2018

Neuromuscular Reflex Control for Prostheses and Exoskeletons

Sandra K. Hnat
Cleveland State University

NEUROMUSCULAR REFLEX CONTROL

FOR PROSTHESES AND EXOSKELETONS

SANDRA K. HNAT

Bachelor of Science in Mechanical Engineering

Cleveland State University

May 2013

submitted in partial fulfillment of requirements for the degree

DOCTOR OF ENGINEERING IN MECHANICAL ENGINEERING

at

CLEVELAND STATE UNIVERSITY

MAY 2018

ACKNOWLEDGMENTS

I've been a fairly lucky person throughout my graduate career. Amidst the throngs of graduate-school-woes, having a mean advisor was *never* something I had to worry about. Ton, thanks for taking me on as one of your students and giving me this fantastic opportunity. Thanks for your patience, kindness, and understanding. Thanks for teaching me so many things—from academics to social graces.

I also wanted to thank my committee members for providing me with many different perspectives to my research: Dr. Eric Schearer, Dr. Jason Halloran, Dr. Michael Hammonds, and Dr. Richter. In particular, thanks Dr. Richter for guiding a goalless, recent undergrad (me) to grad school.

I'd like to thank all my funding sources, including the National Science Foundation under Grant No. 1344954 and by the Ohio Department of Development, Third Frontier Commission. Also, to Parker Hannifin for many reasons: 1) the Human Motion and Control Laboratory at Cleveland State University, 2) the Graduate Research Fellowship Program, 3) loaning the Indego to us, and 4) to Ryan Farris and Skyler Dalley for helping us in all of our questions regarding the Indego.

To all my lab-mates and colleagues: Anne, Milad, Ravi, Huawei, Farzad, Sabrina, Farbod, Roman, Samin, Brad, Nicole, Michael, Ben, Jason, and Obinna. We've shared laughter, pain, disappointment, silliness, Froyo, and knowledge. As for our quote board, I've a permanent one for all of you: Thanks for making grad school so much fun.

A big thanks to my entire family for their support (grandparents, Aunt Barb, Steph, Bill, etc.) Also, to my friend Andy, who might as well be lumped together into the family section (and not the horse section). To my parents, a few obligatory remarks: I wouldn't be here without you both so thanks for making me (literally) and also making me into the person I am now. As a small repayment for everyone's endless support and encouragement, maybe a few more people can see our last name and scratch their head, wondering if it's a typo or not.

I wanted to thank my high school biotechnology teacher, Mr. Gary Maurer. He drove me to pursue a career in science. I was the stereotypical, wallflower student hoping to slink by without being noticed. Yet, on the last day of my senior year, Mr. Maurer wrote in my yearbook "Sandy, the cream of the crop always rises to the top." What a thing to say! Whenever I've found myself discouraged at any point, I fall back to these words. Thanks, Mr. Maurer, for a seemingly small encouragement that has bolstered my spirit for years—and always will.

Lastly, to my husband, Rick. There are so many things I could say that it would probably extend longer than this dissertation. In summary, when I was trying to figure out what I wanted to do with my life, Rick told me I should be an engineer. He wasn't telling me what to do—he just knew I needed some encouragement. To me, engineers were brainiacs, but he told me, "If I can do it, then you can do it." That's a sweet thing to say, because he's a smart guy! He's also the best support system (and proofreader) I could have ever hoped to have. Rick, the future is always bright with you and I look forward to whatever it holds (though, hopefully this includes a few little ones just like you).

NEUROMUSCULAR REFLEX CONTROL

FOR PROTHESES AND EXOSKELETONS

SANDRA K. HNAT

ABSTRACT

Recent powered lower-limb prosthetic and orthotic (P/O) devices aim to restore legged mobility for persons with an amputation or spinal cord injury. Though various control strategies have been proposed for these devices, specifically finite-state impedance controllers, natural gait mechanics are not usually achieved. The goal of this project was to invent a biologically-inspired controller for powered P/O devices. We hypothesize that a more muscle-like actuation system, including spinal reflexes and vestibular feedback, can achieve able-bodied walking and also respond to outside perturbations. The outputs of the Virtual Muscle Reflex (VMR) controller are joint torque commands, sent to the electric motors of a P/O device. We identified the controller parameters through optimizations using human experimental data of perturbed walking, in which we minimized the error between the torque produced by our controller and the standard torque trajectories observed in the able-bodied experiments.

In simulations, we then compare the VMR controller to a four-phase impedance controller. For both controllers the coefficient of determination (R^2) and root-mean-square (RMS) error were calculated as a function of the gait cycle. When simulating the hip, knee, and ankle joints, the RMS error and R^2 across all joints and all trials is 15.65 Nm and 0.28 for the impedance controller, respectively, and for the VMR controller, these values are 15.15 Nm and 0.29, respectively. With similar performance, it was concluded that the VMR controller can reproduce characteristics of human walking in response to perturbations as effectively as an impedance controller.

