ABSTRACT

KHADE, AJINKYA SANJAY. Gait Phase Parameter Estimation and Model Predictive Control of Active Transfemoral Prosthesis. (Under the direction of Dr. Fen Wu.)

Limb loss disrupts the lives of hundreds of thousands of individuals every year. And the disruption it causes is even more severe if a lower limb is lost. Loss of mobility drastically reduces the quality of life of any individual. Lot of prosthetic devices are available these days that restore basic locomotion capability. But most of these devices either require a large amount of effort from the user or are severely restricted in terms of the capabilities they restore. This thesis seeks to remedy some of these issues by examining the viability of Model Predictive Control for controlling Active Transfemoral Prosthesis. The experimental setup used for collecting the data used in this research will be introduced, followed by a description of the nonlinear gait model of the prosthesis. The thesis will develop an optimization framework for estimating the parameters of the system, which are essential for designing a controller. This optimization framework is composed of several smaller optimization sub-problems that are solved using a combination of global optimization algorithms like Genetic Algorithm and local algorithms. This yields a reasonable set of estimates for the system parameters which are used for designing a Model Predictive Controller with the goal of tracking the reference gait trajectory. The process gives a controller which is capable of tracking reference trajectories for both hip and knee angles during the gait cycle with high accuracy for the base scenario of level-ground walking and while reducing the effort required from the user as well as prosthesis. However it leaves some room for improvement in terms of its robustness. The thesis will close with a discussion of the possible future work that could help to improve upon the results obtained.

© Copyright 2016 by Ajinkya Sanjay Khade

All Rights Reserved

Gait Phase Parameter Estimation and Model Predictive Control
of Active Transfemoral Prosthesis

by
Ajinkya Sanjay Khade

A thesis submitted to the Graduate Faculty of
North Carolina State University
in partial fulfillment of the
requirements for the Degree of
Master of Science

Mechanical Engineering

Raleigh, North Carolina

2016

APPROVED BY:

Dr. Aranya Chakrabortty
Minor Advisor

Dr. Gregory Buckner

Dr. Fen Wu
Chair of Advisory Committee

DEDICATION

To my parents and my sister.

BIOGRAPHY

Ajinkya Khade was born in India and has spent the early part of his life living in various parts of the country. He earned his Bachelor of Technology in Mechanical Engineering from Visvesvaraya National Institute of Technology, Nagpur, India in 2013. He joined the Master of Science program in the Department of Mechanical and Aerospace Engineering at North Carolina State University in 2014. During this period he has been working under the guidance of Dr. Fen Wu, and has also earned a minor in Electrical Engineering.

ACKNOWLEDGEMENTS

The two years of graduate school at North Carolina State University have been a huge learning experience for me. I feel confident that everything I have learned here is going to help me in the grand scheme of things, even if it might not be obvious right now how all the dots connect. And for that I have a lot of people to thank. First and foremost, I would like to thank my advisor Dr. Fen Wu for giving me the chance to work under his guidance. His insightful approach and wise judgment have helped me navigate the tricky waters of conducting research. His help with my technical problems and his patience while I dealt with personal matters have gone a long way in making my experience during this journey enjoyable. But most importantly, I would like to thank him for introducing me to this domain. The work I have done has strengthened my interest in this field of work while also helping me to expand my horizons. This interest in allied fields lead me to apply for a minor in Electrical Engineering.

I would like to thank Dr. Aranya Chakrabortty for serving as my *Minor Advisor*. The courses he helped me choose for my minor have complimented my domain knowledge very well, and have provided me with a very broad background. This has allowed me to take a very holistic view of all technical problems and tackle them with a unique perspective. The course taught by him has helped me a lot in my research while also giving me a taste of real-world problems. I am grateful to Dr. Gregory Buckner for serving on my committee. His course has been one of the most enjoyable ones during my Masters, and his guidance and support have proven to be valuable in this endeavor.

I would also like to thank Dr. Paul Ro for giving me the chance to serve as a Teaching Assistant for his course. It was a unique experience indeed; it has helped me look at the student-instructor relationship in a new light and has given me a fresh perspective on how to approach learning. I am also thankful to my peers and lab mates. Watching them excel in their endeavors has motivated me to set a high bar for myself and to constantly push myself.

As important as academic endeavors are for success, their true value can be achieved only in conjunction with personal development. And while it can be difficult to maintain the balance between work and personal life in graduate school, family and friends make it a little easier. I have been blessed to have great friends over the course of my life and their help and support is a big factor that has kept me going.

Everyone has role models that they look up to, and I am no different in that regard. Growing up in a nation where cricket is not merely a sport but very much a religion, Sachin Tendulkar - its foremost protagonist - has had a deep influence on me. Watching him wield his magic on the field has given me pleasure very few things can, and has helped me get through some tough days. But the more important lessons I have learned by watching him are the value of humility, dedication and hard work. He has taught me that no amount of talent or skill can make up for the value of hard work; and if you set your mind to it, nothing is impossible. And I will be forever grateful for these values.

I would like to thank my brother-in-law Abhishek for all the academic and professional advice. His success has been a source of inspiration for me ever since I have known him. I am also grateful for having a great sister Ashwini, who has been like a third parent for me. She has been friend when I needed one, and walked me through some of my silly and irrational fears and questions. Lastly, and *most importantly*, I would like to thank my parents for - quite simply - everything. They have made countless sacrifices over the years for my sake. The values they have imbibed in me should stand me in good stead for the rest of my life. The love and support of my family, and the faith they have in me, is what gives me the strength to get up every morning and try to make a dent in the universe.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

Limb loss has devastating effects on any individual who has the misfortune of experiencing it. The problem is much greater if the limb lost is a lower limb. Locomotion is key to any human endeavor and this ability is something most individuals take for granted. When that is impaired, it reduces a person's independence and affects their quality of life.

A study [ZG08] estimated that there were over 600,000 major lower limb amputees living in the US alone in 2005. Of these, approximately half are expected to be individuals with a transfemoral amputation. And the total number of people with limb loss of any sort is expected to double by 2050. It is difficult to find the equivalent worldwide statistics, since many countries do not keep a record of amputees or cause and level of amputation. Clearly, this is a problem that deserves a lot of attention and resources.

1.1 Types of Knee Prosthesis

The loss of knee joint dramatically reduces the way an individual can manipulate any prosthesis. Quite a few prosthetic devices have been developed over the years which try to improve the quality of life for amputees. Most of these devices make it possible for the amputee to walk again on level-ground, although they require varying levels of training and energy. These knee prostheses can be divided into three categories [Har13], [SP09] -

1. Mechanically Passive Prostheses -

   As the name suggests, these devices are energetically passive and provide no net power input. The movement of the joints relies on the properties of its mechanical components. And although no sensors are required, the users must perform compensatory movements with their other body parts such as trunk and residual limb. Such devices require a large amount of energy from the user and greatly restrict the types of maneuvers they can perform.

2. Microprocessor-Controlled Mechanically Passive Prosthesis -

   Such devices use sensory input to determine the gait phase [Liu14] and modulate the mechanical impedance accordingly. The joint impedances for every gait phase must be determined offline such that the prosthesis mimics human gait as best as possible. They offer various advantages over the mechanically passive prosthesis.

The advantages include greater flexibility, ability to adapt to different walking speeds and greater knee stability. They also reduce the amputee energy consumption and improve smoothness of gait. Although these devices carry on-board power for sensors and modulating impedance, they still can not produce net positive mechanical power. This remains a big limiting factor that impairs their ability to perform maneuvers like sit-to-stand, and climbing up slopes or stairs, which require net positive power input at some or all of the joints.

3. Microprocessor-Controlled Mechanically Active Devices -
   The limitations of the mechanically passive types have lead to a lot of interest in developing a self-contained powered prosthesis that is humanlike in its physical characteristics while being energetically efficient. These devices use a higher level control strategy similar to that of the previous types. Gait cycle is divided in different phases and sensory information is used to determine current phase. A lower-level controller then calculates the joint torques that are required in order to mimic natural gait. Different types of designs have been proposed for this class of prosthesis. One of them [Rou13] tries to leverage the passive dynamics of the leg more like a human muscle does. However some variation of the design proposed by [Sup08] has proven to be popular among researchers, by virtue of having a similar mechanical structure to a human leg.

The class of devices which produce net positive power at the knee joint is known as *Active Transfemoral Prosthesis (ATP)* and is focus of this research.

1.2 Control of Prosthesis

Once the mechanical design of prosthesis is finalized, an appropriate control algorithm needs to be designed. Without a suitable control scheme, it can be very tough to perform regular walking maneuvers. This research area has received a lot of attention lately.

Most of the current control schemes tend to use Finite-State Impedance Control. This scheme uses the same higher-level architecture described previously, where the current state (gait phase) of the system is detected first. This division of gait cycle into different phases or modes will be elaborated upon in subsection 2.1.3. Within each phase, the control law is modeled like a passive spring damper system. The mechanical impedance of this spring damper system is tuned such that the prosthetic leg closely mimics the motion of a human knee. This form of control will be explored in further detail in subsection 2.1.2.

While this provides a predictable and safe way for the user to interact with the prosthesis, it has some limitations of its own. Firstly, because this control scheme does not leverage the hip dynamics, it can often result in the user having to work against the prosthesis instead of working with it. This results in an increased energy requirement for the user, as compared to that of a healthy individual. As a result, the amputee tends to get tired quickly. Secondly, because this approach requires that the impedance parameters be tuned to generate a torque output profile that is similar to that of a human knee, it is not easily adaptable to different types of locomotion activities. The torque profile for different types of activities needs to be studied for a normal human knee and then tuning needs to be performed for every individual activity. Moreover, since the kinematic constraints are not factored-in while calculating the control input, such schemes are inherently unable to handle disturbances and uncertain scenarios.

All these reasons mean that a lot is left to be addressed in terms of improving the quality of life for amputees. This prompts the need for a model based control scheme that is designed using an accurate model of gait dynamics.

1.3 Why MPC

Model Predictive Control (MPC) is an advanced algorithm that calculates the control input by predicting how the system states will evolve with time. MPC achieves this by solving an online optimization problem and calculating the optimal control input required to track any given reference trajectory.

However, this requires an accurate mathematical model of the system. Since the gait dynamics model is highly nonlinear, solving nonlinear constraints in real time for fast evolving processes can prove to be challenging, and requires a large amount of computational resources. For this reason, the MPC has traditionally been used for slow and stationary processes such as controlling a chemical plant. However, with the explosion in availability of cheap and powerful microprocessors, this constraint is fading away very quickly.

Using a model based approach makes the controller inherently more adaptable and more capable of handling disturbances and uncertain scenarios. Moreover, since the gait dynamics model used also accounts for the effects of hip, the controller can leverage it to reduce the energy requirement of the prosthetic device, or the amputee, or both.

1.4 Proposed Solution

The thesis seeks to provide a framework for designing a Model Predictive Controller for Active Transfemoral Prosthesis. In order to do this, however, some reference data is needed for conventional prostheses. This data will be used for validating the gait model and control design. The experimental setup and data collection methodology will be first explained in chapter 2. A brief summary of the model derivation process will then be presented, along with the final gait model used.

However, this model of the gait dynamics has some parameters that are not directly measurable, but are inherent to the system. Phase 1 of the research focuses on estimating these parameters and validating the obtained results. An optimization framework has been presented for carrying out this parameter estimation.

Once reasonable degree of confidence can be expressed in the estimated parameters, the research moves on to Phase 2. This phase focuses on the design of the Model Predictive Controller for mimicking a human knee. The results will then be analyzed to determine the most important factors affecting the performance of the controller.

The thesis finishes with a discussion of the possible future direction of this research and the continued studies that can help improve the obtained results further.

CHAPTER

2

FUNDAMENTALS OF ATP

Before trying to estimate the unknown gait phase parameters or designing a Model Predictive Controller, it is important to understand the gait dynamics and the underlying principles of the Active Transfemoral Prosthesis. This chapter will first present the basic construction of the prosthetic prototype and explain the details of experimental setup and data collection process that are relevant to this research. The division of gait phase into different stages will then be discussed, followed by the modeling process for deriving the nonlinear kinematic equations that describe the gait dynamics. It should be noted that this work was originally performed by other researchers and subsequently presented in papers which have been cited at relevant places in this chapter. Nevertheless, explaining how the elements of their research were adapted in the present context is essential to understanding the novel work presented in this thesis.

2.1 Experimental Setup

The data used for this research was collected in experiments performed by Liu *et al* and presented in [Liu14]. A brief summary of the relevant aspects of the experimental setup and data collection methods is now provided. Further details can be found in the paper cited above.

2.1.1 Prototype Construction

The objective of the ATP prototype was to mimic the biomechanics of human knees as best as possible. Off-the-shelf components were used in order to be able to prototype quickly. Thus, all the actuators and sensors are not necessarily as high-accuracy as may be available in a commercial prosthetic device. This prototype can either be worn directly by an amputee, or by a healthy individual with the help of a bypass adapter. The prototype design and experimental setup has been depicted in Fig. 2.1.

The knee moment was provided by a DC motor in conjunction with a slider-crank mechanism. A potentiometer connected to the knee joint measured the knee angle (β), and an encoder connected to the motor measured the angular velocity ($\dot{\beta}$). The knee torque (τ_{knee}) was measured by a force sensor connected to the moment arm, and the

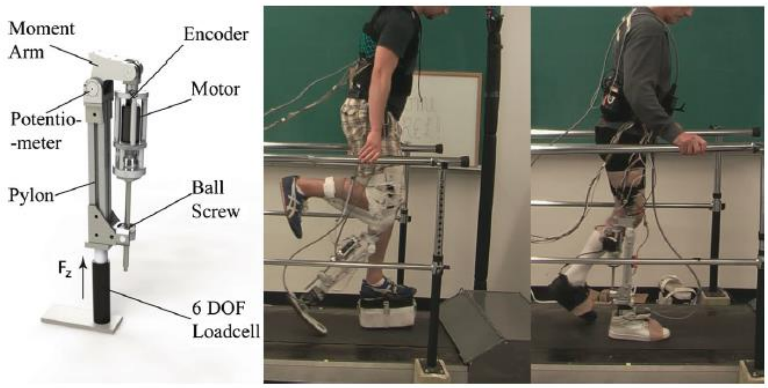

Figure 2.1 Solidworks model of prototype (*left*); Experimental setup for healthy subject (*center*); Experimental setup for Transfemoral amputee (*right*). Adapted from [Liu14]

Ground Reaction Force (GRF) was measured by a load cell attached below the knee unit.

This prototype did not have an actuated ankle joint. Although it is desirable to be able to manipulate the ankle joint since it provides an extra degree of freedom, this experimental setup can still provide adequate evidence as to whether the proposed algorithm is a feasible and practical solution. The effect of the presence of an actuated ankle joint on the results obtained will also be explained later.

2.1.2 Knee Impedance Control

The ATP prototype uses *Finite-State Impedance Control* in order to produce the desired effect at the knee joint. As explained previously, *Impedance Control* has proven to be a popular choice among researchers in recent years. This is because it provides the user with a way to interact with the prosthesis that is predictable and similar to human gait [MM80].

The gait cycle is first divided into multiple states. This division will be discussed in greater detail in subsection 2.1.3. It has been proven using regression analysis of gait data that knee torques can be adequately characterized in the form of passive spring and damper behavior within each of these states. The torque within these states can be expressed in the form -

$$\tau_{knee} = K \cdot (\beta(t) - \bar{\beta}) + d \cdot \dot{\beta(t)} \tag{2.1}$$

where, K = linear knee stiffness
d = linear knee damping
$\bar{\beta}$ = equilibrium angle for current state

The parameters $(K, d, \bar{\beta})$ are collectively referred to as *Control Parameters*. If they are constrained to be positive, the spring-damper system is guaranteed to be passive, and the joint angle β should converge to the stable equilibrium $\bar{\beta}$.

Before the ATP can be put into regular use, these control parameters need to be tuned carefully for each state such that the controller mimics a human knee well enough. A procedure similar to [Liu14], [Sup09] can be adopted for this purpose. During regular operation, the finite state machine first detects the current state depending on the sensory measurements and selects the control parameters to be used. The controller then uses these control parameters and the measured values of β and $\bar{\beta}$ to compute the torque that needs to be applied. This control architecture has been depicted in Fig. 2.2.

2.1.3 Gait Modes

In order for the Finite State Machine to determine the correct set of control parameters, the various gait modes and the switching criteria between them needs to be well-defined. The

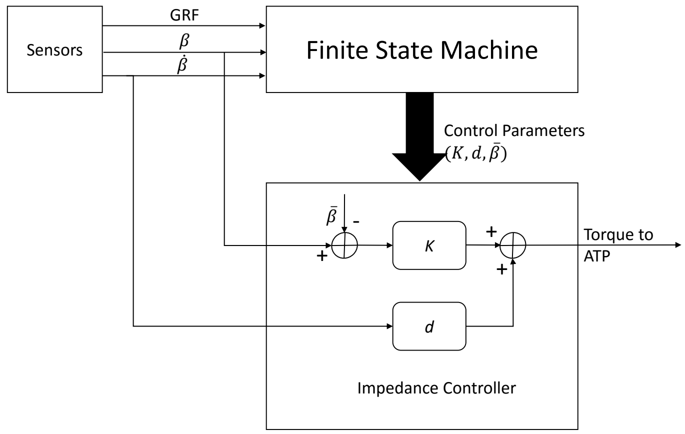

Figure 2.2 Control Architecture of ATP Prototype

number of gait modes required is not unique, and some different approaches have been previously [Liu14], [Zla02] proposed. The cycle can be broadly divided into two phases - (a) *Stance Phase* and (b) *Swing Phase*.

Gait Cycle starts with the stance phase when the leading foot touches the ground, and continues till it is in contact with the ground. The swing phase starts when this foot, which is now trailing, leaves the ground. This event is known as *'toe-off'* and is used as the dividing point between stance and swing phase. The swing phase again comes to an end when this foot touches the ground again, known as *'heel-strike'*.

Various methods have been used by different researchers for further dividing the stance phase into sub-modes. From a clinical gait analysis standpoint, it can be divided into three sub-modes. However, this research only focuses on designing a controller for the swing phase of the gait cycle and thus this subdivision of stance does not need to be explored.

The swing phase is further subdivided into two sub-modes, namely - *swing flexion* mode and *swing extension* mode. GRF sensor measurements are used to detect *toe-off*. GRF magnitude falling below a threshold marks the start of swing flexion. Swing extension starts and swing flexion ends when the shank starts its forward swing, and is characterized by knee angle reaching a specified threshold value. Swing phase ends when GRF magnitude goes above its threshold, marking a *heel-strike*. The subdivision of gait can be visualized as depicted in Fig. 2.3.

