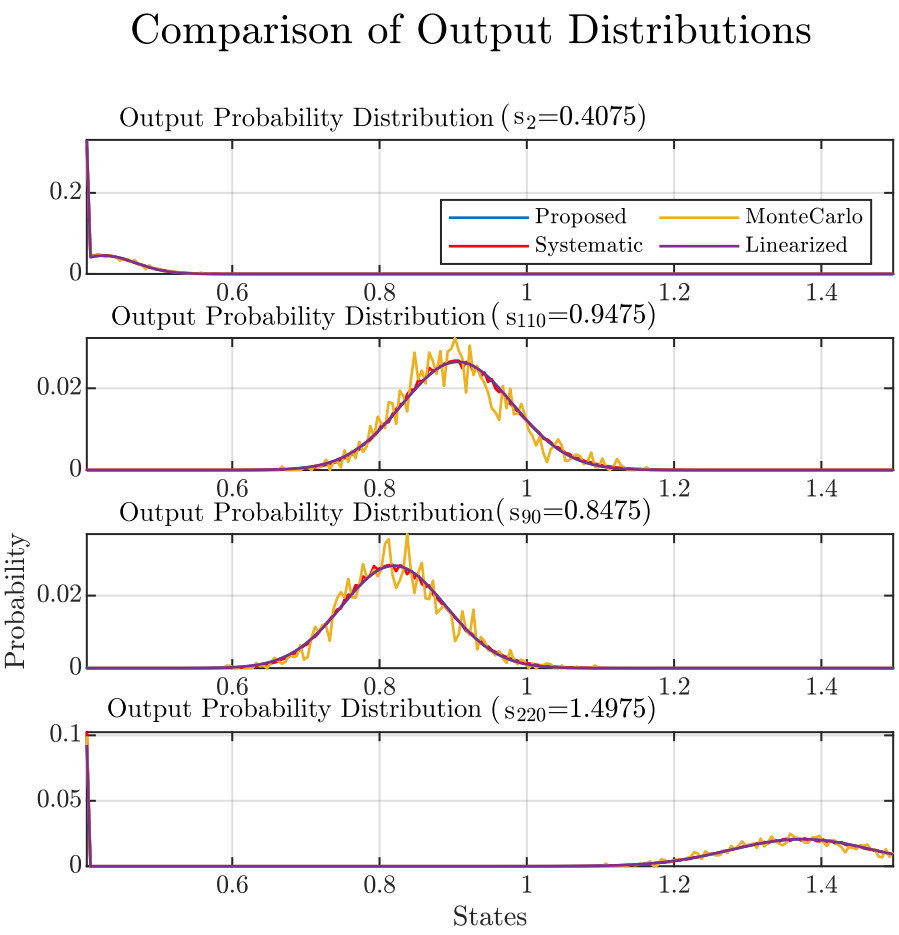

Figure 3.11: Comparison of output probability distributions for some selected Markov states

To assess the performance of the estimation methods, properties of the estimated output distributions and the results of systematic experiments should be compared. Figures 3.12 and 3.13 show the comparison of the estimated mean and variances and their difference from the proposed method for better visibility. As seen from all three Figures 3.11, 3.12 and 3.13, mean and variances are very close to each other. Estimation results by the Monte Carlo experiments are not smooth. However, they are expected to get smoother and be identical to the results of systematic experiments

as the number of random trials increases.

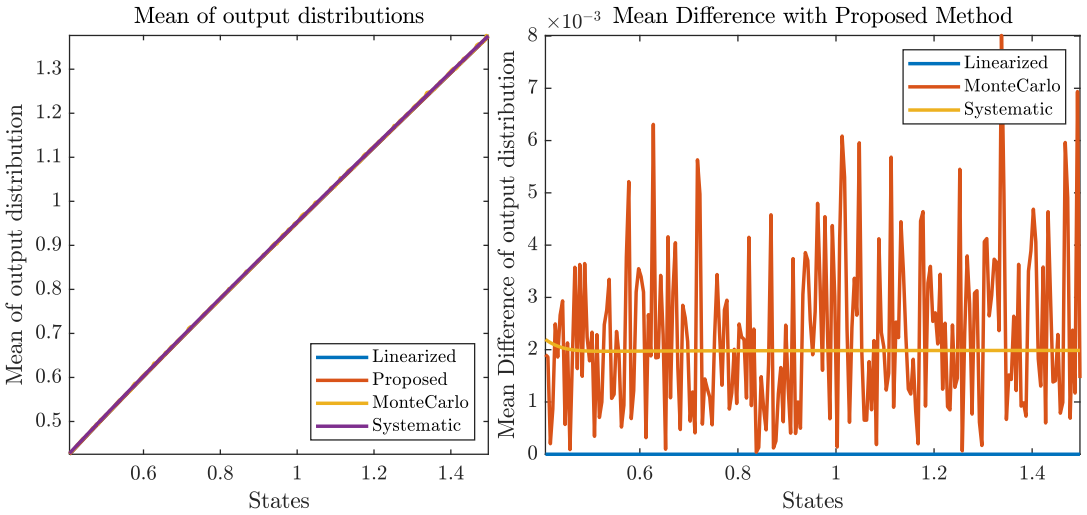

Figure 3.12: Comparison of mean of Output PDF's and their absolute difference with the results of the proposed method for better visibility

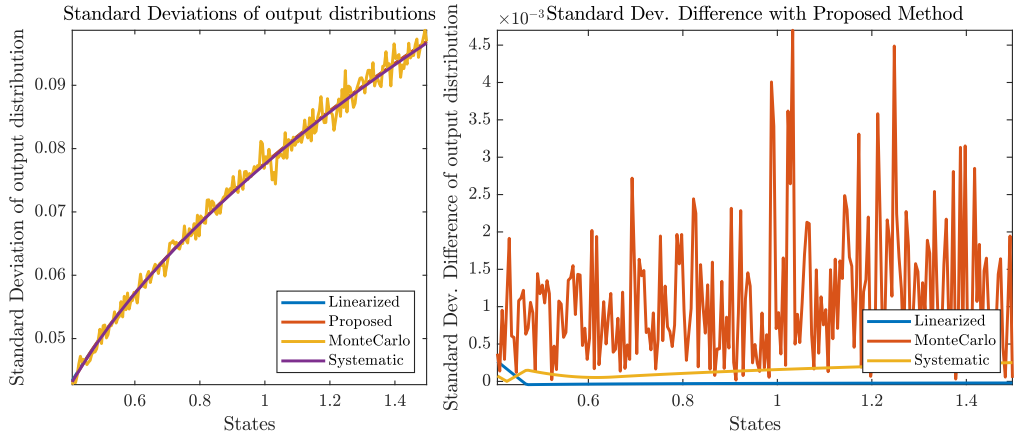

Figure 3.13: Comparison of variances of Output PDF's and their absolute difference with the results of the proposed method for better visibility

That similarity mainly comes from the quasi-linear nature of the simulated one-dimensional hopper. The linearization-based method is expected to work as well as the proposed method, which deals with nonlinearities. The results are also expected to be close enough to the systematic experiment method, which has no further assumptions on the system and represents the true output distribution.

For a deeper comparison, one can examine the other properties extracted from the

state transition matrices. Eigenvalues, for example, contain essential information about the Markov chain and subsequently the metastable system itself. From the second largest magnitude eigenvalue, system-wide mean first passage time is calculated to use as an indicator for stochastic stability. Figure 3.14 shows the first four eigenvalues of the state transition matrices for impact velocity noise variance of 0.05 for comparison, which are also listed in Table 3.1. The methods based on unscented transformation and linearization give almost the same result and slightly deviated results from Monte Carlo and systematic experiments. The linearization-based estimation method is observed to give almost identical results to the proposed method.

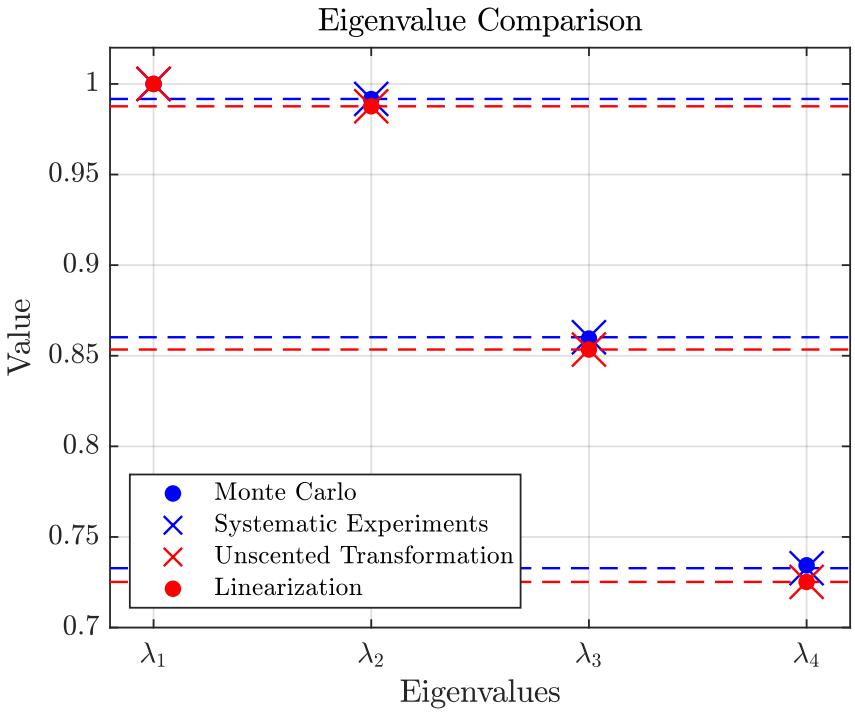

Figure 3.14: Visualization of first 4 eigenvalues of the state transition matrices for impact velocity noise variance of 0.05.

Additionally, the eigenvectors corresponding to the second largest magnitude eigenvalues of the state transition matrices obtained by different methods are also very close to each other, as depicted in Figure 3.15. The second eigenvector actually relates the state transition matrix with the system's metastable distribution. If the first element of the vector is replaced by zero and normalized, the metastable distribution is obtained. Through this definition, it can be concluded that starting from an initial

Table 3.1: Comparison of different transition matrix estimation methods

Method	# of experiment	λ_1	λ_2	λ_3	λ_4
Systematic Experiments	$6.6 \cdot 10^6$	1.0000	0.9917	0.8602	0.7328
Monte Carlo	$3.3 \cdot 10^5$	1.0000	0.9916	0.8595	0.7343
Unscented Transform	$6.6 \cdot 10^2$	1.0000	0.9877	0.8534	0.7252
Linearization	$4.4 \cdot 10^2$	1.0000	0.9877	0.8534	0.7252

condition, the robot will likely move around $0.5m - 0.8m$ unless it falls.

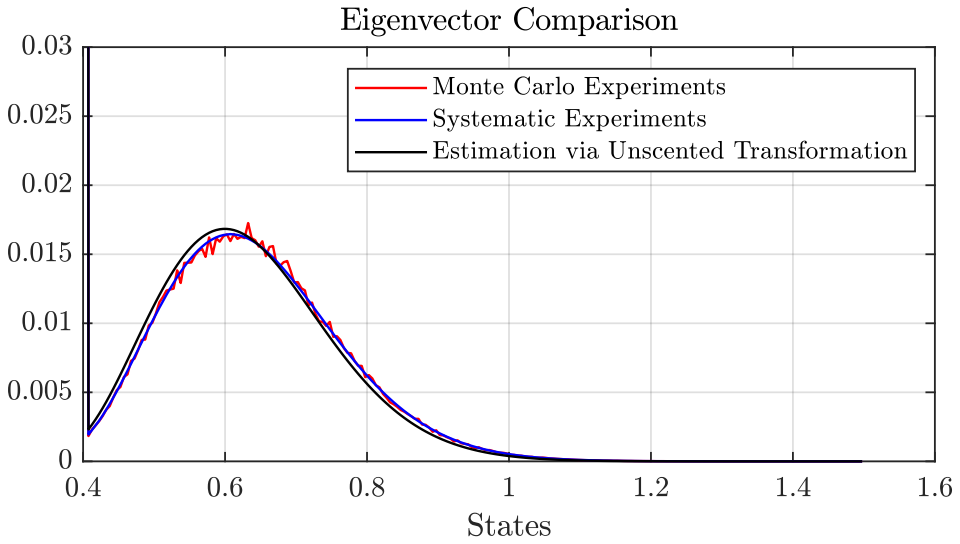

Figure 3.15: Comparison of 2^{nd} eigenvectors of state transition matrices for impact velocity noise variance of 0.05.

In the end, it is important to mention properties to extract from the state transition matrix to emphasize the contribution behind a better and faster estimation of it. State-dependent mean first passage time curves are one of the properties to discuss. The S state dependent MFPT vector m collects the expected passage time from the state s_i to the absorbing state s_1. This vector is computed as in (3.27).

$$m = \begin{bmatrix} 0 \\ (\mathbf{I} - \bar{\mathbf{T}})^{-1}\mathbf{1} \end{bmatrix} \tag{3.27}$$

where $\bar{\mathbf{T}}$ is the state transition matrix without its first row and first column, $\mathbf{I}$ is the

identity and **1** is a vector with all elements equal to 1.

State-dependent MFPT curve is plotted in Figure 3.16, using the state transition matrix estimated by the proposed method. Each initial condition has a particular state-dependent MFPT $m(s_i)$ and $m(s_i)$ quantifies the relative stability for each point. Different from the rimless wheel (RW) in [64], for this system, state-dependent MFPT curve is far from flat. Therefore, the objective system can be inferred as highly sensitive to initial conditions. In addition, the same conclusion can be reached by investigating the eigenvalues: $\lambda_1 = 1$, $\lambda_2 = 0.9917$, $\lambda_3 = 0.8602$, $\lambda_4 = 0.7328$. The value of λ_3 means that almost 14% of the contribution to the probability function at the initial condition is lost ("forgotten") with each successive step. Again, this was not the case for the rimless wheel in [64]. RW system has its third eigenvector near 0.5 and forgets 50% of the initial condition. As a result, within a few steps, initial conditions for any wheel beginning in the range of analysis have therefore predominantly evolved into the metastable output distribution unless it fails. Analogously, the motion of the one-dimensional hopper will converge to its metastable distribution after more steps but eventually it will.