We then implemented the VMR controller on the Parker Hannifin Indego® powered exoskeleton and performed standard isokinetic and isometric knee rehabilitation exercises to observe the behavior of the virtual muscle model. In the isometric results, RMS error

iv

between the measured and commanded extension and flexion torques are 3.28 Nm and 1.25 Nm, respectively. In the isokinetic trials, we receive RMS error between the measured and commanded extension and flexion torques of 0.73 Nm and 0.24 Nm. Since the onboard virtual muscles demonstrate similar muscle force-length and force-velocity relationships observed in humans, we conclude the model is capable of the same stabilizing capabilities as observed in an impedance controller.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER I

OVERVIEW

Recent powered prosthetic and orthotic (P/O) devices aim to improve the quality of life for persons with an amputation or spinal cord injury (SCI). With the prevalence of cardiovascular disease and diabetes [1], along with physical trauma from work-related accidents or wars [2], the number of individuals who require a P/O device is projected to steadily increase. The National Spinal Cord Injury Statistical Center expects an increase of approximately 17,700 new cases of Spinal Cord Injuries from 2017 [3]. Additionally, over 185,000 new lower-limb amputations are reported every year, resulting in over two million individuals who require a prosthetic device [4]. Such a large population of affected individuals promotes the need to advance the performance of available assistive devices.

To restore legged mobility to these populations, the design of modern, powered lower-limb P/O devices focuses on developing sophisticated controllers capable of closely replicating able-bodied gait. The majority of these commercially available devices are equipped with torque-driven electric motors at the joints, which can be controlled to reproduce the typical walking patterns observed in human locomotion [5]. Prosthetic devices are outfitted in series to the residual part of the patient's limb, in which the motors are designed to replicate the function of the missing joint [6]. For individuals with a SCI, powered orthoses operate mechanically in parallel to the body to guide the paralyzed or weakened limbs through sitting, standing, and walking motions [7].

1.1 Powered Prosthetics

Passive prosthetic devices have existed for thousands of years and can restore basic function and mobility to persons with lower-limb amputations [8, 9]. In general, passive prostheses operate by modulating between the swing and stance phases of gait by allowing the leg to swing and by offering support during the weight-bearing stance phase. A few passive trans-femoral prostheses have been commercialized and are readily prescribed to persons with an amputation. The Mauch® Knee by Össur® is a hydraulic system that can be set to either lock or free the knee during swing and can also modulate between different speed settings [10]. Another hydraulic system is the Otto Bock C-Leg [11], which adjusts damping by measuring angular positions and torques. Hugh Herr and Ari Wilkenfeld designed the user-adaptive RHEO-knee®, commercialized by Össur [12, 13]. The RHEO is equipped with a magnetoheological fluid in the braking system, in which iron particles are magnetically charged by an electromagnet to produce torque. Other passive models achieve walking through mechanical compliance in physical hardware, such as springs and clutches at the ankle [14] or the knee [15].

Though recent passive prostheses are far more advanced than a wooden leg, there are several disadvantages in using these devices, including: the inability to produce positive net mechanical work, provide feedback to the user as to the state of the device, or respond to outside perturbations [6]. Persons who are prescribed a passive prosthesis overcompensate with their trunk and intact limbs to address these limitations, leading to an overall increase in oxygen consumption, energy cost, and gait asymmetry [16, 17].

As an alternative, actuated prostheses can not only replicate more features of able-bodied walking, but can also adapt to environmental changes using feedback control. Though some designs are pneumatically [18] or hydraulically powered [19], electrically-actuated prostheses are an emerging trend due to improvements to the physical hardware, including battery life and microcontroller processors [20–22].

There are several electrically-powered prostheses which are either currently in develop-

2

ment or commercially available. Some examples include the Össur Power Knee[TM] which directly drives the knee joint using a single actuator-transmission design [23, 24]. Another single-joint prosthesss was developed by Hugh Herr et al. and the BionX company, named the BiOM ankle [25, 26], which is controlled by a neuromuscular model including simulated muscles and reflexes [27]. Herr also developed an active knee prosthesis [28] and tuned a powered ankle-foot prosthesis for level-ground walking and stair descent which can reduce the metabolic cost of walking [29, 30]. A transfemoral prosthesis was developed by Sup et al. at Vanderbilt University, in which both the knee and ankle joints are actuated [31, 32]. Other prostheses use a variety of different control approaches, including electromyography (EMG) [33] or designing the system with energy regeneration capabilities [34].

Impedance Control Methods

One of the earliest control strategies for a powered P/O device is Echo Control [35] by Grimes in 1979. In this method, gait is separated into many modes between stance and swing. In swing phase, the leg behaves like a spring and damper, while in the stance phase, the prosthetic is controlled to reproduce the same motion of the user's intact knee.