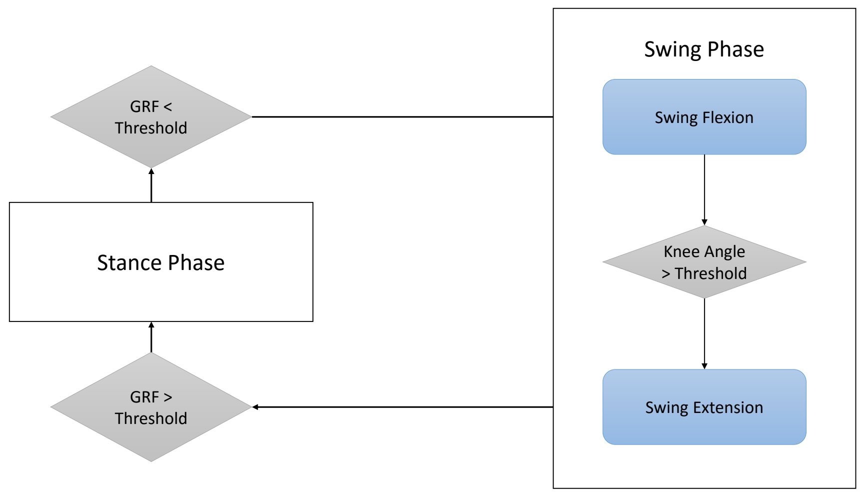

Figure 2.3 Gait Modes

2.1.4 Data Collection

Data was collected from a healthy subject wearing a bypass adapter on their right leg, as depicted in Fig. 2.1. The subject wore a height-adjustable shoe on the left leg to avoid discrepancies in leg lengths. The speed of walking was gradually increased to 0.6 m/s over

20 s and then held constant. Data collected over multiple cycles was averaged in order to remove the effect of minor intra-subject variations or measurement errors. Further details of the data collection methodologies can be found in [Liu14].

The *shank mass* and *shank inertia* in the mathematical model derived later, are the mass and inertia of the ATP. An important distinction should be noted here. Since the data was collected from a health subject wearing an adapter, the *thigh mass* is in fact the mass of the entire leg segment. The *thigh inertia* corresponding to this segment is also calculated accordingly. If however the data had been collected for an amputee, the thigh mass would be their actual thigh mass. This distinction is especially important when selecting initial guesses for these parameters in the estimation process.

2.2 Gait Modeling

Before the unknown system parameters can be estimated or the controller designed, having a good mathematical model of the swing phase is essential. Since this research seeks to study how hip dynamics can be leveraged while designing the control algorithm, it is necessary to incorporate all the hip torques in the model derivation. This model configuration has been termed as *Thigh-Knee-Shank (TKS)* System. A brief summary of the model derivation is now presented, followed by the nonlinear state equations that describe the model. Further details of the model derivation can be found in [Yan15].

2.2.1 Assumptions

The forces and torques experienced by a leg in everyday conditions are primarily restricted to the saggital plane. The effects on the leg outside this plane are negligible. Thus, this model only considers the motion of the leg in the saggital plane.

This system comprising the leg wearing ATP can be represented by a planar 2-link biped model. During the swing phase, the foot can be simplified into a point. The shank segment can be considered to be an ATP. The length of thigh (l_t) and shank (l_s) are considered to be known, since these can be measured very easily. The thigh and shank segments are also assumed to be Euler-Bernoulli beams of uniform densities. This does not introduce a significant error since the inertias of both segments are set independently in a reference range. This biped model has been depicted in Fig. 2.4.

2.2.2 Problem Formulation

The swing phase of gait has been modeled using Lagrangian Dynamics. Lagrange's Equation can be expressed as follows -

$$\frac{\mathrm{d}}{\mathrm{d}t}\frac{\partial T}{\partial \dot{q}_h} - \frac{\partial T}{\partial q_h} = Q_h, \quad h = 1, 2, \cdots, k \tag{2.2}$$

where, T = kinetic energy of leg
q_h = generalized coordinate
Q_h = generalized force
k = dimension of generalized coordinate

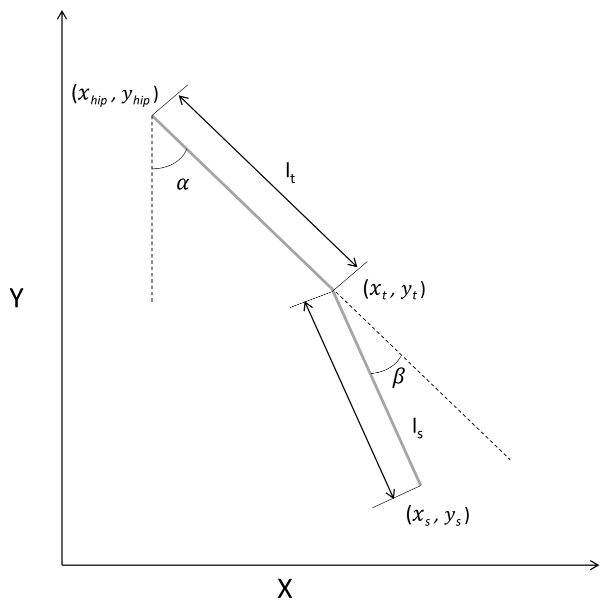

Figure 2.4 Biped Model

For the gait model, the Kinetic Energy is composed of -

$$
\begin{aligned}
T &= (Total\ Kinetic\ Energy)_{hip} + (Total\ Kinetic\ Energy)_{knee} \\
&= (Translational\ KE + Rotational\ KE)_{hip} + (Translational\ KE + Rotational\ KE)_{knee} \quad (2.3)
\end{aligned}
$$

And the Generalized force can be represented as -

$$\begin{aligned} Q_h &= \frac{\partial}{\partial q_h}\big(\textit{Total Work Done by system}\big) \\ &= \frac{\partial}{\partial q_h}\big(Work_{hip} + Work_{knee}\big) \\ &= \frac{\partial}{\partial q_h}\big(\textit{(Rotational Work + Work against gravity)}_{hip} + \textit{(Rotational Work + Work against gravity)}_{knee}\big) \end{aligned} \tag{2.4}$$

The generalized coordinates for the biped model are -

$$q = (\alpha, \beta, x_{hip}, y_{hip}) \tag{2.5}$$

$$\begin{aligned} \text{where, } \alpha &= \text{hip angle} \\ \beta &= \text{knee angle} \\ x_{hip}, y_{hip} &= \text{horizontal and vertical position of hip} \end{aligned}$$

2.2.3 ODE Gait Model

The mathematical expressions are derived for Eq. 2.3 and Eq. 2.4. These expressions are then substituted in Eq. 2.2 and differentiated with respect to the generalized coordinates described in Eq. 2.5. This gives the set of four differential equations that describe the dynamics of the system about $(\alpha, \beta, x_{hip}, y_{hip})$.

However, the equations corresponding to x_{hip}, y_{hip} are only useful if the hip position is also to be controlled. Moreover, these equations contain force terms corresponding to horizontal and vertical interaction of hip. These quantities are not easily measurable. On the other hand, it is sufficient to use the equations corresponding to α , β to control the joint angles. These two equations are now presented.

Torque Equation 1, TE$_1$

$$\begin{aligned}
&\left(J_a + J_b + \frac{1}{3} m_s l_s^2 + m_s l_s l_t \cos\big(\beta(t)\big) + m_s l_t^2 + \frac{1}{3} m_t l_t^2\right) \ddot{\alpha}(t) \\
&-\left(J_b + \frac{1}{3} m_s l_s^2 + \frac{1}{2} m_s l_s l_t \cos\big(\beta(t)\big)\right) \ddot{\beta}(t) \\
&+\left(\frac{1}{2} m_s l_s \cos\big(\alpha(t) - \beta(t)\big) + m_s l_t \cos(\alpha(t)) + \frac{1}{2} m_t l_t \cos(\alpha(t))\right) \ddot{x}_{hip}(t) \\
&+\left(\frac{1}{2} m_s l_s \sin\big(\alpha(t) - \beta(t)\big) + m_s l_t \sin(\alpha(t)) + \frac{1}{2} m_t l_t \sin(\alpha(t))\right) \ddot{y}_{hip}(t) \\
&-m_s l_s l_t \sin\big(\beta(t)\big) \dot{\alpha}(t) \dot{\beta}(t) + \frac{1}{2} m_s l_s l_t \sin\big(\beta(t)\big) \dot{\beta}(t)^2 \\
&-\frac{1}{2} m_s l_s g \sin\big(\alpha(t) - \beta(t)\big) - m_s l_t g \sin(\alpha(t)) - \frac{1}{2} m_t l_t g \sin(\alpha(t)) \\
&-\tau_{knee}(t) - \tau_{hip}(t) = 0
\end{aligned} \tag{2.6}$$

Torque Equation 2, TE$_2$

$$\begin{aligned}
&-\left(J_b + \frac{1}{3} m_s l_s^2 + \frac{1}{2} m_s l_s l_t \cos\big(\beta(t)\big)\right) \ddot{\alpha}(t) + \left(J_b + \frac{1}{3} m_s l_s^2\right) \ddot{\beta}(t) \\
&-\frac{1}{2} m_s l_s \cos\big(\alpha(t) - \beta(t)\big) \ddot{x}_{hip}(t) - \frac{1}{2} m_s l_s \sin\big(\alpha(t) - \beta(t)\big) \ddot{y}_{hip}(t) \\
&+\frac{1}{2} m_s l_s l_t \sin\big(\beta(t)\big) \dot{\alpha}(t)^2 + \frac{1}{2} m_s l_s g \sin\big(\alpha(t) - \beta(t)\big) \\
&+\tau_{knee}(t) = 0
\end{aligned} \tag{2.7}$$

CHAPTER

3

PHASE 1 - PARAMETER ESTIMATION

Before a model based controller can be designed, knowledge of all system parameters is essential. However, all the system parameters are not readily available. And neither is it possible to directly measure these parameters. Phase 1 of this research deals with the estimation of these system parameters. This chapter is organized as follows. First the motivations and objectives of this phase are discussed. Then a high level overview of the process is provided, followed by a detailed description of the two major approaches used and their pros and cons. This is followed by an introduction of the final hybrid approach adopted. The chapter is summed up by a discussion of the results of this phase and future work that could help improve the results.

3.1 Motivations and Objectives

One of the requirements for effectively implementing a Model Predictive Controller is that a fairly accurate mathematical model of the system is needed, and a close approximation of all the system parameters involved is essential. Without a good approximation of the model or it's parameters, the results produced by the Model Predictive Controller can be way off the mark from the desired results. In fact, such a controller can destabilize the system quickly, depending on how inaccurate our estimates are.

The Thigh-Knee-Shank Model describing the human gait, henceforth referred to as the *System Model* or *TKS Model*, has four parameters which are directly dependent on the amputee wearing the prosthesis, who is henceforth referred to as the *subject*. Namely, *Thigh Mass* (m_t), *Shank Mass* (m_s), *Thigh Inertia* (J_a) and *Shank Inertia* (J_b). These four parameters are collectively referred to as the *inertial parameters*.
These parameters are inherent to the system and a good estimate is essential before we can start designing a MPC based controller. However, as is apparent, it is not possible to directly measure these parameters. Neither is it possible to determine these parameters from some simple calculations based on data of the physical process. Estimation of these inertial parameters is the primary motivation behind this stage of the research.

In addition to these inertial parameters, when the control law is of the form previously discussed, the torque input equation involves three other parameters which are collectively referred to as the *control parameters*. These parameters are not inherent to the system. In fact the control system designer has freedom to choose these parameters depending on the desired performance characteristics. The aforementioned control technique has been used for collecting the data which will be used for estimating the system parameters. Thus, the problem of parameter estimation also involves estimating these control parameters. This provides us with a few extra degrees of freedom while determining estimates for the inertial parameters. However, it is desirable to have an estimate of the actual values of these parameters which were used for collecting the data. Success in getting control parameter estimates close to the originally used values would provide added confidence about the validity of the inertial parameter estimates.

3.2 Overview

The problem of estimating the system parameters has been treated as an optimization problem with the goal of fitting the calculated values of gait phase to measured data as best as possible. Before diving into the details of the optimization problem, it is beneficial to provide a high-level overview of the approach and present a brief discussion of the primary considerations behind the chosen approach. The inertial and control parameters to be estimated are treated as the optimization parameters. The nonlinear differential equations representing the system dynamics are used to calculate the gait phase values for the entire gait cycle. The deviation of these calculated parameter values from the measured parameter values is measured and this deviation is attempted to be minimized. Ideally, these deviations should be reduced to very small values as the parameter estimates reach a region very close to the actual values.

However, this may not be the case in reality due to a number of factors. Firstly, the actuators might have unmodeled nonlinearities, which is easily possible in relatively inexpensive actuators. As a result, the actual torque produced by the actuators may not follow the desired control law very closely. Secondly, quantitative information regarding the interaction of the leg with the hip and upper body might be unavailable. The effects of these and some other factors influencing the results will be discussed in greater detail in later sections. However, as can be seen already, these factors make it practically impossible to follow the measured data very accurately. So it can be tough to determine when we have reached the minima of this optimization problem. The problem is further complicated by the existence of many local minima since the design space is highly non-convex.

This precludes the sole use of *local* optimization algorithms like *Gradient Descent*. Although these algorithms can solve optimization problems locally in a very efficient manner, they often get stuck in local minima. This forces the use of *global* optimization algorithms like *Genetic Algorithm*. Although these algorithms can identify the promising regions in design space reliably, convergence to the global minima can be very slow. In fact, even though they can get very close to the global minima, final convergence to the absolute global minima can not be guaranteed. And even these *global* algorithms are overwhelmed by the sheer size of the optimization problem and the non-convexity of the design space. All these factors lead to the need for an optimization process which is specifically designed for this problem, as against formulating it like a general optimization problem and applying a standard procedure.

3.3 Problem Formulation

As discussed in previous section, this optimization problem needs to be broken down into smaller steps such that the individual problems are of a manageable size for the optimization algorithms. At each of the steps in the optimization process, a hybrid algorithm composed of a *global* and *local* solver are used, unless otherwise noted. This hybrid algorithm makes it possible to quickly search over large regions of the design space using a *global solver* and identify regions which could potentially provide good solutions. Once a few good regions are identified, the optimization is then handed over to a *local solver* which can then quickly identify the best solutions within these regions. The optimization process is agnostic to the actual global and local solvers used at every step.

Two different optimization strategies were examined for this problem, and then a combination of the two was finally implemented such that the strengths of both the strategies are combined. The following sections and subsections explain some common terminology used henceforth, followed by an explanation of the two major strategies used and their respective advantages and disadvantages. The final optimization process is then presented followed by the actual local and global optimization solvers, and other implementation details.

3.3.1 Terminology Used

Although it might generally be the practice to define the terminology used in the Appendix, lot of the terminology used henceforth might not be standard or unfamiliar to the readers, in the very least. They have been extensively used to explain the optimization strategies and process and are essential to understanding the material that follows. Hence, it would be appropriate to discuss the terminology used in advance.

Subject Parameters

Subject Parameters collectively refers to the set of parameters which are either inherent to the subject or are defined during the control design process. These parameters remain constant during the entire gait cycle. They can be further subdivided as follows -

1. **Inertial Parameters** -

   These parameters are inherent to the subject and are the primary goal of this phase of the research.

   - m_t - Mass of the thigh segment of the amputee, as defined by Dempster Body Segment Data [Win09]. However, as described previously, a healthy subject was used for collecting the data while wearing a prosthetic adapter. Hence, for this research, m_t will actually be the mass of the entire lower limb. For future work, if data collection is done for an amputee subject, this parameter should be used as originally defined.
   - m_s - Mass of the shank segment of the amputee, as defined by Dempster Body Segment Data [Win09].
   - J_a - Mass moment of Inertia of the thigh segment.
   - J_b - Mass moment of Inertia of the shank segment.

2. **Control Parameters** -

   These parameters are determined during the control design process. The designer has the freedom to choose these parameters such that the system meets the performance requirements. Even though they are not the primary goal for this phase of the research, including them as optimization parameters provides extra degrees of freedom. Also, getting a good approximation of these values, as compared to the actual values used, provides reaffirmation regarding the validity of the inertial parameter estimates.

- K_{flex} , K_{ext} - Proportional gain for the knee angle, for flexion and extension cycle respectively.
- d_{flex} , d_{ext} - Proportional gain for the knee angular velocity, for flexion and extension cycle respectively.
- $\bar{\beta}_{\text{flex}}$, $\bar{\beta}_{\text{ext}}$ - Reference knee angle, for flexion and extension cycle respectively.

Cycle Kinematic Parameters

Cycle Kinematic Parameters collectively refers to the states of the system and their derivatives. Thus, values for these parameters are time dependent and are stored separately for all the time steps in the gait cycle. They can be further subdivided as follows -

1. **Knee Angular Parameters** -
   - β - Knee Angle
   - $\dot{\beta}$ - Knee Angular Velocity
   - $\ddot{\beta}$ - Knee Angular Acceleration
2. **Hip Angular Parameters** -
   - α - Hip Angle

- $\dot{\alpha}$ - Hip Angular Velocity
- $\ddot{\alpha}$ - Hip Angular Acceleration

3. **Hip Coordinate Parameters** -

   - x - Hip Horizontal Position
   - $\dot{x}$ - Hip Horizontal Velocity
   - $\ddot{x}$ - Hip Horizontal Acceleration
   - y - Hip Vertical Position
   - $\dot{y}$ - Hip Vertical Velocity
   - $\ddot{y}$ - Hip Vertical Acceleration

Evaluated Parameters

Evaluated Parameters collectively refers to the parameters that are not known at the outset and are determined from the gait data by performing some calculations. These parameters are -

- C_{eq} - A vector containing the constraint violations for all the equality constraints of the optimization problem.
- τ_{knee} - The torque applied at the knee by the actuator of the prosthetic leg.
- τ_{hip} - The torque applied by the subject at the hip joint.

3.3.2 Optimization Parameters

Optimization Parameters collectively refers to the set of parameters which are optimized by the solver at any step of the optimization process. They are also alternatively referred to as

Design Parameters. Which of the previously described parameters are used as optimization parameters varies with the strategy being used and will be explained separately for each strategy. They are represented as -

$$d_i = i^{th} \text{ optimization parameter}, \forall i = 1 \text{ to } n$$
$$\text{where, } n = \text{number of optimization parameters}$$

Design refers to any combination of values for the optimization parameters that is being evaluated as a possible solution to the optimization problem. The order of the optimization parameters within a design is irrelevant. When the design is expressed in vector form, it is also referred to as the *Design Vector* and can be represented as -

$$D = [d_1, \quad \dots \quad , d_n] \tag{3.1}$$

3.3.3 Constraints

Constraint Equations refers to the nonlinear differential equations which need to be satisfied by any design in order to be considered a feasible solution. As explained previously, only the torque equations from the system model will be used for this phase of the research. They are referred to as -

- TE_1 - Refers to Eq. 2.6. This equation is used for calculating α and it's derivatives.
- TE_2 - Refers to Eq. 2.7. This equation is used for calculating β and it's derivatives.

Depending on the strategy, either or both of these equations will be used as optimization constraints.

