

**ADAPTIVE DYNAMIC PROGRAMMING FOR  
HUMAN POSTURAL BALANCE CONTROL**

---

**THESIS**

**Submitted in Partial Fulfillment of  
the Requirements for  
the Degree of**

**MASTER OF SCIENCE (Electrical Engineering)**

**at the**

**NEW YORK UNIVERSITY  
TANDON SCHOOL OF ENGINEERING**

**by**

**Eric Mauro**

**May 2018**

**ADAPTIVE DYNAMIC PROGRAMMING FOR  
HUMAN POSTURAL BALANCE CONTROL**

**THESIS**

**Submitted in Partial Fulfillment of  
the Requirements for  
the Degree of**

**MASTER OF SCIENCE (Electrical Engineering)**

**at the**

**NEW YORK UNIVERSITY  
TANDON SCHOOL OF ENGINEERING**

**by**

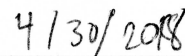
**Eric Mauro**

**May 2018**

Approved:



Department Head Signature



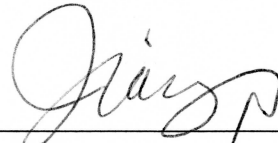
Date

Copy No. \_\_\_\_\_

University ID#: N10683188

Approved by the Guidance Committee:

Major: Electrical Engineering



**Zhong-Ping Jiang**

Professor of Electrical and Computer Engineering

4/30/2018

Date



**Ivan Selesnick**

Professor and Department Chair of  
Electrical and Computer Engineering

4/30/2018

Date



**Quanyan Zhu**

Assistant Professor of  
Electrical and Computer Engineering

05/03/2018

Date



**Ludovic Righetti**

Associate Professor of  
Mechanical and Aerospace Engineering,  
Electrical and Computer Engineering

4/30/2018

Date

## Vita

Eric was born in Willingboro, New Jersey in 1994. He received his Bachelor of Science degree in Electrical Engineering in 2016 at The College of New Jersey. While at TCNJ, he participated in undergraduate research in control systems theory and was awarded First Place Outstanding Undergraduate Student Poster Award and Outstanding Computational-Related Undergraduate Student Poster Award at the American Society of Engineering Education 2015 Northeast section conference.

Eric was admitted to the Master of Science in Electrical Engineering program at New York University's Tandon School of Engineering under the Samuel Morse M.S. fellowship. There, he has worked on human sensorimotor control research in NYU's Control and Networking (CAN) lab under the guidance of Professor Zhong-Ping Jiang. His current research interests include optimal control, signal processing, and machine learning.

## Acknowledgements

I first want to thank my advisor, Professor Zhong-Ping Jiang, for his guidance. He always challenged me and encouraged me to work to my fullest potential.

I would also like to thank all of my fellow lab mates for creating a fun and supportive work environment. I especially want to thank Tao Bian for being my mentor when I first started at NYU, Vijay Yadav for the immense amount of time and effort he put into programming and executing the experiments for this research, and Weinan Gao & Mengzhe Huang for all of their help.

I want to thank Professors Ivan Selesnick, Quanyan Zhu, and Ludovic Righetti for taking the time to review my thesis.

I want to thank my undergraduate advisor, Dr. Ambrose Adegbege, for sparking my interest in control systems research and giving me my first research opportunity. I am so grateful for all of the time he spent teaching me and preparing me for graduate school.

I want to express profound gratitude to my parents and to my girlfriend Nicole for providing me with unfailing support and continuous encouragement throughout my years of study and through the process of researching and writing this thesis.

Finally, I want to thank everyone who has shown me love and kindness along this journey. I could not have done this alone.

Eric Mauro

May 2018

**ABSTRACT****ADAPTIVE DYNAMIC PROGRAMMING FOR HUMAN  
POSTURAL BALANCE CONTROL****by****Eric Mauro****Advisor: Prof. Zhong-Ping Jiang, Ph.D.****Submitted in Partial Fulfillment of the Requirements for  
the Degree of Master of Science (Electrical Engineering)****May 2018**

This work provides a basis for studying human postural balance control about upright stance using adaptive dynamic programming (ADP) theory. Previous models of human sensorimotor control rely on *a priori* knowledge of system dynamics. Here, we provide an alternative framework based on the ADP theory. The main advantage of this new framework is that the system model is no longer required, and an adaptive optimal controller is obtained directly from input and state data. We apply this theory to simulate human balance behavior, and the obtained results

are consistent with the experiment data presented in the past literature.

We further analyzed the learning process for the system with unknown external conditions. More specifically, we tested human balance on a moving cart with an unknown applied force. The ADP algorithm has been shown to model the learning process, and in this case, we compared a simulation to real experimental measurements. After several trials, human learning seemed to converge in a similar manner to the ADP algorithm. However, the human sensorimotor system is incredibly complex, and ADP only paints a part of the whole picture. Humans use a combination of feedback and feed-forward control strategies, and the off-line trial-based policy update that was used can only find the optimal feed-forward control after the feedback control has already been found. While a better variant of ADP might be beneficial to future research, the current framework is still adequate for studying human sensorimotor control.

# Contents

Vita . . . . .	iii
Acknowledgements . . . . .	iv
Abstract . . . . .	v
List of Figures . . . . .	x
<b>1 Introduction</b>	<b>1</b>
1.1 Traditional models . . . . .	2
1.2 Objectives . . . . .	3
1.3 Contributions . . . . .	4
1.4 Thesis Outline . . . . .	5
<b>2 Dynamic Model</b>	<b>6</b>
2.1 Multi-segmented inverted pendulum model . . . . .	7
2.2 Inverted pendulum on a cart . . . . .	9
2.3 Conversion to segment angles . . . . .	12
<b>3 Adaptive Dynamic Programming</b>	<b>15</b>
3.1 Problem formulation . . . . .	15
3.2 Policy iteration . . . . .	16
3.3 ADP for continuous-time linear systems . . . . .	17



	viii
3.4 Adaptive optimal output regulation . . . . .	20
<b>4 Experimental Results and Discussion</b>	<b>27</b>
4.1 Preliminary simulation . . . . .	28
4.2 Selection of parameters . . . . .	28
4.3 Human balance on a cart . . . . .	31
4.4 Comparison to previous methods . . . . .	47
<b>5 Conclusions</b>	<b>54</b>
Bibliography . . . . .	56

# List of Figures

2.1	Three-segment Inverted Pendulum . . . . .	7
2.2	Three-segment Inverted Pendulum on a Cart . . . . .	10
2.3	Three-segment Inverted Pendulum with Segment Angles . . . . .	13
4.1	Trajectories of segment angles with added noise (4.1) for ten trials.	29
4.2	Comparison of linear and nonlinear trajectories with input $u = -K_{10}x$ .	29
4.3	Example balance trajectories on ground . . . . .	32
4.4	Example balance trajectories on cart without applied force . . . . .	33
4.5	Simulation trial 1 trajectories . . . . .	36
4.6	Simulation trial 50 trajectories . . . . .	37
4.7	Quadratic convergence of 10 simulated ADP trials . . . . .	38
4.8	Angle mean-squared error . . . . .	38
4.9	Angular velocity mean-squared error . . . . .	39
4.10	Example Kinect joint detection program . . . . .	40
4.11	Trial 1 trajectories for experiment with only pushes . . . . .	42
4.12	Trial 50 trajectories for experiment with only pushes . . . . .	43
4.13	Angle MSE for experiment with only pushes . . . . .	44
4.14	Angular velocity MSE for experiment with only pushes . . . . .	45

	x
4.15 Individual MSE's for experiment with only pushes . . . . .	46
4.16 Trial 1 trajectories for experiment with pattern of pushes and pulls	48
4.17 Trial 50 trajectories for experiment with pattern of pushes and pulls	49
4.18 Angle MSE for experiment with pattern . . . . .	50
4.19 Angular velocity MSE for experiment with pattern . . . . .	51
4.20 Individual MSE's for experiment with pattern . . . . .	52

# Chapter 1

## Introduction

Optimal control models have been used extensively to study human sensorimotor control [1, 2, 8, 9, 11, 16, 17, 18, 20, 22, 27, 30, 31, 32, 34, 36, 39, 40, 41, 42]. The human body is an adaptive system that works to use behavioral predictions and sensory feedback to continually improve performance. The aim of optimal control theory is to achieve a certain performance criterion by searching for the best control strategy. There are many motivations within medical and robotics research to study human sensorimotor control, such as understanding the degeneration of the human control system caused by aging and neuromuscular disorders such as Parkinsons disease [8, 37]. In the control of human balance, postural instability leads to additional health risks, especially in the associated risk of falling [1, 2]. Human models of control can also be used to better design biologically-inspired robotics [9, 20] and better understand human-in-the-loop systems such as driving [30, 31, 32]. While biological systems are inherently complex and difficult to model, adaptive control algorithms can help further the knowledge of human learning and sensorimotor dynamics.

## 1.1 Traditional models

Typical models of sensorimotor control explore specific everyday tasks such as finger pointing and arm movement [11, 22], postural balance [8, 17, 18, 27, 36, 39], standing up from a chair [24, 26], and walking [9, 39]. These processes require a combination of physical and sensory information. For example, human balance relies on feedback from vision for planning movement and seeing any obstacles that might be in the way, the vestibular system for sensing accelerations, and the somatosensory system for proprioception. Since there is a certain amount of redundancy, humans can still maintain balance when one or more systems fail, and some studies have shown the effects of partial sensory information [18, 27, 39]. However, sensory and motor uncertainty affects how humans process this information. It is not possible for an individual to get exact system information, but humans are able to make approximations. Some models explain this ability through state estimation from a known internal model [18, 40] or through decision theory [41, 42]. Humans base their movements on statistical averages over repeated trials.

Traditionally, sensorimotor control was studied through rigid cost functions based on movement trajectories but has since been looked at through more flexible cost functions based on energy, accuracy, and smoothness of the observed movement. Humans base their actions on multiple goals specific to the task, body, and environment and use every opportunity to improve their performance [22, 34]. Applied perturbations are used to see the responses and ranges of motion in sensorimotor control [39]. Internal perturbations are characterized by voluntary movements such as reaching or bending over. Since the person has knowledge on how to react, the disturbances from these actions test anticipatory responses. External

perturbations, on the other hand, are applied without the person’s knowledge and test reactive responses.

## 1.2 Objectives

Since humans are always learning and improving, sensorimotor processes closely resemble iterative optimal control [11, 16, 22]. While optimal control traditionally relies on *a priori* knowledge of system dynamics, human sensorimotor tasks are constantly updating. It can also be difficult to apply a generalized model to many people with different personalized dynamics. Adaptive dynamic programming (ADP) is one iterative learning approach for optimal control in a system with unknown dynamics [10, 13, 14, 21, 38]. It combines the concepts of reinforcement learning and optimal control to find an adaptive optimal controller using only online state and input data.

Sensorimotor control is especially important for studying human-in-the-loop tasks, such as semi-autonomous driving, for forming more precise and more personalized models. When driving, humans use visual and kinesthetic perception to control their steering actions [30, 32]. Their actions are both anticipatory in knowing the direction to go and reactive in avoiding obstacles and collisions. Every driver also uses a unique set of parameters for control, so sensorimotor control models are used not only for predicting human behavior but also for distinguishing between different drivers [31]. Better models of human behavior lead to better driver assistance and fewer collisions.

Overall, biological processes are complex and unique from person-to-person but can be modeled through optimal control theory. Adaptive optimal control algo-

rhythms can also describe the iterative nature of human development and learning. The objective of this work is to show that ADP can be used to create better models for human sensorimotor control, specifically in maintaining upright balance. The upright position is an inherently unstable equilibrium position and requires constant control, and one hypothesis is that the CNS uses feedback with minimal muscle forces to automatically correct posture [25]. While the equations that define the motion are complex and nonlinear, it is relatively simple to analyze using linearization about the vertical stance as an operating point. Many researchers have studied human balance using principles of optimal control [7, 17, 18, 39]. However, most existing optimal control design methods rely on precise knowledge of system dynamics. These methods are not entirely suited for the problem of posture control because human dynamics are complex and vary from person to person. Flexible, more personalized models help with more accurately describing sensorimotor tasks and further medical and robotic research.

### 1.3 Contributions

For this work, we have modeled and simulated the learning and control processes for upright balance, and we have performed several preliminary experiments to show that ADP is a good fit for studying biological processes like maintaining balance. We have provided a basis for further research into this and related processes, and this work can be used to look into medical research for diseases like Parkinson’s, more human or learning-based robotics, and better, more personalized prostheses.

## 1.4 Thesis Outline

This work is presented as follows: Chapter 2 discusses the model formulation and linearization for human balance; Chapter 3 outlines the ADP algorithm; Chapter 4 includes simulation and experimental results as well as discussion and comparison to previously studied methods; and Chapter 5 contains conclusions and possible future work for this project.



## Chapter 2

# Dynamic Model

When viewed from the side with only forward and backward motion, the human body acts like a multi-segmented pendulum with actuators at each joint. However, there are some challenges to modeling this kind of system. First of all, the upright equilibrium position is unstable and the CNS always needs to apply control. For the human body, this is a much easier task compared to an under-actuated system. Second, the system is made up of nonlinear equations which make system modeling and control more complicated. Lastly, each additional segment adds more complexity to the pendulum dynamics as there are more degrees of freedom. The model for the human system is made up of three segments characterized by movement about the ankle, knee, and hip. The following covers the derivation of the nonlinear and linearized models under normal conditions and in our new learning environment, on a cart with unknown applied force.

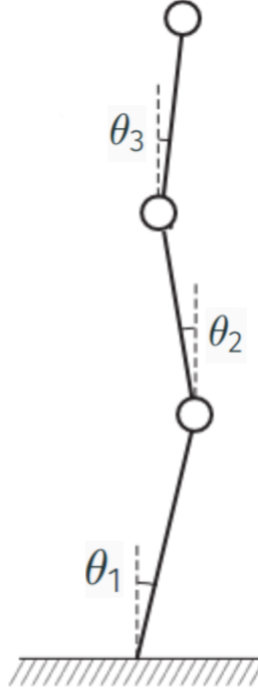


Figure 2.1: Three-segment Inverted Pendulum

## 2.1 Multi-segmented inverted pendulum model

The system dynamics can be derived from Lagrangian mechanics. Indeed, to find the system equations, the Euler-Lagrange equation must be set to equal to the external forces [36]. For the human body, the joint torques act as the control input  $u$ . Namely,

$$\frac{d}{dt} \frac{\partial L}{\partial \dot{\theta}} - \frac{\partial L}{\partial \theta} = u \quad (2.1)$$

where

$$L = \sum_{i=1}^3 \left( \frac{1}{2} m_i v_i^2 + \frac{1}{2} I_i \dot{\theta}^2 - m_i g y_i \right)$$

is the Lagrangian function, and  $m_i$ ,  $v_i$ ,  $I_i$  and  $y_i$  correspond to the mass, velocity, moment of inertia, and vertical position with  $i = 1, 2, 3$  representing the ankle, knee, and hip respectively.