This concept has been replaced by impedance control, which describes the relationship between input flow and output effort [36]. For a prosthetic device, the input is defined as the angular velocity, while the output is the torque. Stiffness and damping properties are often included in impedance controllers, approximately representing the passive, spring-like properties of muscle architecture [37]. By separating the gait amongst many phases, different sets of spring and damper controllers are used to achieve walking.

In the controller of Sup et al. [31], the torque for each joint is modeled as a function of the joint angle and velocity, including stiffness term, k_i, damping term, b_i, and equilibrium

position of the virtual spring, θ_{ki}, where i denotes the number of joints:

$$\tau = -k_i(\theta - \theta_{ki}) - b_i\dot{\theta} \qquad (1.1)$$

The controller is then split throughout four finite states, defined by transition criteria as shown in Figure 1, which promotes a trend towards more passive behavior throughout each gait cycle. Separate stiffness and damping parameters are tuned during each of the four phases.
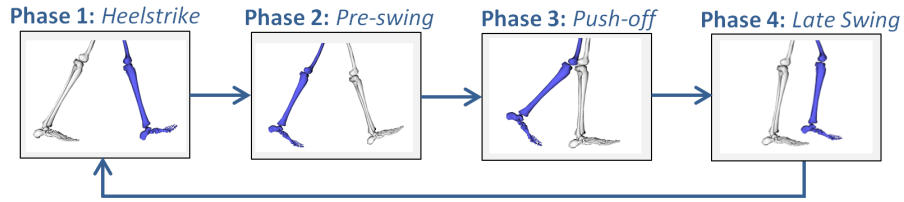
[31, 38]

Figure 1: Finite states for the impedance controller of Sup et al.

The results of Sup et. al suggest that an impedance controller with this structure is capable of replicating able-bodied human locomotion at the knee and ankle joint, as well as respond to some changes in the environment, after tuning the parameters for these conditions [39, 40].

Neuromuscular Reflex Control Methods

Instead of approximating the dynamics of muscles through a single stiffness and damping term, neuromuscular reflex controllers take advantage of the stabilizing ability of muscles by directly modeling them as part of the controller. In the autonomous, positive force feedback controller of Geyer and Herr, spinal reflexes generate the control signal that would activate the muscle instead of relying on the more complex workings of the cerebral cortices [27]. Though positive feedback is usually avoided in classical control design, it has been shown that humans use positive feedback as a stabilizing mechanism during walking, particularly in the plantarflexors during push-off and also for stability during stance

[41, 42].

However, reflexes alone cannot describe the aspects of balance that are regulated by the cerebellum, brain stem, and visual and vestibular cortices of the cerebrum [43]. To address the limitations of reflexes, Geyer and Herr added proportional-derivative (PD) control to maintain postural stability of the trunk. PD control can be loosely correlated to human control strategies as neurological tracts in the spinal cord carry sensory input to the cerebellum to correct for positional errors [44].

This combination of an autonomous, neuromuscular reflex controller and stabilizing PD trunk control has been shown to produce stable walking in simulations of a walking model [27]. The model has been expanded to be able to perform walking, running, stair climbing, and respond to perturbations [45]. This control strategy was implemented in an actuated knee and ankle prosthesis and compared to an impedance controller [46], with results suggesting that reflex control is a viable alternative to an impedance controller. Also, reflex control is implemented in Herr's BiOM single-joint ankle prosthesis (BionX, [25, 26]). Recently, a reflex controller was evaluated on an exoskeleton with an actuated ankle, resulting in a reduction of net metabolic cost during rehabilitation exercises [47].

1.2 Powered Orthotics and Exoskeletons

Exoskeletons are anthropomorphic devices worn by the user which enhance or assist the operator's performance, while *powered orthoses* are classified as a type of exoskeleton that restores function to individuals with a SCI or other type of pathological condition which affects ambulation [48]. The first known wearable exoskeleton is credited to Nicholas Yagn in the year 1890, who intended to amplify power through energy storage in springs [49, 50]. Since then, many exoskeletons have been designed for military applications to increase the strength of the user in order to carry heavier loads, including the Hardiman hydraulic machine designed by General Electric Research in the 1960s [51] and the BLEEX exoskeleton developed at the University of Berkeley [52]. More recently, both active and

passive exoskeletons have been designed to reduce the metabolic energy cost of walking for able-bodied operators [53–55].

In the last few decades, several exoskeletons have emerged for assisting during rehabilitation and gait training for individuals suffering from neurological conditions. In lower-extremity rehabilitation, these exoskeletons can replace the therapist and allow more extensive training while measuring the patient's progress throughout the treatment, namely force production and joint angle patterns [56]. Many of these devices, including the ALEX I and II [57, 58] and LOPES robot [59], are used as treadmill gait trainers to reduce the strain on the therapist. Recently, a neuromuscular reflex controller has been applied to a powered rehabilitative ankle exoskeleton [47].