3.3.4 Weighted Parameters

Weighted Parameters collectively refers to the set of parameters whose deviations are measured and weighted in the cost function. They are independent of the optimization parameters, since deviations are only relevant for cycle parameters. Explicitly specifying weighted parameters also provides the flexibility of including only some of the cycle parameters. Whereas, optimization parameters may include some or all of subject and cycle parameters. The set of weighted parameters can be represented as -

$$\Omega = [\omega_1, \quad \dots \quad , \omega_m], \tag{3.2}$$

$$\text{where, } m = \text{ number of weighted parameters}$$

$$\text{and, } \omega_i = \text{ variable representing any generic system parameter}$$

Each weighted parameter needs to be assigned a numerical weight. The absolute magnitude of these variables is not significant. But their magnitude relative to each other represents the degree to which their accurate matching is important. The weights can be represented in vector form as follows -

$$W = [w_1, \quad \dots \quad , w_m] \tag{3.3}$$

$$\text{where, } w_i = \text{weight of } i^{th} \text{ weighted parameter}$$

3.3.5 Cost Function

The objective of the optimization problem is to find a design that minimizes the cost function. The cost function value is evaluated for every design with a lower value representing a better design. Since the goal of the optimization is to fit the calculated data to the measured data as best as possible, the cost function is defined such that it returns the weighted squared deviation of the calculated data. However, the individual parameter deviations need to be normalized with respect to its maximum measured value. This ensures that the difference in scale of the individual parameters doesn't skew the cost function value. The cost function can be expressed as follows -

$$J = \sum_{i=1}^{m} \frac{w_i \cdot (\widehat{\mathbf{x}}_i - \bar{\mathbf{x}}_i)^2}{\max|\bar{\mathbf{x}}_i|} \tag{3.4}$$

$$\begin{aligned} \text{where, } \widehat{\mathbf{x}}_i &= \text{vector containing calculated value of } \omega_i \text{ for entire gait cycle} \\ \bar{\mathbf{x}}_i &= \text{vector containing measured value of } \omega_i \text{ for entire gait cycle} \\ \max|\bar{\mathbf{x}}_i| &= \text{maximum absolute value of } i^{th} \text{ parameter from measurements} \\ &\text{and, both are vectors containing the parameter value for all time steps} \end{aligned}$$

Although it should be noted that there isn't enough quantitative information available regarding the τ_{hip} profile during the gait cycle. Existing studies mostly focus on the qualitative nature of how τ_{hip} varies over the cycle and its interplay with other factors. There were certainly no τ_{hip} measurements available for this study. However, analysis of gait data for healthy individuals shows that τ_{hip} values are relatively low compared to τ_{knee}. This indicates that having lower magnitude of τ_{hip} might be more energetically efficient. And even though the magnitude of τ_{hip} for amputees might turn out to be higher as compared to τ_{knee} for amputees, it is desirable to minimize τ_{hip} as much as possible. For this reason, whenever τ_{hip} is to be included as a weighted parameter, instead of measuring the deviations from some measured value, its norm will be penalized. However, the corresponding weight will be maintained at a relatively low value, since accurately tracking hip and knee parameters is of greater concern.

3.4 Optimization Strategies

As described previously, the following two distinct optimization strategies were adopted for the purpose of this research -

1. Differential Equation Solver Method
2. Discrete Optimization Method

It should be noted that these names have been coined to easily maintain the distinction in the following discussion regarding them. There are no directly relevant standard terms to adequately describe these methods. Both of these strategies involved some variations of their own. The following section explains these strategies along with their adopted variations. After discussing their advantages and disadvantages, the final adopted optimization roadmap will be presented.

3.4.1 Differential Equation Solver Method

The *Optimization Parameters* for this method are chosen from the set of subject parameters. When the inertias of thigh or shank are included as Optimization Parameters, their values are set independently of the mass values of the respective segments, albeit in a range that is still linked to the respective mass values. If these parameters are not included however, their values are coupled to the respective mass values using the following relation -

$$j_a = k_t m_t l_t^2 \qquad j_b = k_s m_s l_s^2 \tag{3.5}$$

where, k_t and k_s are the radii of gyration of the thigh and shank segments respectively, and their values are obtained from statistical models presented in [Win09].

The solver generates new designs by picking values for the chosen optimization parameters within the specified range. And as the name suggests, the nonlinear state equations are then solved over the entire gait cycle in order to generate values for the state variables, which we refer to as cycle parameters, for all time steps. This calculated data is then used to evaluate the design based on the cost function value. The major advantage offered by this method is that it eliminates the need for satisfying nonlinear constraints at all time steps. This reduces the computational burden significantly and provides the ability to quickly scan large areas of the design space.

However, since the main optimization problem is broken down into several smaller steps, the specific details of this method also vary depending on the sub-problem being solved. The variations are now discussed in detail.

3.4.1.1 Optimizing with coupled State Equations

As the name suggests, this method uses the coupled system of differential equations to solve for all cycle parameters. α , β and their first derivatives are treated as the state variables. The measured data is only used for setting the initial conditions. After that, the derivative values obtained from the system of state equations (TE_1 and TE_2) are used for updating the values of state variables at each time step. Since all the cycle parameters are being solved for, all their deviations needs to be measured in addition to τ_{hip} , which is treated as an independent optimization parameter. However, depending on the priority, the magnitude of these weights can be adjusted such that greater emphasis is placed on matching some of these parameters instead of trying to match all of them equally.

Intuitively, this method seems a very elegant way of solving the optimization problem since it combines all the cycle parameters into one problem without having to satisfy nonlinear constraints at all time steps. However, this method presents many challenges. Firstly, τ_{hip} being an independent optimization parameter needs to be estimated at all time steps in the gait cycle such that it satisfies TE_1 . And since all the other cycle parameters are not being controlled independently at all time steps, it is incredibly tough to get a good estimate for τ_{hip} without having a closed-form solution, or at least a reference curve to track. Moreover, the high degree of coupling between the two equations causes any small tracking error in one of the state variables to quickly blow up and manifest itself in all the state variables as the equations are solved over time. This makes it tough to analyze the tracking performance of individual state variables and determine the cause of the errors in the absence of good starting estimates for the subject parameters. Even a slight perturbation in one of the parameters might result in a huge tracking error in some or all of the state variables, and provide misleading indications about the direction that some parameters need to take. This problem can be tackled by decoupling the state equations, and as a result, the state variables. The decoupled state equations can then be used to solve different aspects of the problem.

3.4.1.2 Optimizing with TE_2 only

This method only uses TE_2 to solve for $(\beta, \dot{\beta}, \ddot{\beta})$. Measured values of $(\alpha, \dot{\alpha}, \ddot{\alpha})$ are used at all time steps for performing the required calculations. TE_1 is used to calculate the value of τ_{hip} at all time steps that would perfectly satisfy the equation.

Since, α and its derivatives are not being calculated, there is no need to measure their deviations from the measured values. Although, as explained in subsection 3.3.5, it is required to minimize τ_{hip} as much as possible. Thus, the weighted parameters include β, $\dot{\beta}$, $\ddot{\beta}$ and τ_{hip}. This method separates the problem of estimating the subject parameters for optimal tracking of β and its derivatives, which is the more important goal for this problem.

3.4.1.3 Optimizing with TE_1 only

Similar to the method described in subsubsection 3.4.1.2, this method only uses TE_1. However, TE_1 contains an additional unknown variable in the form of τ_{hip}. Thus, unlike the previous method, this can not be used to solve for $(\alpha, \dot{\alpha}, \ddot{\alpha})$. Instead, measured values of all cycle parameters are used to solve for τ_{hip} which would satisfy the equation perfectly.

Since none of the cycle parameters are being calculated, there is no need to measure their deviations from the measured values. Instead the norm of τ_{hip} is the sole quantity penalized in the cost function. Thus, this method presents a suitable way for estimating the subject parameters for minimizing the τ_{hip} that a subject needs to apply during the gait cycle, thus minimizing the energy spent by the subject.

3.4.2 Discrete Optimization Method

The *Optimization Parameters* for this method include a combination of subject parameters, and some or all of cycle parameters. The inertias of the thigh and shank segments can be independent or coupled to respective masses, as described in subsection 3.4.1.

For every design, the solver generates a value for each subject parameter in their specified ranges. Similarly, it generates a value for all cycle parameters included in the set of optimization parameters, at each time step. Generally, only one out of α (β) and its derivatives is chosen, and the other derivatives are calculated using finite difference method. These cycle parameters are subject to constraints which limit the maximum change in any given parameter between successive time steps. The design is also required to satisfy the nonlinear constraints (TE_1 and/or TE_2 , depending on variation) at each time step. And measured values are used for all cycle parameters that are not included as optimization parameters.

This method increases the computational burden significantly due to multiple reasons. Firstly, since cycle parameters need to be determined at all time steps, the number of optimization parameters is significantly larger than before. In addition, these optimization parameters also need to satisfy lot of linear inequality constraints between successive time steps and nonlinear equality constraints at all time steps. However, it provides an added advantage by providing extra degrees of freedom in trying to find a solution for this highly non-convex optimization problem. Thus it can prove to be of huge benefit when used judiciously.

As with *DE Solver Method*, this method also has its own variations which are now discussed in detail.

3.4.2.1 Optimizing with coupled State Equations

This method presents the most obvious variation of the method explained above. *Optimization Parameters* include both α and β , in addition to the subject parameters. The design needs to satisfy TE_1 as well as TE_2 at every time step. And since both sets of cycle parameters are being optimized, both of them need to be weighted as well. Whether τ_{hip} is penalized in the cost function depends on the method chosen for handling it.

Although this method is the most broadly applicable, it also carries with it the most amount of computational burden. And the sheer number of optimization parameters mean that it is not feasible to get any meaningful results using this method directly. However once a decent estimate is available for the design, this method can prove to be handy while trying to refine the estimate. As was the case with subsection 3.4.1, the methods described next will help in overcoming some of these difficulties.

3.4.2.2 Optimizing with TE_2 only

For every design generated by the solver, this method only imposes TE_2 as the nonlinear equality constraint, in addition to the linear inequality constraint previously described. Besides the subject parameters, *Optimization Parameters* may include any combination of α , β and their derivatives. And since cycle parameters are independently determined at each time step, this method also provides the ability to include x , y and their derivatives.

This was not possible with the *DE Solver Method* since the state equations corresponding to x and y are not being considered due to lack of knowledge regarding hip forces. Measured values are used for whichever cycle parameter is not included as an optimization parameter.

3.4.2.3 Optimizing with TE_1 only

This method is very similar to subsubsection 3.4.2.2. It imposes TE_2 as the nonlinear equality constraint instead of TE_1 . However, it does have an extra parameter in the form of τ_{hip} .

There are two different ways for handling τ_{hip} though. Like the other cycle parameters, τ_{hip} can be treated as an independent optimization parameter, and estimated at each time step such that it satisfies the constraints as closely as possible. Alternatively, τ_{hip} can be determined by solving for it's value at each time step from TE_1 based on the values determined for other cycle parameters by the solver. When this method is used for determining τ_{hip} , it needs to be included as a weighted parameter in the cost function. Otherwise, τ_{hip} would absorb all the error in the equation and would return a cost function value of zero for any design. In the absence of a reference curve for τ_{hip} , this method is much more effective, direct and computationally efficient. In fact, keeping α and β constant and solving solely for τ_{hip} can be used as a quick way to explore the design space of the subject parameters.

3.5 Optimization Roadmap

It should be clear by now that the highly non-convex nature of this optimization problem makes it impossible to solve by solely using one of the methods discussed so far. All of them present their own set of strengths and challenges. However the problem can be broken down into smaller sub-problems and the strengths of these methods can then be exploited at different stages. Further, it turns out that the dynamics of the extension phase of the gait cycle are much more complicated and tougher to satisfy as compared to the flexion phase. Thus, estimating parameters for flexion phase is much easier. The result is an elaborate procedure with many steps along the way such that the different subject parameters are estimated sequentially and their estimates refined gradually, instead of trying to estimate all of them at once.

Estimating Parameters for Flexion Phase

For this phase of the optimization process, only data corresponding to the flexion phase is used. Thus, the relevant subject parameters only include the *inertial parameters* and *flexion control parameters*.

(a) Use DE Solver Method with TE_2 only -

This step yields a decent estimate for m_s and J_b . Also, it helps to evaluate the design space for flexion control parameters (K_{flex} , d_{flex} , $\bar{\beta}_{\text{flex}}$) and determine which regions could provide promising results.

(b) Use Discrete Optimization Method with TE_2 only -

Since decent estimates are already available for m_s and J_b , this step helps to further refine these estimates. This should result in an approximation that is pretty close to the real values of these parameters. This step also serves to narrow down the regions for flexion control parameters. But these estimates are still relatively rough and leave room for improvement.

(c) Use DE Solver Method with TE_1 only -

The values for m_s , J_b and flexion control parameters are fixed to the center of their estimated ranges for now. The system performance is now evaluated by varying m_s and J_a over relatively wide ranges in order to try and minimize τ_{hip} to estimate the regions in design space that would yield promising results for m_s and J_a .

(d) Use Discrete Optimization Method with TE_1 only -

This step serves to fine-tune the estimates of flexion control parameters. It also helps to narrow down the range for m_t and J_a , but these estimates are still relatively rough and will be improved upon later.

(e) Use Discrete Optimization Method with coupled State Equations -

Now that estimates for m_s , J_b and the flexion control parameters have been considerably fine-tuned, only m_t and J_a still remain with rough estimates. This is the perfect type of scenario for exploiting the strengths of this method. Thus all subject parameters (inertial and control), along with cycle parameters, are included as optimization parameters and the estimates of m_t and J_a are refined.

By this point in the optimization process, the estimates obtained for the inertial parameters and flexion control parameters should be in fairly nnarrow regions around the actual values. Using this knowledge, estimation of extension control parameters can now be performed.

Estimating Parameters for Extension Phase

For this phase of the optimization process, only data corresponding to the extension phase is used. Thus, the relevant subject parameters only include the *inertial parameters* and *extension control parameters.*

(a) Use DE Solver Method with TE_1 only -

All the inertial parameters are fixed to the center of their respective range, which should be very narrow now. Similar to flexion cycle, the system performance is now evaluated by varying extension control parameters over relatively wide ranges and attempting to minimize τ_{hip}. This helps to estimate the regions in design space which would provide promising results for the extension control parameters

(b) Use Discrete Optimization Method with TE_1 only -

This step serves to narrow down estimates of extension control parameters.

(c) Use Discrete Optimization Method with coupled State Equations -

As with flexion cycle, the last step is to solve for the coupled state equations while including all subject parameters as optimization parameters. However there is a small difference here. It turns out that it is really tough to get a good fit for TE_2 for a variety of reasons which will be discussed later. Thus, β is not included in optimization parameters, whereas α is. The method then tries to refine the estimates by minimizing τ_{hip} for TE_1 and the constraint violation for TE_2.

Thus, by breaking the problem down into multiple steps, all the subject parameters can be estimated with reasonable accuracy.

3.6 Optimization Solver - Implementation Details

The optimization procedure explained in the previous section is independent of the specific optimization solver or the optimization algorithm adopted at every step in the process. Using different solvers can provide comparable results as long as the choice of solver adheres to certain principles. This section examines some of the considerations while choosing a solver and tuning its parameters.

3.6.1 Choice of Solver

As explained previously in section 3.2, the strengths of a *global* and *local* optimization solver can be combined by using a *hybrid* scheme. Such a hybrid scheme would use a global solver initially to scan large regions of the design space and identify regions which could provide promising designs. It would then hand over the optimization in these regions to a local solver which can efficiently and accurately determine the solutions in these regions. The choice then boils down to choosing a global and a local solver to be a part of this hybrid solver.

Although multiple global solvers were tried for the purpose of this research, the one used most predominantly was *Genetic Algorithm*. It was chosen mainly because it provides a great amount of flexibility. Also, it tends to be agnostic to the problem being optimized, as long as a cost function is provided which can evaluate every design created by the solver. The solver does not need any knowledge of the internal workings of the problem either, like a gradient, which would be tough to obtain for this problem. Also, since most of the operations to be performed on designs within a generation are independent, it provides the opportunity to parallelize most of these computations. The MATLAB Global Optimization Toolbox [Mat16d] was used for this research since it provides a great implementation of this solver with a wide variety of options and customizations. The working of Genetic Algorithm will not be discussed in a lot of detail since it is a well established algorithm and justice can not be done to it within the framework of this thesis. One such explanation can be found here [Mat16a]. Instead the forthcoming sections will focus on how this solver is adapted for the needs of the problem at hand. When used in conjunction with the MATLAB Parallel Computing Toolbox [Mat16j], lot of the computations at each generation can be parallelized. This results in huge gains in efficiency and reduces the computational time by a large margin.

The MATLAB Optimization Toolbox [Mat16i] provides a wide variety of options of local solvers as well. As explained by [Mat16h], the best option for nonlinear constrained optimization problem is *fmincon*. It essentially provides an implementation of the interior point algorithm as described in [Wik16]. An elaborate explanation of the working of *fmincon* has been provided in [Mat16c].

3.6.2 Initial Estimate

Genetic algorithm can find an optimal, or nearly optimal solution, regardless of the initial estimate. Nevertheless, providing a good initial estimate can be of huge benefit in a highly non-convex problem like this.

[Win09] provides body segment parameter data for 2D studies collected from statistical models. These parameters can be used to calculate the expected values of the masses of shank and thigh segments. The parameters obtained from these studies can also be used to calculate the approximate lengths and radii of gyration of the two segments, which are then used to calculate the respective inertias. Admittedly, the final values obtained for the inertial parameters can deviate from their statistically expected values, especially for amputees for whom the gait dynamics may be slightly altered in order to adapt to the change in biomechanical cost of walking. Nevertheless, these estimates provide a good starting estimate, and definitely better than any initial estimate that can obtained from another source.

For control parameters, one needs to rely on the estimate provided by the researchers performing the data collection experiments. These estimates can come with caveats of their own. Depending on the type of equipment used for control of the prosthesis, precise values of these control parameters may not be available to these researchers themselves. Additionally, depending on the quality and nature of actuation devices, it may not be able to follow the desired control law accurately, as explained in section 3.2. Thus, greater flexibility can be allowed in the estimation of these control parameters as long as their estimates lie in similar regions.

3.6.3 Parameter Weights

The relative magnitudes of the parameter weights represents the emphasis laid on tracking the individual parameters. Since the measurements are generally made for the angle parameters, they provide the greatest confidence regarding the correctness of their values. On the other hand, the first and second derivatives are calculated from the cycle parameters by difference method, even a small error in their measurement can quickly cascade into high magnitudes in the derivatives. Thus, the greatest emphasis is laid on tracking the angle parameters accurately, and hence the ratio of magnitude of their weights is to be inversely proportional to the time step of measurement. It can be represented by a simple equation in the following form where x represents any generic cycle parameter -

$$\frac{x}{\dot{x}} \propto \frac{1}{\Delta t}$$

where, $\Delta t =$ sampling time for measured data

The deviations of the calculated parameters from measured parameters should be normalized with respect to the maximum measured value of that parameter. Otherwise, the difference in scale of the various parameters would unfairly skew the quantities being penalized in the cost function and would thus have the undesirable effect of shifting the emphasis of optimization.

Table 3.1 Reference Values for Optimization Parameter Weights

Optimization Parameter	**Optimization Weight**
Cycle Kinematic Parameters	
α	500
$\dot{\alpha}$	250
$\ddot{\alpha}$	100
β	500
$\dot{\beta}$	250
$\ddot{\beta}$	100
$\ddot{x}$	100
$\ddot{y}$	100
Evaluated Parameters	
τ_{hip}	25
C_{eq}	20

It should be noted that since no reference measurements are available for τ_{hip} , the calculated values of this parameter can not be normalized. Thus, the weight for τ_{hip} needs to be manually calibrated taking into account its magnitude.