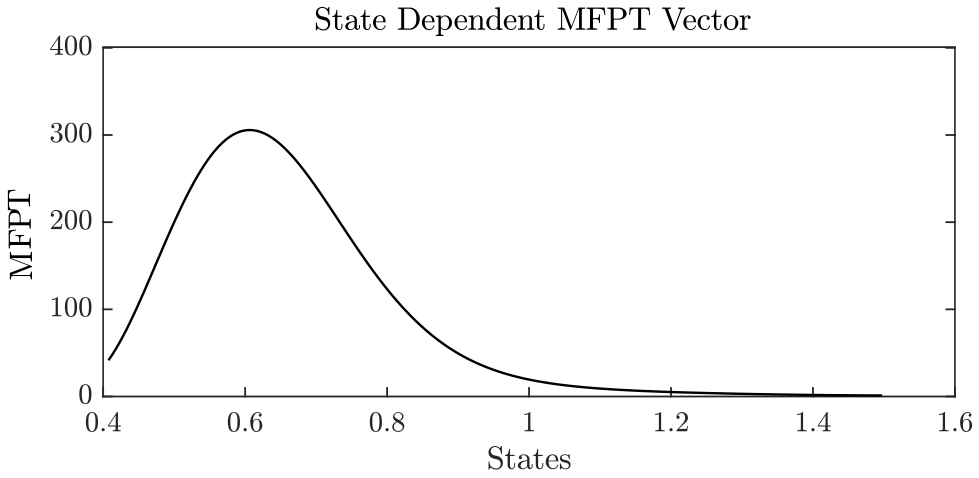

Figure 3.16: State dependent MFPTs, quantifies the relative stability of each point in state space, for impact velocity noise variance of 0.05.

After the assessment of the estimation performance, proposed estimation method is utilized to examine stochastic behavior of the system under different levels of noises. Figure 3.17 shows the output distribution of fixed point state for each of several values

of impact velocity noise variance. One can calculate the mean first passage time (MFPT) value from this estimated output distributions.

The comparison of the MFPT values based on noise variance helps the examination of the system behavior. Figure 3.18 shows the mean first passage time values with respect to different noise variances. The blue curve is drawn for this particular system with the specified controller and system parameters. One can conclude more about the affect of the different parameters in the system to the overall stochastic behavior through this analysis. If the stiffness value is changed to 3000, the MFPT values becomes as in the red curve meaning that the system become more resilient to the impact noise. However, increasing stiffness value does not always mean an increase in robustness, as can be deduced from the location of the yellow curve.

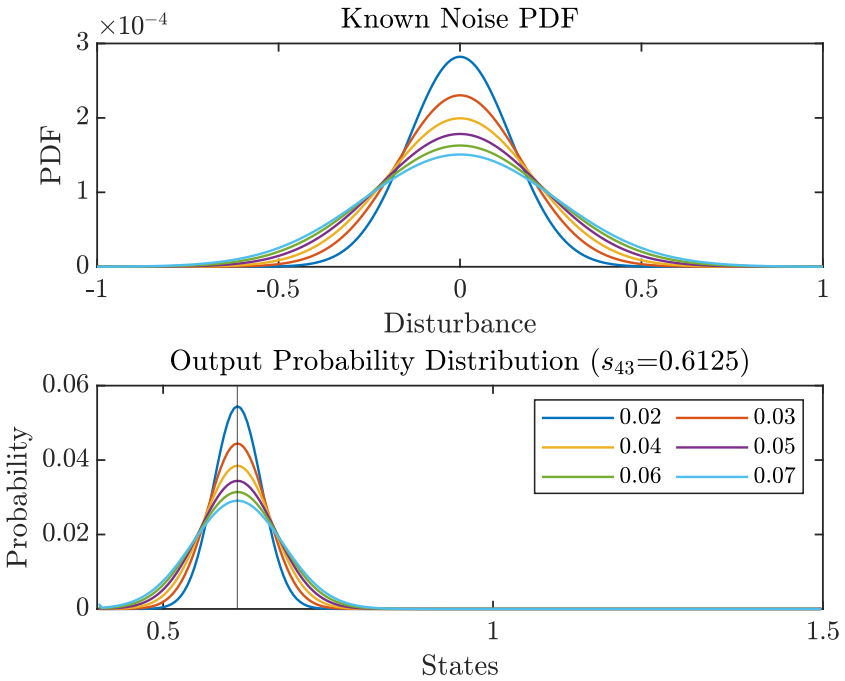

Figure 3.17: PDF of known noise and resulting output distribution of fixed point state, for each of several values of impact velocity noise variance. Each distribution is estimated by running the system using sigma points, i.e. unscented transformation. States are the 220 slices between 0.4m and 1.5m

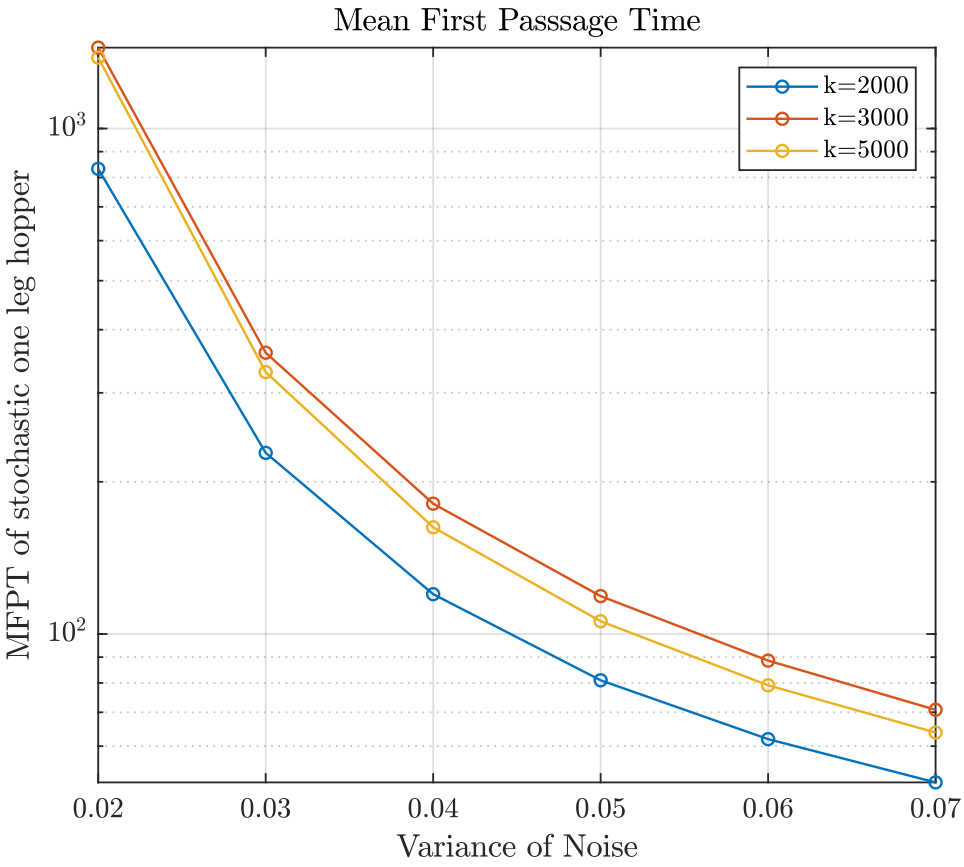

Figure 3.18: MFPT for the one leg hopper as a function of noise variation, σ^2, obtained by proposed method

3.4 Extension to High-Dimensional Systems: Dimensionality Reduction

The main problem in extending this stochastic analysis to multidimensional systems stems from the requirement of meshing of multidimensional state spaces. In Saglam's studies [69], meshing the hybrid zero dynamics (HZD) surface is presented as an alternative to finding and meshing the reachable state space for a bipedal walker operated by an HZD controller. In this way, a switching mechanism between multiple HZD controllers becomes possible to increase stability. Despite the effort to decrease the complexity of the meshing process, the issue still exists and grows with the increasing degree of freedom and the variety of noise sources.

Linearization is a candidate method to decrease system order to identify and control systems. Numerical Jacobian calculations with variable step size can be conducted exactly in Section 2.5 [54], or other methods such as analytical linearization, forward and center difference approximations can be used, noting that the linearization method will influence the result. After linearization, by investigating the eigenvector associated with the largest magnitude eigenvalue, the state can be identified such that the system is the most sensitive against a change in that state. Eigenvalues of the system give a picture of stability around the chosen operating point.

The selected state can be used as the indicator state for the stability conditions in the stochastic stability analysis. Nevertheless, calculating the linearized system matrix with variable step size gets more difficult as the dimension increases. Choosing a fixed step size to tackle complexity diminishes the accuracy of the calculation.

Alternatively, this analysis for the most vulnerable state can also be done with stochastic tools. For dimension reduction, there are many methods such as Principal Component Analysis (PCA), Heatmaps, t-SNE plots, and Multidimensional Scaling (MDS). Although all the methods aim to reduce the dimensionality of the dataset, their objective is different. Each technique has its own trade-off while mapping high-dimensional data into low dimensions and preserving the information. PCA identifies linear combinations of variables that provide maximum variability within a data set. i.e., PCA aims to maximize the variance in the data.

This study features PCA to reduce the objective dimensions to assess the legged sys-

tem's stochastic stability. PCA is used as a preprocessing technique to reduce dimensionality. It aims to increase interpretability while minimizing information loss [89] and allows to use of previously collected data rather than conducting experiments for numerical simulation. A PCA plot converts the correlations among all cells into a 2D graph and allows to comment on the features in the dataset. The mathematical details of the method are not in the scope, so they are not covered in this thesis. More detailed information can be referred from [89].

In order to explain the basic principles of the PCA, this toy example is inserted. Assume that there is a dataset containing mice and their selected genes, as in Table 3.2.

	Mouse A	Mouse B	Mouse C	Mouse D	Mouse E	Mouse F
Gene 1	10	11	8	3	2	1
Gene 2	6	4	5	3	2.8	1
Gene 3	12	9	10	2.5	1.3	2
Gene 4	5	7	6	2	4	7

Table 3.2: Mouse - Genes Dataset

If the raw data from the table is plotted as in Figure 3.19, one may not see the relations to classify the mice according to the genes that they have. PCA can be conducted to reduce the dimensionality of the gene data.

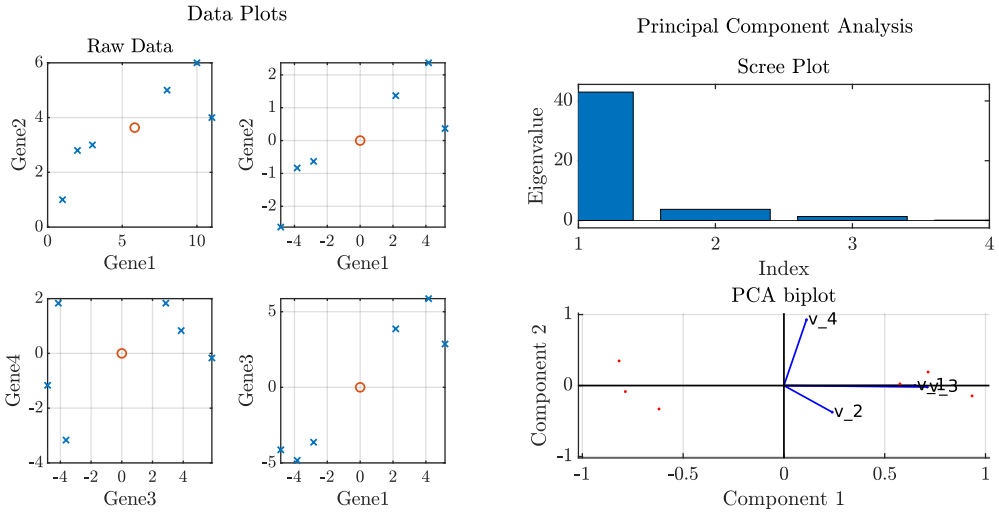

Figure 3.19: Data from the table and PCA plots of the dataset

Scree plot, in Figure 3.19, states how much variation each principal component (PC) captures from the data. The first principal component (PC1) obviously has the largest eigenvalue. That is why PC1 can be considered enough to describe the data. Since v_1 and v_3 has the largest projections over the PC1 axis, one can distinguish the mouses using only those two genes. This approach is generalized to select "the most important state" in a multi-dimensional legged system.

The most critical limitation of PCA is its reliance on linear models and sensitivity towards outliers. Because PCA is a linear projection, it assumes a linear relationship between features and cannot capture the nonlinear dependencies. Its goal is to find the directions (i.e., principal components) that maximize the variance in a dataset.

In this thesis, the system's input-output relation is assumed to be known. While having the system that generates the dataset, analyzing the dynamics based on just the data might seem controversial. However, this model reduction method can be generalized for a model reduction on experimental legged setups when this methodological study goes forward with the actual implementation.

3.5 Stochastic Analysis of Bipedal Walking

This section presents the higher dimensional system example featuring an existing bipedal locomotion structure. Analysis of this more complex system strengthens the argument of this thesis by showing the applicability of proposed method. The stochastic analysis is again conducted by following the methodology proposed in Section 3.2. The model dynamics are explained briefly and the model is reduced to extend the methodology to this higher dimensional hybrid rhythmic dynamical system. The comparison of different methods and the implications of stochastic analysis are also presented in following sections.

3.5.1 Model and Dynamics

Proposed method is demonstrated on a 5-link bipedal locomotion model, the RABBIT [90]. This simulation testbed has a controller design based on optimization of the hybrid zero dynamics (HZD) following the same steps in [91, Chapter 6.6.2.1]. The details about the HZD controller implementation is not in the scope of this thesis and treated like a black-box during the analysis. Briefly, as in the implementation in [91], the system dynamics are defined as

$$\dot{x} = f(x) + g(x)u \tag{3.28}$$

where ten dimensional state $x := [q^T \ \dot{q}^T]^T$ collects the configuration variables $q := [q_1 \ q_2 \ q_3 \ q_4 \ q_5]^T$, as shown in Figure 3.20, along with their velocities and system dynamics are represented by f and g.

HZD ensures that relative angles $h_0(q)$ track desired trajectory $h_d(q)$. Definition of tracking error (y) is

$$y = h(q) := h_0(q) - h_d(q). \tag{3.29}$$

Control input applied by the HZD controller takes the following form

$$u = (\mathcal{L}_g \mathcal{L}_f h)^{-1}(-\mathcal{L}_f^2 h + v) \tag{3.30}$$

where $\mathcal{L}_g \mathcal{L}_f h$ and $\mathcal{L}_f^2 h$ represents Lie derivatives of tracking error with respect to system dynamics f and g in (3.28). The control input is saturated in the implementation to make the simulation studies more realistic.