However, these devices are intended to work in concert with the patient's capabilities and cannot provide enough support for those with complete paraplegia. To address this population, a few passive orthoses were designed to act as stiff braces which lock the knee and ankle joints, though walking is only achieved by cumbersome movement of the upper body [60]. As an alternative, there are few portable, commercially available powered orthoses that can entirely facilitate locomotion.

The first commercialized powered orthoses was the Hybrid-Assistive-Leg® (HAL®), invented by Cyberdyne Inc. in Japan [61, 62]. Though initially used to increase the strength of the user in load-bearing applications, it is now widely used as a rehabilitation device for patients following stroke, SCI, or other neurological pathologies. Though various versions of HAL of been developed, the current HAL-3 weighs approximately 15 kg and uses brushless DC motors at the hip and knee with harmonic gear drives [63–65], though preliminary designs included activation at all joints in the lower limb [66]. Similarly, Ekso Bionics® designed the 20 kg Ekso GT™ [67, 68] for rehabilitation purposes.

Only a few powered orthoses have been developed, commercialized, and marketed towards individuals with complete paraplegia. One is the 23 kg, battery-powered ReWalk™ by ReWalk Robotics™, which has a passive ankle foot orthosis (AFO) and actuated hip

6

and knee joints [69–71]. Likewise, The Parker Hannifin Indego® [72] is a battery-powered, relatively lightweight (12 kg) device which uses brushless motors at the hip and knee joints and an AFO. The motors of the Indego are capable of operating within the range of motion typically observed during able-bodied walking [5]. Few studies have have been published which describe the performance of these devices in clinical settings. Likewise, the design specifications and details of the control strategies are proprietary and minimally described in literature.

Control Strategies

Exoskeleton control can be classified as either the dynamic model or the muscle based control model. Dynamic models consider the physical system itself, modeling the links of the exoskeleton as rigid bodies, and deriving the equations of motion through classical dynamics and robotics equations (mass, inertia, coriolis) [7]. Deriving a dynamic model of the entire exoskeleton is challenging since a theoretical model requires accurate estimations of its parameters for use in physical hardware. Methods for identifying these parameters greatly vary between exoskeletons, though neural networks have been used to estimate parameters and predict gait phases throughout a walking cycle [73]. The controller for the BLEEX exoskeleton was derived using system identification and linear least squares between the joint angle and joint torque to obtain the parameters for the dynamic model [52].

Muscle-based control strategies use EMG to increase assistance based on user-intent [74]. Since EMG produces an electrical signal, cyclical changes in the user's muscle activation can trigger the controller to produce walking. Other approaches use EMG to estimate muscle forces [75], though deriving a model between muscle force and EMG is challenging due to signal noise. EMG-based methods have also been implemented in HAL [62, 63], in which a calibration estimates a joint torque from increased muscle activity to provide additional assistance whenever required.

Alternatively, position controllers are used as the low-level controller in rehabilitation exoskeletons, including the LOPES [59] and HAL [65]. This is to ensure the leg maintains a specific joint angle trajectory throughout the gait cycle. For commercially available products, such as the ReWalk and Ekso, information regarding the control systems are not published. However, some details regarding the Indego's controller were available [72]. The Indego uses phase-based proportional-derivative (PD) control, in which the gains are adjusted to follow joint angle trajectories of the hip and knee joints based on collected experimental data. Similar to the impedance controller of a prosthesis, the PD controller during the walking state is divided into four separate phases of the gait cycle: single-support, two phases of double-support, and the swing phase [72]. Separate controller gains are tuned to follow the joint angle trajectory throughout the four phases.

1.3 Problem Statement

Though various control strategies have been proposed for P/O devices, natural gait mechanics are not usually achieved. Typically, controllers for these devices are manually tuned until walking-like motions are obtained in either simulation or hardware. These unnatural and asymmetric gait patterns can cause other health-related concerns for prosthetic users. Studies consistently report an elevated metabolic energy cost when compared to able-bodied walking at similar speeds [76–78] and increased prevalence of pain or osteoarthritis in the joints of the intact limb [79–81]. For powered orthoses users, onboard controllers have yet to produce able-bodied walking at natural speeds and the discrete gain switching between gait phases causes jerky transitions. Users are incapable of maintaining balance during walking and require the use of crutches or walkers for stabilizing the upper body. Therefore, manually tuning controller parameters may not be sufficient enough to obtain smooth, able-bodied walking and maintain balance while being able to respond to outside perturbations.

8

1.4 Specific Aims

We hypothesize that a more biologically inspired actuation system is capable of replicating able-bodied locomotion in powered P/O devices. In this dissertation, we design a new Virtual Muscle Reflex (VMR) controller, based on Geyer and Herr's model [27], that combines muscles, spinal reflexes, and vestibular feedback (Virtual Muscle Reflex, VMR). The VMR system produces a joint torque which can be used to drive the electric motors of a P/O device. Because muscles and reflexes demonstrate the same joint position and velocity dependence of an impedance controller, the VMR system can be described as a biologically-inspired subset of impedance control. An overview of the VMR controller is shown in Figure 2.