There is no unique value of the parameter weights that return the best results and a lot of freedom is available for choosing their exact values. They need to be determined individually for the different steps in the optimization roadmap by trial. Nevertheless, some guidance regarding their relative magnitudes and ranges can prove to be useful. Values provided in Table 3.1 can be used for reference.

3.6.4 Additional Adaptations for Speed

As mentioned several times, this optimization process is very computationally intensive. Even after parallelizing the computations on a local machine, one optimization run can easily take several days to complete in order to get any meaningful results. This is clearly not a desirable scenario. It would take up valuable computing resources on a local machine and make it very tough to try out new variations of existing strategies.

This problem can be addressed, at least partially, if access is available to a distributed computing cluster. MATLAB Distributed Computing Toolbox [Mat16b] automates lot of the additional work needed to set up the problem to run on a distributed computing cluster. Time and resources permitting, these efforts can be taken a step further to design custom code for the problem being solved, in order to fully exploit the advantages offered by such a setup. But even with the use of a readily available solution, this approach can result in huge time savings. In addition, it frees up the computing resources on the local machine for the researcher to try out other strategies.

The High Performance Computing Cluster of NC State University [Uni16] was used for the purpose of this research.

3.7 Discussion of Results

The final estimates obtained for the *Subject Parameters* are now presented along with a visualization of the output of the simulated gait cycle. This will be followed by a discussion of the possible factors affecting the results and the factors that could help to improve these results.

3.7.1 Subject Parameter Estimates

The final estimates of the subject parameters that returned the best possible match for the hip and knee angles are listed in Table 3.2.

The important parameters to note here are the *Inertial Parameters*. Estimating these was the primary goal of this phase of the research, since their knowledge is essential for designing a Model Predictive Controller. The values of *Control Parameters* are essential for getting a good estimate of the knee torques and thus satisfying the nonlinear constraints. However, they are not essential for the control design process.

The accuracy of these estimates will now be analyzed.

3.7.2 Analysis of Results

In order to analyze the accuracy of the results, the system of differential equations need to be solved over time. However, solving the coupled system of equations poses a challenge. There

Table 3.2 Final estimates of Subject Parameters

Parameter	Estimate
Inertial Parameters	
Mass of Thigh (m_t)	4.492 kg
Mass of Shank (m_s)	13.5 kg
Inertia of Thigh (J_a)	0.4057 kg.m^2
Inertia of Shank (J_b)	0.0359 kg.m^2
Control Parameters	
Flexion Phase	
K_{flex}	74.50 N.m/rad
d_{flex}	18.5 N.m.s/rad
$\bar{\beta}_{\text{flex}}$	1.06 rad
Extension Phase	
K_{ext}	65.34 N.m/rad
d_{ext}	19.5 N.m.s/rad
$\bar{\beta}_{\text{ext}}$	0.0728 rad

is a high degree of coupling present between TE_1 and TE_2 . Any small error in calculation of $\ddot{\alpha}$ in TE_1 manifests itself in TE_2 leading to an error in $\ddot{\beta}$ as well. The converse of this is also true, with an error in $\ddot{\beta}$ leading to an error in $\ddot{\alpha}$. Thus, the error cascades into a very large value in short amount of time, and it becomes tough to isolate the cause of the error. It can also give misleading indications about the degree of inaccuracy of the parameter estimates.

To overcome this problem, and to be able to gauge the accuracy of estimates, the decoupled state equations are solved instead. In order to visualize the matching of α , only TE_1 is solved over time using α and its derivatives as the state variables. The measured values of β and its derivatives are used in this equation. The τ_{hip} values required to satisfy this equation are also calculated and plotted.

Similarly, to visualize the matching of β , only TE_2 is solved over time using β and its derivatives as the state variables. The measured values of α and its derivatives are used in this equation. The τ_{knee} values required to satisfy this equation are also calculated and plotted. Estimates of τ_{hip} and τ_{knee} can not be compared to their actual values since no measurements are available for them. Nevertheless, these estimates will prove useful in order to compare the performance of the controller in the next phase of the research.

Flexion Phase

The method explained above was used for visualizing the accuracy of the estimated parameters for the flexion phase and the results are presented in Fig. 3.1. It should be noted that the scales used for plotting α and β are different. Similarly, the scales for τ_{hip} and τ_{knee} are different. This is done because forcing the quantities to be plotted on the same scale would not allow their variation to be visualized properly. As can be observed from the plots, the values of α and β are matched with very high accuracy and providing confidence about the accuracy of these estimates.

Extension Phase

The accuracy of estimated parameters for extension phase was visualized similarly, and has been presented in Fig. 3.2. It can be observed that the matching of parameters is not as good as it was in the flexion phase. However, this is to be expected since the dynamics of extension phase are much more complex as compared to the flexion phase, and the matching of joint angles can be considered reasonable.

These results leave some room for improvement, though. One of the limitations is the

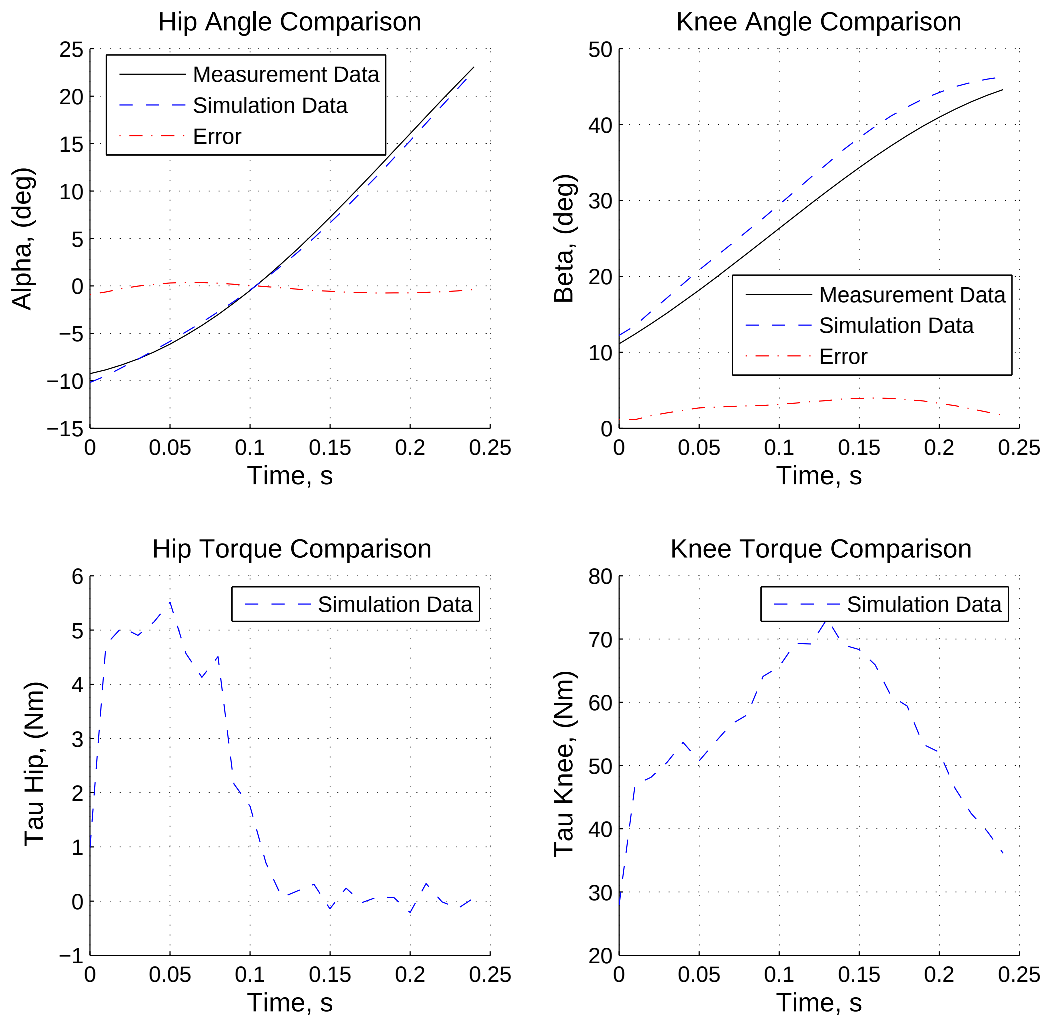

Figure 3.1 Simulation of *flexion phase* with estimated parameters

spike in values of both τ_{hip} and τ_{knee} at the start of the cycle. This can be understood by observing the rate of change of β . Simulated value of β starts decreasing at a greater rate as compared to the measured value. Since the control law for τ_{knee} is linearly dependent on $\dot{\beta}$, this results in a greater value of τ_{knee} . This in turn leads to an increase in τ_{hip} , in order to compensate for the error produced in TE_1 .

Another possible reason of concern is the higher value of τ_{hip} than that observed in healthy individuals. Although this may seem to be a limitation of this result, it is in fact a logical outcome. Studies [LF08] have shown that there exists a strong interplay between the hip torque and ankle push-off. A higher ankle push-off at the start of the swing phase provides greater momentum to the leg, which carries it through the swing phase and thus requires a reduced effort at the hip. Whenever there is a reduction in ankle push-off due to physical injury, some medical condition, or with progression of age, humans automatically compensate for it by increasing the hip torque. Since the prototype used for data collection did not have an actuated ankle joint, there was no ankle push-off available. Thus, there is no initial momentum available and the hip joint has to perform greater amount of work to *pull* the leg through.

Future studies that measure the applied hip torque can help to test the validity of this hypothesis. Performing data collection using a prosthesis with an actuated ankle joint can also help in this regard.

Since the goal of the prosthesis was to quick prototyping, the sensors and actuators used may not be comparable to those found in commercial devices. The actuator used might have unmodeled nonlinearities, and it might not be reasonable to expect it to exactly produce the desired output. This can be another mitigating factor to consider.

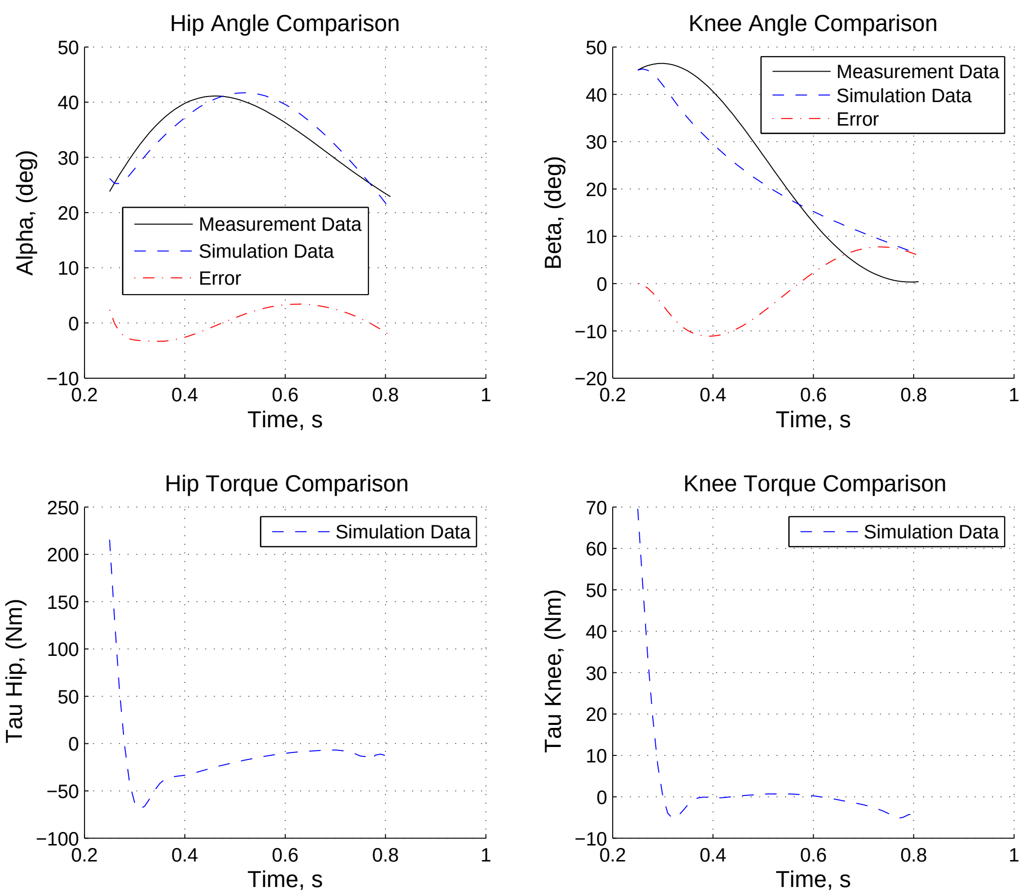

Figure 3.2 Simulation of *extension phase* with estimated parameters

Considering all these factors, reasonable confidence can be expressed in the estimated parameters and the design of the Model Predictive Controller can now be tackled.

CHAPTER

4

PHASE 2 - CONTROL DESIGN

Now that the parameters for the prosthesis have been estimated to a reasonable degree of accuracy, the next step is to design the controller. Some work has been done previously by various researchers with regard to designing a control scheme for Active Transfmeoral Prosthesis. However, most of the proposed schemes only consider the dynamic effects of the knee and ankle. This chapter will examine the motivations for an advanced control scheme that incorporates the effect of the hip in greater detail. An overview of the control design is then presented, along with a brief introduction to *Model Predictive Control (MPC)*. The control design problem is then formulated followed by the design itself and a discussion of the primary criteria behind the choices of design parameters.

4.1 Motivations and Objectives

As discussed previously, most of the currently proposed control schemes for ATP rely on a knee impedance control approach. Some advanced approaches also factor in the effect of ankle, while still relying on impedance control for controlling the torque input at the ankle. While the effects of ankle on human gait are important, so are the effects of the hip and hip torque. Not accounting for τ_{hip} and its effect can result in the amputee working against the prosthetic leg, instead of the two working in harmony. This means an increase in the metabolic cost of walking for the amputee. This is certainly not a desirable outcome, considering that the comfort of amputee is a primary objective that the prosthesis needs to accomplish.

The prosthetic leg used for the data collection for this research did not have an actuated ankle joint. Thus, the ankle effects are not accounted for in this research. However, it still serves to demonstrate how this approach can lead to a reduction in τ_{hip} compared to a knee impedance control approach. And as will be explained later, extending this research to incorporate the ankle effects should only improve the results further.

In addition to this, a more severe constraint is the inability of the impedance control approach to adapt to different gaits or even disturbances in the normal gait. Many of the current approaches only focus on level ground walking. And for any given gait trajectory, the *control parameters* need to be tuned carefully in order to produce the desired result. Thus, it is impossible to directly port a given controller over to another walking scenario. This is because the torque profile generated by the control law is only applicable for a specific gait. For the same reason, an impedance control technique can not help an amputee recover from any major disturbance or an uncertain scenario.

MPC, however, does not depend on a predefined torque input profile for tracking any desired trajectory. It uses the mathematical model of the system to calculate the torque input required to track the desired gait trajectory. This makes it inherently more adaptable to different gait scenarios, and more capable of tracking desired trajectory in the face of uncertainty and disturbance.

4.2 Guiding Principle

The gait model of the system has two quantities that are external to it and are not a function of any other system parameters. These external inputs are τ_{knee} and τ_{hip} . While τ_{knee} is directly determined by the controller, τ_{hip} is delivered by the amputee.

There have been some studies to analyze the hip torque profile during some routine walking activities. However, obtaining this data for every amputee can be very difficult and time-consuming. Also, as mentioned in section 4.1 in the context of τ_{knee} , a controller based on tracking reference input profiles can be limited in scope and not adaptable to different scenarios. τ_{hip} measurements were certainly not available in the dataset used for this research. Neither is it feasible to measure τ_{hip} for an amputee in real-time during normal operation. So the controller can not calculate the required τ_{knee} that would satisfy the kinematic constraints at every time instant.

This research hypothesizes that this problem can be solved by optimizing the values of τ_{hip} and τ_{knee} such that the energy consumption is minimized for tracking any given reference trajectory. The weights of the individual control inputs can be further tuned to place greater emphasis on improving the comfort of the amputee, or to minimize the energy requirements of the prosthesis. Studies like [LF08] have demonstrated that humans can automatically adapt their hip torque to react to changes in other joint torques, with minimum to no training. In fact, most humans adapt their joint torques during their lifetime. This can happen either due to temporary impairment of physical abilities from injury. Or due to some medical conditions like diabetes or with the progression of age, as demonstrated by [LF08].

The details of this hypothesis will become clearer in the following sections.

4.3 Problem Formulation

The problem of controlling an ATP can be considered as a two-stage process and can be solved using a hierarchical control architecture. The higher-level controller generates the reference trajectory depending on the gait mode or the activity being performed by the amputee. These trajectories can be generated from measured data for various standard activities or gait modes, and the controller can choose the appropriate trajectory depending on sensor measurements [Liu14], and by detecting amputee intent based on EMG signals [Har13]. The lower-level controller determines the control input required to track the desired reference trajectory. This research project mainly deals with the lower-level controller design based on MPC.

4.3.1 Reference Trajectory

The level-ground walking scenario has been used for testing the controller implementation. The design approach presented can be extended to other similar gait trajectories without much additional effort.

Some researchers have proposed using a phase-parameterized trajectory for reference in the context of an Ankle-Knee Model. It was proposed to use the knee angle as a function of the ankle angle, as a reference trajectory. However, in the present scenario of a Thigh-Knee-Shank (TKS) Model, this approach proves to be infeasible. There is a high degree of coupling between the various cycle parameters in TE_1 and TE_2 . As a result, it is not possible to parameterize one cycle parameter solely in terms of another cycle parameter and eliminate their dependence on the other cycle parameters.

Thus, the time-parameterized trajectories of the state variables(α , $\dot{\alpha}$, β , $\dot{\beta}$) are treated as the reference trajectory. For the purpose of this research, the data collected for the subject over multiple gait cycles was averaged since it reduces the effect of any measurement errors. This averaged data, depicted in Fig. 4.1, was then used as the reference trajectory.

4.3.2 Manipulated Variables

Manipulated Variables refer to the set of control variables which the MPC algorithm optimizes online. τ_{knee} is directly applied by the MPC based controller, and is included in this set. And even though τ_{hip} can not be directly influenced by the controller, as explained in section 4.2, it needs to be included in the online optimization problem. Hence, τ_{hip} is also included in this set of manipulated variables.