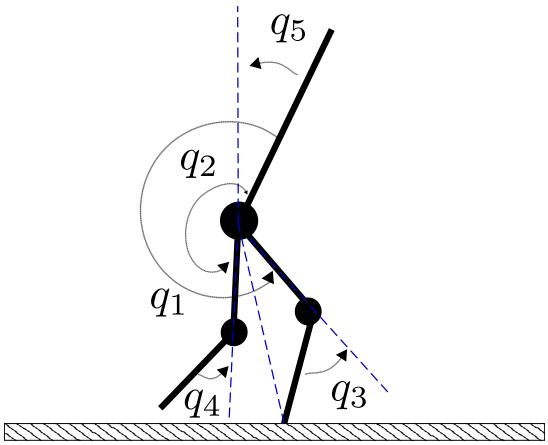

Figure 3.20: Illustration of 5-link bipedal robot

Study in [91] proves that, with a basic PD controller, the solution of the closed-loop system converges to an exponentially stable periodic orbit of the hybrid zero dynamics. Therefore, it is preferred to utilize a PD controller in (3.31) for v to force h in (3.29) to zero.

$$v = K_D \mathcal{L}_f h + K_P h \tag{3.31}$$

Table 3.3 shows different parameter choices for diagonal entries of K_P and K_D pairs to analyze the closed-loop behavior later with the proposed method.

Table 3.3: Different controller parameter pairs, values of the table represents the diagonal entries of K_P and K_D values

	K_P	K_D
C_1	[60 90 90 50]	[10 20 20 10]
C_2	[5 5 5 5]	[5 5 5 5]
C_3	[40 40 40 40]	0
C_4	[40 40 40 40]	[1 1 1 1]
C_5	[10 89 83 50]	[5.4 21 21 9]

The first step towards the stochastic analysis of bipedal locomotion is building the reachable state space. Reachable state space should be built by Monte Carlo sampling or meshing the space by predefined ranges [66]. In the case of a 5-link bipedal robot,

each Markov state s_i can be chosen as a 10×1 vector, containing each link's angular positions and velocities. Nevertheless, it is too complicated to specify the Markov states and build the state transition matrix either by applying Monte Carlo sampling to 10D space or meshing a 10D space. The underactuated 5-link bipedal testbed needs a different approach for stochastic analysis.

The system's controller follows a trajectory such that the unactuated link shows the desired behavior, i.e., actuated degree of freedoms indirectly control the body angle. As shown in Figure 3.20, the position and velocity of stance and swing legs of the walker are defined relative to the body of the system. This coordinate configuration strengthens the idea that high bandwidth actuated joints are expected to be around their desired trajectory as long as the unactuated joint is close to its desired evolution. That is why observing body angle provides strong information about other joints' evolution and stability. Furthermore, for the reachable state space construction, the underactuated body angle $s_i = q^i_5$ and body angular velocity $s_i = q^i_{10}$ are suitable candidates to focus, search the vicinity of the fixed point and define the reachable limits assuming noises for all five states representing velocities. Nevertheless, this model reduction should be justified quantitatively. The next section explains the model reduction process for this 5-link bipedal system.

3.5.2 Model Reduction

The way to chose the objective state should be justified. Although the body angle can be concluded as the indicator of stability by the intuition resulting from the comments on underactuated nature. A candidate approach is the analysis by linearization as stated in Section 3.4. The linearized system matrix is extracted by numerical Jacobian calculation [54] while the bipedal system is tuned with the controller parameter set

C_1 on Table 3.3. Resulting linearized system matrix becomes as follows:

$$\mathbf{J} = \begin{bmatrix} -0.0155 & -0.2900 & -0.0054 & -0.1065 & -0.2273 & 0.0008 & 0.0011 & 0.0002 & -0.0001 & -0.0005 \\ 0.0559 & 0.0968 & -0.0015 & 0.0561 & 0.0920 & 0.0044 & -0.0004 & -0.0003 & 0.0001 & -0.0009 \\ 0.0239 & 0.0631 & 0.0169 & 0.1088 & 0.1008 & 0.0035 & -0.0046 & 0.0029 & 0.0005 & -0.0008 \\ 0.0128 & -0.3144 & -0.0217 & -0.1639 & -0.3134 & -0.0021 & 0.0001 & 0.0088 & -0.0001 & -0.0021 \\ -0.0253 & 0.1104 & -0.0012 & 0.0039 & 0.0647 & -0.0035 & 0.0012 & -0.0014 & -0.0002 & 0.0010 \\ 0.5528 & 5.0936 & 0.1411 & 2.3864 & 5.2884 & 0.0853 & -0.0401 & 0.0461 & 0.0038 & 0.1568 \\ -0.2237 & 0.8440 & -0.0446 & 0.1693 & 0.5534 & 0.0633 & 0.0021 & 0.0411 & -0.0002 & 0.1097 \\ 0.0074 & 1.5831 & -0.1939 & 0.0218 & 0.9167 & 0.1223 & 0.0287 & 0.0444 & -0.0023 & 0.1731 \\ -0.2970 & 2.1255 & -0.3576 & 1.3694 & 2.4911 & -0.1077 & -0.0005 & -0.1599 & 0.0001 & -0.1273 \\ -1.1268 & -3.5229 & -0.3458 & -1.2075 & -3.6862 & 0.4914 & 0.0185 & 0.2414 & 0.0006 & 0.5523 \end{bmatrix} \tag{3.32}$$

If the linearized system matrix is extracted with a different set of PD coefficients on Table 3.3, λ_1 and ν_1 results remains approximately the same. The eigenvalues of the linearized system matrix (**D**) is calculated.

$$\mathbf{D} = \begin{bmatrix} 0.7476 \\ 0.0535 \\ -0.0363 + 0.0567i \\ -0.0363 - 0.0567i \\ -0.0337 + 0.0299i \\ -0.0337 - 0.0299i \\ 0.0223 \\ -0.0000 \\ -0.0000 \\ 0.0000 \end{bmatrix} \longrightarrow \quad \lambda_1 = 0.7476 \quad \nu_1 = \begin{bmatrix} -3.54e-08 \\ -6.37e-08 \\ -1.32e-07 \\ -2.9e-08 \\ 9.12e-08 \\ -0.2186 \\ -0.1646 \\ -0.2652 \\ 0.2404 \\ -0.8928 \end{bmatrix} \tag{3.33}$$

The eigenvector (ν_1) associated with the largest eigenvalue (λ_1) indicates that the system is the most sensitive against the change in the 10^{th} state.

The second approach is using Principal Component Analysis. The first step of this analysis is to generate a dataset that contains the state values at the beginning of each gait, just before the impact. Controller set is the same as in previous analysis: C_1 on Table 3.3. The bipedal walker is run for 100 steps, and a 10×100 dataset is generated. Next, using the built-in function for PCA in MATLAB, the dataset is visualized. Figure 3.21 demonstrates the PCA biplot, scree plot, and score plot for the dataset. From the scree plot, it can be concluded that the first principle component (PC1) can be concluded as enough to describe the data. Investigating states' projections onto principal components, one can see that the 10^{th} state has the largest projection on

the first principal component, meaning that one can roughly characterize the motion using the 10^{th} state.

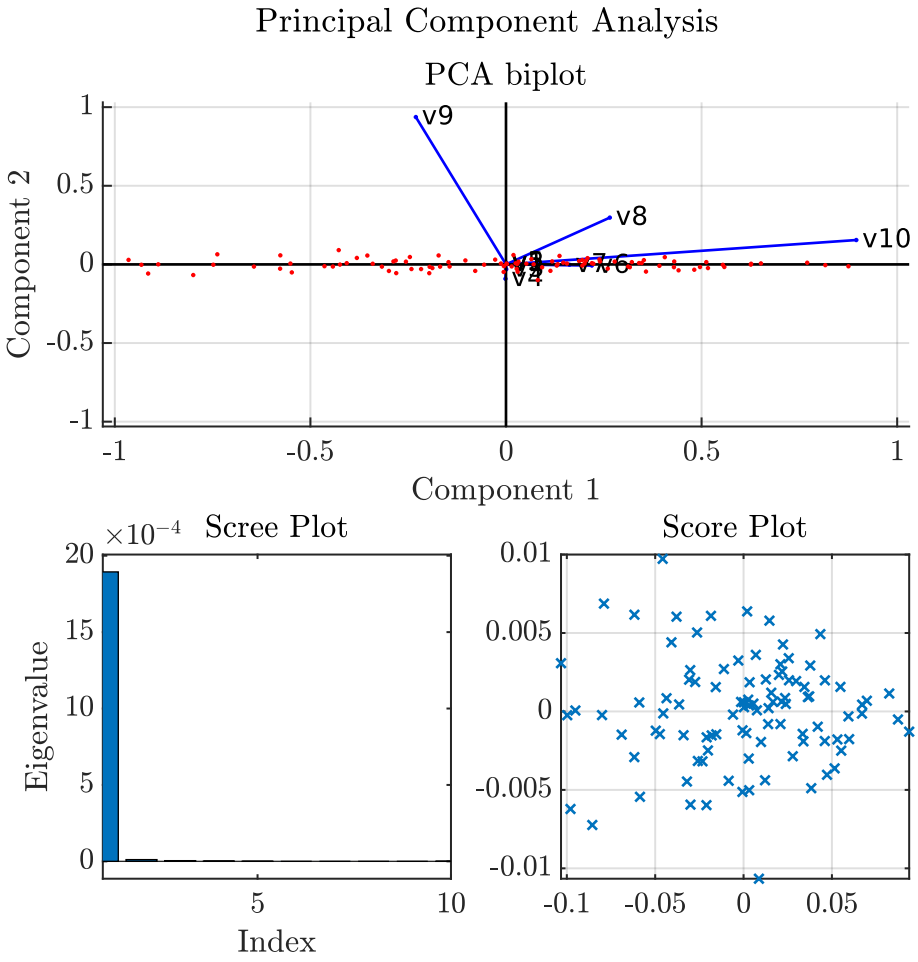

Figure 3.21: Principal Component Analysis for the 5-link bipedal system

While on the subject, it is no surprise that both linearization and PCA give the same result about the 10^{th} state because they both rely on the linearity assumption for the state relations. And they both produced results that are compliant with our intuition in the previous section.

3.5.3 Stochastic Analysis

In this section, stochastic behavior of 5-link bipedal walker's torso is examined. This stochastic analysis will confirm the estimation capability of the proposed method on multi-dimensional rhythmic hybrid dynamical systems.

In absence of noise, deterministic return map for the angular velocity of torso becomes as in Figure 3.22. As clearly seen from the Figure, body angle has a nonlinear behavior in some interval ($[-1.12, -0.98] rad/s$) and linear elsewhere.

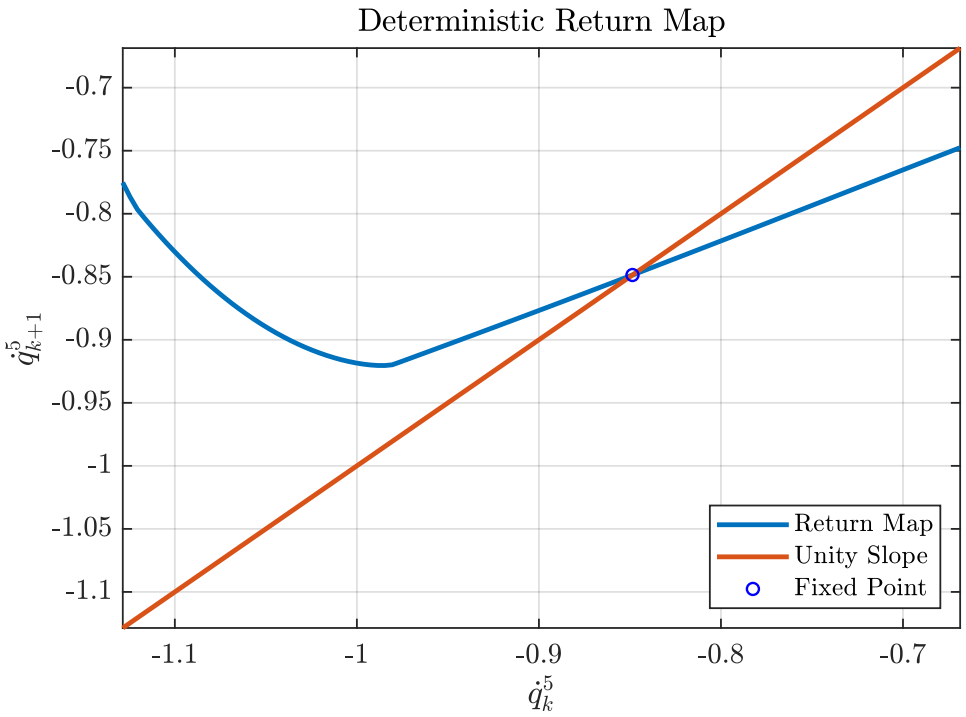

Figure 3.22: Return map and fixed point for the angular velocity $\dot{q}^5$ of 5-link bipedal system's torso without external disturbance, (Fixed point ≈ -0.8486 rad/s)

During stochastic analysis, disturbance is applied to all states representing velocities ($\dot{q} := [\dot{q}^1 \ \dot{q}^2 \ \dot{q}^3 \ \dot{q}^4 \ \dot{q}^5]^T$) at the impact and drawn from a zero mean Gaussian distribution with a variance of 0.001. Conducting systematic experiments is in this case not feasible because it is not trivial to mesh and sample a hypercube in 5 dimensions just to slice noise values. That's why, state transition matrix is calculated by random experiments. Calculated state transition matrix can be visualized in Figure 3.23 together with its deterministic return map.

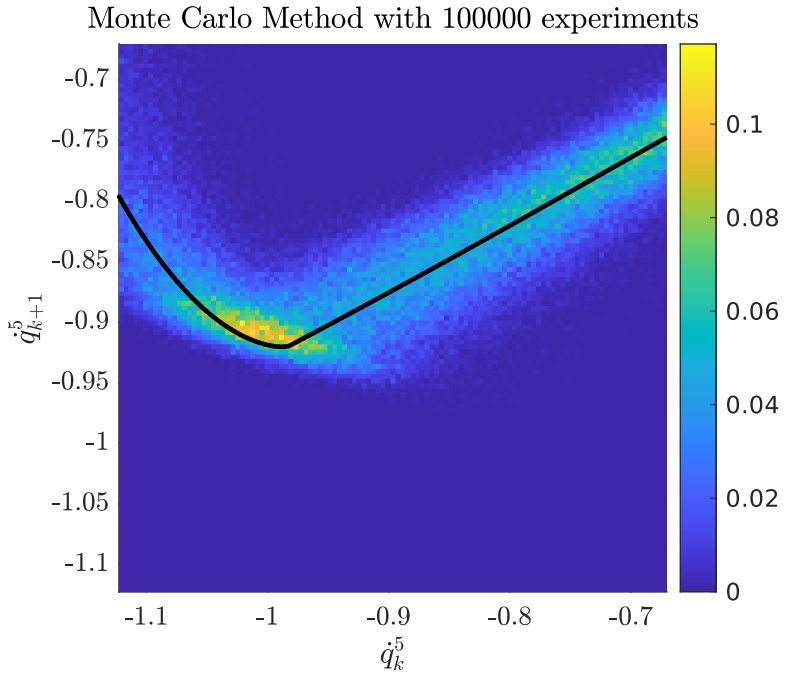

Figure 3.23: Return distribution for one dimensional hopper with an impact disturbance with variance 0.001 and zero mean. Colorful surface plot represents the 115×115 state transition matrix of body angular velocity for a zero mean Gaussian noise with variance of 10^{-3} on each derivative state ($\dot{q}$).The black line represents the deterministic return map of body angle. Controller is C_1 on Table 3.3.