Equation (2.1) can be rewritten in matrix notation as in [36]:

$$M(\theta)\ddot{\theta} + N(\theta)\dot{\theta}^2 - G(\theta) = u \quad (2.2)$$

where  $M$  is a symmetric inertia matrix

$$M(\theta) = \begin{bmatrix} I_1 + h_1 l_1^2 & k_2 l_1 l_2 \cos(\theta_1 - \theta_2) & k_3 l_1 l_3 \cos(\theta_1 - \theta_3) \\ k_2 l_1 l_2 \cos(\theta_1 - \theta_2) & I_2 + h_2 l_2^2 & k_3 l_2 l_3 \cos(\theta_2 - \theta_3) \\ k_3 l_1 l_3 \cos(\theta_1 - \theta_3) & k_3 l_2 l_3 \cos(\theta_2 - \theta_3) & I_3 + h_3 l_3^2 \end{bmatrix},$$

$N$  is an anti-symmetric matrix

$$N(\theta) = \begin{bmatrix} 0 & k_2 l_1 l_2 \sin(\theta_1 - \theta_2) & k_3 l_1 l_3 \sin(\theta_1 - \theta_3) \\ -k_2 l_1 l_2 \sin(\theta_1 - \theta_2) & 0 & k_3 l_2 l_3 \sin(\theta_2 - \theta_3) \\ -k_3 l_1 l_3 \sin(\theta_1 - \theta_3) & -k_3 l_2 l_3 \sin(\theta_2 - \theta_3) & 0 \end{bmatrix},$$

and  $G$  is a gravitation vector

$$G(\theta) = \begin{bmatrix} k_1 l_1 g \sin(\theta_1) \\ k_2 l_2 g \sin(\theta_2) \\ k_3 l_3 g \sin(\theta_3) \end{bmatrix}$$

with constants  $h_i$  and  $k_i$  based on the mass and center-of-mass of each segment

$$h_1 = \frac{1}{4}m_1 + m_2 + m_3, \quad h_2 = \frac{1}{4}m_2 + m_3, \quad h_3 = \frac{1}{4}m_3$$

$$k_1 = \frac{1}{2}m_1 + m_2 + m_3, \quad k_2 = \frac{1}{2}m_2 + m_3, \quad k_3 = \frac{1}{2}m_3.$$

The model (2.2) can be easily linearized around the upright equilibrium position

( $\theta = 0, \dot{\theta} = 0$ ) using the segment angles as the state of the linearized model,  $x \triangleq [\theta_1, \theta_2, \theta_3, \dot{\theta}_1, \dot{\theta}_2, \dot{\theta}_3]^T$  where  $\theta_1, \theta_2$ , and  $\theta_3$  represent the joint angles at the ankle, knee, and hip respectively. The linearization of (2.2) becomes

$$\hat{M}\ddot{\theta} - \hat{G}\theta = u \quad (2.3)$$

where  $\hat{M}$  is a constant symmetric inertia matrix and  $\hat{G}$  is a constant diagonal gravitation matrix with constants  $k_i$  and  $h_i$  such that

$$\hat{M} = \begin{bmatrix} I_1 + h_1 l_1^2 & k_2 l_1 l_2 & k_3 l_1 l_3 \\ k_2 l_1 l_2 & I_2 + h_2 l_2^2 & k_3 l_2 l_3 \\ k_3 l_1 l_3 & k_3 l_2 l_3 & I_3 + h_3 l_3^2 \end{bmatrix}, \quad \hat{G} = \begin{bmatrix} k_1 l_1 g & 0 & 0 \\ 0 & k_2 l_2 g & 0 \\ 0 & 0 & k_3 l_3 g \end{bmatrix}.$$

Furthermore, (2.3) can be rewritten in state-space form as

$$\dot{x} = Ax + Bu \quad (2.4)$$

where

$$A = \begin{bmatrix} 0 & I_3 \\ \hat{M}^{-1}\hat{G} & 0 \end{bmatrix}, \quad B = \begin{bmatrix} 0 \\ \hat{M}^{-1} \end{bmatrix}.$$

## 2.2 Inverted pendulum on a cart

With the additional dynamics of cart and some applied force, (2.1) still holds with

$$L = \frac{1}{2}m_0 v_0^2 + \sum_{i=1}^3 \left( \frac{1}{2}m_i v_i^2 + \frac{1}{2}I_i \dot{\theta}^2 - m_i g y_i \right)$$

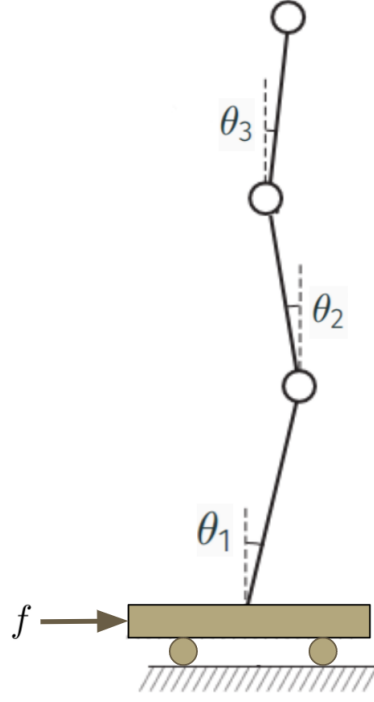


Figure 2.2: Three-segment Inverted Pendulum on a Cart

where  $m_0$  and  $v_0$  represent the mass and velocity of the cart respectively. In this case, an additional state  $x_0$ , the position of the cart, must also be considered. Assuming there is no friction, the equations that govern this system are

$$M(\theta)\ddot{\theta} + N(\theta)\dot{\theta}^2 - G(\theta) + W(\theta)\ddot{x}_0 = u, \quad (2.5)$$

$$\left(\sum_{i=0}^3 m_i\right)\ddot{x}_0 + W^T(\theta)\ddot{\theta} - (\sin(\theta))^T \dot{\theta}^2 = f \quad (2.6)$$

where

$$W(\theta) = \begin{bmatrix} k_1 l_1 \cos(\theta_1) \\ k_2 l_2 \cos(\theta_2) \\ k_3 l_3 \cos(\theta_3) \end{bmatrix},$$

$\ddot{x}_0$  represents the acceleration of the cart, and  $f$  is an exogenous force applied to the cart.

Equations (2.5), (2.6) can also be linearized around the upright position to obtain a model described by

$$\hat{M}\ddot{\theta} - \hat{G}\theta + \hat{W}\ddot{x}_0 = u \quad (2.7)$$

$$\left(\sum_{i=0}^3 m_i\right)\ddot{x}_0 + \hat{W}^T\ddot{\theta} = f \quad (2.8)$$

where

$$\hat{W} = \begin{bmatrix} k_1 l_1 \\ k_2 l_2 \\ k_3 l_3 \end{bmatrix},$$

Assuming that the position of the cart  $x_0$  does not need to be controlled to maintain balance, (2.7) and (2.8) can be combined into a single equation

$$\left(\hat{M} - \frac{1}{\sum m_i} \hat{W} \hat{W}^T\right) \ddot{\theta} - \hat{G}\theta + \frac{1}{\sum m_i} \hat{W} f = u \quad (2.9)$$

which can be represented in state-space form with a disturbance term as

$$\dot{x} = Ax + Bu + Df \quad (2.10)$$

where

$$A = \begin{bmatrix} 0 & \mathbf{I}_3 \\ \left(\hat{M} - \frac{1}{\sum m_i} \hat{W} \hat{W}^T\right)^{-1} \hat{G} & 0 \end{bmatrix}, \quad B = \begin{bmatrix} 0 \\ \left(\hat{M} - \frac{1}{\sum m_i} \hat{W} \hat{W}^T\right)^{-1} \end{bmatrix},$$

$$D = \begin{bmatrix} 0 \\ (\hat{M} - \frac{1}{\sum m_i} \hat{W} \hat{W}^T)^{-1} (\frac{-1}{\sum m_i} \hat{W}) \end{bmatrix}.$$

## 2.3 Conversion to segment angles

While an inverted pendulum can be modeled with joint angles with respect to the vertical, it might be useful to also model the system with segment angles instead. To be specific, the state variable  $\theta$  can be converted from joint angles referenced to the vertical to segment angles  $\phi$  (lower leg, thigh, and torso) referenced to each other using a transformation matrix  $F$  such that

$$\theta = F\phi = \begin{bmatrix} 1 & 0 & 0 \\ 1 & 1 & 0 \\ 1 & 1 & 1 \end{bmatrix} \phi. \quad (2.11)$$

Then, the linear model (2.3) in terms of segment angle becomes

$$\hat{M}F\ddot{\phi} - \hat{G}F\phi = u \quad (2.12)$$

and has state-space representation (2.4) with  $x \triangleq [\phi_1, \phi_2, \phi_3, \dot{\phi}_1, \dot{\phi}_2, \dot{\phi}_3]^T$  and

$$A = \begin{bmatrix} 0 & I_3 \\ F^{-1}\hat{M}^{-1}\hat{G}F & 0 \end{bmatrix}, \quad B = \begin{bmatrix} 0 \\ F^{-1}\hat{M}^{-1} \end{bmatrix}.$$

Similarly, (2.9) in terms of segment angles becomes

$$(\hat{M} - \frac{1}{\sum m_i} \hat{W} \hat{W}^T)F\ddot{\phi} - \hat{G}F\phi + \frac{1}{\sum m_i} \hat{W}f = u \quad (2.13)$$

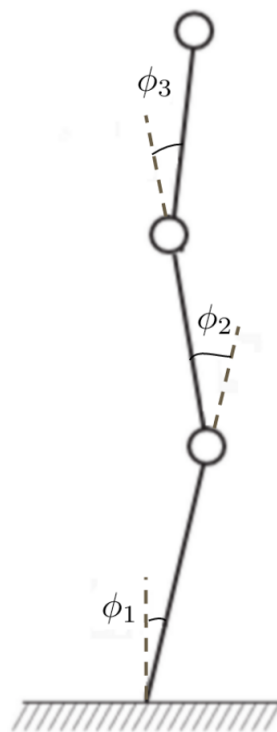


Figure 2.3: Three-segment Inverted Pendulum with Segment Angles



and has state-space representation (2.10) with

$$A = \begin{bmatrix} 0 & \mathbf{I}_3 \\ F^{-1}(\hat{M} - \frac{1}{\sum m_i} \hat{W} \hat{W}^T)^{-1} \hat{G} F & 0 \end{bmatrix}, \quad B = \begin{bmatrix} 0 \\ F^{-1}(\hat{M} - \frac{1}{\sum m_i} \hat{W} \hat{W}^T)^{-1} \end{bmatrix},$$

$$D = \begin{bmatrix} 0 \\ F^{-1}(\hat{M} - \frac{1}{\sum m_i} \hat{W} \hat{W}^T)^{-1} (\frac{-1}{\sum m_i} \hat{W}) \end{bmatrix}.$$

## Chapter 3

# Adaptive Dynamic Programming

While the system matrices  $A$  and  $B$  in (2.4) can be measured, they vary from person to person. In practice, these system matrices are unknown, and a control policy must be computed without exact knowledge of the system dynamics. Under normal conditions, this control policy is trivial to healthy adults, but in a new environment, it must be relearned and adjusted to more optimally perform the required movements. First, we review an online, off-policy ADP algorithm which will be used to study the human balance control. The design philosophy is based on the principles of policy iteration [15] and reinforcement learning [33]. The ADP algorithm presented here is found in more detail in [6, 10, 13].

### 3.1 Problem formulation

The objective is to find a control input  $u$  for system (2.4) that minimizes the cost function

$$J = \int_0^{\infty} (x^T Q x + u^T R u) dt \quad (3.1)$$

with  $Q = Q^T \geq 0$ ,  $R = R^T > 0$ , and  $(A, Q^{1/2})$  observable.

When  $A$  and  $B$  are accurately known, the minimum cost is  $x^T(0)P^*x(0)$  where  $P^* \in \mathbb{R}^{n \times n}$  is a unique, symmetric, positive-definite solution to the algebraic Riccati equation (ARE)

$$A^T P + PA + Q - PBR^{-1}B^T P = 0. \quad (3.2)$$

The optimal controller is

$$u = -K^* x \quad (3.3)$$

where the feedback matrix  $K^*$  in (3.3) can then be determined by

$$K^* = R^{-1}B^T P^*. \quad (3.4)$$

## 3.2 Policy iteration

$P^*$  appears nonlinearly in (3.2) and thus is difficult to solve directly, especially for high-order systems. To tackle this issue, the author of [15] has proposed a successive approximation algorithm, also known as policy iteration, to approximate  $P^*$ :

1. Choose  $K_0$  such that  $A - BK_0$  is Hurwitz. Let  $k = 0$ .
2. Solve  $P_k$  from

$$(A - BK_k)^T P_k + P_k(A - BK_k) + Q + K_k^T R K_k = 0. \quad (3.5)$$

3. Improve control policy by

$$K_{k+1} = R^{-1}B^T P_k. \quad (3.6)$$

4. Let  $k \leftarrow k + 1$ , go to Step 2.

It has also been shown in [15] that  $\{P_k\}$  and  $\{K_k\}$  solved iteratively from (3.5) and (3.6) hold the following properties:

- 1)  $A - BK_k$  is Hurwitz for all  $k$ ,
- 2)  $P^* \leq P_{k+1} \leq P_k \leq \dots \leq P_0$ , and
- 3)  $\lim_{k \rightarrow \infty} K_k = K^*, \lim_{k \rightarrow \infty} P_k = P^*$ .

While this policy iteration is useful for solving  $P^*$ , it still relies on precise system knowledge. Therefore, it is not well suited for modeling human balance control where the system varies from person to person.

### 3.3 ADP for continuous-time linear systems

While the policy iteration relies on full knowledge of the  $A$  and  $B$  matrices, when  $A$  and  $B$  are unknown, equivalent iterations can be achieved using online state and input measurements [10]. Assuming an initial stabilizing  $K_0$  is known, the system (2.4) can be rewritten as

$$\dot{x} = A_k x + B(K_k x + u) \quad (3.7)$$

where

$$A_k = A - BK_k.$$

Then from (3.7) and the policy iteration defined by (3.5) and (3.6), it follows that

$$\begin{aligned}
& x(t + \delta t)^T P_k x(t + \delta t) - x(t)^T P_k x(t) \\
&= \int_t^{t+\delta t} [x^T (A_k^T P_k + P_k A_k) x + 2(u + K_k x)^T B^T P_k x] d\tau \\
&= - \int_t^{t+\delta t} x^T Q_k x d\tau + 2 \int_t^{t+\delta t} (u + K_k x)^T R K_{k+1} x d\tau \tag{3.8}
\end{aligned}$$

where

$$Q_k = Q + K_k^T R K_k.$$

Given a stabilizing  $K_k$ , a pair of matrices  $(P_k, K_{k+1})$ , with  $P_k = P_k^T > 0$ , satisfying (3.5) and (3.6) can be found without any knowledge of the  $A$  and  $B$  matrices.

Define the following notation:

$$P \in \mathbb{R}^{n \times n} \rightarrow \hat{P} \in \mathbb{R}^{\frac{1}{2}n(n+1)}, \quad x \in \mathbb{R}^n \rightarrow \bar{x} \in \mathbb{R}^{\frac{1}{2}n(n+1)}$$

where

$$\begin{aligned}
\hat{P} &= [p_{11}, 2p_{12}, \dots, 2p_{1n}, p_{22}, 2p_{23}, \dots, 2p_{n-1,n}, p_{nn}]^T, \\
\bar{x} &= [x_1^1, x_1 x_2, \dots, x_1 x_n, x_2^2, x_2 x_3, \dots, x_{n-1} x_n, x_n^2]^T
\end{aligned}$$

Also define the matrices (for a positive integer  $l$ )  $\delta_{xx} \in \mathbb{R}^{l \times \frac{1}{2}n(n+1)}$ ,  $I_{xx} \in \mathbb{R}^{l \times n^2}$ ,

$I_{xu} \in \mathbb{R}^{l \times mn}$ ,  $\Theta_k \in \mathbb{R}^{x[\frac{1}{2}n(n+1)+mn]}$ ,  $\Xi_k \in \mathbb{R}^l$  such that

$$\begin{aligned}\delta_{xx} &= \left[ \bar{x}(t_1) - \bar{x}(t_0), \bar{x}(t_2) - \bar{x}(t_1), \dots, \bar{x}(t_l) - \bar{x}(t_{l-1}) \right]^T, \\ I_{xx} &= \left[ \int_{t_0}^{t_1} x \otimes x d\tau, \int_{t_1}^{t_2} x \otimes x d\tau, \dots, \int_{t_{l-1}}^{t_l} x \otimes x d\tau \right]^T, \\ I_{xu} &= \left[ \int_{t_0}^{t_1} x \otimes u d\tau, \int_{t_1}^{t_2} x \otimes u d\tau, \dots, \int_{t_{l-1}}^{t_l} x \otimes u d\tau \right]^T\end{aligned}$$

where  $0 \leq t_0 < t_1 < \dots < t_l$  and the *vec* operator denotes the vectorization of an  $m \times n$  matrix into an  $mn \times 1$  column vector by concatenating the column vectors of the matrix. Then (3.8) implies

$$\Theta_k \begin{bmatrix} \hat{P}_k \\ \text{vec}(K_{k+1}) \end{bmatrix} = \Xi_k \quad (3.9)$$

where

$$\begin{aligned}\Theta_k &= [\delta_{xx}, -2I_{xx}(I_n \otimes K_k^T R) - 2I_{xu}(I_n \otimes R)], \\ \Xi_k &= -I_{xx} \text{vec}(Q + K_k^T R K_k).\end{aligned}$$

If  $\Theta_k$  has full column rank, then  $P_k$  and  $K_{k+1}$  can be updated from

$$\begin{bmatrix} \hat{P}_k \\ \text{vec}(K_{k+1}) \end{bmatrix} = (\Theta_k^T \Theta_k)^{-1} \Theta_k^T \Xi_k \quad (3.10)$$

The off-policy policy-iteration-based ADP algorithm is given below [10]:

0. Choose  $K_0$  as a stabilizing feedback matrix. Set  $k = 0$ .
1. Apply  $u = -K_k x + e$  as the input, where  $e$  is the exploration noise. Compute

$\delta_{xx}$ ,  $I_{xx}$ , and  $I_{xu}$  until  $\Theta_k$  has full column rank.