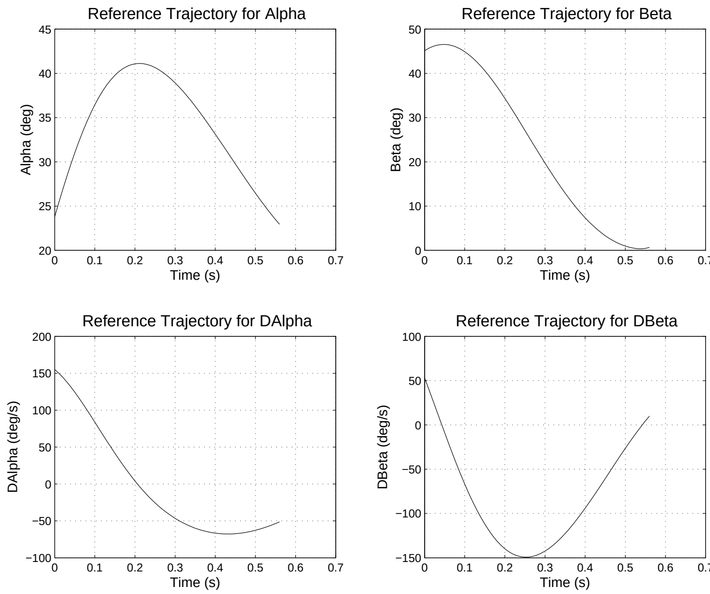

Figure 4.1 Reference Trajectory for Model Predictive Controller

4.3.3 Output Variables

Output Variables refers to the system state variables which the controller needs to track. Although it would be ideal to be able to track (α , β) as well as their derivatives ($\dot{\alpha}$, $\dot{\beta}$), it is not possible to eliminate the steady-state error in all four of them with just two control inputs. Also, requiring the controller to perfectly track the joint angles as well as angular velocities can have the effect of excessively constraining the controller and reducing its ability to recover from uncertain and emergent scenarios.

Moreover, the angular velocities of the joints are not measured directly from the physical system. They are calculated from the angles using difference method. As a result, even small errors in measurement of joint angles can magnify the error in the value of their derivatives.

Hence, less degree of confidence can be expressed regarding the values of ($\dot{\alpha}$, $\dot{\beta}$).

For all the aforementioned reasons, the controller will only be required to track α and β as the output variables. This can be achieved in two different ways while implementing the controller. The most obvious and direct way is to only return α and β as the outputs from the plant model. Another way is to include $\dot{\alpha}$ and $\dot{\beta}$ in the outputs returned by the plant model, but placing zero, or minimal weight, on tracking these parameters while formulating the online optimization problem. The strategies are equivalent to each other in terms of the controller performance and they only vary in terms of the convenience one offers over the other while implementing the control logic. However, the second strategy can make it convenient to visualize the performance of $\dot{\alpha}$ and $\dot{\beta}$, just to make sure that their behavior is not completely unacceptable for the physical system. Having the values of $\dot{\alpha}$ and $\dot{\beta}$ easily available can also be helpful while switching between multiple controllers, as will be explained later.

4.3.4 Plant Model

In order to design a MP Controller, a plant model needs to be specified. To this end, the nonlinear plant model needs to be linearized before it can be used for designing the MPC algorithm. Traditionally, MPC has been used for controlling processes that evolve slowly over time. As a result, the system model can be linearized at its steady state value and the results obtained as a result are reasonable.

This presents a significant challenge for the nonlinear gait model, though. The system evolves rapidly with time. Moreover, the state equations representing the system are highly nonlinear and there is a high degree of coupling between the various state variables. As a result, it is not feasible to linearize the system at a single operating point over the entire gait cycle.

This problem can be solved by linearizing the prosthesis model at multiple operating points over the gait cycle and designing an MP Controller at each of these operating points. During runtime, a switching logic is used to determine the appropriate controller that would provide the best results for the current operating point. This logic then selects between the various controllers as the system evolves with time. Using an adequately high number of linearization points provides an approximation of the system that is close enough to the real nonlinear system. As the results will demonstrate later, choosing a sufficient number of linearization points proves to be a critical factor affecting the performance of the control algorithm.

4.3.5 Switching Logic

Another factor crucial to the performance of the control algorithm is choosing the correct controller in runtime, since the system dynamics vary greatly depending on its current operating region. Choosing an appropriate controller at any given time involves determining which controller linearization point represents the closest approximation of the current operating point. This can be done in two different ways which are now presented.

Switching Logic 1 (SW_1)

The main philosophy behind this approach is to require the system to track the point on the reference trajectory corresponding to the current time step. Following steps need to be followed for this logic -

- The operating range is determined within which β for the current time step lies.
- The controller corresponding to this operating range is selected.
- The reference trajectory for the future prediction horizon starts at the current time instant and uses the measured values of the subsequent time steps.
- The simulation is stopped when the final time step of measured data is reached.

Switching Logic 2 (SW_2)

The main philosophy behind this approach is to require the system to follow the general shape of the reference trajectory and not necessarily track the value corresponding to the current time step. This approach is especially popular in path-planning of robotics, where the robot is only required to follow the space-parameterized trajectory, and not the time-parameterized trajectory. This logic has the effect of relaxing the constraints on the system to an extent and allowing it to track more extreme trajectories which would not have been possible otherwise. Following steps need to be followed for this logic -

- The sum of the squared distance of the current values of (α , β) is determined from those at all the linearization points.
- The controller corresponding to the linearization point with the minimum distance is chosen.
- The reference trajectory for the future prediction horizon starts at the point on the curve which is closest to the current operating point and uses the measured values of the subsequent time steps.
- The simulation is stopped when the system reaches a point that is within a specified threshold of the final measured data.

4.3.5.1 Resolving Conflicts

There can be cases, however, when either of the two switching logics returns multiple controller regions within which the current operating point possibly lies. The following sequence of checks will be applied in such cases -

- The sign of $\dot{\beta}$ for each of the operating regions corresponding to the possible controllers, is compared against the current sign of $\dot{\beta}$. This check should help to eliminate some of the possible choices of candidate controllers.
- However, if there still are multiple controllers which could be applicable, the controller representing the smallest shift from the last used controller will be used.

4.4 Design Implementation

This section presents the implementation of the control algorithm discussed in the previous section. There are a couple of software toolboxes available that automate the underlying tasks involved in controller implementation and solve the online optimization problem to return the control input. MATLAB provides a great software package in the form of MPC toolbox [Mat16e], which was used in this research. In addition, Simulink [Mat16k] was used for simulating the performance of the prosthesis system.

4.4.1 Basics of MPC Theory

MPC is based on performing a finite-horizon optimization of a plant model at every time step. The plant is sampled at every time step t and a control strategy is computed for the *Control Horizon* $[t, t+N_C]$ that minimizes the cost of tracking the reference trajectory over the *Prediction Horizon* $[t, t+N_P]$. The control input is held constant after N_c control moves for the remaining prediction horizon. An internal mathematical model of the system is used for predicting the evolution of the system states. And a numerical minimization algorithm is used for minimizing the control cost. However, at every time step t, only the first control input from the calculated sequence is applied. The whole process is then repeated at the next time step. The Control and Prediction Horizons keep shifting forward at every time step. This process has been illustrated in Fig. 4.2.

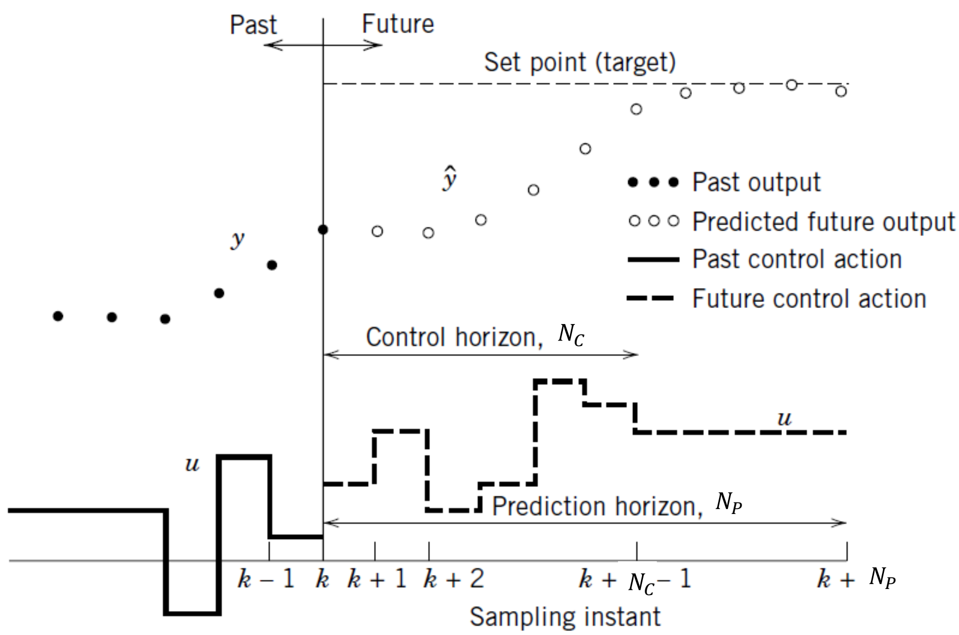

Figure 4.2 Basic concept of Model Predictive Control. Adapted from [Seb11]

A more detailed description of MPC can be found in [Mat16g] or [Seb11]. Various aspects affecting the controller implementation and the considerations behind choosing various parameters are now presented.

4.4.2 Sampling Time

Having a sufficiently low sampling time is crucial to the controller performance for a rapidly evolving system such as this. Simulations have shown that good performance can be obtained for a sampling time of approximately 1 ms - 5 ms. Anything greater than this seriously deteriorates the controller performance.

4.4.3 Prediction Horizon (N_p)

Prediction Horizon refers to the number of time steps, starting from the current time step, over which the controller predicts the evolution of the system states. A large enough prediction horizon ensures that the controller takes into account how the applied control input will affect the system states in the future, and can prevent excessive control input that could prove to be destabilizing. At the same time, an extremely large prediction horizon can add a large computational burden. This can be especially crucial for mobile systems where computational power may be limited.

Since MPC has typically been used for systems that evolve slowly over time, most of the current design guidance regarding choosing a prediction horizon indicates that a value in the region of 20-50 should be sufficient. However, this value proves to be grossly insufficient. A value in the region of 180-200 for N_p proves to be adequate.

4.4.4 Control Horizon (N_c)

Control Horizon refers to the number of time steps, starting from the current time step, over which the controller calculates a sequence of control inputs in order to find the optimal solution. Since MPC predicts the evolution of the system over a limited window, having N_c close to N_p can make the system marginally stable. A situation can arise where the system is stable over this short prediction horizon, but becomes unstable soon after. To avoid this, sufficient gap needs to be maintained between N_p and N_c . This allows the controller to evaluate the impact of the control input sequence for sufficiently long after the input has been removed. Best results are obtained when N_c is approximately 10-20.

4.4.5 Online Optimization Cost

The objective of the online optimization problem is to find a control input sequence such that the tracking error for output variables is minimized. In addition, the optimization function also seeks to keep the cost of control reasonable. These two objectives are achieved by calculating the weighted squared sum of the deviations of output variables over the entire prediction and adding it to the weighted squared sum of the control input. Eq. 4.1 provides a simple representation of the optimization cost function. A detailed mathematical expression can be found here [Mat16f].

$$J = \sum_{i=1}^{N_P} (r_i - x_i)^2 \cdot w(x)^T + \sum_{j=1}^{N_C} \left((u_j)^2 \cdot w(u)^T + (\Delta u_j)^2 \cdot w(\Delta u)^T \right) \tag{4.1}$$

$$
\begin{aligned}
\text{where, } x_i &= \text{output variable vector for } i^{\text{th}} \text{ step in prediction horizon}\\
r_i &= \text{reference variable vector for } i^{\text{th}} \text{ step in prediction horizon}\\
u_j &= \text{manipulated variable vector for } j^{\text{th}} \text{ step in control horizon}\\
w(x) &= \text{vector containing weights for output variables}\\
w(u) &= \text{vector containing weights for manipulated variable}\\
w(\Delta u) &= \text{vector containing weights for changes in manipulated variable}
\end{aligned}
$$

So the problem of choosing an appropriate cost function reduces to choosing an appropriate set of weights for all the output and control variables.

Optimization Weights

The relative magnitude of these weights represents the relative importance of tracking these individual variables, with a higher weight representing a higher priority. The exact parameters need to be determined by tuning the weights to achieve the specific performance objectives. Nevertheless, some guidance regarding the tuning process can come in handy.

- **For Output Variables** ($w(\alpha)$, $w(\beta)$, $w(\dot{\alpha})$, $w(\dot{\beta})$) -

  Tracking β is the sole responsibility of the controller. Whereas, the amputee has some degree of control over α because of the presence of τ_{hip} which, as explained before, is adaptable. Also, a small tracking error in α would result in a change in the stride length of the amputee. Whereas a tracking error in β could result in the amputee stumbling due to the toes hitting the ground in swing. It could also result in the heel striking the ground at an incorrect angle at the end of gait, thus causing discomfort for the amputee.

  Thus, it can be argued that greater emphasis should be placed on tracking β as compared to α . At the same time, a high tracking error in α would manifest itself in the tracking performance of β as well due to the high degree coupling between the two terms. Thus, $w(\alpha)$and $w(\beta)$need to have a similar order of magnitude. Simulations have shown that best results are obtained when,

$$w(\beta) \in [w(\alpha), 1.5w(\alpha)] \tag{4.2}$$

As explained in subsection 4.3.3, it is not feasible to track $\dot{\alpha}$ and $\dot{\beta}$ in addition to α and β . Thus,

$$w(\dot{\alpha}) = w(\dot{\beta}) = 0 \tag{4.3}$$

- **Terminal Weights** -

  Terminal Weights refers to the weights placed on the final value of output variables. Achieving the desired final value can make it easier for the system to transition to the next phase of operation. And even if achieving a specific value at the end of gait cycle is not very important, placing high terminal weights can have a stabilizing effect on the system. *Terminal Weight Ratio* (w_{term}) is the multiplying factor of ($w(\alpha)$, $w(\beta)$) that returns the terminal weight values to be used. Best results are obtained when $w_{term} \sim 5-10$.

- **For Manipulated Variables** ($w(\tau_{knee})$, $w(\tau_{hip})$) -

  It might seem intuitively that much lesser emphasis should be placed on minimizing the control cost as compared to tracking the output variables accurately. However, if the corresponding weights are tuned based on this principle, the controller tries to reach reference trajectory very quickly, without regard to the control effort that would be required to get there. This sudden change in system states may also lead to instability. Simulations have shown that best results are obtained when,

$$(w(\tau_{knee}),\ w(\tau_{hip})) \sim 5 \times (w(\alpha), w(\beta)) \tag{4.4}$$

An equal magnitude for $w(\tau_{knee})$and $w(\tau_{hip})$would represent a good starting point, since it has also been observed to return good tracking performance. If the goal is to reduce amputee effort, $w(\tau_{hip})$needs to be increased relative to $w(\tau_{knee})$. However, if the goal is to reduce the power requirements of the prosthetic leg, $w(\tau_{knee})$needs to be increased.

- **For Manipulated Variables Rate** $(w(\Delta\tau_{knee}),\ w(\Delta\tau_{hip}))$ -
  In order to avoid instabilities in the system due to rapid change in control input, the rate of change of the manipulated variables also needs to be moderated. However, these rates should not be penalized excessively. The controller would otherwise be unable to respond to severe disturbances. Best results are obtained when,

$$(w(\Delta\tau_{knee}),\ w(\Delta\tau_{hip})) \sim 0.5 \times (w(\tau_{knee}),\ w(\tau_{hip})) \tag{4.5}$$

4.4.6 Number of Controllers

As mentioned previously, choosing an adequate number of controllers is very important to the performance of this control algorithm. In case of MPC, switching between controllers represents a change in the online optimization problem. And since the optimization problem solved at any time step is essentially independent of that solved at the previous time step, switching between controllers should not result in the transient behavior associated with switching of conventional feedback systems. Thus an increase in the number of controllers means that the linearization represents a much closer approximation of the actual system behavior. Hence higher the number of controllers, better will be the tracking performance of the controller.

However, excessive number of controllers also leads to a larger memory requirement for storing all the matrices related to the controller. This could be a limiting constraint in mobile systems with limited computing resources. Thus, the choice of number of controllers is directly dependent on the memory constraints. Approximately 8 controllers for the flexion phase and 14 controllers for the extension phase have proven to be adequate in the simulations.

4.4.7 Simulating Prosthesis System Output

It should be noted that even though the system model was linearized for the purpose of designing the controller, the full-order nonlinear system model is used for simulating its output. This ensures that the simulated output accurately captures the complete system dynamics and is close to the expected output of the actual system. S-Functions [Mat16l] provided by Simulink were used for implementing the nonlinear state equations of the prosthesis system. The Simulink model representing the control architecture is depicted in Fig. 4.3.

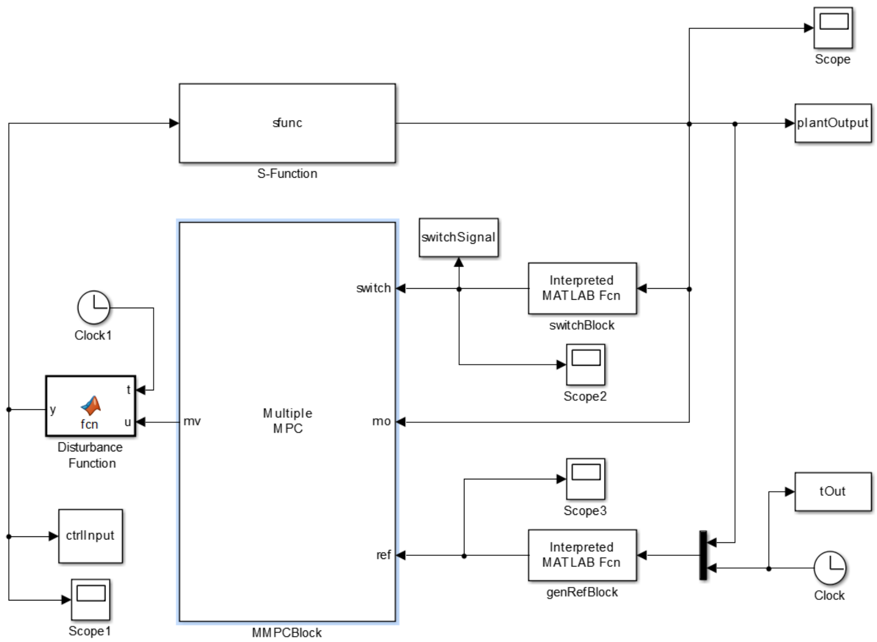

Figure 4.3 Simulink Model of the Control Architecture of Prosthesis System

4.5 Results

The previously explained control architecture was used for simulating the output of the system and studying the controller performance. The test cases were used to validate the design considerations presented in previous sections. These results will now be presented.

4.5.1 Baseline Scenario

The baseline scenario represents the level-ground walking scenario for which data has been collected. The most important parameters used while designing this control scheme have been listed in Table 4.1. The result of the simulation of baseline scenario with SW_1, depicted

Table 4.1 Parameters used for MPC design for baseline scenario

Parameter	**Value**
Horizons	
N_p	200
N_c	20
Optimization Weights	
Output Variables	
$w(\alpha)$	500
$w(\beta)$	500
$w(\dot{\alpha})$	0
$w(\dot{\beta})$	0
w_{term}	7.5
Manipulated Variables	
$w(\tau_{knee})$	2500
$w(\tau_{hip})$	2500
Manipulated Variables Rate	
$w(\Delta\tau_{knee})$	1000
$w(\Delta\tau_{hip})$	1000
Number of Controllers	22

in Fig. 4.4, shows very good matching of both the α and β curves, with the maximum error in both of them being less than 5° simultaneously. Moreover, the joint torques see a huge reduction as compared to the estimated values obtained for the impedance control approach. Both τ_{hip} and τ_{knee} see a significant improvement, which indicates that the hypothesis that MPC can better leverage hip dynamics is likely true.