The objectives of this dissertation are as follows:

Objective 1: To design human walking experiments under the effect of mechanical perturbations. The approach to developing a more bio-inspired controller is to use perturbed human walking data collected from able-bodied subjects. Through recorded kinematic and kinetic perturbed gait data, optimizations will minimize the difference between the subject's joint torque pattern (obtained from standard 2-D inverse dynamics [5]) and the torque generated from the VMR system.

Objective 2: To develop a virtual muscle model that generates real-time joint torque commands. In place of the spring-damper characteristics of an impedance controller, we will model virtual muscles to produce the joint torque which would operate the electric motors of the P/O device. To be used on a P/O device, this virtual muscle model must be able to be simulated in real-time.

9

Objective 3: To design a neuromuscular reflex controller to generate muscle activation signals. Muscles are stimulated by a neural activation signal which causes contraction. Instead of modeling more complex cerebral processes, we will modify Geyer and Herr's reflex controller [27] in order to more closely resemble the joint torques measured in the perturbed walking experiments. We then compare its performance to a standard four-phase impedance controller.

Objective 4: To implement the VMR system in the Indego. To observe the VMR controller in hardware, we will implement the system on the Indego. We measure the behavior of the virtual muscles through standard isokinetic and isometric rehabilitation exercises to determine if the VMR system exhibits similar force-length and force-velocity properties of human muscles under the same conditions.
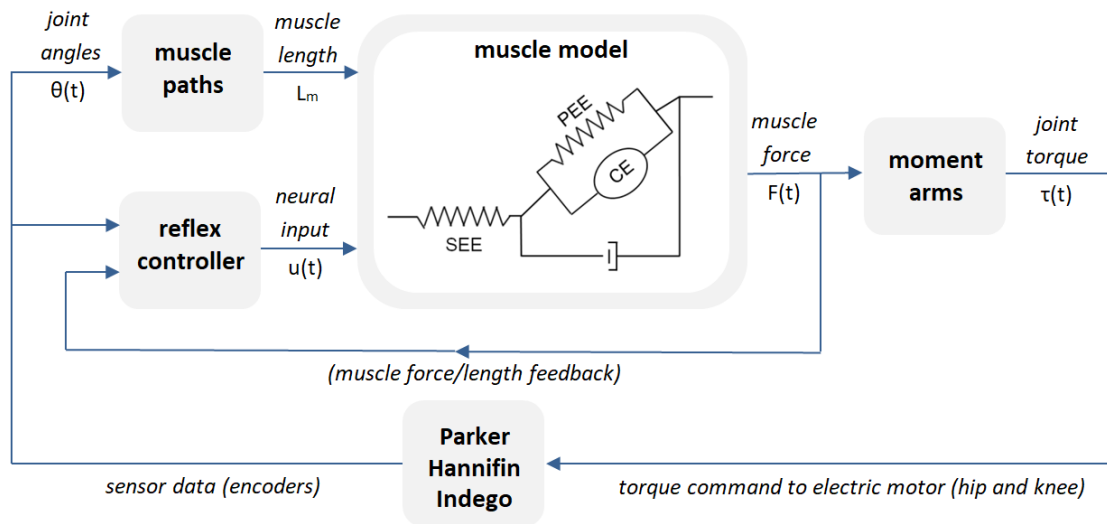
Figure 2: Block diagram of the Virtual Muscle Reflex (VMR) controller. Inputs to the muscle model are the muscle length L_m (determined from joint angles from either an encoder or from walking experiments), and the neural excitation signals generated by the reflex controller $u(t)$. The output is the muscle force which can be multiplied by the moment arm to obtain the joint torque, $\tau(t)$, used to control the electric motors of the Indego. Muscle force and length are used as feedback for the reflex controller.

1.5   Outline

This dissertation is divided into three parts.

Part I addresses **Objective 1** of this dissertation, where:

- *Chapter II:* we design perturbation experiments to obtain the human joint torque in order to identify the nature of our VMR controller.

- *Chapter III:* we invent an inertial compensation method for reducing force plate measurement errors which can occur when varying the belt speed for surface perturbations

- *Chapter IV:* though our experimental protocol did not include platform movement, we invent a separate inertial compensation method for minimizing force plate measurement errors from these type of perturbations.

Part II concerns both **Objective 2** and **Objective 3** for designing the VMR system, where:

- *Chapter V:* we detail the muscle model and its formulation for real-time applications. First, we verify the behavior of the muscles through a simple proof-of-concept test using step-input neural commands, and then we calculate the speed and accuracy of the model.