2. Solve  $P_k$  and  $K_{k+1}$  from (3.10).
3. Let  $k \leftarrow k + 1$  and repeat Steps 1 and 2 until  $\|P_k - P_{k-1}\| \leq \epsilon$  for  $k \geq 1$  where the constant  $\epsilon > 0$  is a predefined small threshold.
4. Use  $u = -K_k x$  as the approximated optimal control policy.

### 3.4 Adaptive optimal output regulation

This section covers the ADP algorithm covered in [5, 6] on a class of continuous-time linear systems described by

$$\dot{x} = Ax + Bu + Dv, \quad (3.11)$$

$$\dot{v} = Ev, \quad (3.12)$$

$$e = Cx + Fv \quad (3.13)$$

where  $d = Dv$  is the exogenous disturbance,  $y = Cx$  is the output of the plant,  $y_d = -Fv$  is the reference signal, and  $e$  represents the tracking error. Again, it is assumed that the system is stabilizable. It is also assumed that though  $v$  is unmeasurable, the minimal polynomial of  $E$  is available. Under this condition, a vector  $w \in \mathbb{R}^{q_m}$  and a matrix  $\hat{E}$  can always be found where

$$\dot{w} = \hat{E}w(t) \quad (3.14)$$

$$v = Gw(t), \quad \forall t \geq 0 \quad (3.15)$$

with unknown constant matrix  $G$ . From this, (3.11) and (3.13) are equivalent to

$$\dot{x} = Ax + Bu + \hat{D}w, \quad (3.16)$$

$$e = Cx + \hat{F}w \quad (3.17)$$

where  $\hat{D} = DG$  and  $\hat{F} = FG$ .

This system can be used for the linear output regulation and linear optimal output regulation problems. The goal of these output regulation problems is to design a controller of the form

$$u = -Kx + Lw \quad (3.18)$$

such that the closed-loop system is globally exponentially stable, the tracking error  $e(t)$  converges to 0,  $K$  is the feedback control gain matrix, and  $L$  is the feed-forward control gain matrix. The linear output regulation problem is solvable by controller (3.18) if there exist solutions  $X, U$  of the regulator equations

$$X\hat{E} = AX + BU + \hat{D}, \quad (3.19)$$

$$0 = CX + \hat{F} \quad (3.20)$$

with

$$L = U + KX. \quad (3.21)$$

To find the optimal solution  $(X^*, U^*)$ , the controller must be designed to minimize a certain predefined cost, and for the ADP algorithm described in [6], the controller



is designed by solving the optimization problem

$$\begin{aligned} \min_{(X,U)} \quad & \text{Tr}(X^T \bar{Q} X + U^T \bar{R} U) \\ \text{subject to} \quad & (3.19) - (3.20) \end{aligned} \quad (3.22)$$

where  $\hat{Q} = \hat{Q}^T > 0, \hat{R} = \hat{R}^T > 0$ .

To find the optimal feedback control, the error system can be written as

$$\dot{\bar{x}} = A\bar{x} + B\bar{u}, \quad (3.23)$$

$$e = C\bar{x} \quad (3.24)$$

with  $\bar{x} = x - X^*w, \bar{u} = u - U^*w$ , and then solved by minimizing (3.1) subject to (3.23). The goal of designing the optimal controller  $u = -K^*x + L^*$  then comes down to first minimizing (3.1) to find  $K^*$  and then minimizing (3.22) to find  $L^* = U^* + K^*X^*$ .

If all system parameters are known  $K^*$  can be solved from (3.2), (3.4). To find  $L^*$ , we can first define a Sylvester map

$$\mathcal{S}(X) = X\hat{E} - AX. \quad (3.25)$$

A constant matrix  $X_1$  is chosen such that  $CX_1 + \hat{F} = 0$ , and then matrices  $X_i$  for  $i = 2, 3, \dots, h+1$  are chosen such that all the vectors  $\text{vec}(X_i)$  form a basis for  $\ker(I_{q_m} \otimes C)$  where  $h$  is the dimension of the null space of  $I_{q_m} \otimes C$ . A general solution to (3.20) can be described by a sequence

$$X = X_1 + \sum_{i=2}^{h+1} \alpha_i X_i \quad (3.26)$$

with  $\alpha_i \in \mathbb{R}$ . Then, (3.19) implies

$$\mathcal{S}(X) = \mathcal{S}(X_1) + X = X_1 + \sum_{i=2}^{h+1} \alpha_i \mathcal{S}(X_i) = BU + \hat{D} \quad (3.27)$$

and both of these equations (3.26) and (3.27) can be rewritten as

$$\mathcal{A}\mathcal{X} = b \quad (3.28)$$

where

$$\begin{aligned} \mathcal{A} &= \begin{bmatrix} \text{vec}(\mathcal{S}(X_2)) & \dots & \mathcal{S}(X_{h+1}) & 0 & -I_{q_m} \otimes B \\ \text{vec}(X_2) & \dots & \text{vec}(X_{h+1}) & -I_{nq_m} & 0 \end{bmatrix}, \\ \mathcal{X} &= \begin{bmatrix} \alpha_2, & \dots, & \alpha_{h+1}, & \text{vec}(X)^T, & \text{vec}(U)^T \end{bmatrix}^T, \\ b &= \begin{bmatrix} \text{vec}(-\mathcal{S}(X_1) + \hat{D}) \\ -\text{vec}(X_1) \end{bmatrix}. \end{aligned}$$

For  $i = 2, \dots, h+1$ ,  $\text{vec}(X_i)$  are linearly independent column vectors, so (3.28)

can be described as

$$\begin{bmatrix} \bar{\mathcal{A}}_{11} & \bar{\mathcal{A}}_{12} \\ \bar{\mathcal{A}}_{21} & \bar{\mathcal{A}}_{22} \end{bmatrix} \mathcal{X} = \begin{bmatrix} \bar{b}_1 \\ \bar{b}_2 \end{bmatrix} \quad (3.29)$$

where  $\bar{\mathcal{A}}_{21}$  is a nonsingular matrix. Using this, a pair  $(X, U)$  is a solution to the regulator equations (3.19), (3.20) if and only if it solves

$$\mathcal{M} \begin{bmatrix} \text{vec}(X) \\ \text{vec}(U) \end{bmatrix} = \mathcal{N} \quad (3.30)$$

where  $\mathcal{M} = -\bar{\mathcal{A}}_{11}\bar{\mathcal{A}}_{21}^{-1}\bar{\mathcal{A}}_{22} + \bar{\mathcal{A}}_{12}$ ,  $\mathcal{N} = -\bar{\mathcal{A}}_{11}\bar{\mathcal{A}}_{21}^{-1}\bar{b}_2 + \bar{b}_1$ .

If the system parameters are unknown, we can solve for the optimal controller in a similar way as described in Section 3.3. Defining  $\bar{x}_i = x - X_i w$  for  $i = 0, 1, 2, \dots, h+1$  with  $X_0 = 0_{n \times q_m}$ , we have

$$\begin{aligned}\dot{\bar{x}}_i &= Ax + Bu + (\hat{D} - X_i \hat{E})w \\ &= A_j \bar{x}_i + B(K_j \bar{x}_i + u) + (\hat{D} - \mathcal{S}(X_i))w\end{aligned}\tag{3.31}$$

where  $A_j = A - BK_j$ . Then from the policy iteration (3.5),

$$\begin{aligned}&\bar{x}_i(t + \delta t)^T P_j \bar{x}_i(t + \delta t) - \bar{x}_i(t)^T P_j \bar{x}_i(t) \\ &= \int_t^{t+\delta t} [\bar{x}_i^T (A_j^T P_j + P_j A_j) \bar{x}_i + 2(u + K_j \bar{x}_i)^T B^T P_j \bar{x}_i \\ &\quad + 2w^T (\hat{D} - \mathcal{S}(X_i))^T P_j \bar{x}_i] d\tau \\ &= - \int_t^{t+\delta t} \bar{x}_i^T (Q + K_j^T R K_j) \bar{x}_i d\tau + 2 \int_t^{t+\delta t} (u + K_j \bar{x}_i)^T \\ &\quad \times R K_{j+1} \bar{x}_i d\tau + 2 \int_t^{t+\delta t} w^T (\hat{D} - \mathcal{S}(X_i))^T P_j \bar{x}_i d\tau.\end{aligned}\tag{3.32}$$

Then for some positive integer  $s$ , define

$$\begin{aligned}\delta_{\bar{x}_i \bar{x}_i} &= [\text{vecv}(\bar{x}_i(t_1)) - \text{vecv}(\bar{x}_i(t_0)), \text{vecv}(\bar{x}_i(t_2)) - \text{vecv}(\bar{x}_i(t_1)), \\ &\quad \dots, \text{vecv}(\bar{x}_i(t_s)) - \text{vecv}(\bar{x}_i(t_{s-1}))]^T, \\ \Gamma_{\bar{x}_i \bar{x}_i} &= \left[ \int_{t_0}^{t_1} \bar{x}_i \otimes \bar{x}_i d\tau, \int_{t_1}^{t_2} \bar{x}_i \otimes \bar{x}_i d\tau, \dots, \int_{t_{s-1}}^{t_s} \bar{x}_i \otimes \bar{x}_i d\tau \right]^T, \\ \Gamma_{\bar{x}_i u} &= \left[ \int_{t_0}^{t_1} \bar{x}_i \otimes u d\tau, \int_{t_1}^{t_2} \bar{x}_i \otimes u d\tau, \dots, \int_{t_{s-1}}^{t_s} \bar{x}_i \otimes u d\tau \right]^T, \\ \Gamma_{\bar{x}_i w} &= \left[ \int_{t_0}^{t_1} \bar{x}_i \otimes w d\tau, \int_{t_1}^{t_2} \bar{x}_i \otimes w d\tau, \dots, \int_{t_{s-1}}^{t_s} \bar{x}_i \otimes w d\tau \right]^T\end{aligned}$$

where  $t_0 < t_1 < \dots < t_s$  are positive integers. Then, (3.32) implies

$$\Psi_{ij} \begin{bmatrix} \text{vecs}(P_j) \\ \text{vec}(K_{j+1}) \\ \text{vec}((\hat{D} - \mathcal{S}(X_i))^T P_j) \end{bmatrix} = \Phi_{ij} \quad (3.33)$$

where

$$\begin{aligned} \Psi_{ij} &= [\delta_{\bar{x}_i \bar{x}_i}, -2\Gamma_{\bar{x}_i \bar{x}_i}(I_n \otimes K_j^T R) - 2\Gamma_{\bar{x}_i u}(I_n \otimes R), -2\Gamma_{\bar{x}_i w}], \\ \Phi_{ij} &= -\Gamma_{\bar{x}_i \bar{x}_i} \text{vec}(Q + K_j^T R K_j). \end{aligned}$$

When the matrix  $\Psi_{ij}$  has full column rank, (3.33) can be uniquely solved from

$$\begin{bmatrix} \text{vecs}(P_j) \\ \text{vec}(K_{j+1}) \\ \text{vec}((\hat{D} - \mathcal{S}(X_i))^T P_j) \end{bmatrix} = (\Psi_{ij}^T \Psi_{ij})^{-1} \Psi_{ij}^T \Phi_{ij}. \quad (3.34)$$

Thus,  $P_j$  and  $K_{j+1}$  are known, and by computing  $\hat{D}$  for  $i = 0$  and  $\mathcal{S}(X_i)$  for  $i = 1, 2, \dots, h+1$  and knowing that  $B = P_j^{-1} K_{j+1}^T R$ ,  $\mathcal{M}$  and  $\mathcal{N}$  are also known.

The optimization problem (3.22) can also be reformulated as

$$\begin{aligned} \min_{(X,U)} & \left( \begin{bmatrix} \text{vec}(X) \\ \text{vec}(U) \end{bmatrix} \right)^T \begin{bmatrix} I_{q_m} \otimes \bar{Q} & 0 \\ 0 & I_{q_m} \otimes \bar{R} \end{bmatrix} \begin{bmatrix} \text{vec}(X) \\ \text{vec}(U) \end{bmatrix} \\ & \text{subject to} \quad (3.30) \end{aligned} \quad (3.35)$$

The ADP-based algorithm for the linear optimal output regulation problem is described below [5, 6]:

1. Compute matrices  $X_0, X_1, \dots, X_{h+1}$ . Utilize  $u = -K_0x + \xi$  on  $[t_0, t_s]$  with bounded exploration noise and stabilizing  $K_0$ . For  $i = 0, 1, \dots, h+1$ , compute  $\delta_{\bar{x}_i \bar{x}_i}, \Gamma_{\bar{x}_i \bar{x}_i}, \Gamma_{\bar{x}_i u}$ , and  $\Gamma_{\bar{x}_i w}$  until  $\Psi_{ij}$  has full column rank. Let  $i = 0, j = 0$ .
2. Solve  $P_j, K_{j+1}$  from (3.34).
3. Let  $j \leftarrow j + 1$ , repeat Step 2 until  $\|P_j - P_{j-1}\| \leq \epsilon$ , for  $j \leq 1$  and for some small positive constant  $\epsilon$ .
4. Let  $j^* \leftarrow j, i \leftarrow i + 1$ , repeat solving  $\mathcal{S}(X_i)$  from (3.34) until  $i = h + 1$ . Find  $(X^*, U^*)$  by solving (3.35).
5. Letting  $L_{j^*} = U^* + K_{j^*}X^*$ , the approximated optimal controller is  $u = -K_{j^*}x + L_{j^*}w$ .

# Chapter 4

## Experimental Results and Discussion

In this section, we conduct a series of simulation studies to investigate how the CNS learns and controls the balance of the human body. The initial simulations were only performed under normal conditions for the linearized model as presented in [23]. While this was a somewhat trivial case, it showed that ADP worked with the control model. The next simulations included the unknown dynamics with the human on a cart. More recently, we have also obtained data from several preliminary experiments performed in the lab. While we need a more substantial data set from several more subjects, the preliminary results match the expected outcome of the ADP model.

## 4.1 Preliminary simulation

The first simulation was separated into two stages: learning and implementation. In the learning stage, several trials were performed on the linearized model (2.4) to obtain state data  $x$  with the following exploration noise on the input

$$e = 10 \sum_{i=1}^3 \sin(\omega_i t) \quad (4.1)$$

where  $\omega_i$  for  $i = 1, 2, 3$  were equally spaced frequencies selected in the range  $[0, 200]$ . After each trial, the feedback matrix was updated for the next trial. After several trials, the final learned feedback matrix was implemented on the input of the nonlinear model (2.2) (in terms of segment angles).