It might be useful to visualize how the MPC algorithm switches between different controllers during runtime. For this purpose, the plot of switching signal with time is depicted in Fig. 4.5. The value of the switching signal indicates the index of the controller being used at any time instant. As expected, the switching logic gradually transitions from the first controller to the last as the system states evolve with time.

The prosthesis system was also simulated with $SW_2$and the result has been depicted in Fig. 4.6. However, SW_2does not require the system states to follow the time-parameterized trajectory. The prosthesis system might reach the end of cycle in lesser or greater amount of time. Thus, plotting the system states against time can only help to visualize if the system states qualitatively follow the general shape.

This calls for a different way of visualizing the performance of the system. This can be achieved by plotting the evolution system states with respect to each other (α vs β), instead of against time. This plot will be referred to as the *phase plot*. The error in matching this trajectory is calculated by finding the minimum distance between the simulated trajectory and the reference trajectory. This provides a much better way of quantitatively measuring the performance of the controller and prosthetic system. The *phase plot* for this simulation has been depicted in Fig. 4.7.

For the sake of comparison, the phase plot for the simulation with SW_1has also been

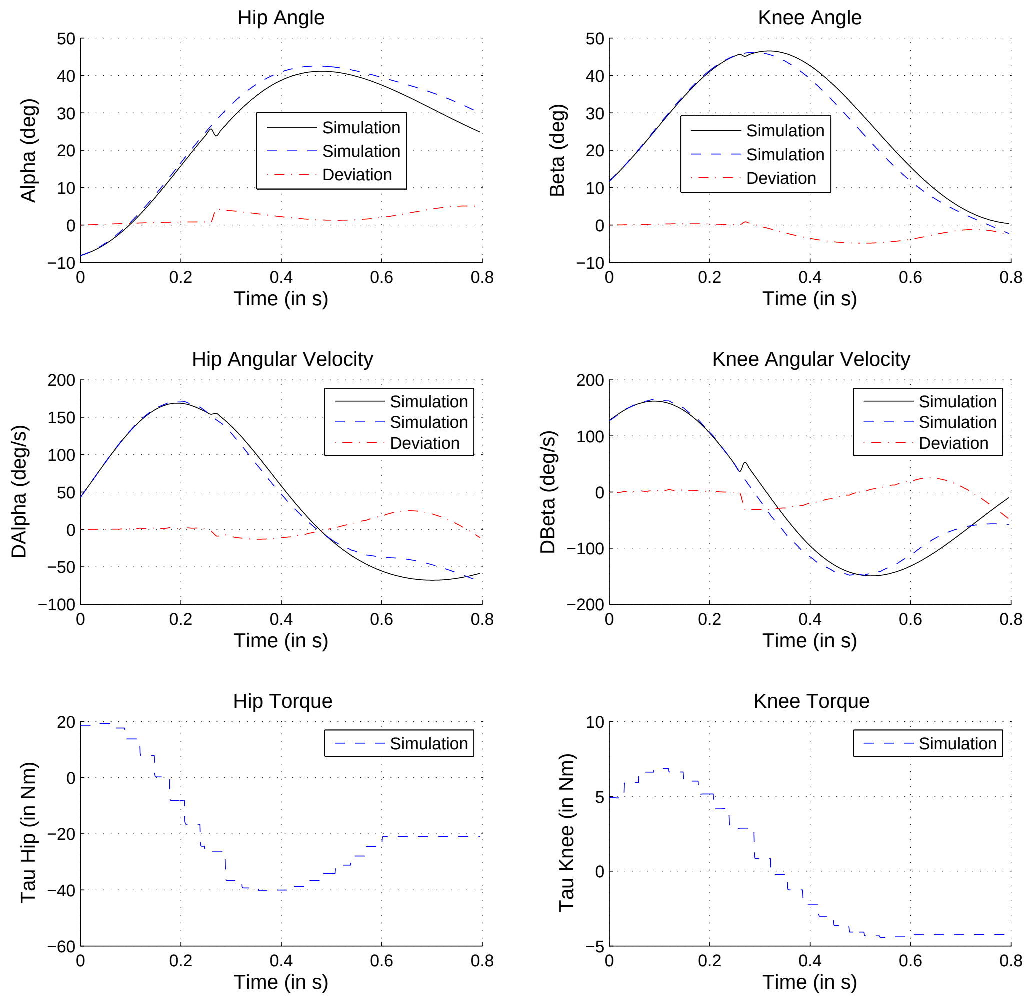

Figure 4.4 Simulation of Baseline Scenario with SW_1

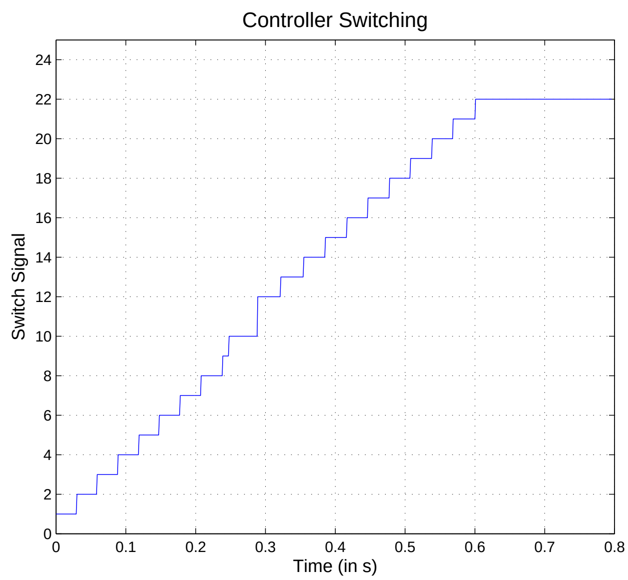

Figure 4.5 Switching Signal for Baseline Scenario with SW_1

depicted in Fig. 4.8.

Although the maximum deviation from the reference trajectory is comparable for both the scenarios, the difference in nature of responses is apparent from the plots. The response for SW_1 starts tracking the reference trajectory perfectly. However, midway through the cycle, after the switching of gait cycle to extension phase introduces a slightly abrupt change, the response begins to gradually deviate from the reference trajectory. While for this case, the simulation response is still very reasonable until the end of the cycle, it may not be the case when more severe disturbances are experienced.

The response for SW_2, on the other hand, starts deviating slightly from the reference trajectory at the start of the cycle. However, the error reduces again, even after the gait cycle has switched to the extension phase. This indicates that SW_2 might be a better candidate

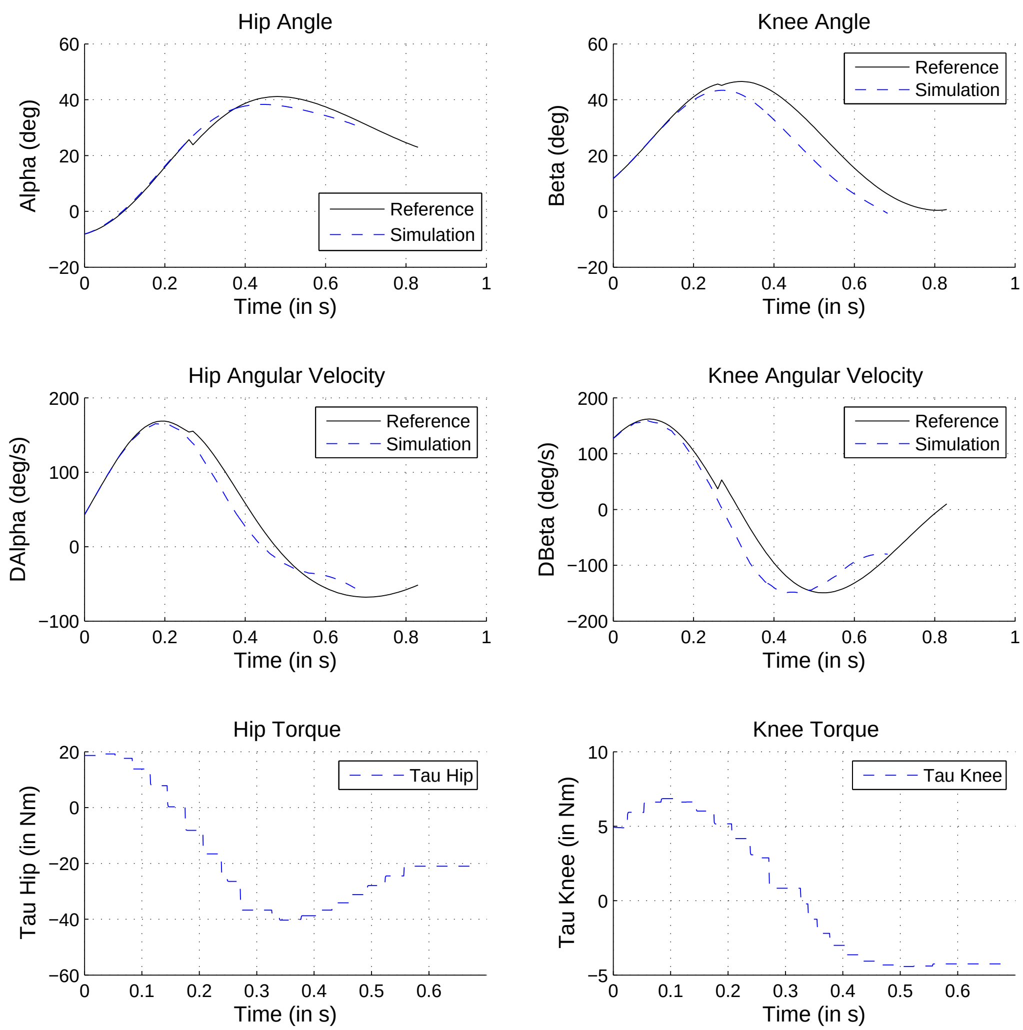

Figure 4.6 Simulation of Baseline Scenario with SW_2

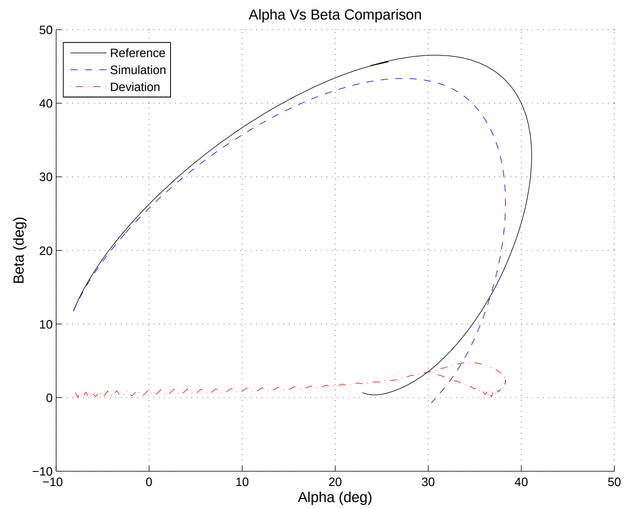

Figure 4.7 Phase plot for Baseline Scenario with SW_2

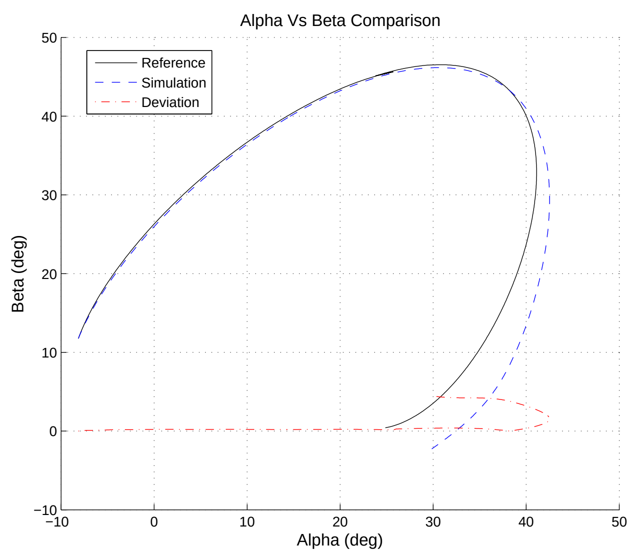

Figure 4.8 Phase plot for Baseline Scenario with SW_1

for handling scenarios where disturbance is experienced. It is also expected to be better suitable for adapting to different gait speeds from the user.

4.5.2 Disturbance Rejection

In a real-world scenario, the prosthesis system can experience a number of disturbances or uncertain scenarios. Since it is the goal of this research to design a controller that is adaptable to such scenarios, it is important to analyze the behavior of the controller in the presence of disturbance. This test was performed by applying 50 times the optimal value of τ_{hip} and τ_{knee} calculated by the controller at t = 0.2 s. Fig. 4.9 depicts the time response of the state variables for the $SW_1$and SW_2. A large spike can be seen in the values of τ_{hip} and τ_{knee}.

However, as explained previously, plotting the time response does not provide a fair visualization of the performance of SW_2. For this purpose the phase plot for both the cases has been depicted in Fig. 4.10. It should be noted that the scales for the trajectory plot and the deviation plot are not the same.

As expected, SW_2performs much better than SW_1. The deviation for SW_1keeps increasing, with a maximum value of $\sim 17°$. Whereas, the maximum deviation for SW_1reaches $\sim 10°$ momentarily before decreasing again. The deviation from the reference trajectory is only $\sim 5°$ at the end of the cycle. It might also be insightful to visualize the switching of controllers for both the switching logics, as depicted in Fig. 4.11.

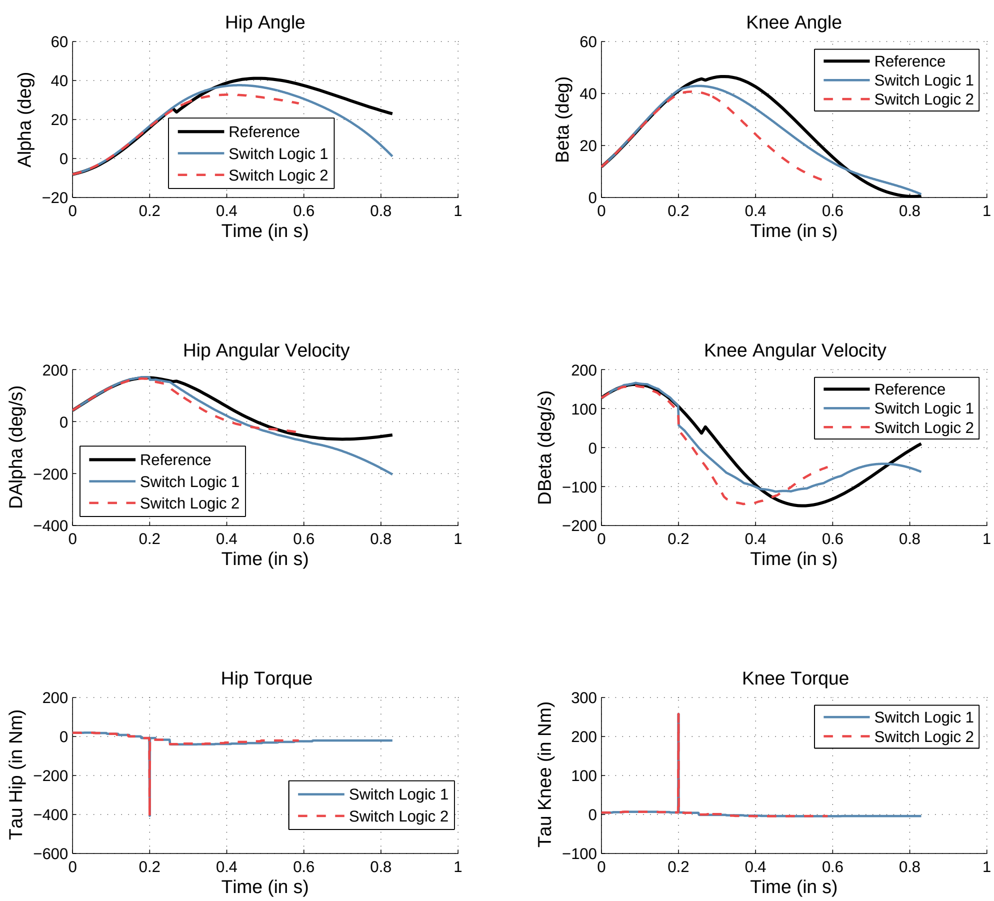

Figure 4.9 Time response plot when disturbance is applied at t = 0.2 s

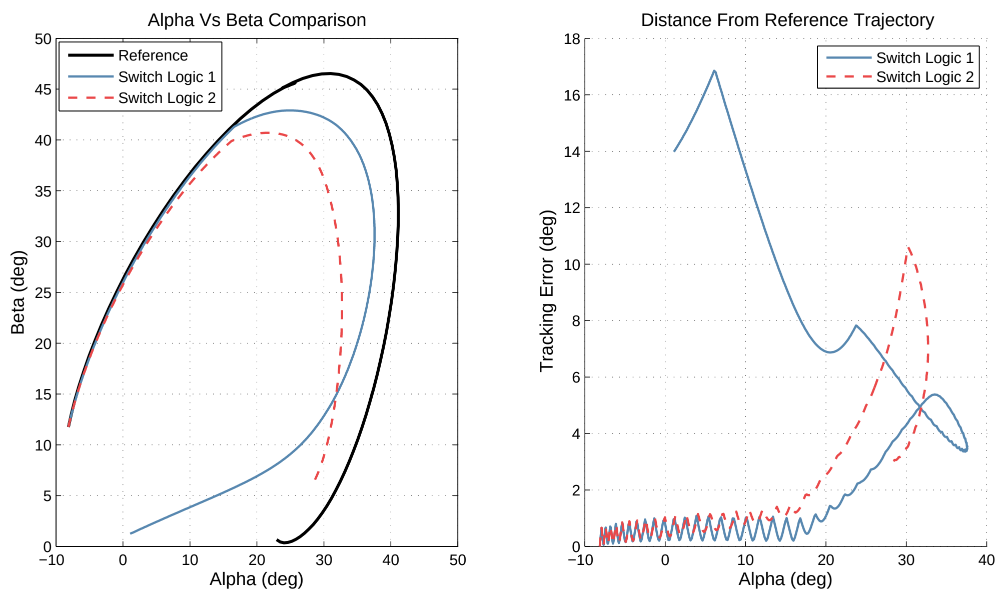

Figure 4.10 Phase plot when disturbance is applied at t = 0.2 s

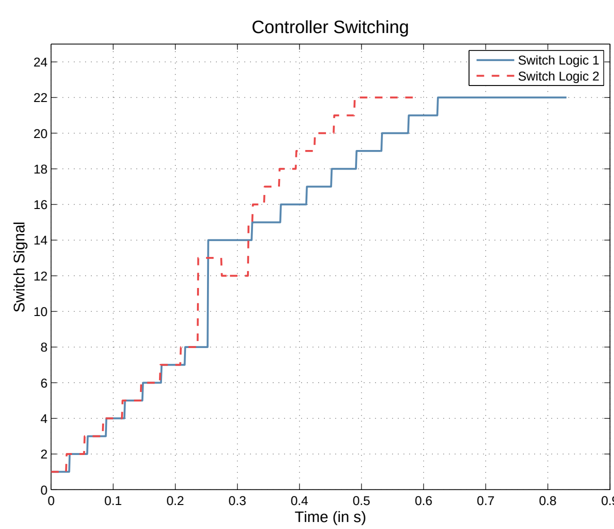

Figure 4.11 Comparison of Switching signal when disturbance is applied at t = 0.2 s

4.5.3 Variation in Number of Controllers

Fig. 4.12 and Fig. 4.13 depict the effect of variation of the number of controllers used. As explained in subsection 4.4.6, the performance of the controller starts degrading with a reduction in the number of controllers.