Building a state transition matrix with Monte Carlo experiments requires a repetitive calculation of many experiments (For Figure 3.25, it is 10^4 for one Markov state, 1.15×10^6 in total.) for the mid-point of each state of the Markov chain as the initial condition of body angular velocity. Suppose the Monte Carlo sampling method is chosen to build matrices. In that case, due to its random nature, no two matrices will be the same, and increasing the number of trials per Markov state, for example, from 10^4 to 10^6 leads to a lower variance among the produced matrices. In other words, Monte Carlo simulations result in different matrices for each experiment set, requiring significant computational power and time. In a different approach, if the systematic experiments are conducted for each Markov state-noise value pair to get a result with lower variance, the objective system should be run for ν^{10} times, where ν is the size of the set of noise values. Apparently, both systematic and Monte Carlo experiments are infeasible for high-dimensional cases, as also clearly stated in [66].

The proposed method includes additional information to the experimental procedure to choose initial conditions to estimate each output state distribution. It requires 41 experiments for each Markov state in a 5-link bipedal walking case. Figure 3.24 depicts the constructed state transition matrix for body angle together with a deterministic return map. The output distribution of body angle is represented if all the states are subject to noise with known characteristics.

The goal is to estimate this stochastic return map with fewer experiments to reach basic conclusions on stochastic stability easier. At first, the linearization-based method and then the proposed methodology is employed. After implementation, the results from the estimated matrices and the comparison of different estimation methods are presented in the following sections.

3.5.3.1 Estimation with Linearization and Proposed Method

This thesis proposes a novel methodology for estimating the output distributions for hybrid multi-dimensional nonlinear systems. After choosing the Markov chain states as the body angular velocity by model reduction, the next step is to estimate the output distributions for each Markov state and build the state transition matrices.

As an alternative, estimation of the state transition matrix by linearizing the system around each Markov state is formulated in Section 3.1.2.1. Applying for the bipedal simulation, the state transition matrix for the body angular velocity state is visualized in Figure 3.24. As a result, linearizing the system allows us to build a state transition matrix with fewer experiments but seems only to work for quasi-linear parts. Around the nonlinear region, this method does not seem to capture output variances correctly. This difference stands obviously in Figure 3.26. Evidently, the methodology based on linearization of the system fails for the highly nonlinear return maps. In order to improve the estimation, the proposed methodology in Section 3.2 is followed. Figure 3.24 also shows the results of the proposed method, the stochastic return map, i.e., state transition matrix for the body angular velocity state, which is chosen by principal component analysis in the model reduction step.

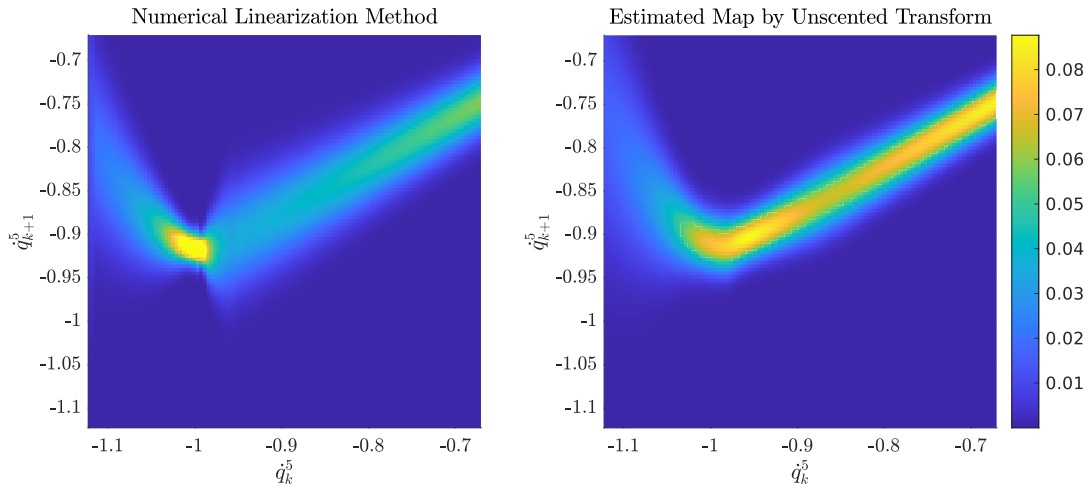

Figure 3.24: State Transition Matrix for body angular velocity (stochastic return map) estimated based on linearization (on the left) and proposed method (on the right). Colorful surface plot represents the 115×115 state transition matrix of body angular velocity for a zero mean Gaussian noise with variance of 10^{-3} on each derivative state. Controller is C_1 on Table 3.3

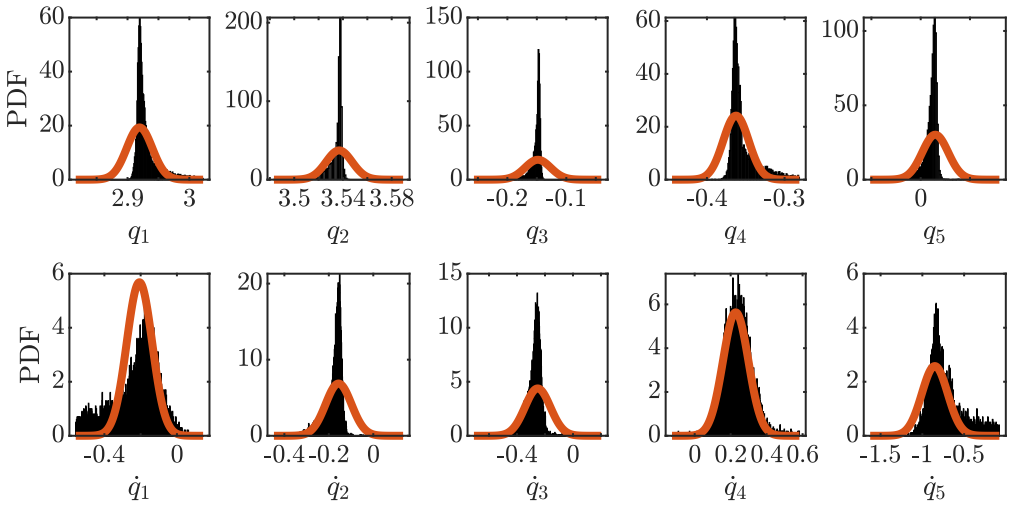

Figure 3.25: Comparison of experimental results and estimation with unscented transformation. Each subplot represents the output distribution of states of the 5-link bipedal robot under disturbance. Histogram of experimental results is a product of 10^4 experiments and the proposed method run 41 simulations for estimation.

The experimental observation supports the claim that under Gaussian noise, output

distributions' shapes approach Gaussian shape, shown as histograms in Figure 3.25. Also, the proposed estimation is capable of capturing each ten state's variance and mean under independent noise effects for each state. The advantage of the proposed method stands out with this example because the previously implemented methods are based on systematic experiments; in order to estimate all states, one must run too many experiments to cover the effect of multiple state noises. Unscented transformation provides minimal information loss to estimate the output distributions, thanks to its nonlinear nature (sample-and-propagate methodology).

3.5.3.2 Comparison and Results

This section deals with the comparison of different estimation methods and presents the conclusions made from the estimated state transition matrices. Figure 3.26 shows some selected rows of the state transition matrices obtained by three different methods. Even if the results of the Monte Carlo experiments are not smooth, since it has no assumption over the system dynamics, the blue curve can be treated as the true distribution for this particular figure. As can be seen, the two estimation methods introduced in this thesis cannot be compared just by looking at those selected rows, but they seem to represent satisfactory results.

The state transition matrix represents a stochastic return map and includes valuable information about the system behavior. For example, Figure 3.24 shows that the deterministic return map tends to be linear for some intervals, including the fixed point. That means, near the fixed point, the system can be assumed as a linear system. Under stochastic disturbance, estimation with the linearity assumption works very well, as seen in Figure 3.27. In addition, both deterministic and stochastic return maps indicate that the linearity assumption is not valid for a small region, so the body's behavior cannot be generalized as linear. The stochastic return map implicates the same facts as the deterministic one and brings the output variance information for different initial conditions of the body angular velocity.

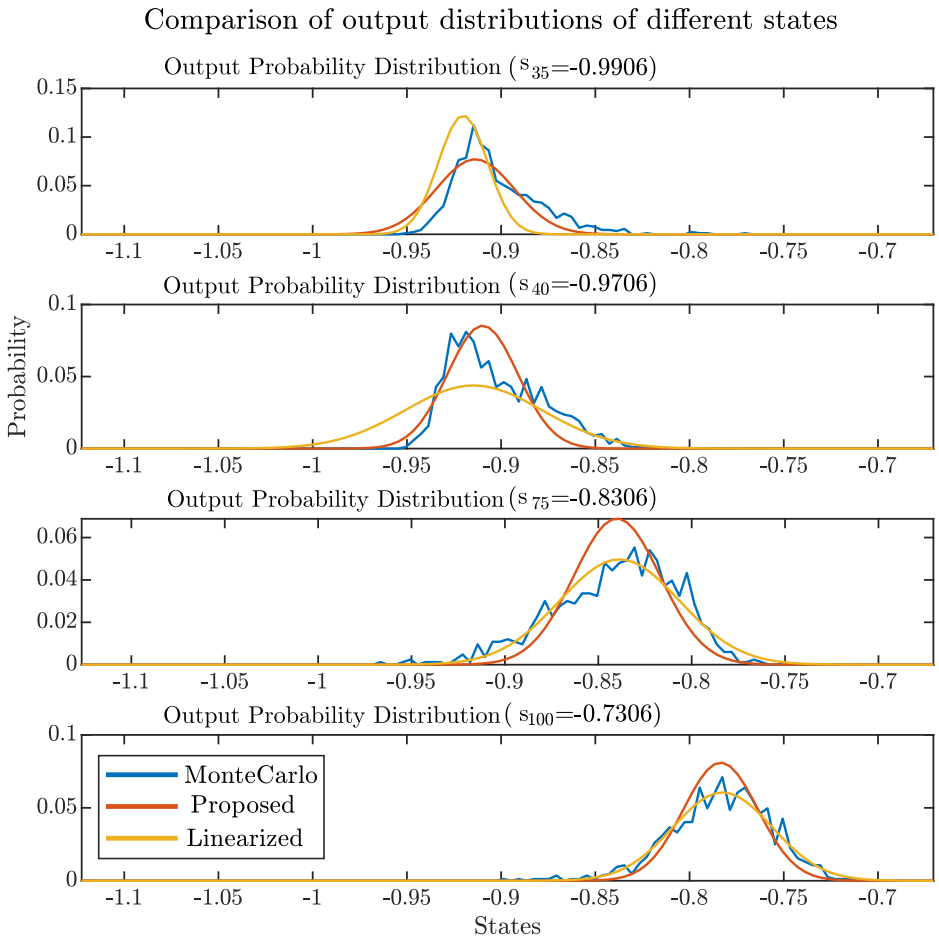

Figure 3.26: Comparison of output probability distributions i.e. selected rows of matrices

Mean and variances of estimated PDFs are plotted in Figure 3.27 in order to assess the estimation performance. Around the region where the nonlinear behavior is dominant ($[-1.12, -0.95] rad/s$), the proposed method with unscented transformation works better than the linearization-based method as expected. In the linear region, both methods have a satisfactory performance; however, the linearization-based method works better. This may caused from the asymmetrical shape of the output distribution, which can be observed in Figure 3.25. The estimation can be improved by tuning the weights in the unscented transformation up to some level. After all, the aim is to find an approach with fewer assumptions and generalizable for highly nonlinear systems. Therefore, the unscented transformation is adopted for further investigation of the stochastic behavior of the bipedal system.

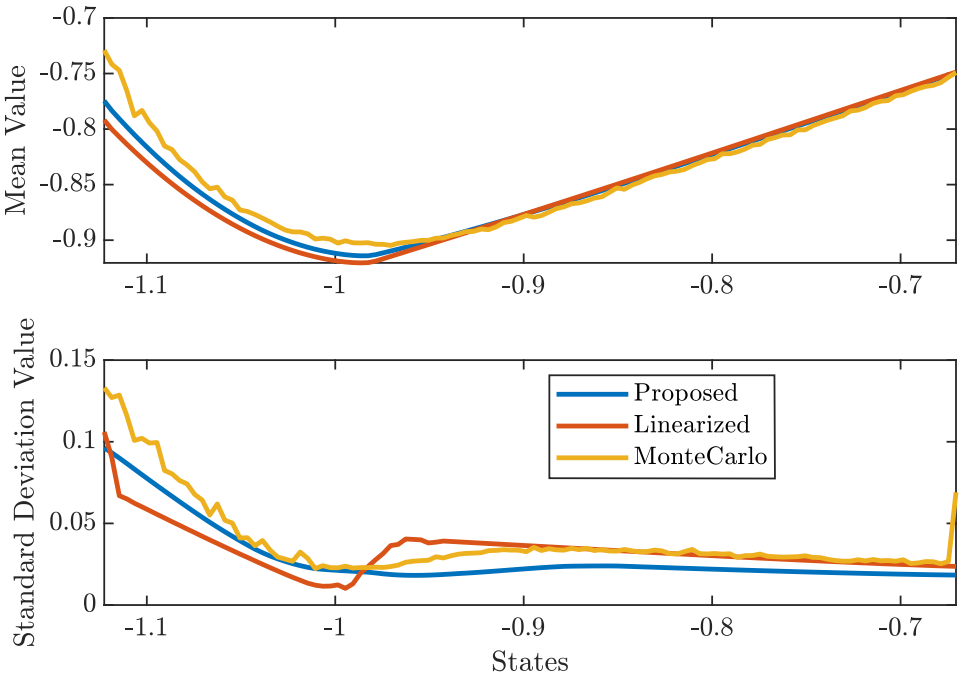

Figure 3.27: Comparison of mean and variances of Output PDF's

After evaluating the success of estimation methods, the deductions on the stochastic properties of the system are made by the estimated state transition matrices by the proposed method. As stated before, the eigenvector associated with the second largest magnitude eigenvalue is used to calculate metastable distribution. Metastable distribution in Figure 3.28 states that if the body angular velocity starts from a random point, it will more likely be around $[-0.9, -0.8] rad/s$ unless it fails.