## 4.2 Selection of parameters

Human model parameters from [36] were used for the simulation. The  $Q$  and  $R$  matrices are weighting matrices that limit the variations in the state and input respectively, and they can be chosen to model the human CNS. For this case,  $R$  was set to the identity matrix, and  $Q$  was selected through observation of the 2-D simulation.  $Q$  could however be selected to penalize movements from the center of mass or upright stance [18].

Selection of an initial stabilizing  $K_0$  is difficult for an inverted pendulum as the dynamics are inherently unstable. However, it is assumed that the CNS in healthy adults has some estimate of the control policy. Mathematically, such initial estimation can be obtained in several ways. For example, recent work in value iteration-based ADP has provided a framework for modeling sensorimotor learning

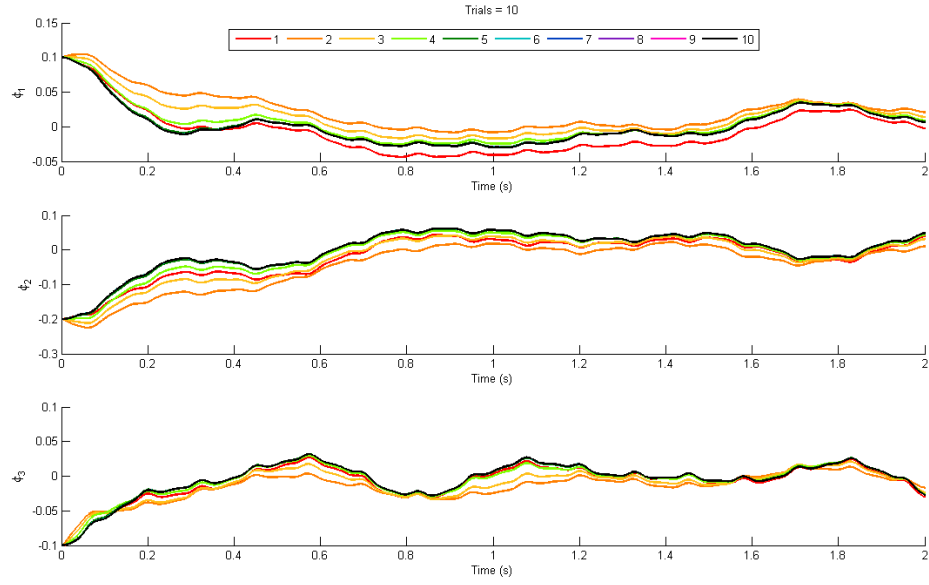


Figure 4.1: Trajectories of segment angles with added noise (4.1) for ten trials.

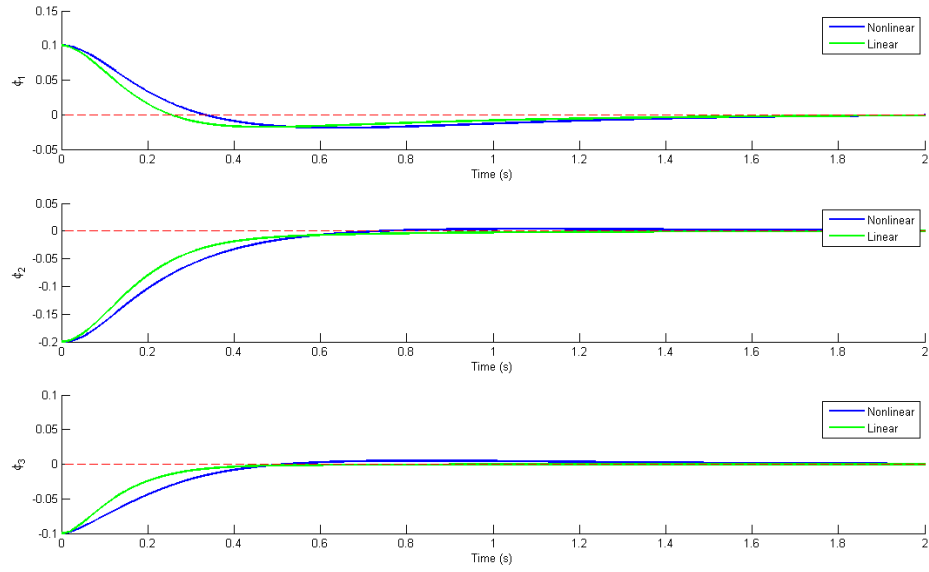


Figure 4.2: Comparison of linear and nonlinear trajectories with input  $u = -K_{10}x$ .



in an initially unstable environment [3, 4]. For simplicity, in our simulation,  $K_0$  was preselected such that  $A - BK_0$  was stable.

Figure 4.1 shows the trajectories for ten trials. One can see that after only a few learning trials with noise (4.1), the iterated solution  $P_k$  and feedback matrix  $K_k$  converge to their optimal LQR values of  $P^*$  and  $K^*$ . It is well-known that the policy iteration algorithm converges quadratically [15]. This models the learning process of the human CNS. After each trial, the control policy updates and gets closer to optimal.

Figure 4.2 shows the comparison of both the linear and nonlinear trajectories implemented with feedback matrix

$$K_{10} = \begin{bmatrix} 738.405 & 81.027 & 100.673 & 273.625 & 138.175 & 53.162 \\ 460.986 & 486.143 & -187.443 & 178.788 & 109.914 & 32.722 \\ 272.517 & 379.454 & 527.543 & 116.680 & 93.356 & 69.356 \end{bmatrix}$$

which is reasonably close to the optimal feedback matrix

$$K^* = \begin{bmatrix} 738.4050 & 81.027 & 100.673 & 273.625 & 138.175 & 53.162 \\ 460.9864 & 486.143 & -187.442 & 178.788 & 109.914 & 32.722 \\ 272.5167 & 379.454 & 527.543 & 116.680 & 93.356 & 69.356 \end{bmatrix}$$

from the LQR solution. For an initial starting position close to the upright equilibrium position, both models' trajectories are very similar. For this localized region, it is possible to study human balance with these methods.

## 4.3 Human balance on a cart

### 4.3.1 Simulation

To simulate human balance on a cart, we first looked at a healthy individual balancing without any exogenous forces acting on the system. In this case the models (2.3) and (2.9) have the same structure but slightly different constant matrix values. As a result, the feedback  $K$  learned prior to standing on the cart is stabilizing but not optimal when balancing on a cart. Figure 4.3 shows trajectories for a healthy person balancing on solid ground and Figure 4.4 shows the trajectories for a healthy person balancing on a cart without any exogenous forces and with the same control. Both trajectories look incredibly similar, but as stated before, the control used in Figure 4.4 is stabilizing but not optimal.

Next, we simulated fifty learning trials. As stated for the initial simulations, it is known that the feedback control  $K_k$  converges quadratically with each trial [15]. Assuming the force can be accurately estimated, the optimal feed-forward control  $L$  can be calculated from  $K_k$  when  $\|P_k - P_{k-1}\| < \epsilon$  for some predefined error threshold  $\epsilon$ . To implement this in a trial-based simulation, the off-line ADP for output regulation policy based on [5, 6] was used. Only feedback control was used for the initial trials, and after a specified number of trials such that  $K_k$  became close enough to its optimal value  $K^*$ , the control input was updated to include both feedback and feed-forward strategies. The simulation considers the ideal system in which there is no delay from human reaction time, and the initial conditions are all zero ( $\theta = \dot{\theta} = 0$ ).

The force function was found through some trial and error. We wanted the the cart to be pushed at some time delay during the window such that the initial push

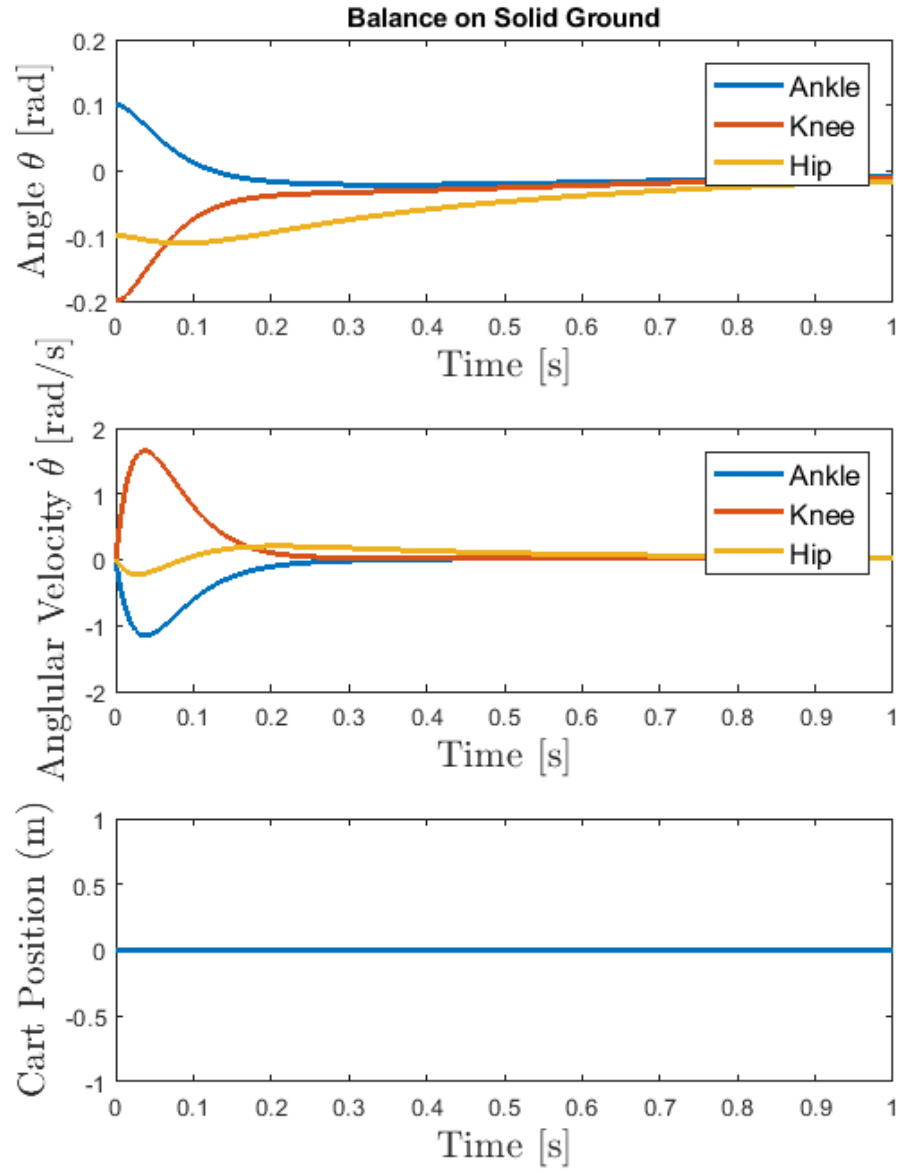


Figure 4.3: Example balance trajectories on ground

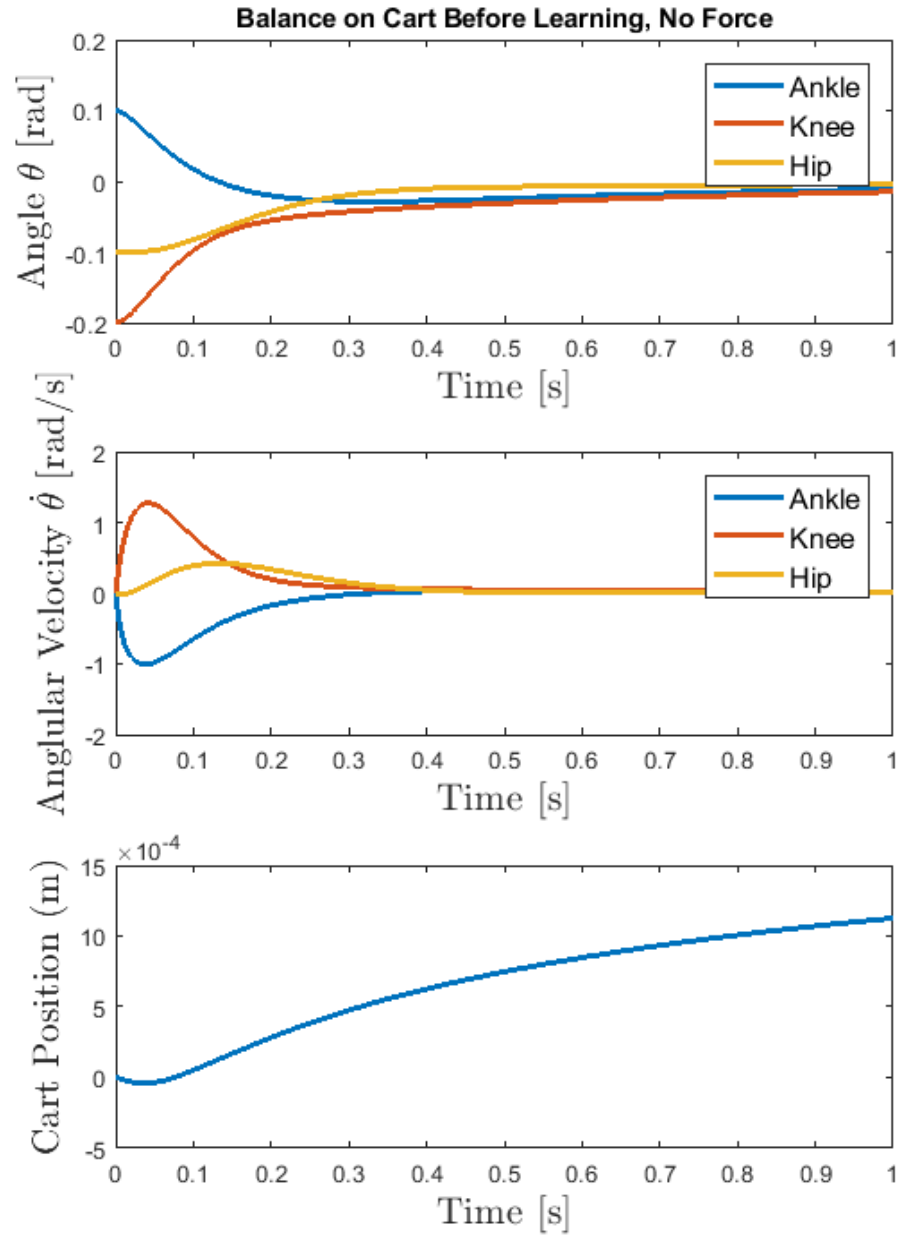


Figure 4.4: Example balance trajectories on cart without applied force

was large and then decayed to an offset value. In the simulation, this corresponds to a large push and then a somewhat constant speed throughout the remaining portion of the trial. To get the desired response from the pendulum and cart position, the force was modeled as

$$\dot{f}(t) = \begin{cases} 0, & 0 < t \leq t_1 \\ c_1(t - t_2)f, & t_1 < t < t_2 \\ c_2f, & t \geq t_2 \end{cases} \quad (4.2)$$

where  $c_1$  and  $c_2$  are constant values that correspond to the constant matrix  $\hat{E} \in \mathbb{R}^{1 \times 1}$  in (3.14) and  $t_1, t_2$  are time delay values with  $0 \leq t_1 < t_2$ . For this simulation, the parameters were set to  $c_1 = -40$ ,  $c_2 = -1.4$ ,  $t_1 = 1.5\text{s}$ ,  $t_2 = 2\text{s}$ . To specify the magnitude of the force, a non-zero random magnitude initial value  $f_0$  was chosen on the range  $[0.1, 0.2]$ .

To quantify the learning process, the convergence of  $P_k$  and  $K_k$  is typically used. However, since it is difficult to measure muscle torques of the human sensorimotor system, a different metric needed to be observed. We decided to look at the mean-squared error (MSE) of the states because the  $x$  term in the LQR cost function (3.1) aims to minimize the weighted sum of squares of the states. The upright stance, zero angle trajectory was used as the reference for the MSE, so the equation could be written as mean of the sum of squares, ie.

$$\text{MSE}(x_i) = \frac{1}{N} \sum_{k=1}^N x_i(k)^2 \quad (4.3)$$

for time sample  $k$ ,  $N$  number of time samples, and system states  $i = 1, 2, \dots, 6$ .