4.5.4 Variation in Prediction Horizon

Fig. 4.14 depicts the effect of variation of prediction horizon. The time response of the system remains qualitatively same and has not been depicted since it does not add any value in trying to understand the system response. As explained in subsection 4.4.3, the performance of the controller starts degrading with a reduction in the prediction horizon. Moreover, the number of feasible solutions for the optimization problem reduces, leading to an increase in computation time.

4.5.5 Variation in Relative Weights of Output Variables

Fig. 4.15 and Fig. 4.16 depict the effect of variation of the emphasis placed on tracking β relative to α . As explained in subsection 4.4.3, the performance of the controller is highly dependent on the emphasis placed on β . The system response becomes unstable when $w(\beta)$is reduced to a very low value. Whereas, the tracking accuracy starts to reach its best value as $w(\beta)$is increased.

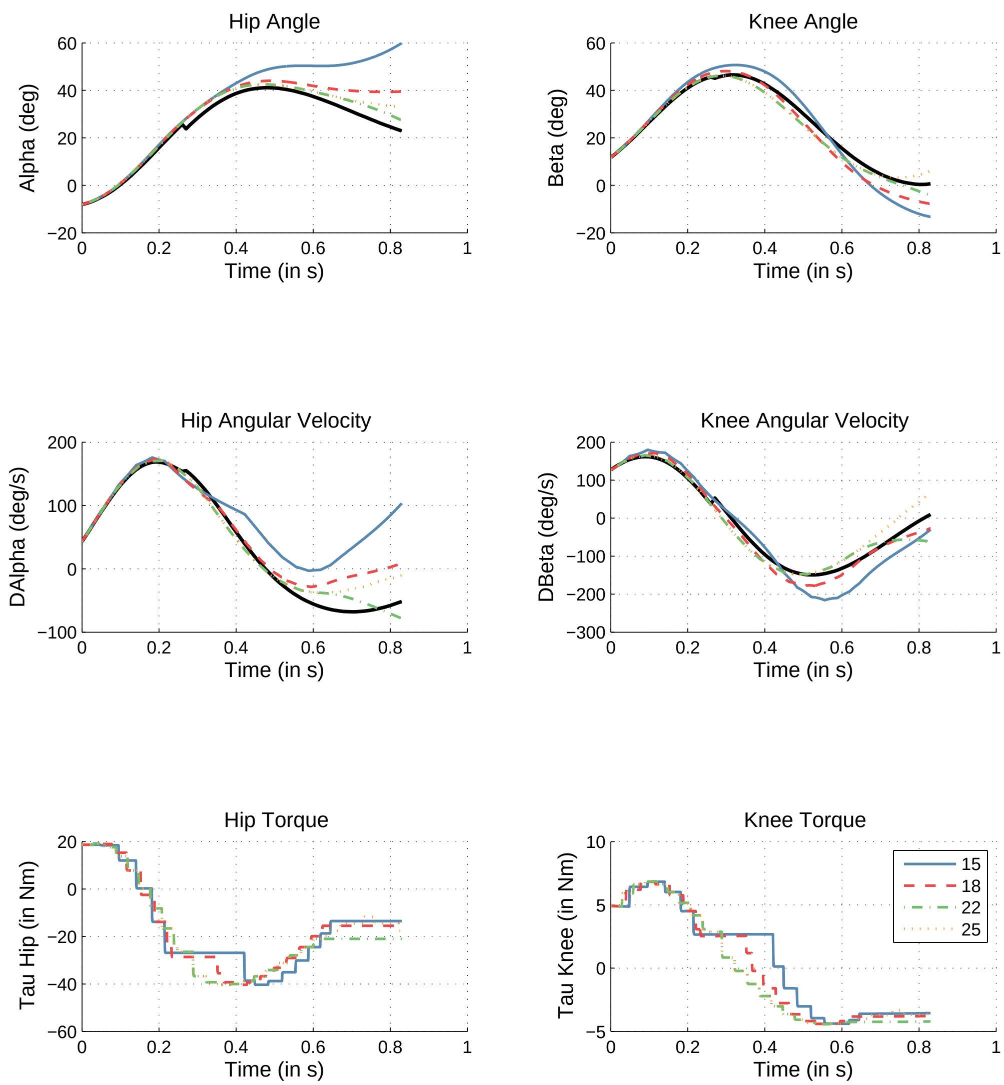

Figure 4.12 Time response plot for varying of number of controllers

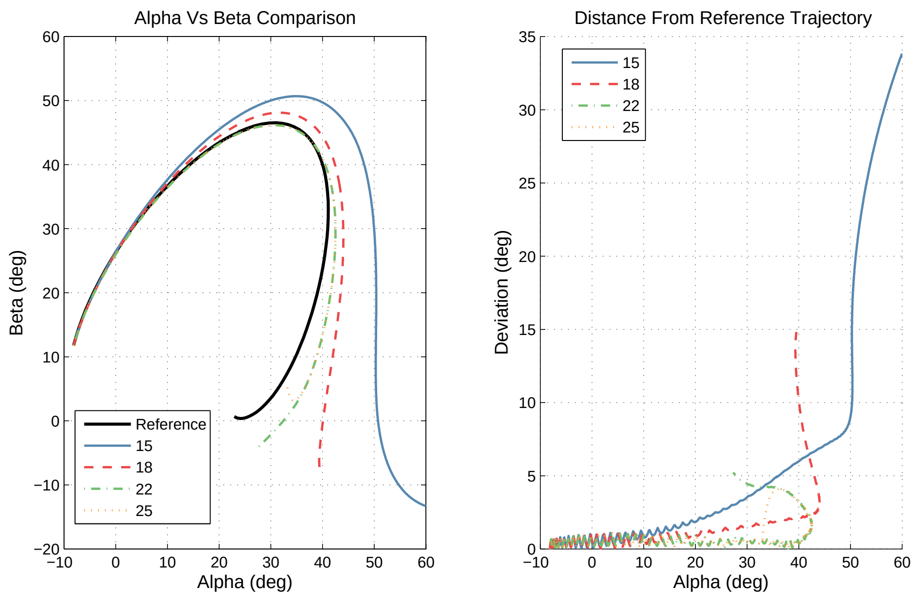

Figure 4.13 Phase plot for varying of number of controllers

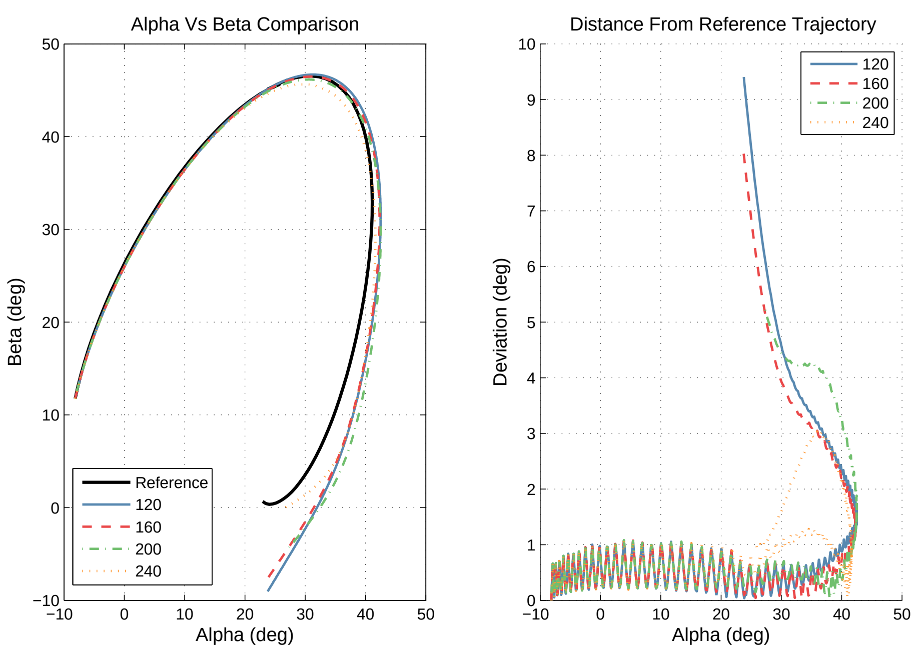

Figure 4.14 Phase plot for varying prediction horizon

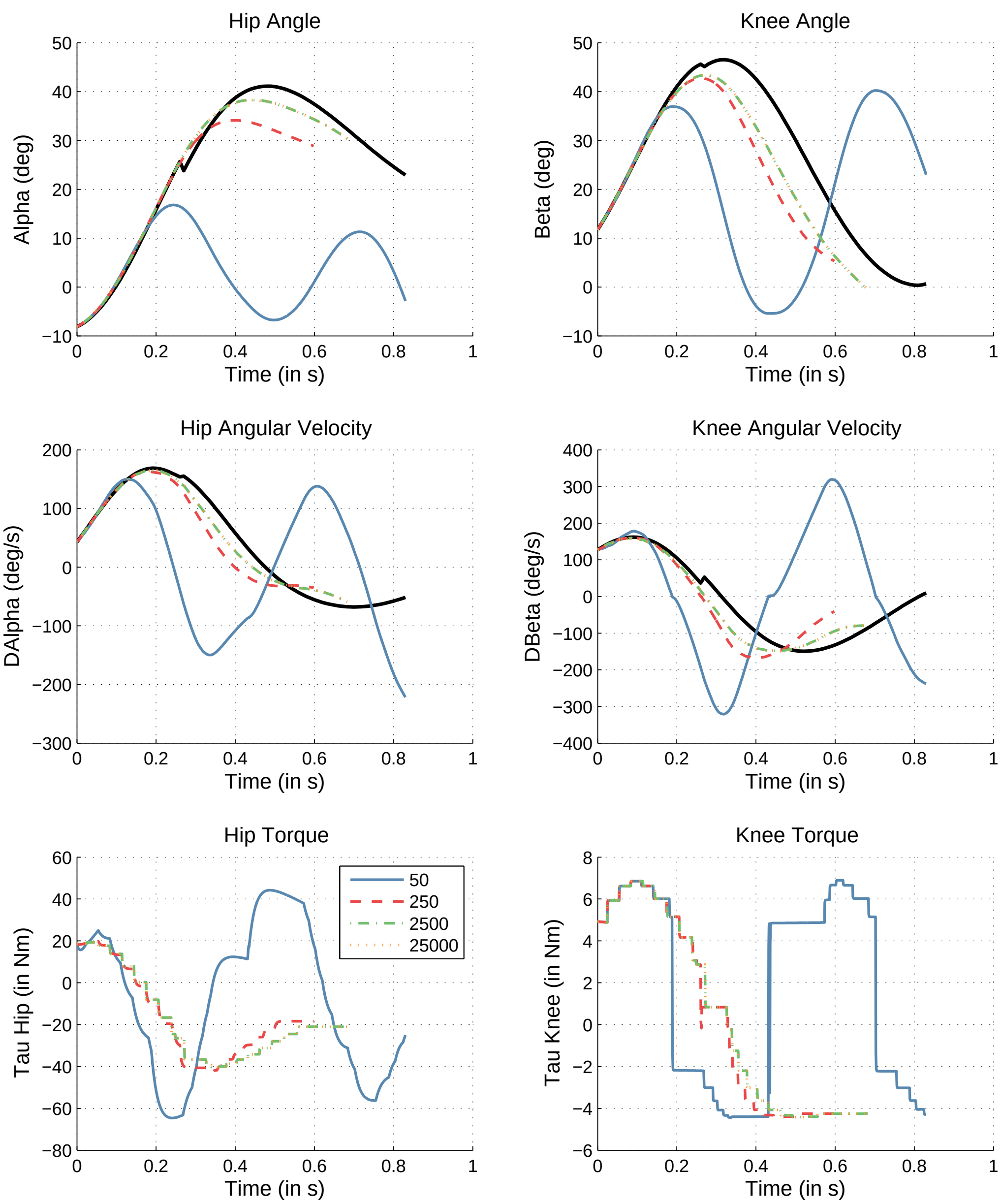

Figure 4.15 Time response plot for varying relative weight of β

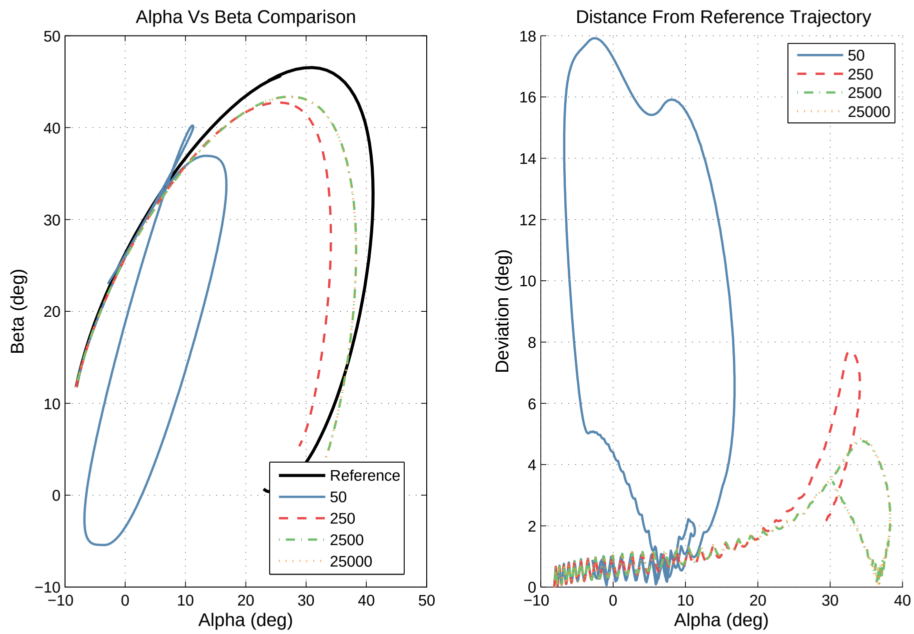

Figure 4.16 Phase plot for varying relative weight of β

4.5.6 Variation in Relative Weights of Manipulated Variables

Similar to the previous subsection, the performance of the controller is highly dependent on the relative emphasis placed on the two control inputs τ_{hip} and τ_{knee} . For very low relative weight placed on τ_{hip} $(w(\tau_{hip}))$, the system response tends to become unstable as a result of the excessive control action taken for small perturbations in system states. The system response improves with an increase in $w(\tau_{hip})$. However, excessively increasing $w(\tau_{hip})$does not result in a rapid decrease in τ_{hip} since the emphasis then shifts on accurate tracking of system states. The control input values thus tend to saturate near their nominal values for each of the linearization points. Fig. 4.17 depicts this effect adequately and thus the time response of the system has not been added.

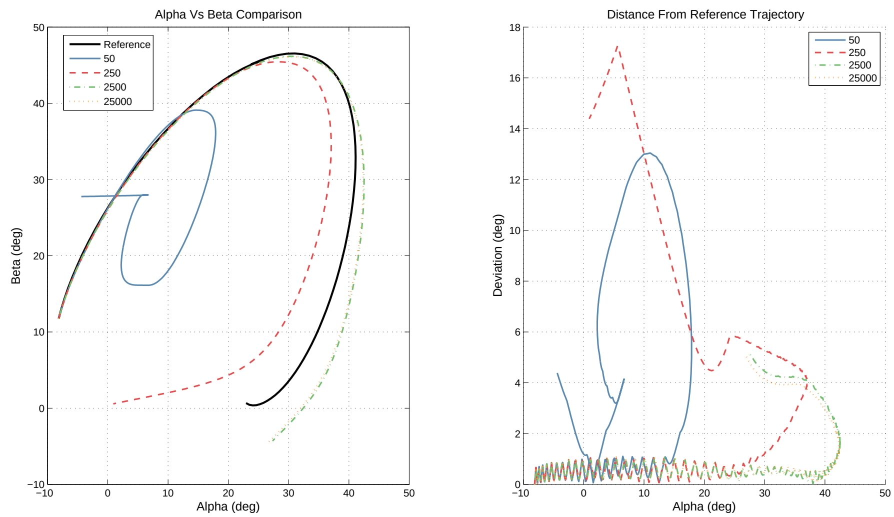

Figure 4.17 Phase plot for varying relative weight of τ_{hip}

4.5.7 Effect of Parameter Uncertainty

In a physical system, it may not be possible to determine all the required system parameters with perfect accuracy. This may lead to uncertainty in the model parameters, which can have a major effect on the controller performance. Thus it is necessary to study the effect of uncertainty in the parameter estimates for the prosthesis system model.

For this test, the controller was first designed using the best estimates for all the inertial parameters. Uncertainty of $\sim \pm 2.5\%$ was introduced in each of the system parameters. The prosthesis system was then simulated for various combinations of the system parameter values in these ranges. The output of this test is presented in Fig. 4.18.