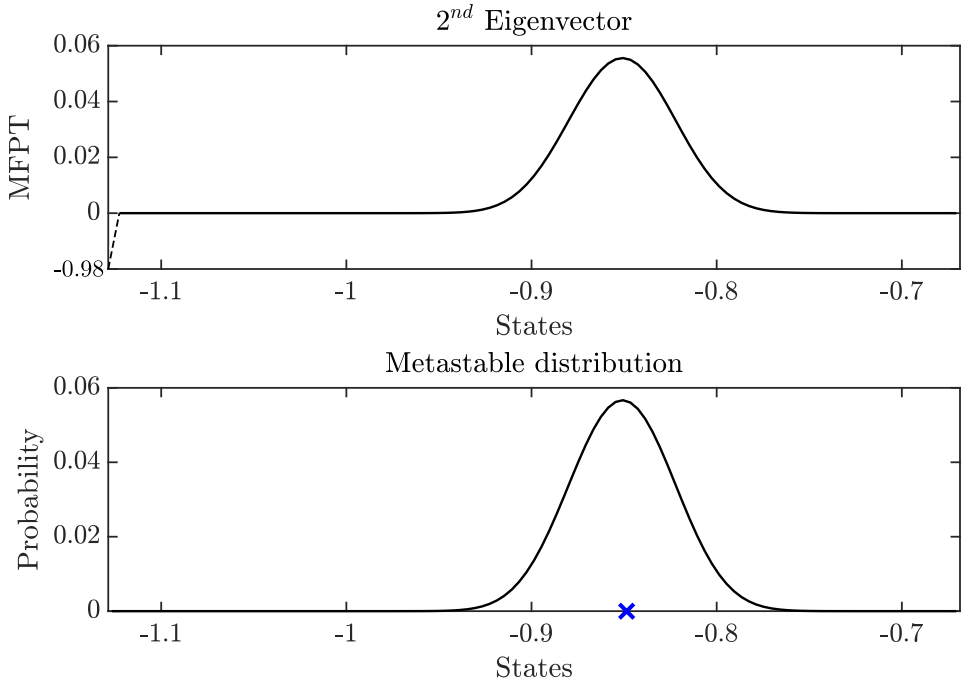

Figure 3.28: Second eigenvector and metastable distribution for the bipedal system for noise variance of 10^{-3}

The state dependent MFPT vector in Figure 3.29 also shows the initial conditions such that the system is more likely to maintain its locomotion under noise. The curves in Figures 3.28 and 3.29 imply the same possibilities.

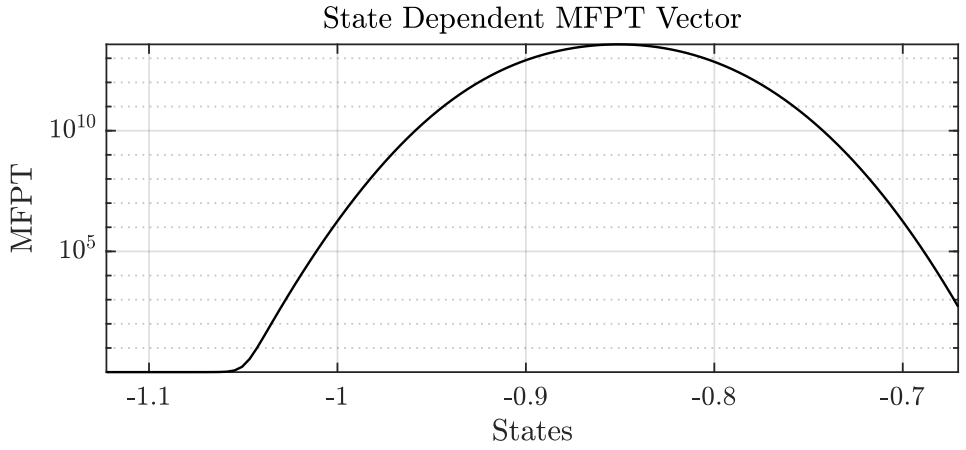

Figure 3.29: State dependent MFPTs for the bipedal system for noise variance of 10^{-3}

A metastable neighborhood map focuses on relating the probabilities of successive

steps. Figure 3.30 shows the metastable neighborhood of the body angular velocity for the objective bipedal system. This neighborhood is actually the stochastic counterpart of the fixed point of the deterministic return map. The metastable neighborhood indicates the joint probability of the two successive body angular velocity value measured just before the impact.

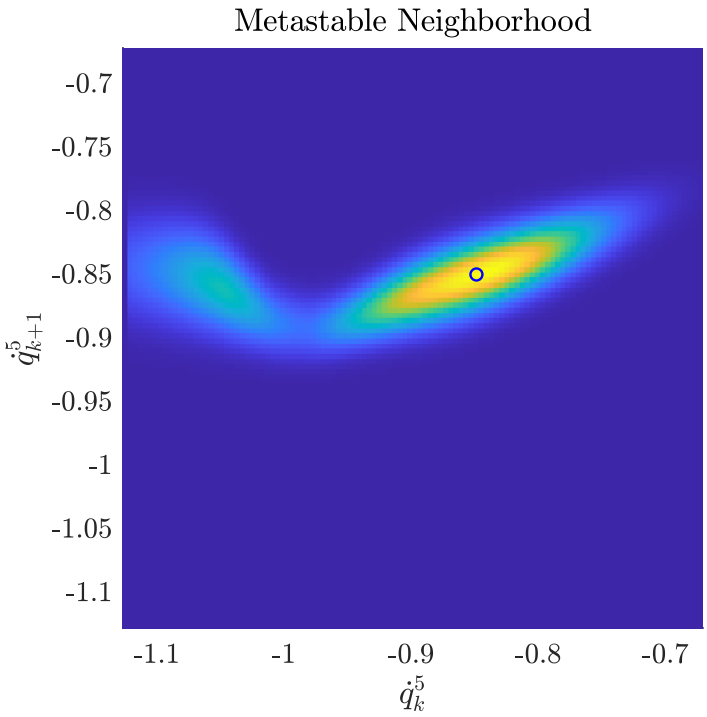

Figure 3.30: Metastable neighborhood of state transitions for the bipedal system

In addition, the system's sensitivity to initial conditions can also be inferred by investigating the eigenvalues of the state transition matrix of the absorbing Markov chain: $\lambda_1 = 1$, $\lambda_2 = 0.9775$, $\lambda_3 = 0.3552$, $\lambda_4 = 0.2918$. The value of λ_3 means that nearly 65% of the contribution to the probability function at the initial condition is lost with each successive step. As noise variance increases, λ_3 decreases, which means the system tends to forget its initial condition more and converges to its metastable distribution in a few steps, faster than the one-dimensional hopper.

The feasibility of building this matrix with unscented transformation allows us to assess the different controllers and analyze the system's stability under different noise levels. Each ten states of the 5-link bipedal are subject to noise with the same variance. Figure 3.31 shows the dependence of system-wide MFPT in (3.26) on noise

standard deviation for different controller parameters in Table 3.3. The MFPT values over 10^{14} are not reliable due to MATLAB's numerical limits. From the figure, it can be deduced that the first controller C_1 is more stable than the other experimented controllers. Also, it can be related that the proportional controller C_5 shows the least stable behavior. Control input saturation prevents making this observation without conducting the stochastic analysis.

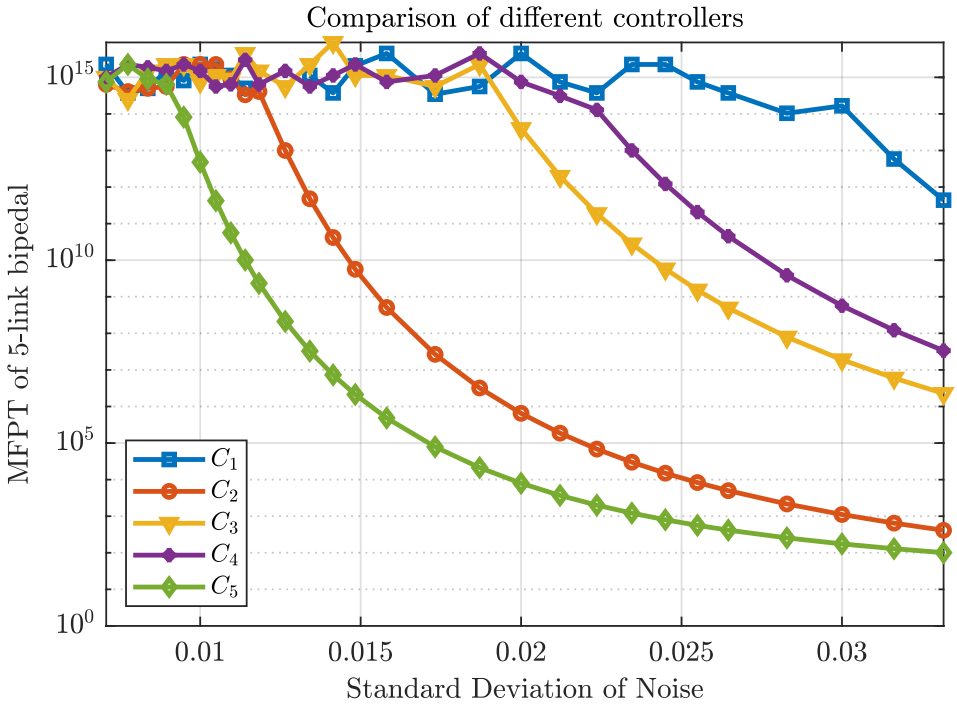

Figure 3.31: System-wide MFPT for the bipedal walker with respect to standard deviation of state noises, σ, obtained by unscented transformation method

3.5.3.3 Two Dimensional Meshing and Estimation

One of the advantages of the proposed method is its generalizability to Markov states with multiple dimensions. If two states are chosen as a result of the model reduction, each Markov state will be tuples containing body angle q^5 and body angular velocity $\dot{q}^5$. After choosing the states representing the behavior of the bipedal system, the state space for those states should be meshed. Figure 3.32 shows the region called the basin of attraction. If the system is started inside this region, keeping all other eight initial states as in the fixed point, the system will maintain its walking motion. This

region is meshed by a simple grid.

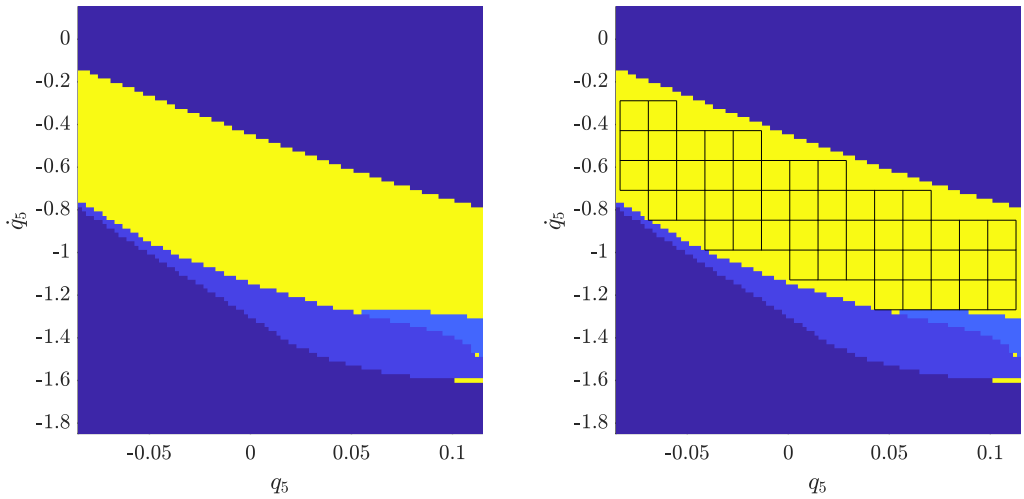

Figure 3.32: Basin of attraction for the bipedal walker's body angle (on the left) and angular velocity and 2 dimensional meshing inside the stable region (on the right)

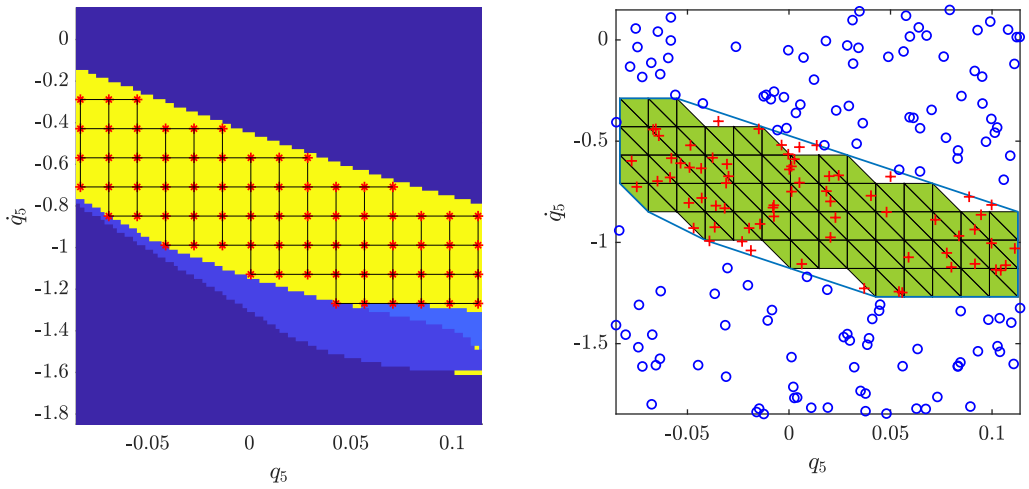

Figure 3.33: Basin of attraction with red marks at the Markov States (on the left) and the samples used in Monte Carlo experiments (on the right)

Then, as Figure 3.33 depicted, the grid points and their vicinity are chosen as the Markov states of the absorbing Markov chain. Outside of the basin of attraction region belongs to the absorbing state in the Markov chain. In order to build the state transition matrix, one can conduct Monte Carlo experiments that sample the initial

conditions as in Figure 3.33.

This implementation aims to show the ability of the proposed method for the Markov chains with multi-dimensional Markov states. As a result of the estimation, Figure 3.35 becomes very close to Figure 3.34. Then, it can be concluded that the proposed method is effective even if the model reduction does not result in a single state representing the system.