Combined MSE's for the angles and angular velocities were unweighted averages of the corresponding states  $x$ .

For the first set of trials, only pushes from behind were simulated. Each force is applied at  $t = 1.5\text{s}$  and decays over time. However, the force is applied to the cart throughout the entire simulation such that the force does not decay to zero and the cart continues to move. While the force of each push is random, it is assumed that each force is consistently within some range of possible values and that the subject can estimate the force fairly accurately. Figure 4.5 shows the first trial in this simulation with the time of the onset of the force function shown as a solid vertical line. Without any feed-forward control to compensate for the applied force, the subject is pushed backward ( $\text{sign}(\theta) = -1$ ) and then tries to return back to the center. However, due to the applied force, the angles do not fully converge to zero and are instead at some offset.

Figure 4.6 shows a simulated trial after the 50-trial learning process. After learning the updated control with both feedback and feed-forward, there is very little movement. While the trajectories do not completely converge to zero, the offset is very minimal. It is also important to note that when using this updated control with the added feed-forward term, the simulated person does not get pushed completely backward. Depending on the force, certain joint angles actually move forward.

The quadratic convergence of the first ten simulated ADP trials is shown in Figure 4.7. With each consecutive trial, the control gets closer to the optimal solution. While it is not exact, the control gets close to the optimal solution with some small error even after just a handful of trials.

Figures 4.8 and 4.9 show the MSE of the angles and angular velocities re-

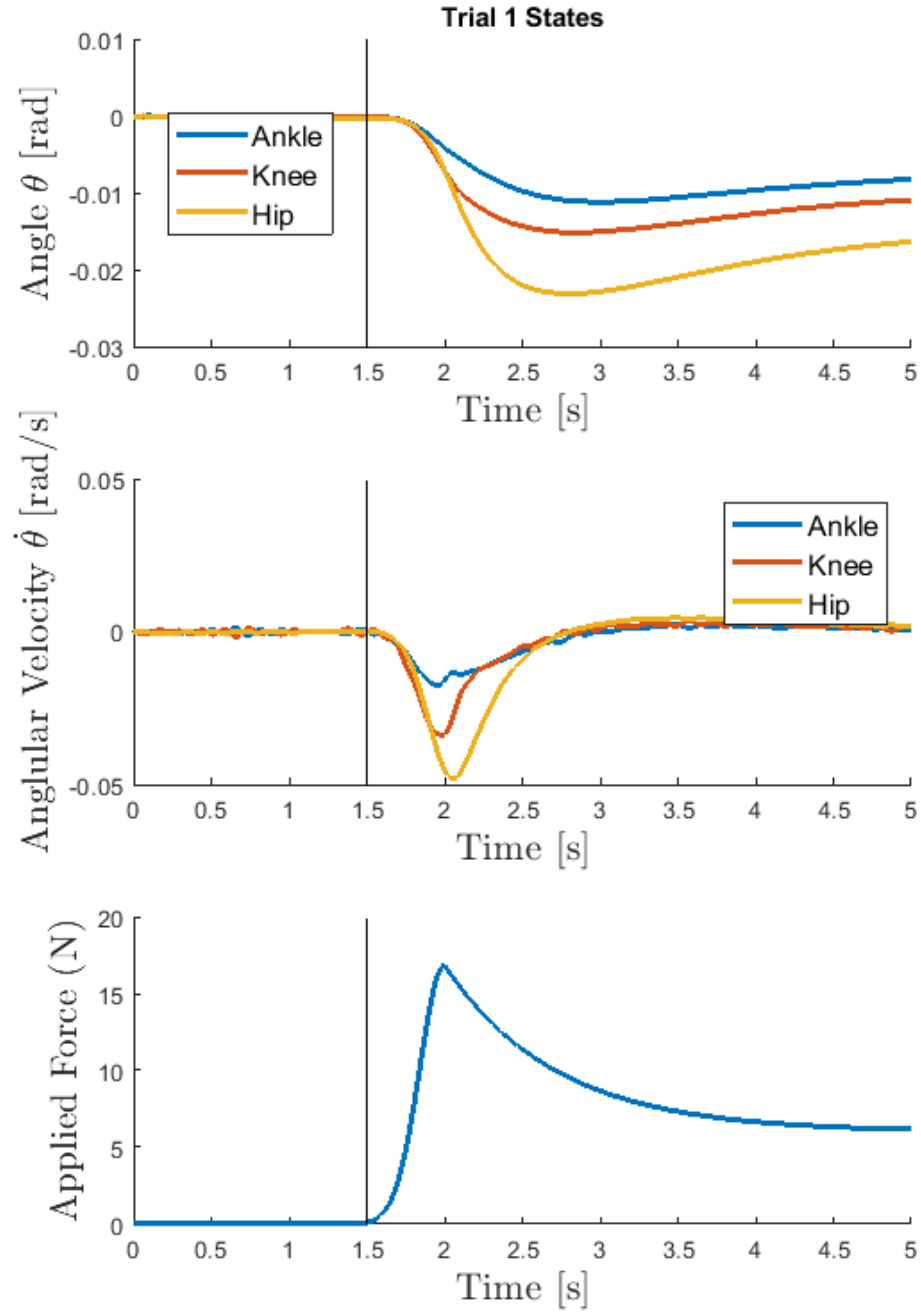


Figure 4.5: Simulation trial 1 trajectories

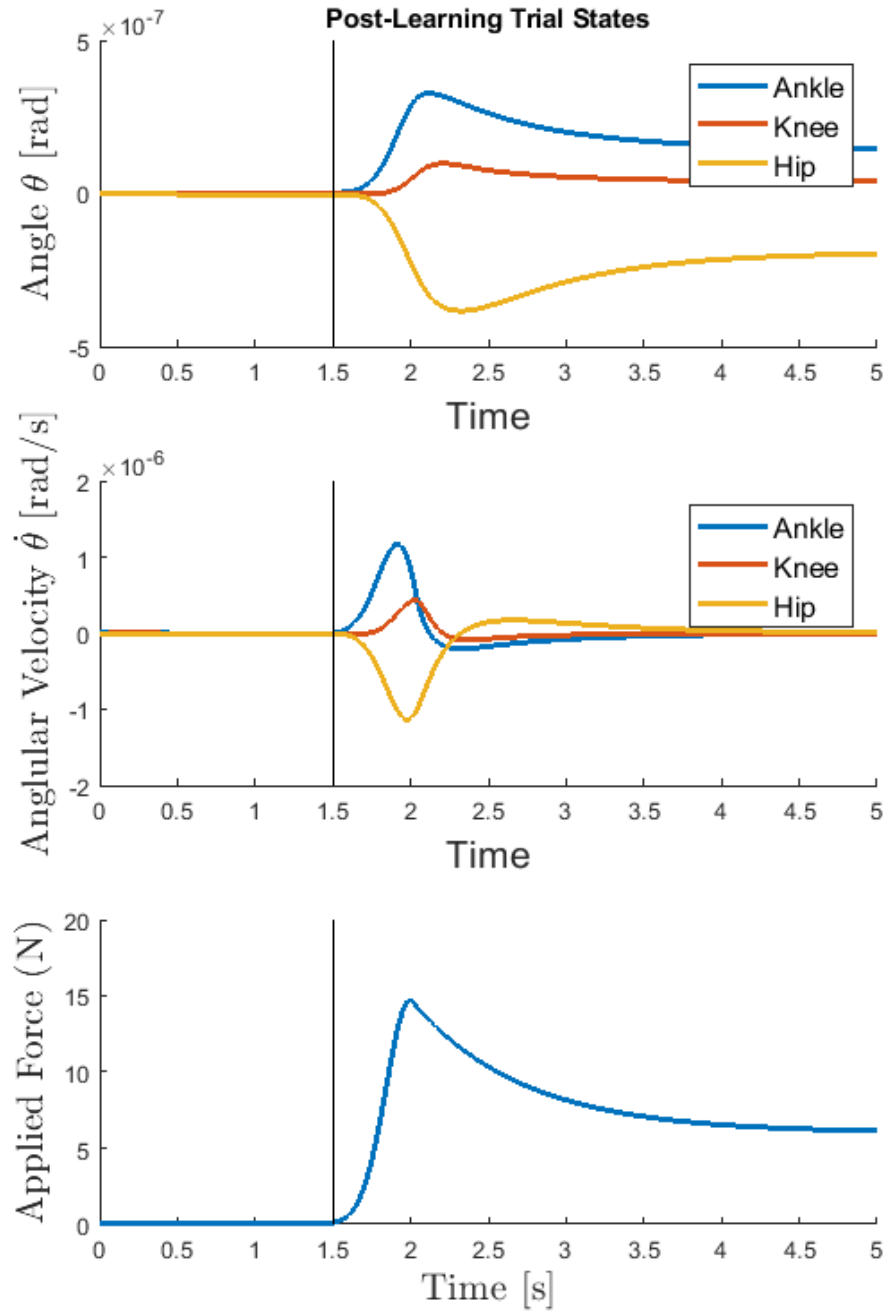


Figure 4.6: Simulation trial 50 trajectories



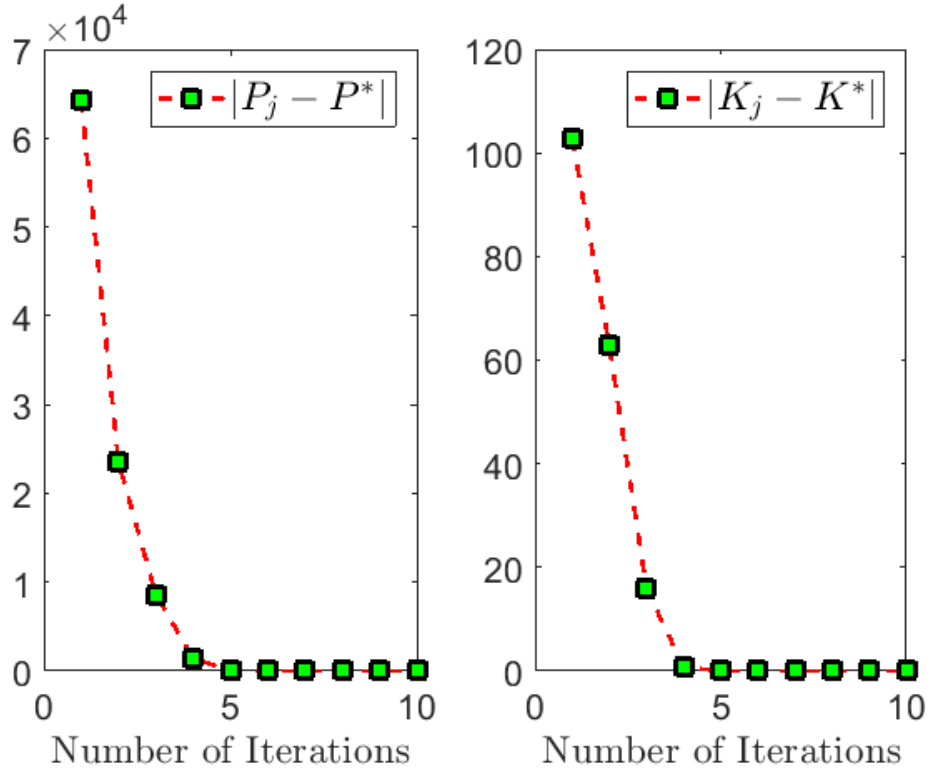


Figure 4.7: Quadratic convergence of 10 simulated ADP trials

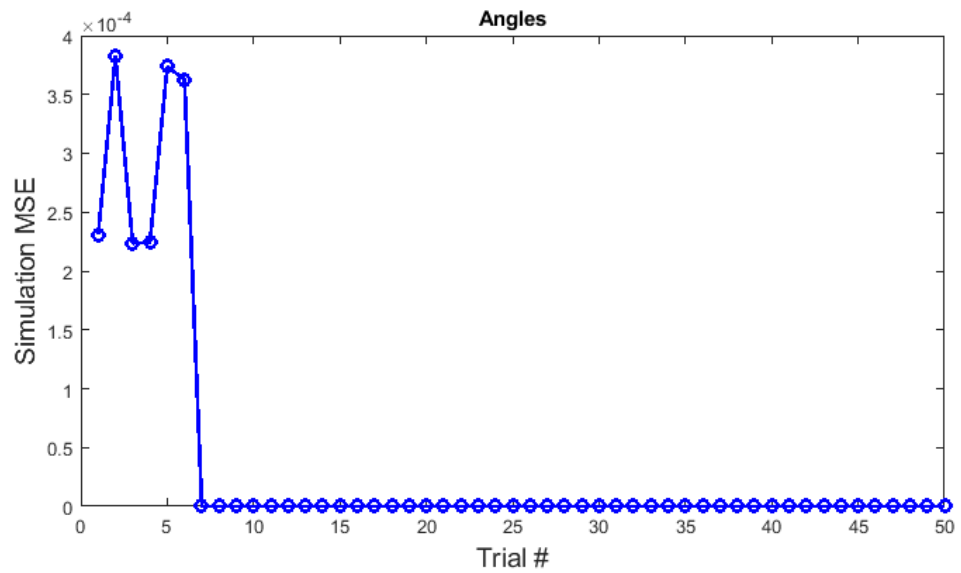


Figure 4.8: Angle mean-squared error

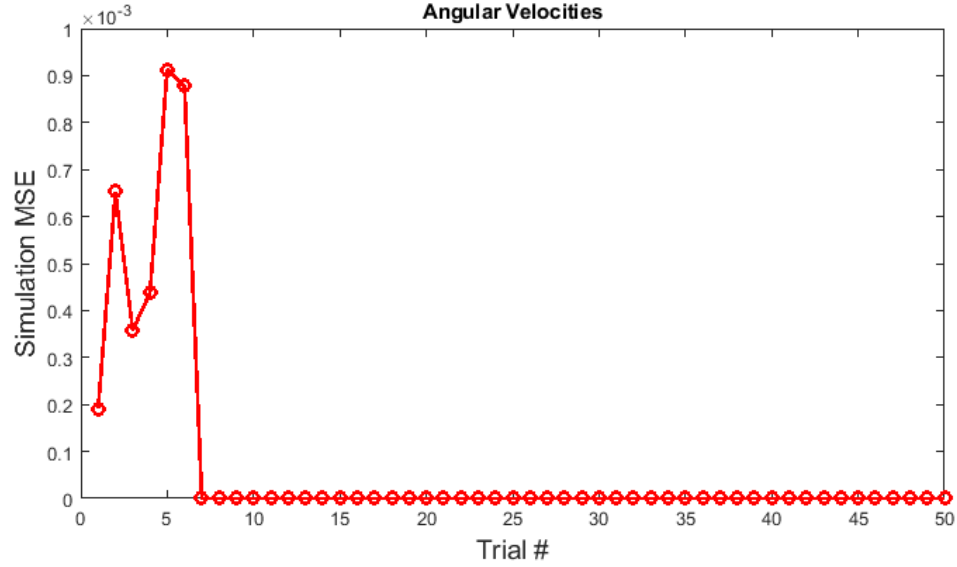


Figure 4.9: Angular velocity mean-squared error

spectively referenced to the center zero trajectory. With the off-line policy, the feedback term  $K_k$  must converge before adding the feed-forward control  $L$ . Before updating, there are some errors based on the magnitude of the force. After a set number of trials (in this case five trials), the control is updated and the forces are completely counteracted. In this simulation, the force is very accurately estimated by the person, but in non-ideal real situations, the MSE might not be zero for the updated trials.

We also simulated trials with a pattern of both forward and backward pushes. The resulting trajectories were similar. It is difficult to determine when, if humans do use some kind of off-line update policy, the person updates their control. It is also difficult to know exactly how they estimate the force and when they can accurately predict the pattern and consequently the direction of the upcoming force.