It can observed from the plot that even with this uncertainty, the controller is able to match the values of β quite well. The matching of α starts to suffer after the halfway

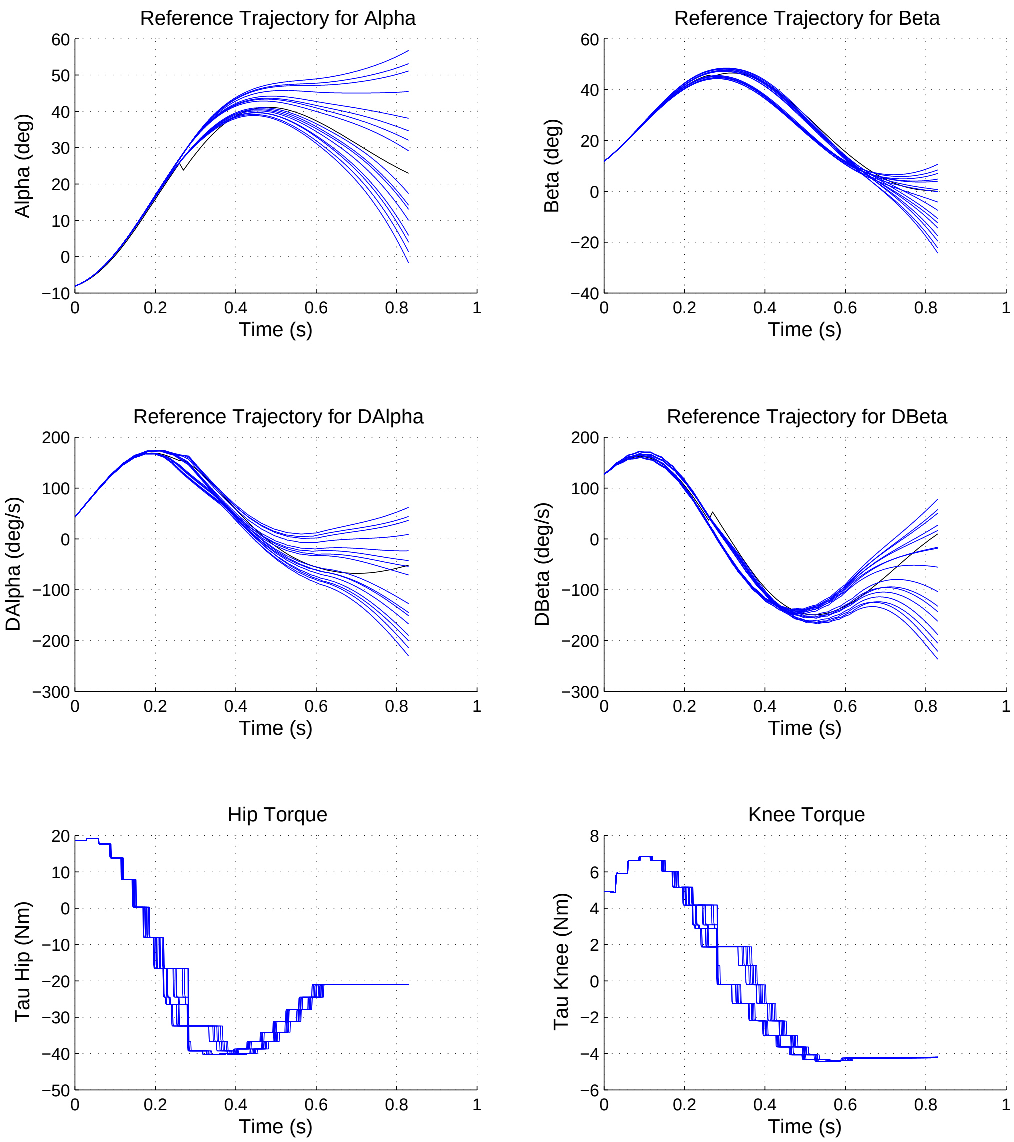

Figure 4.18 Effect of Uncertainty in System Parameters

stage, though. But this is to be expected, since the controller can directly manipulate the knee joint. Whereas, it has no direct control over the hip joint. The hip angle is largely determined by the user input, and enough knowledge is not available regarding the factors affecting it. Thus even though there is room for improving the robustness of the controller to uncertainty, the results obtained at this stage can be considered reasonable and logical.

4.5.8 Concluding Remarks

The goal of this research was to explore the viability of using MPC for motion control of ATP. The results have demonstrated that it is a promising option that needs to be explored in greater depths. The controller was able to track the reference trajectory to a greater accuracy compared to conventional control schemes and while requiring lesser effort from the user as well as prosthesis. The controller was even able to reasonably recover from an extreme disturbance.

Performing more rigorous tests on this controller and simulating it in more extreme scenarios are necessary though. And it is equally important to test this control scheme on a physical prototype. Improving the robustness of the controller would also help to bring this control scheme closer to real-world application.

And while MPC has some constraints in terms of the computational resources it needs, this constraint is reducing in severity with the advent of faster microprocessors and smarter optimization algorithms. Explicit-MPC can further help to reduce this problem, and needs to be explored in greater detail.

CHAPTER

5

CONCLUSION

5.1 Summary and Novel Contributions

The thesis began with the goal of examining a model-based approach for controlling an Active Transfemoral Prosthesis. An overview of the existing research revealed that most of the current control approaches focus on an impedance control approach, where the joint torques are modeled as a spring damper system. Since this approach provides a relatively easy solution to the control problem, lot of efforts have been focused on studying different ways to divide the gait cycle into phases, such that the joint torque is adequately characterized for level-ground walking.

However, this control technique has quite a few limitations of its own, which can result in a reduced quality of life for an amputee. Since the method relies on accurately tuned impedance values, lot of time and effort has to be spent by experienced professionals on tuning these parameters. Moreover, this tuning process has to be performed individually for each phase of the gait. This method also requires very high-accuracy sensors for correctly determining the current gait phase. All these factors lead to a very high cost for these devices and make it unavailable for lot of individuals who really need it.

Neither is this approach easily adaptable to different locomotion scenarios, since the torque requirements can vary greatly for different scenarios like walking backwards or climbing stairs. It would require individual tuning for all cases and complex algorithms to accurately detect the users intent. Also, since this approach does not take into account the hip dynamics, it often results in the user *reacting* to the prosthesis, instead of *interacting* with it. These factors leave a lot to be desired with respect to the goal of restoring natural mobility for amputees.

All these factors urge a model-based control approach. An accurate mathematical model of the system and a good estimate of the parameters inherent to this system are necessary though. It also requires gait data for estimating these system parameters and validating the model. The experimental setup used for collecting this gait data was first studied in the context of how it needs to be adapted for this research. The most important factors to note being the absence of an actuated ankle joint, and data collection being performed on a healthy individual wearing the prosthesis with the help of a bypass adapter. Overview of the process of deriving the Thigh-Knee-Shank Model of gait dynamics was also presented along with the final model used.

The first phase of the research dealt with deriving reasonable estimates of the inherent system parameters. Standard approaches used for nonlinear regression or system identification prove to be inadequate for this problem. Thus, an optimization approach was chosen since it provides the most amount of flexibility. However, it was observed that the optimization space for this problem is highly non-convex with the presence of nonlinear constraints making the problem very difficult to solve as a whole even with heuristic algorithms like *Genetic Algorithm*.

This leads to the need for evolving an elaborate optimization roadmap that has been specifically developed for this problem. The optimization problem was broken down into several smaller optimization problems with the estimate of system parameters being estimated in batches instead of trying to estimate all of them at once. Every step of this roadmap serves as an optimization problem in itself that serves to refine the estimates gradually. For each of these optimization sub-problems, the strengths of global and local solvers were combined using a hybrid optimization scheme. This hybrid scheme makes it possible to quickly search over large regions of the optimization space and identify promising regions with a global solver, and then narrowing down the estimates within these regions to the optimal value using a local solver.

Accurately matching the knee angle was treated as the primary objective of parameter estimation, since the prosthesis has direct control over it and there are no unmodeled effects unlike the hip angle. The factors affecting the hip torque were not entirely clear and measurements were not available for it either. This problem was tackled by trying to minimize the hip torque as much as possible instead of trying to match it to a reference profile. This is because the hip torque is known to be relatively low for healthy individuals because of its higher metabolic cost. However, this may not be the case for amputees especially when an actuated ankle joint is unavailable. Previous studies have shown that there exists a strong interplay between hip and ankle torques. Humans compensate for the reduction in the ankle torque due to injury or other medical conditions with an increase in hip torque. This was attributed as the primary reason for the higher hip torque values than is expected for healthy individuals.

The final results yielded reasonable estimates of the system parameters. Although the estimates obtained were better than those previously available, admittedly there is still room for improvement in the matching of the knee angle. However, this could be attributed to some extenuating circumstances. Since the prototype used for collecting the data was a prototype device and not of commercial quality, some inaccuracies in the sensors and actuators are to be expected. The actuator is likely to have some unmodeled nonlinearities and it may not be reasonable to expect perfect matching of the desired input. Also, the absence of any measurements for the hip torque turned out to be a limiting factor as well. Availability of a reference hip torque profile would certainly help to further improve the estimates in the future.

With reasonable estimates of the system parameters available, the thesis then focused on the phase of controller design. Model Predictive Control was chosen for controlling the ATP. Being a model-based control algorithm, MPC is inherently more capable of handling lot of the limitations discussed for more conventional control schemes. MPC solves an optimization problem at every time step in order to calculate the optimal control input that would result in the closest matching of the reference trajectory over the future horizon. And thus it is much more adaptable to different reference trajectories. It eliminates the need for extensive tuning that was previously required for every maneuver that the user might need to perform. And since the gait dynamics are essentially the same throughout the swing phase, it reduces the dependence on expensive sensors to accurately determine the gait phase. Both these factors can help to reduce the cost of prosthetic devices.

Also, since the hip effects are accounted for in the system model, the controller can leverage the hip dynamics to reduce the user effort, or the power requirement of the prosthesis, or both. This was achieved by penalizing the hip torque in the optimization cost function. The weight placed on the hip torque can be varied relative to the knee torque in order to achieve one of the stated dual objectives. This allows the user to interact with the prosthesis in a more natural way depending on their comfort.

The time-parameterized trajectories of the hip and joint angles and their derivatives were used as the reference trajectories for the system states. However emphasis was placed only on tracking the joint angles because only two system states can be accurately tracked with the limited number of control inputs. Moreover, lesser confidence can be expressed in the accuracy of measurements of the angular velocities. And although accurate tracking of the knee angle is more important, it was observed that placing excessive emphasis on it can have the opposite effect. This is because an error in the hip angle can quickly propagate into the knee angle due to the high degree of coupling between the two.

The nonlinear gait model derived previously was used for simulating the behavior of the system. But the control algorithm involved multiple MP Controllers designed by linearizing the system model at various operating values of knee angle. It was observed that the performance of the control scheme was highly dependent on choosing a sufficient number of controllers and designing a switching logic capable of accurately determining the controller to be used during runtime.

Two different switching logics were examined for this research. While the first switching logic requires the prosthesis to follow the time-parameterized trajectory, the second one only requires it to follow the space-parameterized trajectory. This relaxes the constraints to an extent and provides the ability to track more extreme trajectories than would otherwise be possible. And while the two logics had an equivalent performance for the base scenario of level ground walking, the second logic is more adaptable to varying speed by virtue of being independent of time.

In conclusion, very good trajectory matching was achieved for the level ground walking scenario with the maximum error being of the order of 5 deg. This result was indeed much better than what could be achieved with more conventional control schemes. Reduction in knee torque was also achieved with an additional leverage for adjusting it based on user comfort.

And while availability of sufficient computational resources is essential for MPC, this constraint is reducing in significance with the advent of faster and cheaper microprocessors at a very fast rate. Moreover, implementation of explicit MPC can provide significant reduction in computational requirements.

5.2 Future Work

The work done in this thesis provides some room for improvement and some parts need further validation. It also opens up the possibility for some interesting work in the future.

Some possible directions have been discussed here.

- **Measurement of Hip Torque** -

  Measurement of hip torque in conjunction with the the other data needed for this research would provide a better reference trajectory for comparing its estimated values. This would go a long way in improving the estimates of the system parameters. It would also help to test the validity of the hypothesis that hip torque would have a higher value for amputees in comparison to healthy individuals.

- **Prosthesis Prototype with actuated Ankle Joint** -

  Having a prototype with an actuated ankle joint would help to test the validity of the hypothesis that hip torque requirement should reduce in the presence of ankle joint. It would also help to further reduce the effort required by the amputee. Besides, it would help to validate the unified gait model once it has been developed.

- **Further development of Gait Model** -

  Although some researchers have previously incorporated the effects of ankle joints in gait models, some work is still desirable in order to combine the effects of knee and ankle in a single unified gait model. The interactions of the lower limbs with the upper body at the hip joint can be explored further in order to be able to account for the effects of hip position also in the gait model. This could also improve the description of gait dynamics in a wider variety of locomotion scenarios, thus making the controller more adaptable.

- **Data Collection for Greater Number of Locomotion Scenarios** -

  Collecting gait data for different scenarios such as varying walking speeds, walking on an incline, jogging, running etc can help to test the adaptability of the controller to these situations and tune the parameters for better performance.

- **Hardware Implementation and Rigorous Testing of Controller** -
  A natural way ahead for this research would be to implement the control algorithm on a prototype and test the performance. This would bring up new challenges which were not faced in the simulation phase. Also, doing more rigorous tests on the controller would help in efforts to tune the performance for more adaptability.

- **Implementation of Explicit-MPC** -
  Explicit MPC [Bem02] would help to transfer a lot of the computational burden offline. It involves characterizing the entire state-space offline and determining the controller parameters for each of these regions. This means that MPC would no longer need to perform online optimizations. Instead, it only needs to evaluate the feasible region of the state-space during runtime. This greatly reduces the computational burden, which can be a very important factor in systems like a prosthetic device where computational resources can be limited.

- **Exploring feasibility of Adaptive MPC for parameter estimation** -
  Adaptive MPC can create unnecessary computational burden if implemented during regular use of the prosthesis. Nevertheless, it can be used during the training phase, when the system parameters are still being estimated. If Adaptive MPC is found to be feasible for this purpose, it can eliminate, or greatly reduce the need for an elaborate tuning process. This can increase the ease of adoption of such devices while also reducing the barriers to its accessibility.

[Bem02] Bemporad, A. et al. "The explicit linear quadratic regulator for constrained systems". *Automatica* **38**.1 (2002), pp. 3–20.

[Har13] Hargrove, L. J. et al. "Non-weight-bearing neural control of a powered transfemoral prosthesis". English. *Journal of Neuroengineering and Rehabilitation* **10** (2013). Source type: scholarlyjournals; Object type: Article; Object type: Feature; Copyright: Âl' 2013 Hargrove et al.; licensee BioMed Central Ltd. This is an Open Access article distributed under the terms of the Creative Commons Attribution License (http://creativecommons.org/licenses/by/2.0), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.; DOCID: 3019043681; PCID: 75074212; PMID: 113416; ProvJournalCode: JNRH; DOI: 10.1186/1743-0003-10-62; AccNum: 23782953; PublisherXID: BMDDJNRH20130101174300031062, n/a–62.

[LF08] Lewis, C. L. & Ferris, D. P. "Walking with increased ankle pushoff decreases hip muscle moments". *Journal of Biomechanics* **41**.10 (2008), pp. 2082–2089.

[Liu14] Liu, M. et al. "Improving finite state impedance control of active-transfemoral prosthesis using dempster-shafer based state transition rules". *Journal of Intelligent and Robotic Systems* **76**.3-4 (2014), pp. 461–474.

[Mat16a] Mathworks. *Genetic Algorithm*. 2016. URL: `http://www.mathworks.com/help/gads/genetic-algorithm.html`.

[Mat16b] Mathworks. *MATLAB Distributed Computing Server*. 2016. URL: `http://www.mathworks.com/help/mdce/index.html`.

[Mat16c] Mathworks. *MATLAB fmincon*. 2016. URL: `http://www.mathworks.com/help/optim/ug/fmincon.html?searchHighlight=fmincon`.

[Mat16d] Mathworks. *MATLAB Global Optimization Toolbox*. 2016. URL: `http://www.mathworks.com/help/gads/index.html`.

[Mat16e] Mathworks. *MATLAB Model Predictive Control Toolbox*. 2016. URL: `http://www.mathworks.com/help/mpc/index.html`.

[Mat16f] Mathworks. *MATLAB MPC Optimization Problem*. 2016. URL: `http://www.mathworks.com/help/mpc/ug/optimization-problem.html`.

[Mat16g] Mathworks. *MATLAB MPC Working*. 2016. URL: `http://www.mathworks.com/help/mpc/gs/mpc-modeling.html`.

[Mat16h] Mathworks. *MATLAB Optimization Decision Table*. 2016. URL: `http://www.mathworks.com/help/optim/ug/choosing-a-solver.html`.

[Mat16i] Mathworks. *MATLAB Optimization Toolbox*. 2016. URL: `http://www.mathworks.com/help/optim/index.html`.

[Mat16j] Mathworks. *MATLAB Parallel Computing Toolbox*. 2016. URL: `http://www.mathworks.com/help/distcomp/index.html`.

[Mat16k] Mathworks. *Simulink*. 2016. URL: `http://www.mathworks.com/help/simulink/`.

[Mat16l] Mathworks. *Simulink S-Function*. 2016. URL: `http://www.mathworks.com/help/simulink/sfg/what-is-an-s-function.html`.

[MM80] Mochon, S. & McMahon, T. A. "Ballistic walking". *Journal of Biomechanics* **13**.1 (1980), pp. 49–57.

[Rou13] Rouse, E. J. et al. "Clutchable series-elastic actuator: Design of a robotic knee prosthesis for minimum energy consumption". *Rehabilitation Robotics (ICORR), 2013 IEEE International Conference on*. ID: 1. 2013, pp. 1–6.

[Seb11] Seborg, D. E. *Process dynamics and control*. Ed. by Seborg, D. E. ID: NCSU2713275; Formats: Book; 3rd ed., international student version.; xiv, 394, 44, 9 p. : ill. ; 29 cm.; M2: OCLC Number: 708190340; Includes bibliographical references and index.; Previous ed. published: 2004. Hoboken, N.J.: John Wiley and Sons, 2011.

[SP09] SM, E. C. M.-V. & PhD, H. H. "Agonist-antagonist active knee prosthesis: A preliminary study in level-ground walking". English. *Journal of Rehabilitation Research and Development* **46**.3 (2009). Source type: scholarlyjournals; Object type: Article; Object type: Feature; Copyright: Copyright Superintendent of Documents 2009; DOCID: 1839767561; PCID: 48008941; PMID: 20977; CODEN: JRRDDB; ProvJournalCode: PJHB; AccNum: 19675988; PublisherXID: INNNPJHB0000302825, pp. 361–73.

[Sup08] Sup, F. et al. "Design and control of a powered transfemoral prosthesis". *The International journal of robotics research* **27**.2 (2008), pp. 263–273.

[Sup09] Sup, F. et al. “Preliminary evaluations of a self-contained anthropomorphic transfemoral prosthesis”. *Mechatronics, IEEE/ASME Transactions on* **14**.6 (2009), pp. 667–676.

[Uni16] University, N. S. *Office of Information Technology High Performance Computing*. 2016. URL: `https://ncsu.edu/hpc/`.

[Wik16] Wikipedia. *Interior Point Method*. 2016. URL: `https://en.wikipedia.org/wiki/Interior_point_method`.

[Win09] Winter David A., . *Biomechanics and motor control of human movement*. ID: NCSU2239720; Formats: Book; 4th ed.; xiv, 370 p. : ill. ; 25 cm.; M2: OCLC Number: 318408191; Includes bibliographical references and index. Hoboken, N.J.: Wiley, 2009.

[Yan15] Yang, T. et al. “An Optimization-Based Approach for Prosthesis Dynamic Modeling and Parameter Identification”. *ASME 2015 Dynamic Systems and Control Conference*. American Society of Mechanical Engineers, 2015, V001T16A001–V001T16A001.

[ZG08] Ziegler-Graham, K. et al. “Estimating the Prevalence of Limb Loss in the United States: 2005 to 2050”. *Archives of Physical Medicine and Rehabilitation* **89**.3 (2008), pp. 422–429.

[Zla02] Zlatnik, D. et al. “Finite-state control of a trans-femoral (TF) prosthesis”. *IEEE Transactions on Control Systems Technology* **10**.3 (2002). ID: 1, pp. 408–420.