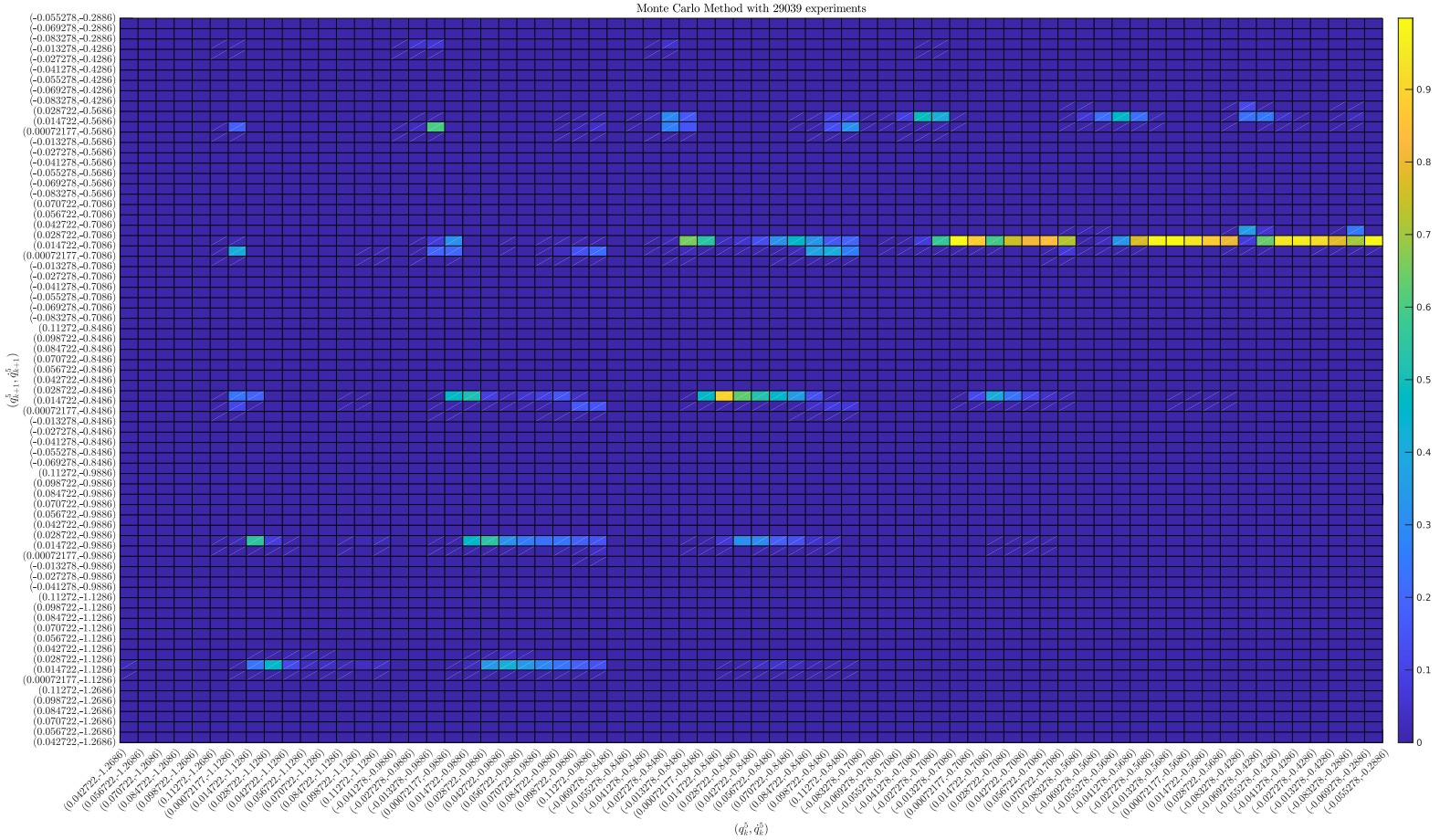

Figure 3.34: State transition matrix with 2D Markov states body angle and angular velocity $(q^5, \dot{q}^5)$ obtained by Monte Carlo experiments

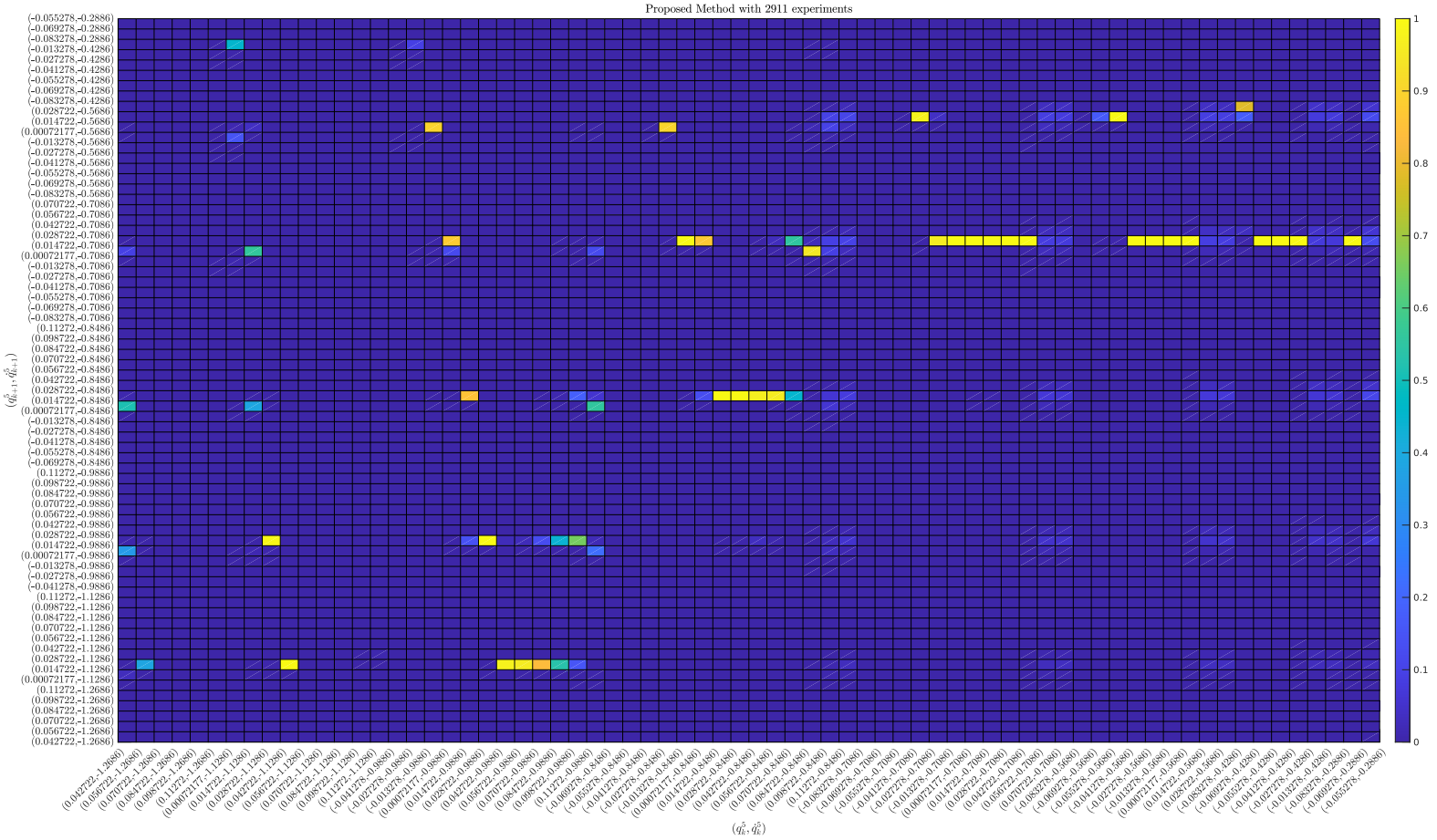

Figure 3.35: State transition matrix with 2D Markov states body angle and angular velocity $(q^5, \dot{q}^5)$ obtained by proposed method

3.6 Future Work

This thesis is aimed to be the groundwork for future studies. There are many possible directions for stochastic analysis to drive into. Some of them are listed as future work for this study.

3.6.1 Experimental Setup

Reproducing the simulation results with actual experimental setups is a way to prove the strength of the proposed methodology. Comparing the mean first passage time values calculated by the estimated matrices and real setup experiments is essential for transferring this methodology to the physical world.

The study in [92] utilizes the mean first passage time metric as a measure of the long-term metastability of a stochastic dynamical system to quantify the effect of haptic feedback in a virtual paddle juggling experiment. Following a similar procedure, replacing the human experiment with robot hopping or walking, the stability of the systems can be estimated and compared with the actual results.

3.6.2 Controller Optimization

Output distribution estimation scheme can also be utilized in the controller parameter tuning as an alternative to the deterministic methods. The proposed methodology can be exploited to find the optimal controller parameters for future work. For example, minimizing the variance of output distribution near a fixed point can be a candidate for the optimization objective.

The initial variance was around 5.7×10^{-4}. After optimization, it is reduced to 4.27×10^{-5} and the minimizing controller parameters are calculated as follows:

$$\begin{aligned} K_P &= \begin{bmatrix} 10.39747 & 89.17644 & 83.06905 & 50.86171 \end{bmatrix} \\ K_D &= \begin{bmatrix} 5.38221 & 21.50242 & 20.79997 & 9.11454 \end{bmatrix} \end{aligned} \tag{3.34}$$

This controller optimization study with a broader implementation with stability in-

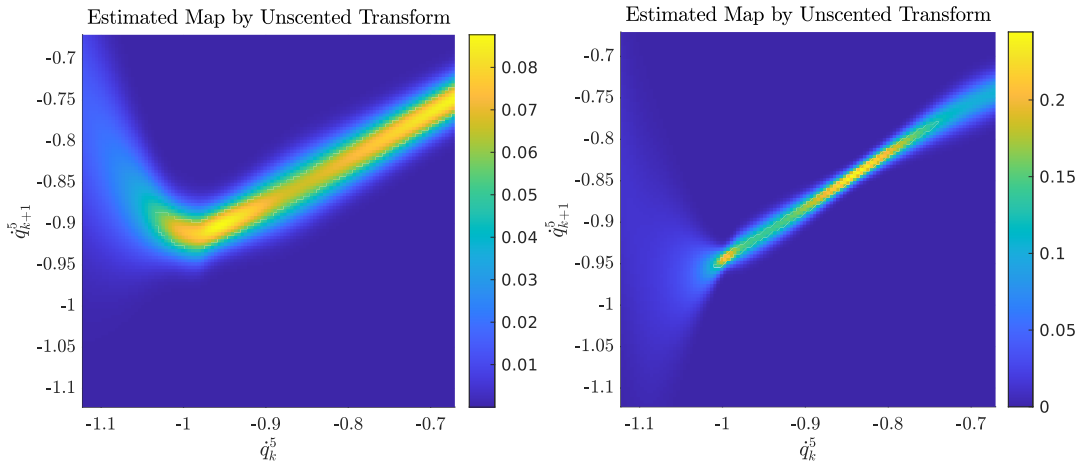

Figure 3.36: Initial return map with controller parameters $Kp = diag([60, 90, 90, 50])$, $Kd = diag([10, 20, 20, 10])$ (on the left) and the return map after optimization based on variance around fixed point (on the right)

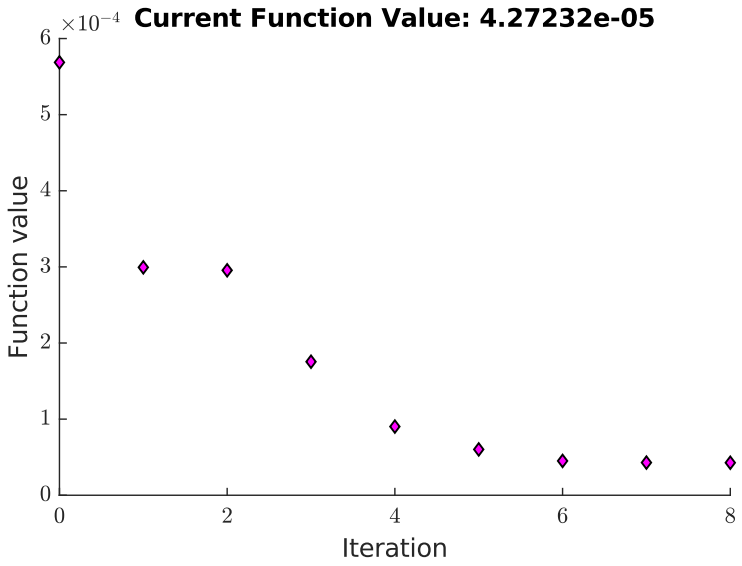

Figure 3.37: Variance values during optimization

vestigation and comparison of different optimization methods promises a practical optimal controller that might be used online. For example, this optimization objective is questionable because its performance results for the MFPT are worse than the initial parameter set. Also, the search algorithm in optimization needs to be studied deeper.

3.6.3 Extension with Other Stochastic Tools

This study focuses on how to estimate the output distribution using the known system dynamics and noise characteristics. In the next step, Gaussian process models are foreseen to approximate system dynamics. Therefore, the cost of running the system decreases significantly. In addition, calculating the metastable distribution of a system without calculating the individual state transition probabilities allows us to approximate the mean first passage time value much more efficiently.

3.7 Conclusion

This chapter compiles the methodology for the stochastic analysis of legged systems. This methodology is mainly based on the metastable nature of the legged locomotion. Under uncertainties, Metastable-legged robots can maintain their locomotion for a reasonable amount of time but eventually will fail. The idea of modeling the system as an absorbing Markov chain comes from this metastability.

Stochastic return maps calculated as the state transition matrices for the absorbing Markov chains represent the system behavior in the existence of uncertainties. It is important to extract the information related to stochastic stability, such as metastable distribution and mean first passage time. In this study, those properties are investigated and used in commenting on the metastable behavior and controller comparison.

Inspiring from Kalman filters, two different estimation method is formulated to estimate state transition matrices and compared with the existing methods. At first, the linearization-based method is inspired by the extended Kalman filters and reformulated for the non-additive noise case. This method relies on the numerically linearized version of the system and is observed to fail for the systems with dominating nonlinear behavior. This limitation is an expected tradeoff originating from the linearization and should be eliminated by extending the methodology for the dynamical systems.

In basic terms, suppose the objective system is a simple nonlinear system with additive noise inside. In that case, the effect of noise on the resulting apex-to-apex map is unknown because the noise input might go through a nonlinear transformation. This fact leads to the use of methods that take the nonlinear dynamics into account. In [92], the closed-loop system identification uses Poincaré theory and assumes the closed-loop juggling behavior as operating near a limit cycle. So, the authors fit a linear system model of apex-to-apex dynamics, assuming that the subject humans remain within a local region where the linear dynamics dominate. Therefore, an autoregressive Gaussian model using the apex height data can model the system. However, this approach cannot be generalized to all hybrid rhythmic dynamical systems, which may have some dominating nonlinear behavior and are not self-stable.