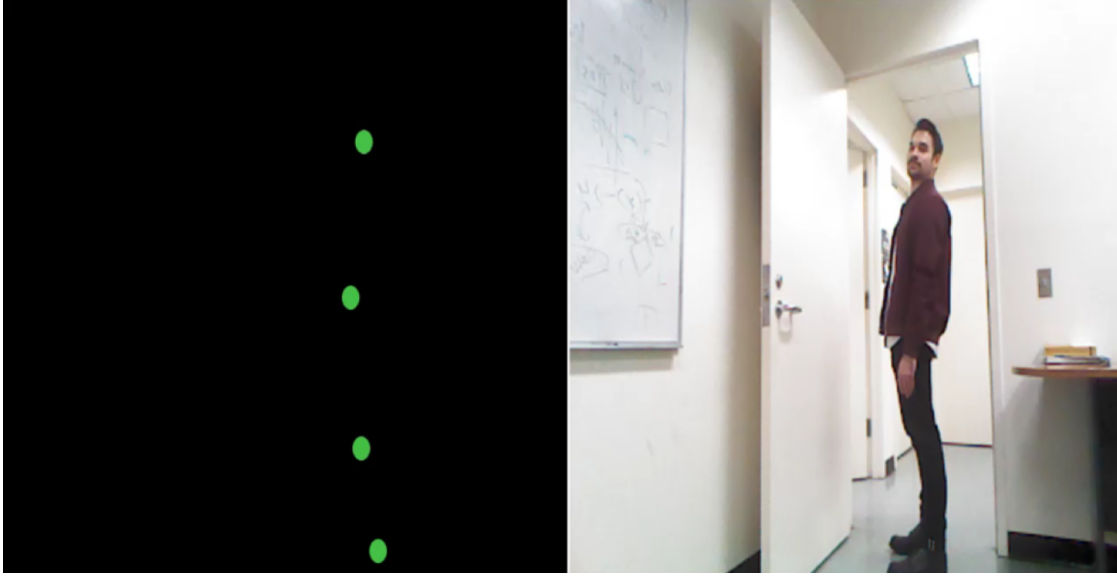


Figure 4.10: Example Kinect joint detection program

### 4.3.2 Experimental data

To collect experimental data, we used the Microsoft Kinect camera to detect the human skeleton. An example of the detection program is shown in Figure 4.10. We were able to isolate the ankle, knee, and hip joints from a side-view in the detection software, and from those joints, the angles were calculated from the x-y pixel position data. Angular velocity was calculated as the first-order difference between position values.

We placed subjects on a wooden board with cylindrical dowels underneath such that when they bent forward or backward, the board would move. A push or pull was also applied to the board for each trial. This force was applied by another person from behind the subject, so the magnitude varied slightly and the subject could not see when the force was applied. This became the new learning environment in which subjects had to reconfigure their balance.

We conducted two sets of fifty trials similar to the simulations in Section 4.3.1.

For the first set, we only applied pushes (single direction) to the board. For the second set, we applied a pattern of pushes and pulls (two directions). Data was collected in separate CSV files and then loaded into MATLAB to be smoothed and plotted.

Figure 4.11 shows the trajectories for the first trial of the single-direction experiment with a solid vertical line indicating approximately where the push began. The angles do not completely converge to zero due to the force, and the postural sway is very apparent, especially in the angular velocity plot. The knee and hip angles both move backward when the cart is pushed as expected, but unlike the simulation trial in Figure 4.5, the ankle actually moves forward. This implies that the subject uses some form of feed-forward control before even beginning the experiment.

Figure 4.12 shows the trajectories for Trial 50. After learning how to balance, the subject only makes minimal adjustments to maintain their posture, and most of the movement is in the ankle. Again, the angles do not completely converge to zero, but compared to Trial 1 in Figure 4.11, the angles are much closer to zero and less spread apart from each other.

Figures 4.13 and 4.14 show the MSE of the angles and angular velocities respectively for the experiment. While the angle MSE is very noisy, there are fewer fluctuations toward the end of the experiment (with a few exceptions). The velocity MSE drops after just a few trials. The subject is able to quickly estimate the applied forces and becomes more efficient. This is very similar to what was observed in the simulation in Figure 4.9. Figure 4.15 shows the separate MSE's for each state. It is apparent that there is very little knee movement, so the knee angle and velocity is minimized very quickly. Most of the movement comes from