- *Chapter VI:* we design the VMR system for the ankle and compare to a four-phase impedance controller. An optimization identifies the parameters of each controller, and their performances are compared using the mean root-mean-square (RMS) error and maximum correlation coefficient R^2 as a function of the gait cycle. The better controller will achieve lower RMS error and higher correlation.

- *Chapter VII:* we repeat the techniques described in Chapter 6 and expand the VMR system to include the knee and hip joint. We then compare to an impedance controller.

Part III addresses **Objective 4** of this dissertation, where:

- *Chapter VIII:* we detail how the VMR system was implemented on the Indego and describe some preliminary tests to ensure the model was successfully built.

- *Chapter IX:* we isolate the virtual muscles without reflex control to verify if the Indego's "muscles" can behave similarly to humans during standard isokinetic and isometric trials on a dynamometer rehabilitation machine.

Figure 3 illustrates Parts I (Objective 1) and Part II (Objective 2-3), where Figure 2 shows the final block diagram for Part III (Objective 4).
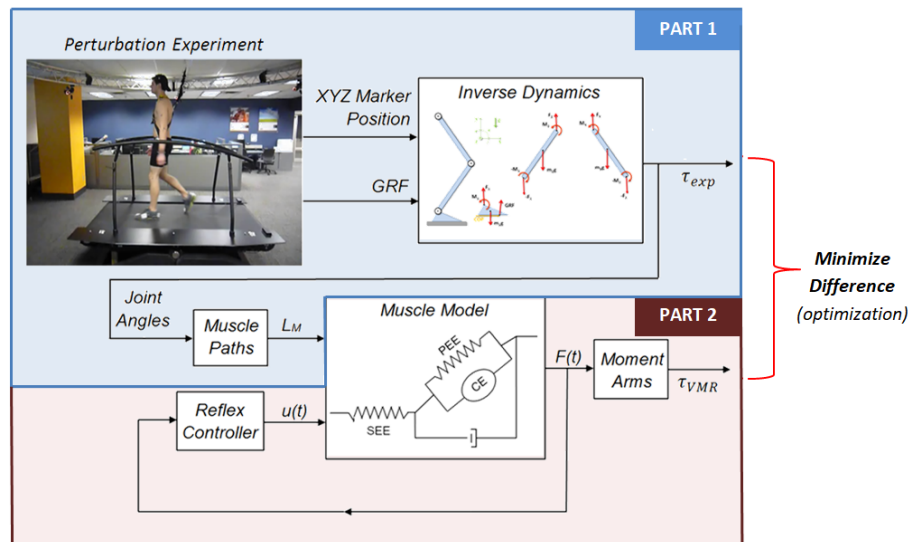
Figure 3: Diagram for Part I in blue: Perturbed walking experiment design and perturbation method. Here we obtain joint torques through inverse dynamics of our able-bodied perturbed walking experiments. Part II in red: Designing the virtual muscle reflex controller using joint angle data as input to the controller. An optimization minimizes the difference between the joint torque measured in the experiment and the joint torque from the VMR controller

1.6 REFERENCES

[1] L. J. Marks and J. W. Michael. Artificial limbs. *BMJ: British Medical Journal*, 323 (7315):732–735, September 2001.

[2] C. Horne and J. Neil. Quality of life in patients with prosthetic legs: A comparison study. *Journal of Prosthetics and Orthotics*, 21(3):154–159, July 2009.

[3] Spinal cord injury facts and injuries at a glance. `https://www.nscisc.uab.edu/Public/Facts%20and%20Figures%20-%202018.pdf`. Accessed: 2018-03-13.

[4] Amputee statistics you ought to know. `http://www.advancedamputees.com/amputee-statistics-you-ought-know`. Accessed: 2017-03-13.

[5] D. A. Winter. *Biomechanics and Motor Control of Human Movement*. John Wiley and Sons, New York, 1990.

[6] M. R. Tucker, J. Olivier, A. Pagel, H. Bleuler, M. Bouri, O. Lambercy, J. del R. Millán, R. Riener, H. Vallery, and R. Gassert. Control strategies for active lower extremity prosthetics and orthotics: A review. *Journal of NeuroEngineering and Rehabilitation*, 12(1):1, December 2015.

[7] K. Anam and A. A. Al-Jumaily. Active exoskeleton control systems: State of the art. *Procedia Engineering*, 41:988–994, December 2012. ISSN 1877-7058.

[8] E. E. Vanderwerker. A brief review of the history of amputations and prostheses. *Inter-Clinic Information Bulletin*, 15(5):15–16, 1976.

[9] K. Norton. A brief history of prosthetics. *InMotion*, 17(7):11–13, November 2007.

[10] H. A. Mauch and D. Ing. Stance control for above-knee artificial legs-design considerations in the S-N-S knee. 10, January 1968.

[11] M. S. Orendurff, A. D. Segal, G. K. Klute, M. L McDowell, J. A. Pecoraro, and J. M. Czerniecki. Gait efficiency using the C-leg. *Journal of Rehabilitation Research and Development*, 43:239–246, April 2006.