Unscented transformation provides a strong tool for estimation, requires much less

simulation time to conduct, and leads to a low variance solution. Most importantly, it considers the full system dynamics and brings no simplification at this step while reducing the number of experiments. However, even if unscented transformation considers the whole system dynamics, some tradeoffs exist. For example, it assumes that the output distribution is a Gaussian. As long as the output is near symmetric, it does not cause a huge problem in capturing the mean and variances of the output distribution. However, symmetry is non-generalizable to every system. In addition, another simplification step in the methodology related to the model reduction of the 5-link bipedal robot extends the stochastic stability analysis to multi-dimensional systems. The limitation comes from making dimension reduction based on methods that adopt linearization, such as Principal Component Analysis. This model reduction may require to be conducted with less assumption for highly nonlinear systems. In the end, despite these limitations of the principal component analysis and unscented transformation-based estimation, simulation results in this chapter showed that the improved stochastic stability analysis methodology provides a much faster analysis that is inclusive for complex legged systems.

To conclude, this thesis discusses the preeminence of the proposed method based on unscented transformation over existing methods. Furthermore, this chapter employs this estimation method for the stochastic stability analysis of legged systems with different specifications. Despite all the limitations, the proposed methodology satisfies the requirements of the analysis. Most importantly, this study promises a wide range of new research questions in this unexplored territory of legged locomotion.

CHAPTER 4

CONCLUSIONS

In this thesis, two independent studies are conducted related to discrepancies in legged locomotion. Specifically, two key aspects are investigated; adaptation and stochastic behavior. New methodologies are introduced, such as an adaptation layer for the adaptive control for a pronking hexapod in Chapter 2 and output distribution estimation methods based on linearization and unscented transformation stochastic analysis in Chapter 3.

In Chapter 2, the study is an incremental work to improve the tracking performance of an existing deadbeat controller structure that is vulnerable to modeling errors. This modeling error can also be replaced with environmental information. After performance analysis, updating only one selected parameter is observed to help maintain the pronking motion. In addition, stability analysis of the closed-loop system with the adaptation layer is novel to adaptive control studies in terms of using a numerical method to inspect stability.

The second project in Chapter 3 aims to build the groundwork for future studies in metastable legged systems. Actually, two new methods are introduced for the output distribution estimation. The first is the estimation based on numerical linearization of the system dynamics inspired by the extended Kalman filter formulation. The second one is an estimation method based on unscented transformation inspired by a different Kalman filter considering the nonlinear nature of the system, unscented Kalman filter. Both one-dimensional hopper and bipedal walker are analyzed with those methods and compared with the Monte Carlo simulations adopted in the literature. Estimation performance analysis based on mean and variance values showed that the method based on unscented transformation is good at capturing the output distribution. Then,

the state transition matrices are obtained using the proposed method with unscented transformation, and in the end, some critical comments are made on the objective rhythmic hybrid dynamical systems.

These studies together conclude the importance of handling unexpected external effects. Moreover, they introduce the methods to make robots more adaptable by basic changes and improve the existing analysis procedures to be computationally feasible.

REFERENCES

[1] "Summary of the rhex robot platform." https://www.rhex.web.tr/. Accessed: 2022-03-17.

[2] "Xrhex - kod*lab." https://kodlab.seas.upenn.edu/robots/rhex/xrhex/. Accessed: 2022-03-17.

[3] "Springbok britannica." https://www.britannica.com/animal/springbok-mammal. Accessed: 2022-03-17.

[4] M. M. Ankarali and U. Saranli, "Control of underactuated planar pronking through an embedded spring-mass Hopper template," *Autonomous Robots*, vol. 30, pp. 217–231, 2011.

[5] U. Saranli and D. Koditschek, "Design and Analysis of a Flipping Controller for RHex," tech. rep., UM, Ann Arbor, MI, 2010.

[6] U. Saranli, A. Rizzi, and D. Koditschek, *Multi-Point Contact Models for Dynamic Self-Righting of a Hexapod*, vol. 17, pp. 409–424. Springer, 2005.

[7] M. H. Raibert, *Legged Robots that Balance*. MIT Press, 1986.

[8] P. Gregorio, M. Ahmadi, and M. Buehler, "Design, control, and energetics of an electrically actuated legged robot," *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 27, pp. 626–634, aug 1997.

[9] M. Ahmadi and M. Buehler, "Controlled passive dynamic running experiments with the ARL-monopod II," *IEEE Transactions on Robotics*, vol. 22, pp. 974–986, oct 2006.

[10] T. McGeer, "Passive bipedal running," *Royal Soc. B*, vol. 213, no. 240, pp. 107–134, 1990.

[11] B. Brown and G. Zeglin, "The bow leg hopping robot," in *Proceedings.*

1998 IEEE International Conference on Robotics and Automation (Cat. No.98CH36146), vol. 1, pp. 781–786, IEEE, 1998.

[12] A. T. Peekema, *Template-Based Control of the Bipedal Robot ATRIAS*. Master's thesis, Oregon State University, 2015.

[13] J. Oehlke, M. A. Sharbafi, P. Beckerle, and A. Seyfarth, "Template-based hopping control of a bio-inspired segmented robotic leg," *Proceedings of the IEEE RAS and EMBS International Conference on Biomedical Robotics and Biomechatronics*, vol. 2016-July, pp. 35–40, 2016.

[14] U. Saranli and D. Koditschek, "Template based control of hexapedal running," in *2003 IEEE International Conference on Robotics and Automation (Cat. No.03CH37422)*, vol. 1, pp. 1374–1379, IEEE, 2003.

[15] V. Kurtz, P. M. Wensing, and H. Lin, "Approximate Simulation for Template-Based Whole-Body Control," *IEEE Robotics and Automation Letters*, vol. 6, pp. 558–565, apr 2021.

[16] U. Saranli, M. Buehler, and D. Koditschek, "RHex: A Simple and Highly Mobile Hexapod Robot," *The International Journal of Robotics Research*, vol. 20, pp. 616–631, jul 2001.

[17] C. D. Fitzgibbon and J. H. Fanshawe, "Stotting in Thomson ' s gazelles : an honest signal of condition," *Behavioral Ecology and Sociobiology*, pp. 69–74, 1988.

[18] D. Mcmordie, *Towards pronking with a hexapod robot*. Phd thesis, McGill University, Montreal, Canada, 2002.

[19] A. M. Johnson and D. E. Koditschek, "Legged self-manipulation," *IEEE Access*, vol. 1, no. 33, pp. 310–334, 2013.

[20] Y. C. Chou, K. J. Huang, W. S. Yu, and P. C. Lin, "Model-based development of leaping in a hexapod robot," *IEEE Transactions on Robotics*, vol. 31, no. 1, pp. 40–54, 2015.

[21] K.-Y. Tseng and P.-C. Lin, "Development of Leaping/Flipping Behaviors in a Quadruped Robot with Passive Compliant Legs," in *2019 IEEE/ASME Interna-*

tional Conference on Advanced Intelligent Mechatronics (AIM), vol. 2019-July, pp. 364–369, IEEE, jul 2019.

[22] P. Holmes, R. J. Full, D. Koditschek, and J. Guckenheimer, "The Dynamics of Legged Locomotion: Models, Analyses, and Challenges," *SIAM Review*, vol. 48, pp. 207–304, jan 2006.

[23] R. Full and D. Koditschek, "Templates and anchors: neuromechanical hypotheses of legged locomotion on land," *Journal of Experimental Biology*, vol. 202, pp. 3325–3332, dec 1999.

[24] R. Blickhan and R. Full, "Similarity in multilegged locomotion: Bouncing like a monopode," *Journal of Comparative Physiology A*, vol. 173, pp. 509–517, nov 1993.

[25] R. Blickhan, "The spring-mass model for running and hopping," *Journal of Biomechanics*, vol. 22, pp. 1217–1227, jan 1989.

[26] C. T. Farley, J. Glasheen, and T. A. McMahon, "Running springs: speed and animal size," *Journal of Experimental Biology*, vol. 185, pp. 71–86, dec 1993.

[27] P. M. Wensing and D. E. Orin, *Control of Humanoid Hopping Based on a SLIP Model*, pp. 265–274. Heidelberg: Springer International Publishing, 2013.

[28] J. A. Grimes and J. W. Hurst, "The Design of ATRIAS 1.0 a Unique Monopod, Hopping Robot," in *Adaptive Mobile Robotics*, pp. 548–554, WORLD SCIENTIFIC, sep 2012.

[29] B. Vanderborght, N. G. Tsagarakis, R. Van Ham, I. Thorson, and D. G. Caldwell, "MACCEPA 2.0: compliant actuator used for energy efficient hopping robot Chobino1D," *Autonomous Robots*, vol. 31, pp. 55–65, jul 2011.

[30] W. J. Schwind, *Spring loaded inverted pendulum running: a plant model*. Phd thesis, UM, Ann Arbor, MI, 1998.

[31] H. Geyer, A. Seyfarth, and R. Blickhan, "Spring-mass running: simple approximate solution and application to gait stability," *Journal of Theoretical Biology*, vol. 232, pp. 315–328, feb 2005.

[32] U. Saranli, Ö. Arslan, M. M. Ankarali, and O. Morgul, "Approximate analytic solutions to non-symmetric stance trajectories of the passive Spring-Loaded Inverted Pendulum with damping," *Nonlinear Dynamics*, vol. 62, pp. 729–742, dec 2010.

[33] J. Faigl and P. Čížek, "Adaptive locomotion control of hexapod walking robot for traversing rough terrains with position feedback only," *Robotics and Autonomous Systems*, vol. 116, pp. 136–147, jun 2019.

[34] Q. Zhu, Y. Mao, R. Xiong, and J. Wu, "Adaptive Torque and Position Control for a Legged Robot Based on a Series Elastic Actuator," *International Journal of Advanced Robotic Systems*, vol. 13, p. 26, jan 2016.

[35] S. Aoi, P. Manoonpong, Y. Ambe, F. Matsuno, and F. Wörgötter, "Adaptive control strategies for interlimb coordination in legged robots: A review," *Frontiers in Neurorobotics*, vol. 11, no. AUG, pp. 1–21, 2017.

[36] Helferty and Kam, "Adaptive control of a legged robot using an artificial neural network," in *IEEE International Conference on Systems Engineering*, pp. 165–168, IEEE, 1989.

[37] E. Massi, L. Vannucci, U. Albanese, M. C. Capolei, A. Vandesompele, G. Urbain, A. M. Sabatini, J. Dambre, C. Laschi, S. Tolu, and E. Falotico, "Combining evolutionary and adaptive control strategies for quadruped robotic locomotion," *Frontiers in Neurorobotics*, vol. 13, no. August, pp. 1–19, 2019.

[38] I. Uyanik, U. Saranli, and O. Morgul, "Adaptive control of a spring-mass hopper," in *2011 IEEE International Conference on Robotics and Automation*, pp. 2138–2143, IEEE, may 2011.

[39] B. D. Miller, D. Cartes, and J. E. Clark, "Leg stiffness adaptation for running on unknown terrains," in *IEEE International Conference on Intelligent Robots and Systems*, pp. 5108–5113, 2013.

[40] P. Bateson, "Adaptability and evolution," *Interface Focus*, vol. 7, p. 20160126, oct 2017.

[41] K. S. Narendra and A. M. Annaswamy, *Stable Adaptive Systems*. USA: Prentice-Hall, Inc., 1989.

[42] M. M. Ankarali, *Control of hexapedal pronking through a dynamically embedded spring loaded inverted pedulum template*. Master's thesis, Middle East Technical University, TR, Middle East Technical University, TR, 2010.

[43] M. M. Ankaralı, U. Saranlı, and A. Saranlı, "Control of underactuated planar hexapedal pronking through a dynamically embedded SLIP monopod," in *2010 IEEE International Conference on Robotics and Automation*, pp. 4721–4727, IEEE, may 2010.

[44] H. Geyer and U. Saranli, "Gait Based on the Spring-Loaded Inverted Pendulum," in *Humanoid Robotics: A Reference*, pp. 923–947, Springer, 2018.

[45] U. Saranli, *SimSect Hybrid DynamicalSimulation Environment*. UM, Ann Arbor, MI: Technical Report CSE-TR-436-00, 2000.

[46] U. Saranli, *Dynamic locomotion with a hexapod robot*. Phd thesis, UM, Ann Arbor, MI, 2002.

[47] P. Holmes, "Poincaré, celestial mechanics, dynamical-systems theory and "chaos"," *Physics Reports*, vol. 193, pp. 137–163, sep 1990.

[48] A. Mohanty and B. Yao, "Indirect Adaptive Robust Control of Hydraulic Manipulators With Accurate Parameter Estimates," *IEEE Transactions on Control Systems Technology*, vol. 19, no. 3, pp. 567–575, 2011.

[49] S. A. Burden, S. Revzen, and S. S. Sastry, "Model Reduction Near Periodic Orbits of Hybrid Dynamical Systems," *IEEE Transactions on Automatic Control*, vol. 60, pp. 2626–2639, oct 2015.

[50] P. Hartman, "A lemma in the theory of structural stability of differential equations," *Proceedings of the American Mathematical Society*, vol. 11, pp. 610–610, apr 1960.

[51] K. J. Astrom and B. Wittenmark, *Adaptive Control*. Mineola, New York: Dover Publications, 2 ed., 1995.

[52] I. Uyanik, *Adaptive control of a one-legged hopping robot through dynamically embedded spring-loaded inverted pendulum template*. Master's thesis, Bilkent University, TR, Bilkent University, TR, 2011.

[53] M. M. Ankarali and U. Saranli, "Stride-to-stride energy regulation for robust self-stability of a torque-actuated dissipative spring-mass hopper," *Chaos: An Interdisciplinary Journal of Nonlinear Science*, vol. 20, p. 033121, sep 2010.

[54] J. H. Mathews and K. D. Fink, *Numerical methods using MATLAB*. USA: Prentice-Hall, Inc., 1999.

[55] M. M. Ankarali and N. J. Cowan, "System identification of rhythmic hybrid dynamical systems via discrete time harmonic transfer functions," in *53rd IEEE Conference on Decision and Control*, pp. 1017–1022, IEEE, 2014.

[56] J. Grizzle, G. Abba, and F. Plestan, "Asymptotically stable walking for biped robots: analysis via systems with impulse effects," *IEEE Transactions on Automatic Control*, vol. 46, no. 1, pp. 51–64, 2001.

[57] B. Griffin and J. Grizzle, "Nonholonomic virtual constraints and gait optimization for robust walking control," *International Journal of Robotics Research*, vol. 36, pp. 895–922, 2017.

[58] H. Dai and R. Tedrake, "Optimizing robust limit cycles for legged locomotion on unknown terrain," *Proceedings of the IEEE Conference on Decision and Control*, pp. 1207–1213, 2012.

[59] S. Aoi, T. Tanaka, S. Fujiki, T. Funato, K. Senda, and K. Tsuchiya, "Advantage of straight walk instability in turning maneuver of multilegged locomotion: a robotics approach," *Scientific Reports*, vol. 6, p. 30199, jul 2016.

[60] S. Mannervik, "Experimental Lifetime Studies of Metastable Levels," *Physica Scripta*, vol. T105, no. 1, p. 67, 2003.