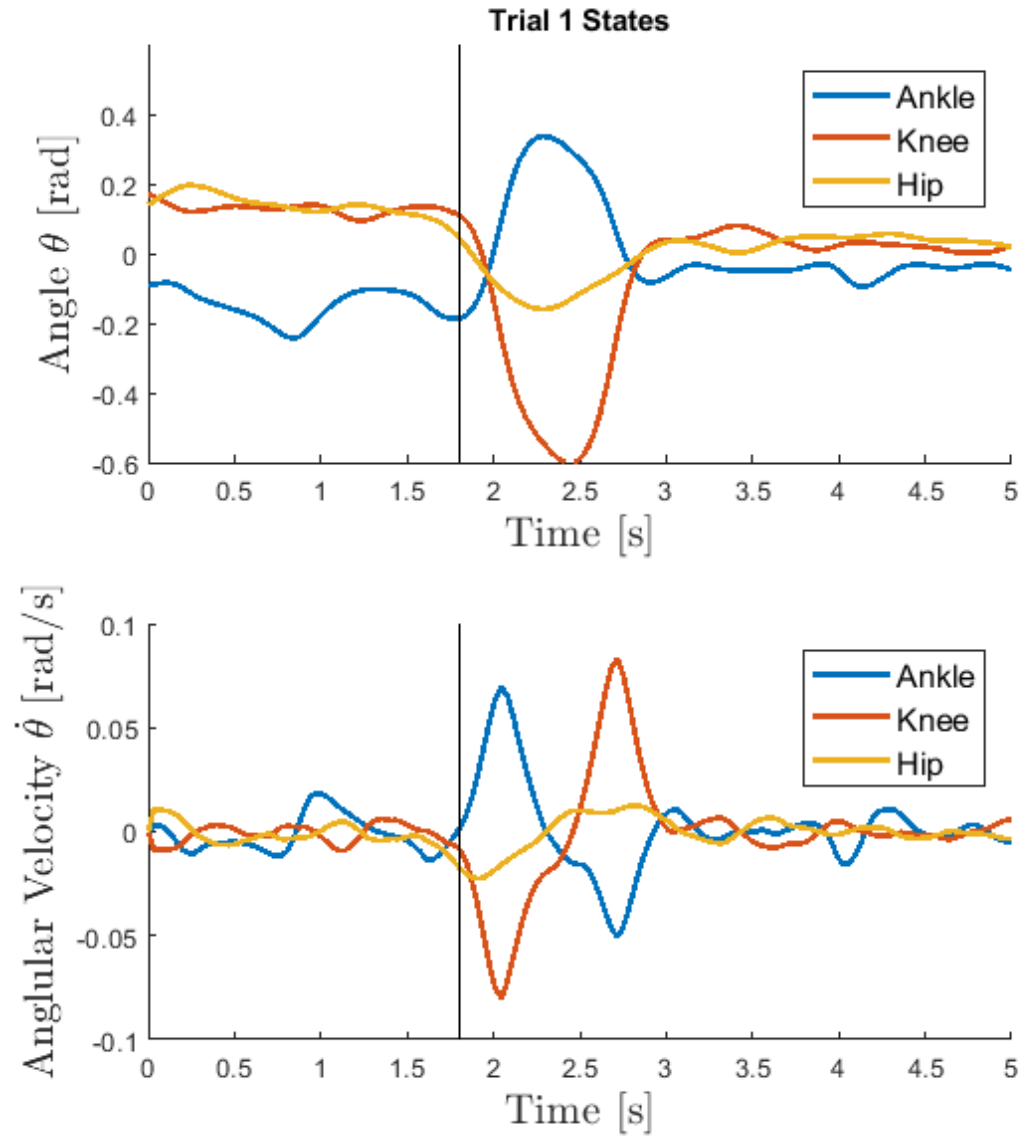


Figure 4.11: Trial 1 trajectories for experiment with only pushes

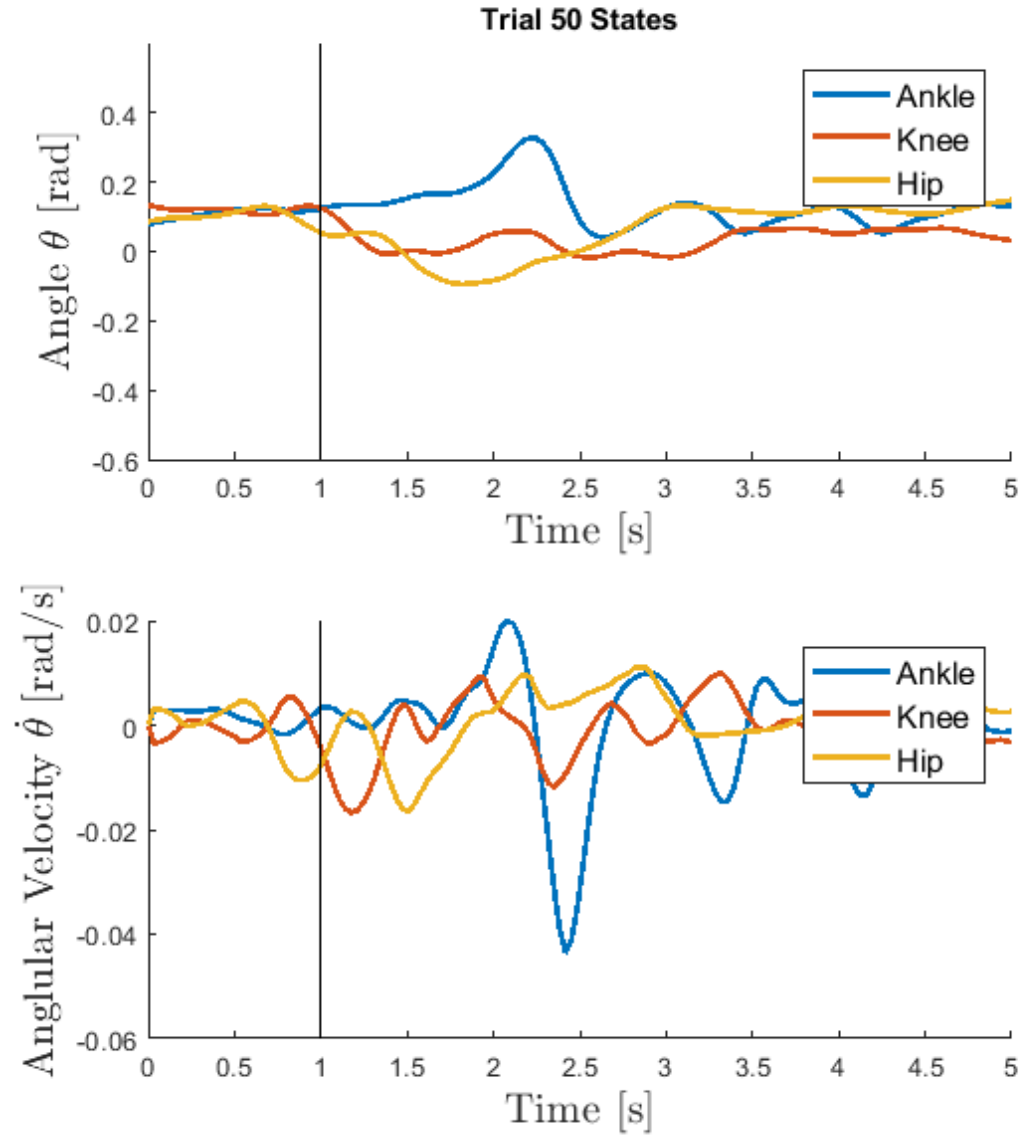


Figure 4.12: Trial 50 trajectories for experiment with only pushes

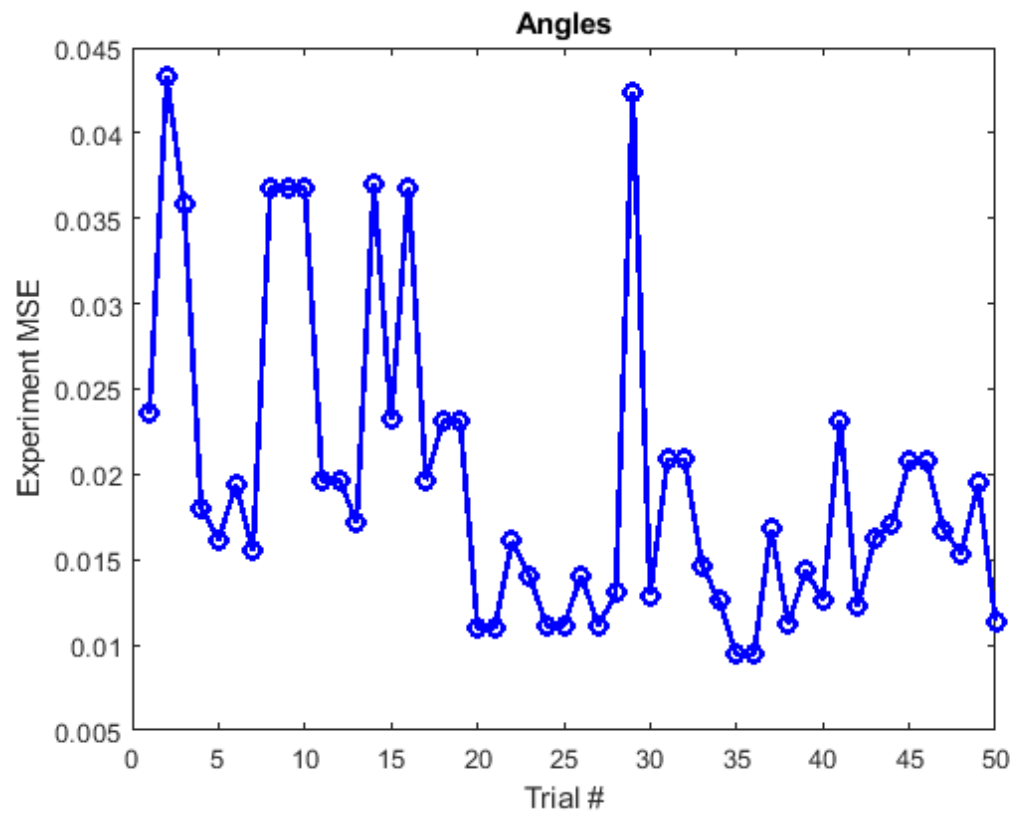


Figure 4.13: Angle MSE for experiment with only pushes

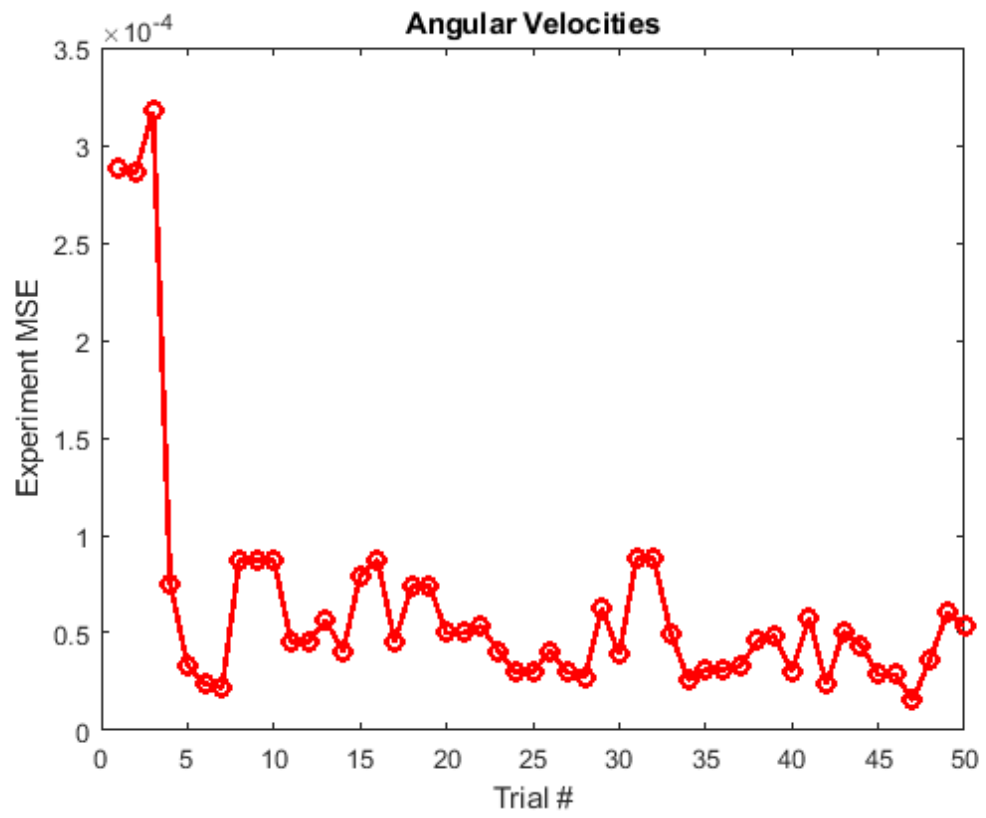


Figure 4.14: Angular velocity MSE for experiment with only pushes



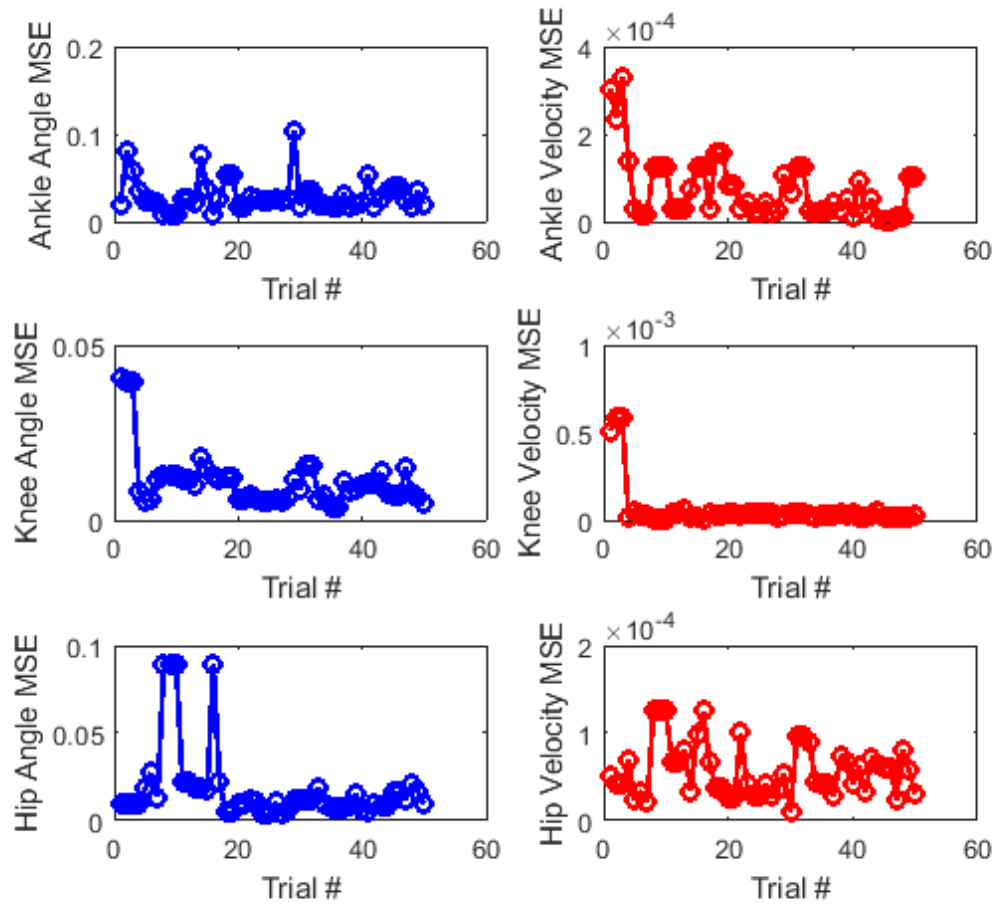


Figure 4.15: Individual MSE's for experiment with only pushes

the ankle and hip.

Next, the experiment was performed with a pattern of pushes and pulls from behind. The pattern of { push, push, pull, push, pull, push } was repeated throughout this experiment. Figure 4.16 shows the first trial. This is very similar to the first trial of the first experiment in Figure 4.11 since they are both pushes. Again, the ankle movement shows signs of some predetermined feed-forward control. Figure 4.17 shows the final trial of this experiment. The movement is similar to the final trial of the previous experiment in Figure 4.12 as well. Although the hip angle does seem to be farther away from the center.

Figures 4.18 and 4.19 show the MSE of the angle and angular velocities respectively. In this case, there is more variation, especially in the velocity MSE, but there is still generally a flattening downward trend. Figure 4.20 shows the separate MSE's for each state. Again, the knee does not contribute much to the movement and is minimized quickly, and the movement comes primarily from the ankle and hip.

## 4.4 Comparison to previous methods

While other methods [7, 17, 18, 36, 39] use similar techniques to model human balance using optimal control, they are typically not flexible in terms of personalized dynamics. These models typically counter some uncertainty by combining state estimation with optimal feedback control [17, 36], but even these models make assumptions about unknown parameters and dynamics. Even though an optimal control policy can be formed using statistically average human parameters, each person responds to perturbations from upright stance in slightly different

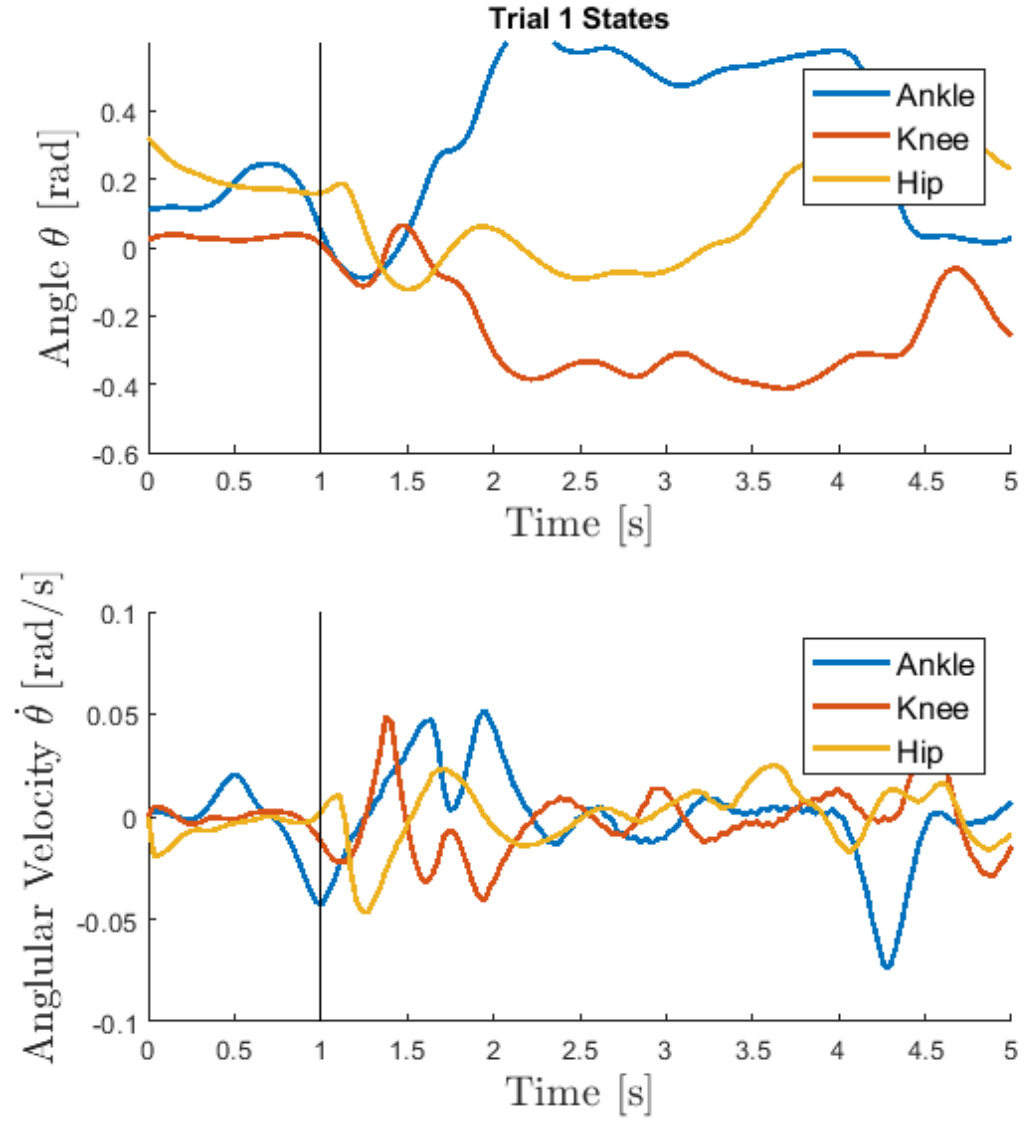


Figure 4.16: Trial 1 trajectories for experiment with pattern of pushes and pulls

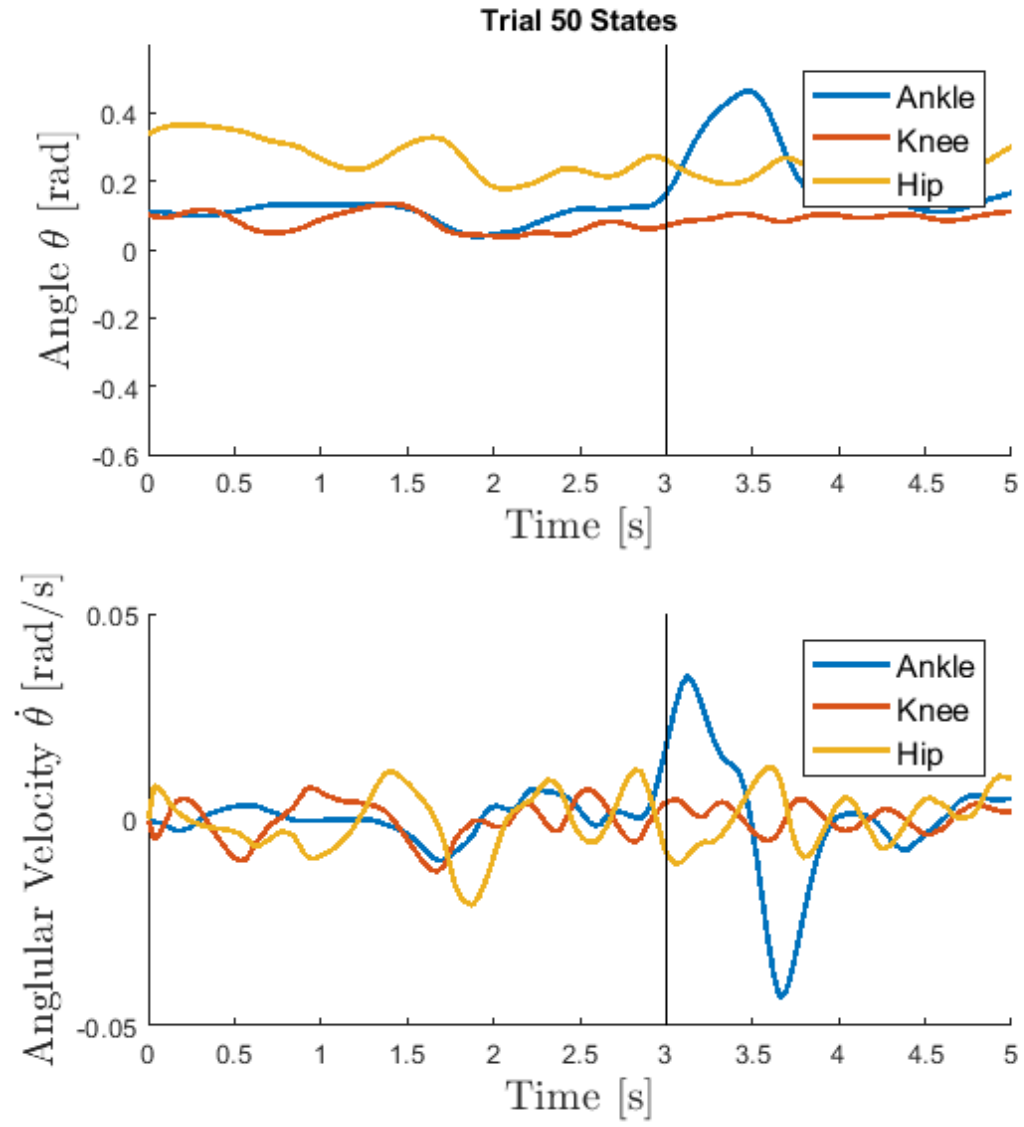


Figure 4.17: Trial 50 trajectories for experiment with pattern of pushes and pulls