[12] H. Herr, A. Wilkenfield, and O. Bleck. Speed-adaptive and patient-adaptive prosthetic knee, October 2007. US Patent 7,279,009 B2.

[13] H. Herr and A. Wilkenfeld. User-adaptive control of a magnetorheological prosthetic knee. 30:42–55, February 2003.

[14] K. E. Zelik, S. H. Collins, P. G. Adamczyk, A. D. Segal, G. K. Klute, D. C. Morgenroth, M. E. Hahn, M. S. Orendurff, J. M. Czerniecki, and A. D. Kuo. Systematic variation of prosthetic foot spring affects center-of-mass mechanics and metabolic cost during walking. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 19:411–419, August 2011.

[15] V. N. Murthy Arelekatti and A. G. Winter. Design of a fully passive prosthetic knee mechanism for transfemoral amputees in India. In *2015 IEEE International Conference on Rehabilitation Robotics*, pages 350–356, August 2015.

[16] R. L. Waters, J. Perry, D. Antonelli, and H. Hislop. Energy cost of walking of amputees: The influence of level of amputation. *The Journal of Bone and Joint Surgery*, 58(1):42–46, January 1976.

[17] S. V. Fisher and G. Gullickson Jr.. Energy cost of ambulation in health and disability: A literature review. *Archives of Physical Medicine and Rehabilitation*, 59(3):124–133, March 1978.

[18] R. Versluys, A. Desomer, G. Lenaerts, O. Pareit, B. Vanderborght, G. Perre, L. Peeraer, and D. Lefeber. A biomechatronical transtibial prosthesis powered by pleated pneumatic artificial muscles. *International Journal of Modelling, Identification and Control*, 4(4):394–405, January 2008.

[19] S. Blumentritt H. W. Scherer, J. W. Michael, and T. Schmalz. Transfemoral amputees walking on a rotary hydraulic prosthetic knee mechanism: A preliminary report. *JPO: Journal of Prosthetics and Orthotics*, 10(3):61–70, January 1998.

[20] J. Yu, M. Tan, J. Chen, and J. Zhang. A survey on CPG-inspired control models and system implementation. *IEEE Transactions on Neural Networks and Learning Systems*, 25(3):441–456, September 2013.

[21] K. Nakada, T. Asai, and Y. Amemiya. An analog CMOS central pattern generator for interlimb coordination in quadruped locomotion. *IEEE Transactions on Neural Networks*, 14(5):1356–1365, November 2003.

[22] C. Torres-Huitzil and B. Girau. Implementation of central pattern generator in an FPGA-based embedded system. In *International Conference on Artificial Neural Networks*, pages 179–187, November 2008.

[23] E. C. Martinez-Villalpando, J. Weber, G. Elliott, and H. Herr. Design of an agonist-antagonist active knee prosthesis. In *BioRob 2008. 2nd IEEE RAS & EMBS International Conference on Biomedical Robotics and Biomechatronics*, pages 529–534, October 2008.

[24] Össur - Life without Limitations. `https://www.ossur.com/prosthetic-solutions/products/dynamic-solutions/power-knee`. Accessed: 2017-03-15.

[25] M. F. Eilenberg, H. Geyer, and H. Herr. Control of a powered ankle-foot prosthesis based on a neuromuscular model. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 18(2):164–173, April 2010. doi: 10.1109/TNSRE.2009. 2039620.

[26] H. Herr and A. M. Grabowski. Bionic ankle–foot prosthesis normalizes walking gait

for persons with leg amputation. *Proceedings of the Royal Society of London B: Biological Sciences*, 279:457–464, February 2012.

[27] H. Geyer and H. Herr. A muscle-reflex model that encodes principles of legged mechanics produces human walking dynamics and muscle activities. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 18(3):263–273, June 2010.

[28] E. C. Martinez-Villalpando and H. Herr. Agonist-antagonist active knee prosthesis: A preliminary study in level-ground walking. *Journal of Rehabilitation Research and Development*, 46(3):361, February 2009.

[29] S. Au, M. Berniker, and H. Herr. Powered ankle-foot prosthesis to assist level-ground and stair-descent gaits. *Neural Networks*, 21(4):654–666, May 2008.

[30] S. K. Au, J. Weber, and H. Herr. Powered ankle-foot prosthesis improves walking metabolic economy. *IEEE Transactions on Robotics*, 25(1):51–66, January 2009.

[31] F. Sup, A. Bohara, and M. Goldfarb. Design and control of a powered transfemoral prosthesis. *International Journal of Robotics Research*, 27(2):263–273, February 2008.

[32] B. E. Lawson, B. Ruhe, A. Shultz, and M. Goldfarb. A powered prosthetic intervention for bilateral transfemoral amputees. *IEEE Transactions on Biomedical Engineering*, 62(4):1042–1050, April 2015.