[61] A. J. Lovinger, *Phase Transitions in Polymers*. Elsevier, 2008.

[62] T. Chaney and C. Molnar, "Anomalous Behavior of Synchronizer and Arbiter Circuits," *IEEE Transactions on Computers*, vol. C-22, pp. 421–422, apr 1973.

[63] K. Byl, *Metastable Legged-Robot Locomotion*. Phd thesis, Massachusetts Institute of Technology 2008., 2008.

[64] K. Byl and R. Tedrake, "Metastable Walking Machines," *The International Journal of Robotics Research*, vol. 28, pp. 1040–1064, aug 2009.

[65] K. Byl and R. Tedrake, "Metastable Walking on Stochastically Rough Terrain," in *Robotics: Science and Systems IV*, vol. 4, pp. 230–237, Robotics: Science and Systems Foundation, jun 2008.

[66] M. Benallegue and J.-P. Laumond, "Metastability for High-Dimensional Walking Systems on Stochastically Rough Terrain," in *Robotics: Science and Systems IX*, Robotics: Science and Systems Foundation, jun 2013.

[67] C. O. Saglam and K. Byl, "Robust Policies via Meshing for Metastable Rough Terrain Walking," in *Robotics: Science and Systems X*, Robotics: Science and Systems Foundation, jul 2014.

[68] C. O. Saglam and K. Byl, "Metastable Markov chains," in *53rd IEEE Conference on Decision and Control*, vol. 16, pp. 2979–2985, IEEE, dec 2014.

[69] C. O. Saglam and K. Byl, "Meshing hybrid zero dynamics for rough terrain walking," *Proceedings - IEEE International Conference on Robotics and Automation*, vol. 2015-June, no. June, pp. 5718–5725, 2015.

[70] C. O. Saglam and K. Byl, "Quantifying and Optimizing Robustness of Bipedal Walking Gaits on Rough Terrain," in *Springer Proceedings in Advanced Robotics*, vol. 3, pp. 235–251, 2018.

[71] R. E. Kalman, "A new approach to linear filtering and prediction problems," *Journal of Fluids Engineering, Transactions of the ASME*, vol. 82, no. 1, pp. 35–45, 1960.

[72] A. Jazwinski, *Stochastic Processes and Filtering Theory*. San Diego, CA: Elsevier, 1st ed., 1970.

[73] H. W. Sorenson, *Kalman Filtering: Theory and Application*. 1st ed., 1985.

[74] M. S. Grewal and A. P. Andrews, "Applications of kalman filtering in aerospace 1960 to the present [historical perspectives]," *IEEE Control Systems*, vol. 30, pp. 69–78, 6 2010.

[75] M. S. Grewal and A. P. Andrews, *Kalman Filtering: Theory and Practice Using MATLAB*. New Jersey: John Wiley & Sons., 4 ed., 2015.

[76] S. Julier and J. Uhlmann, "Unscented Filtering and Nonlinear Estimation," *Proceedings of the IEEE*, vol. 92, pp. 401–422, mar 2004.

[77] E. Wan and R. Van Der Merwe, "The unscented Kalman filter for nonlinear estimation," in *Proceedings of the IEEE 2000 Adaptive Systems for Signal Processing, Communications, and Control Symposium (Cat. No.00EX373)*, vol. 42, pp. 153–158, IEEE, mar 2006.

[78] S. Safaoui, B. J. Gravell, V. Renganathan, and T. H. Summers, "Risk-Averse RRT* Planning with Nonlinear Steering and Tracking Controllers for Nonlinear Robotic Systems Under Uncertainty," in *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 3681–3688, IEEE, sep 2021.

[79] M. Choi, M. Seo, H. S. Kim, and T. Seo, "UKF-Based Sensor Fusion Method for Position Estimation of a 2-DOF Rope Driven Robot," *IEEE Access*, vol. 9, pp. 12301–12308, 2021.

[80] A. Daid, E. Busvelle, and M. Aidene, "On the convergence of the unscented Kalman filter," *European Journal of Control*, vol. 57, pp. 125–134, jan 2021.

[81] P. M. Sieberg, S. Blume, S. Reicherts, N. Maas, and D. Schramm, "Hybrid State Estimation-A Contribution Towards Reliability Enhancement of Artificial Neural Network Estimators," *IEEE Transactions on Intelligent Transportation Systems*, pp. 1–10, 2021.

[82] P. A. Gagniuc, *Markov Chains: From Theory to Implementation and Experimentation.* NJ: John Wiley & Sons., 2017.

[83] S. Meyn and R. L. Tweedie, *Markov Chains and Stochastic Stability*. Cambridge University Presss, 2 ed., 2009.

[84] W. L. Dunn and J. K. Shultis, "The Basis of Monte Carlo," in *Exploring Monte Carlo Methods*, pp. 21–46, Elsevier, 2012.

[85] H. Fischer, *A History of the Central Limit Theorem.* Springer New York, 2011.

[86] S. S. Candan, *Design of Energy Efficient Hopper By Using Parallel Elasticity and Wrapping Cam Mechanisms*. Master's thesis, Middle East Technical University, 2019.

[87] G. Secer, *Control of Spring-Mass Through Virtual Tuning of Leg Damping*. Phd thesis, Middle East Technical University, 2020.

[88] E. Tanfener, *Design and Experimental Verification of a Clutched Parallel Elastic Actuation Mechanism for Legged Locomotion*. Master's thesis, Middle East Technical University, 2022.

[89] I. T. Jolliffe and J. Cadima, "Principal component analysis: a review and recent developments," *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, vol. 374, p. 20150202, apr 2016.

[90] C. Chevallereau, G. Abba, Y. Aoustin, F. Plestan, E. Westervelt, C. C. De Wit, and J. Grizzle, "RABBIT: a testbed for advanced control theory," *IEEE Control Systems*, vol. 23, pp. 57–79, oct 2003.

[91] E. R. Westervelt, J. W. Grizzle, C. Chevallereau, J. H. Choi, and B. Morris, *Feedback Control of Dynamic Bipedal Robot Locomotion*. CRC Press, oct 2018.

[92] M. M. Ankarali, H. Tutkun Şen, A. De, A. M. Okamura, and N. J. Cowan, "Haptic feedback enhances rhythmic motor control by reducing variability, not improving convergence rate," *Journal of Neurophysiology*, vol. 111, pp. 1286–1299, mar 2014.

APPENDIX A

DERIVATION OF THE APEX TO APEX MAP FOR ONE DIMENSIONAL HOPPER

This section aims to collect all the calculations to build a apex-to-apex returnmap **f** for the one dimensional hopper. The input output relation is as follows,

$$y_{next} = \mathbf{f}(y_0) \tag{A.1}$$

For this system, touchdown states are y_{td} and $\dot{y}_{td}$.

$$\begin{aligned} y_{td} &= L_0 \\ \dot{y}_{td} &= -\sqrt{2gy_0 - 2gy_{td}} \end{aligned} \tag{A.2}$$

To extract the apex-to-apex map of this model, we need to integrate of hybrid dynamics of F-SLIP. Force input f is applied as a sine input with periode $\frac{2\pi}{\omega}$ and amplitude L.

$$\begin{aligned} f &= -Lsin(\omega t) \\ m\ddot{y} &= -mg - ky - ky_{td} - d\dot{y} + f \\ m\ddot{y} + d\dot{y} + ky &= -mg + ky_{td} - Lsin(\omega t) \end{aligned} \tag{A.3}$$

Definitions of standard second order systems are the damping ratio, $\xi := \frac{d'}{2\sqrt{km}}$ and natural frequency, $w_n := \sqrt{\frac{k}{m}}$. Solving this forced vibration,

$$
\begin{aligned}
m\ddot{x} + b\dot{x} + kx &= f(t) \\
x(t) &= x_h(t) + x_p(t) \\
x_h(t) &= c_1 x_1(t) + c_2 x_2(t) \\
&... \\
x_h(t) &= Ce^{-\xi\omega_n t} cos(\omega_d t - \phi) \\
x_p(t) &= Acos(\omega t) + Bsin(\omega t) + D \\
\dot{x}_p(t) &= -A\omega sin(\omega t) + B\omega cos(\omega t) \\
\ddot{x}_p(t) &= -A\omega^2 cos(\omega t) - B\omega^2 sin(\omega t) \\
m\ddot{x}_p(t) + d\dot{x}_p(t) + kx_p(t) &= -mg + ky_{td} - Lsin(\omega t)
\end{aligned}
\tag{A.4}
$$

Particular solution $x_p(t)$ can be found as following,

$$
\begin{aligned}
& m\left(-A\omega^2 cos(\omega t) - B\omega^2 sin(\omega t)\right) + d\left(-A\omega sin(\omega t) + B\omega cos(\omega t)\right) + ... \\
& k\left(Acos(\omega t) + Bsin(\omega t) + D\right) = -mg + ky_{td} - Lsin(\omega t) \\
& sin(\omega t)\left(\underbrace{-mB\omega^2 - dA\omega + kB + L}_{=0}\right) + cos(\omega t)\left(\underbrace{-mA\omega^2 + dB\omega + kA}_{=0}\right) + kD = -mg + ky_{td} \\
& -mA\omega^2 + dB\omega + kA = 0 \\
& -mB\omega^2 - dA\omega + kB + L = 0 \\
& \begin{bmatrix} k - m\omega^2 & d\omega & 0 \\ -d\omega & k - m\omega^2 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} A \\ B \\ D \end{bmatrix} = \begin{bmatrix} 0 \\ -L \\ -\frac{mg}{k} + y_{td} \end{bmatrix} \longrightarrow \begin{bmatrix} \alpha & \beta & 0 \\ -\beta & \alpha & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} A \\ B \\ D \end{bmatrix} = \begin{bmatrix} 0 \\ -L \\ -\frac{mg}{k} + y_{td} \end{bmatrix} \\
& A = \frac{\beta}{\alpha^2 + \beta^2} L = \frac{d\omega}{(k - m\omega^2)^2 + (d\omega)^2} L \\
& B = \frac{-\alpha}{\alpha^2 + \beta^2} L = \frac{m\omega^2 - k}{(k - m\omega^2)^2 + (d\omega)^2} L \\
& D = -\frac{mg}{k} + y_{td}
\end{aligned}
\tag{A.5}
$$

Unknowns C and ϕ in homogeneous solution $x_h(t)$ can be found as following,

$$
\begin{aligned}
x(t) &= Ce^{-\xi\omega_n t}cos(\omega_d t - \phi) + Acos(\omega t) + Bsin(\omega t) + D \\
x(0) &= y_{td} = Ccos(\phi) + A + D \\
&= Ccos(\phi) + A - \frac{mg}{k} + y_{td} \longrightarrow 0 = Ccos(\phi) + A - \frac{mg}{k} \\
&\longrightarrow Ccos(\phi) = \frac{mg}{k} - A \\
\dot{x}(t) &= -Ce^{-\xi\omega_n t}\left(\omega_n\xi cos(\omega_d t - \phi) + \omega_d sin(\omega_d t - \phi)\right) - A\omega sin(\omega t) + B\omega cos(\omega t) \\
\dot{x}(0) &= -C\left(\omega_n\xi cos(\phi) + \omega_d sin(\phi)\right) + B\omega \\
&= \left(-\omega_n\xi(\frac{mg}{k} - A) - C\omega_d sin(\phi)\right) + B\omega \\
&= -C\omega_d sin(\phi) - \omega_n\xi(\frac{mg}{k} - A) + B\omega = \dot{y}_{td} \\
Csin(\phi) &= \frac{-\dot{y}_{td} + B\omega - \omega_n\xi(\frac{mg}{k} - A)}{\omega_d} \\
\phi &= atan2(\frac{-\dot{y}_{td} + B\omega - \omega_n\xi(\frac{mg}{k} - A)}{\omega_d}, \frac{mg}{k} - A) \\
C &= \sqrt{(\frac{-\dot{y}_{td} + B\omega - \omega_n\xi(\frac{mg}{k} - A)}{\omega_d})^2 + (\frac{mg}{k} - A)^2}
\end{aligned}
\tag{A.6}
$$

Bottom time t_b cannot be extracted from the solution easily. So, the MATLAB built-in function $fzero()$ is used to find the zero crossing of the velocity function.

$$
t_b = \underset{t}{argmin}\ \dot{x}(t) = fzero(\dot{x}(t)) \tag{A.7}
$$

There are 2 different cases for lift off.

$$
\begin{aligned}
y(t_{lo1}) - y_{td} &= 0 \\
k(y(t_{lo2}) - y_{td}) - d\dot{y}(t_{lo2}) &= 0 \\
t_{lo} &= min(t_{lo1}, t_{lo2})
\end{aligned}
\tag{A.8}
$$

Solving the equation of motion for those conditions analytically is not possible. Hence, by assuming that compression time will be approximately equal to decompression time, the following assumption can be made;

$$
e^{-\xi\omega_n t} \approx e^{-2\xi\omega_n t_b} \tag{A.9}
$$

For t_{lo1};

$$
\begin{aligned}
x(t_{lo1}) - y_{td} &= 0 = Ce^{-2\xi\omega_n t_b}cos(\omega_d t_{lo1} - \phi) + Acos(\omega t_{lo1}) + Bsin(\omega t_{lo1}) - \frac{mg}{k} \\
t_{lo1} &= fzero(x(t) - y_{td})
\end{aligned}
\tag{A.10}
$$

For t_{lo2};

$$
\begin{aligned}
&k(x(t_{lo2}) - y_{td}) - d\dot{x}(t_{lo2}) = 0 \\
&0 = k(Ce^{-2\xi\omega_n t_b}cos(\omega_d t_{lo2} - \phi) + Acos(\omega t_{lo2}) + Bsin(\omega t_{lo2}) + D - y_{td})... \\
&- d\left(-Ce^{-2\xi\omega_n t_b}(\omega_n \xi cos(\omega_d t_{lo2} - \phi) + \omega_d sin(\omega_d t_{lo2} - \phi)) - A\omega sin(\omega t_{lo2}) + B\omega cos(\omega t_{lo2})\right) \\
&t_{lo2} = fzero(k(x(t) - y_{td}) - d\dot{x}(t))
\end{aligned}
\tag{A.11}
$$

The next apex position y_{next} can be found as the following;

$$
\begin{aligned}
t_{lo} &= min(t_{lo1}, t_{lo2}) \\
y_{next} &= \frac{\dot{y}(t_{lo})^2}{2g)} + y(t_{lo})
\end{aligned}
\tag{A.12}
$$

Table A.1 collects all kinematic and dynamic parameters for the one dimensional hopper model.

Table A.1: Parameters of the One Dimensional Hopper Model

Quantity	Symbol	Value	Unit
Body mass	m	2	kg
Leg stiffness	k	2000	N/m
Leg damping	b	5	Nm/s
Rest leg length	l	0.2	m