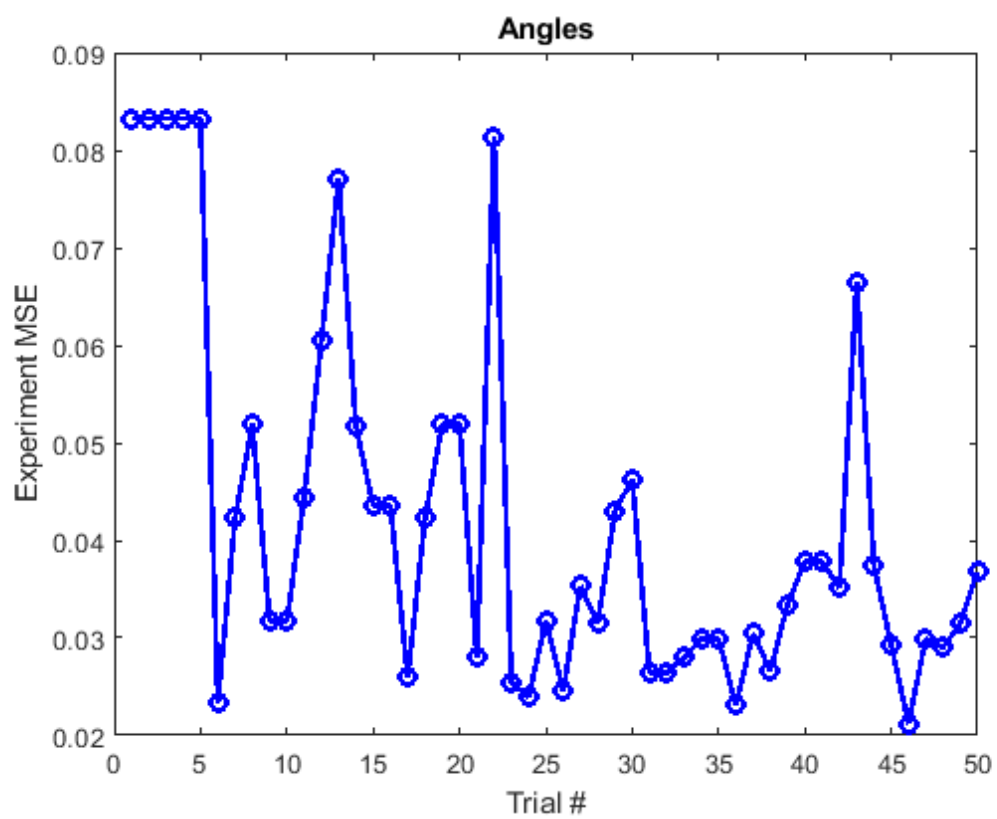


Figure 4.18: Angle MSE for experiment with pattern

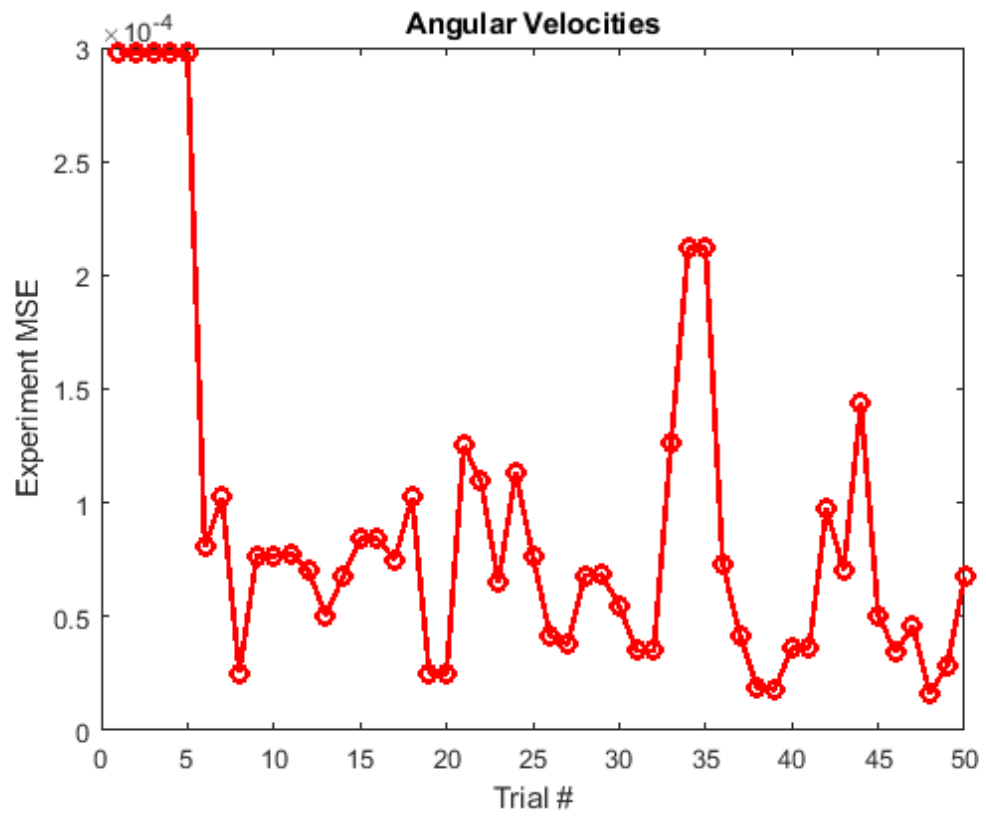


Figure 4.19: Angular velocity MSE for experiment with pattern

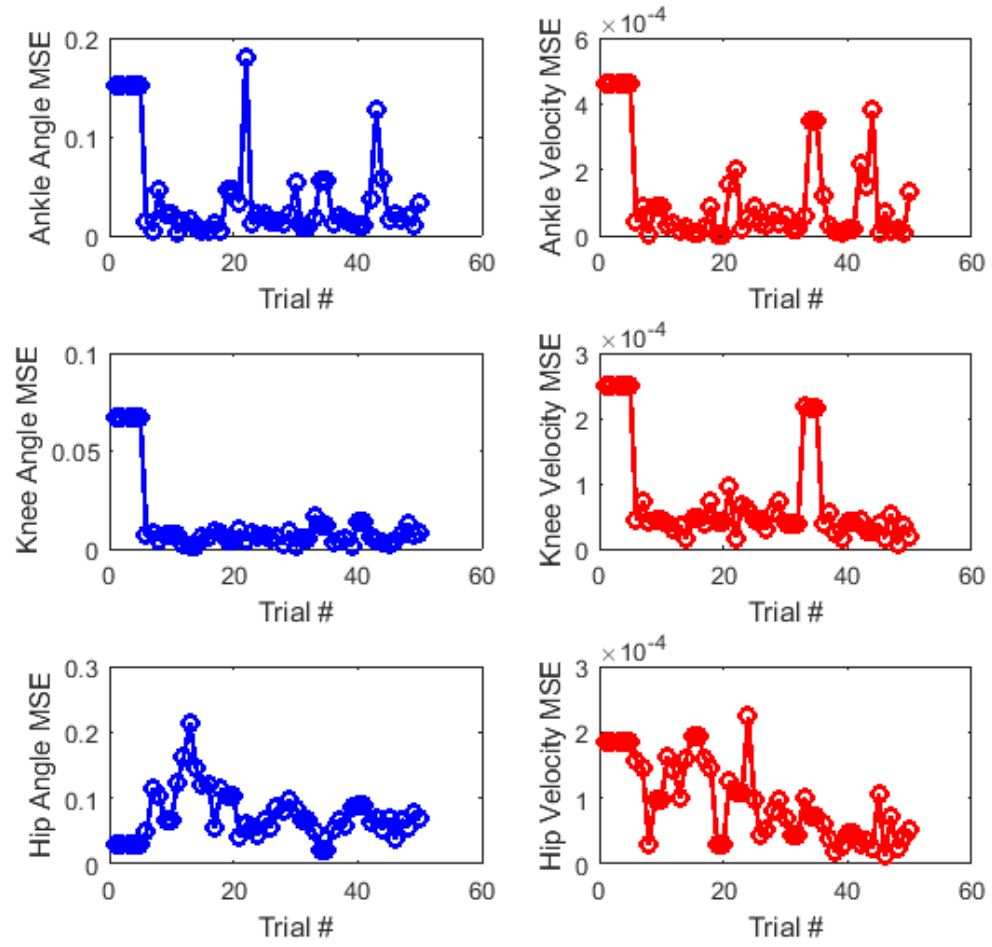


Figure 4.20: Individual MSE's for experiment with pattern

ways. Using ADP, the concepts and biological implications of previous methods can be extended to more personalized cases where physical parameters are not easily known.

Furthermore, the ADP method provides a new model of the human learning process. Not only can it be used in the same manner as previous methods to study dynamics and possibly diagnose medical disorders, it can be used to study how humans adapt to changing environment. However, the off-line ADP policy seems to be an oversimplification of the actual human learning process. People are continuously readjusting and learning. There are many factors that contribute to how people learn, and the sensorimotor system is incredibly complex. The Sensory Organization Test (SOT) is one experimental procedure that has been used to study balance in varying sensory environments [17, 18, 28] and can also be used here to isolate some further conditions.



## Chapter 5

### Conclusions

This work presents only a basis for studying human balance with ADP. While it has shown that balance can be studied in a small localized region about upright stance, the methods presented here are simplified to the deterministic case for simulation. Certain issues with ADP, such as robustness, have already been addressed by previous work [3, 11] and should be applied to this work. Due to the complexity of the human sensorimotor control system, there are many system states and other unknown factors that contribute to human balance. RADP is suited for handling these kinds of systems with an unknown number of states or inputs.

This research is useful for presenting ADP as a feasible method of study for human learning. While the results do not exactly match the ADP algorithm, they show the same trend in learning over trials. The ADP algorithms used here [6, 10] are probably not suited for this exact problem. The observed subjects showed feed-forward action during learning trials, something that is not completely explained by ADP yet. The algorithm should be adjusted to include online learning.

Further work must also be done with more rigorous experimentation. Less

noisy data could probably be obtained with a more consistent method for pushing the subjects and with additional sensors. We were not able to measure the applied force or the velocity directly, the force was not applied at a consistent time within the five-second window. Given more data with more trials, it is likely that our results could be clearer. There are many other ways to test ADP as a theory for human balance as well. Moving platforms and virtual reality devices seem like viable ways of testing balance in new environments. Also, certain factors that are not modeled here, like the contribution of vision, should be isolated in further testing.

There are also several more related problems that can be studied using this work. Sit-to-Stand movement [24, 26] and walking [9, 39] are just two examples of more complex problems that use balance as a starting point. Sit-to-stand is a much larger movement that cannot be linearized around a single point while walking is comprised of several different movements executed consecutively. These more complex motions most likely follow a similar learning trend that can be studied with ADP.

Overall, this problem is best studied using more real data. Compared to previous methods, ADP for human balance is an interesting computational learning mechanism that could be used to study the CNS and medical disorders such as Parkinson’s disease. It is highly suited for systems with unknown dynamics and therefore suited for studying the human sensorimotor system which is very personalized.

# Bibliography

- [1] A. L. Adkin, J. S. Frank, and M. S. Jog. Fear of falling and postural control in parkinson's disease. *Movement Disorders*, 18(5):496–502, 2003.
- [2] N. B. Alexander. Postural control in older adults. *Journal of the American Geriatrics Society*, 42(1):93–108, 1994.
- [3] T. Bian and Z. P. Jiang. Model-free robust optimal feedback mechanisms of biological motor control. In *2016 12th World Congress on Intelligent Control and Automation (WCICA)*, June 2016.
- [4] T. Bian and Z. P. Jiang. Value iteration and adaptive dynamic programming for data-driven adaptive optimal control design. *Automatica*, 71:348–360, 2016.
- [5] W. Gao and Z.-P. Jiang. Global optimal output regulation of partially linear systems via robust adaptive dynamic programming\*\*this work has been partly supported by the u.s. national science foundation grants ECCS-1101401 and ECCS-1230040. *IFAC-PapersOnLine*, 48(11):742–747, 2015.
- [6] W. Gao and Z.-P. Jiang. Adaptive dynamic programming and adaptive optimal output regulation of linear systems. *IEEE Transactions on Automatic Control*, 61(12):4164–4169, dec 2016.

- [7] C. Golliday and H. Hemami. Postural stability of the two-degree-of-freedom biped by general linear feedback. *IEEE Transactions on Automatic Control*, 21(1):74–79, Feb. 1976.
- [8] F. Horak, J. Nutt, and L. Nashner. Postural inflexibility in parkinsonian subjects. *Journal of the Neurological Sciences*, 111(1):46 – 58, 1992.
- [9] Y. Hu and K. Mombaur. Bio-inspired optimal control framework to generate walking motions for the humanoid robot iCub using whole body models. dec 2017.
- [10] Y. Jiang and Z. P. Jiang. Computational adaptive optimal control for continuous-time linear systems with completely unknown dynamics. *Automatica*, 48(10):2699–2704, Oct. 2012.
- [11] Y. Jiang and Z. P. Jiang. Adaptive dynamic programming as a theory of sensorimotor control. *Biological Cybernetics*, 108(4):459–473, June 2014.
- [12] Y. Jiang and Z. P. Jiang. A robust adaptive dynamic programming principle for sensorimotor control with signal-dependent noise. *Journal of Systems Science and Complexity*, 28(2):261–288, 2015.
- [13] Y. Jiang and Z. P. Jiang. *Robust adaptive dynamic programming*. Wiley IEEE Press, 2017.
- [14] Z. P. Jiang and Y. Jiang. Robust adaptive dynamic programming for linear and nonlinear systems: An overview. *European Journal of Control*, 19(5):417–425, 2013.

- [15] D. Kleinman. On an iterative technique for riccati equation computations. *IEEE Transactions on Automatic Control*, 13(1):114–115, Feb. 1968.
- [16] J. W. Krakauer and P. Mazzoni. Human sensorimotor learning: adaptation, skill, and beyond. *Current Opinion in Neurobiology*, 21(4):636–644, aug 2011.
- [17] A. D. Kuo. An optimal control model of human balance: can it provide theoretical insight to neural control of movement? In *Proceedings of the 1997 American Control Conference*, volume 5, pages 2856–2860, 1997.
- [18] A. D. Kuo. An optimal state estimation model of sensory integration in human postural balance. *Journal of Neural Engineering*, 2(3):S235–S249, Aug. 2005.
- [19] K. P. Krding and D. M. Wolpert. Bayesian integration in sensorimotor learning. *Nature*, 427(6971):244–247, jan 2004.
- [20] C. Laschi and R. S. Johansson. Bio-inspired sensory-motor coordination. *Autonomous Robots*, 25(1):1–2, 2008.
- [21] F. L. Lewis, D. Vrabie, and K. G. Vamvoudakis. Reinforcement learning and feedback control: Using natural decision methods to design optimal adaptive controllers. *IEEE Control Systems*, 32(6):76–105, 2012.
- [22] D. Liu and E. Todorov. Evidence for the flexible sensorimotor strategies predicted by optimal feedback control. *Journal of Neuroscience*, 27(35):9354–9368, Aug. 2007.
- [23] E. Mauro, T. Bian, and Z.-P. Jiang. Adaptive dynamic programming for human postural balance control. In *Neural Information Processing*, pages 249–257. Springer International Publishing, 2017.

- [24] A. M. Mughal and K. Iqbal. Physiological lqr design for postural control coordination of sit-to-stand movement. *Cognitive Computation*, 4(4):549–562, 2012.
- [25] L. M. Nashner and G. McCollum. The organization of human postural movements: A formal basis and experimental synthesis. *Behavioral and Brain Sciences*, 8(01):135, Mar. 1985.
- [26] S. Nuzik, R. Lamb, A. VanSant, and S. Hirt. Sit-to-stand movement pattern a kinematic study. *Physical therapy*, 66(11):1708–1713, 1986.
- [27] R. J. Peterka. Sensorimotor integration in human postural control. *Journal of Neurophysiology*, 88(3):1097–1118, sep 2002.
- [28] R. J. Peterka and F. O. Black. Age-related changes in human posture control: sensory organization tests. 1989.
- [29] H. E. B. Russell, L. K. Harbott, I. Nisky, S. Pan, A. M. Okamura, and J. C. Gerdes. Motor learning affects car-to-driver handover in automated vehicles. *Science Robotics*, 1(1):eaah5682, nov 2016.
- [30] D. D. Salvucci and R. Gray. A two-point visual control model of steering. *Perception*, 33(10):1233–1248, oct 2004.
- [31] S. Schnelle, J. Wang, H.-J. Su, and R. Jagacinski. A personalizable driver steering model capable of predicting driver behaviors in vehicle collision avoidance maneuvers. *IEEE Transactions on Human-Machine Systems*, 47(5):625–635, oct 2017.

- [32] C. Sentouh, P. Chevrel, F. Mars, and F. Claveau. A sensorimotor driver model for steering control. In *2009 IEEE International Conference on Systems, Man and Cybernetics*. IEEE, oct 2009.
- [33] R. S. Sutton and A. G. Barto. *Introduction to Reinforcement Learning*. MIT Press, Cambridge, MA, USA, 1st edition, 1998.
- [34] E. Todorov. Optimality principles in sensorimotor control. *Nature Neuroscience*, 7(9):907–915, Sept. 2004.
- [35] K. G. Vamvoudakis. Non-zero sum nash q-learning for unknown deterministic continuous-time linear systems. *Automatica*, 61:274–281, 2015.
- [36] H. van der Kooij, R. Jacobs, B. Koopman, and H. Grootenboer. A multi-sensory integration model of human stance control. *Biological Cybernetics*, 80(5):299–308, May 1999.
- [37] M. Vaugoyeau, H. Hakam, and J.-P. Azulay. Proprioceptive impairment and postural orientation control in parkinson’s disease. *Human Movement Science*, 30(2):405–414, Apr. 2011.
- [38] D. Vrabie, K. G. Vamvoudakis, and F. L. Lewis. *Optimal adaptive control & differential games by reinforcement learning principles*. Institution of Electrical Engineers, 2013.
- [39] D. A. Winter. Human balance and posture control during standing and walking. *Gait & posture*, 3(4):193–214, 1995.
- [40] D. Wolpert, Z. Ghahramani, and M. Jordan. An internal model for sensorimotor integration. *Science*, 269(5232):1880–1882, sep 1995.

- [41] D. M. Wolpert. Probabilistic models in human sensorimotor control. *Human Movement Science*, 26(4):511–524, aug 2007.
- [42] D. M. Wolpert and M. S. Landy. Motor control is decision-making. *Current Opinion in Neurobiology*, 22(6):996–1003, dec 2012.