[33] C. D. Hoover, G. D. Fulk, and K. B. Fite. Stair ascent with a powered transfemoral prosthesis under direct myoelectric control. *IEEE/ASME Transactions on Mechatronics*, 18(3):1191–1200, June 2013.

[34] J. K. Hitt, T. G. Sugar, M. Holgate, and R. Bellman. An active foot-ankle prosthesis with biomechanical energy regeneration. *Journal of Medical Devices*, 4(1):011003, March 2010.

[35] D. L. Grimes. *An active multi-mode above knee prosthesis controller*. PhD thesis, Massachusetts Institute of Technology, August 2005.

[36] N. Hogan. The mechanics of multi-joint posture and movement control. *Biological Cybernetics*, 52(5):315–331, September 1985.

[37] M. Millard, T. Uchida, A. Seth, and S. L. Delp. Flexing computational muscle: Modeling and simulation of musculotendon dynamics. *ASME Journal of Biomechanical Engineering*, 135(2):0210051–02100511, February 2013.

[38] F. C. Sup. *A powered self-contained knee and ankle prosthesis for near normal gait in transfemoral amputees*. PhD thesis, Vanderbilt University, 2009.

[39] F. Sup IV, A. Varol, J. Mitchell, T. Withrow, and M. Goldfarb. Preliminary evaluations of a self-contained anthropomorphic transfemoral prosthesis. In *IEEE/ASME Transactions on Mechatronics: A Joint Publication of the IEEE Industrial Electronics Society and the ASME Dynamic Systems and Control Division*, volume 14, pages 667–676, December 2009.

[40] M. Goldfarb, A. Varol, F. Sup IV, J. Mitchell, and T. Withrow. Powered leg prosthesis and control methodologies for obtaining near normal gait, February 2014. US Patent 8,652,218 B2.

[41] A. and H. Geyer. The 3-D spring-mass model reveals a time-based deadbeat control for highly robust running and steering in uncertain environments. *IEEE Transactions on Robotics*, 29(5):1114–1124, May 2013.

[42] H. Geyer, A. Seyfarth, and R. Blickhan. Positive force feedback in bouncing gaits? *Proceedings of the Royal Society of London B: Biological Sciences*, 270(1529):2173–2183, October 2003.

[43] V. Dietz. Proprioception and locomotor disorders. *Nature Reviews Neuroscience*, 3 (10):781–790, October 2002.

[44] J. Nolte. *The Human Brain: An Introduction to its Functional Anatomy*. Springer-Verlag, New York, 1988.

[45] S. Song and H. Geyer. A neural circuitry that emphasizes spinal feedback generates diverse behaviours of human locomotion. *The Journal of Physiology*, 593(16):3493–3511, August 2015.

[46] N. Thatte and H. Geyer. Toward balance recovery with leg prostheses using neuromuscular model control. *IEEE Transactions on Biomedical Engineering*, 63(4): 904–913, May 2016.

[47] F. Dzeladini, A. R. Wu, D. Renjewski, A. Arami, E. Burdet, E. van Asseldonk, H. van der Kooij, and A. J. Ijspeert. Effects of a neuromuscular controller on a powered ankle exoskeleton during human walking. In *2016 6th IEEE International Conference on Biomedical Robotics and Biomechatronics (BioRob)*, pages 617–622, June 2016.

[48] A. M. Dollar and H. Herr. Lower extremity exoskeletons and active orthoses: Challenges and state-of-the-art. *IEEE Transactions on Robotics*, 24(1):144–158, February 2008.

[49] N. Yagn. Apparatus for facilitating walking, January 1890. US Patent 420,179.

[50] N. Yagn. Apparatus for facilitating walking, November 1890. US Patent 440,684.

[51] B. R. Fick and J. B. Makinson. Hardiman I prototype for machine augmentation of human strength and endurance: Final report. *General Electric Company*, S-71-1056, August 1971.

[52] H. Kazerooni. The berkeley lower extremity exoskeleton. In *Field and Service Robotics: Results of the 5th International Conference*, pages 9–15, Berlin, Heidelberg, 2006. Springer Berlin Heidelberg.

[53] S. H. Collins, M. B. Wiggin, and G. S. Sawicki. Reducing the energy cost of human walking using an unpowered exoskeleton. *Nature*, 522:212–215, June 2015.

[54] S. Diller, C. Majidi, and S. H. Collins. A lightweight, low-power electroadhesive clutch and spring for exoskeleton actuation. In *2016 IEEE International Conference on Robotics and Automation (ICRA)*, pages 682–689, May 2016.

[55] W. Van Dijk, H. van der Kooij, and E. Hekman. A passive exoskeleton with artificial tendons: Design and experimental evaluation. In *2011 IEEE International Conference on Rehabilitation Robotics*, pages 1–6, July 2011.

[56] I. Diaz, J. J. Gil, and E. Sanchez. Lower-limb robotic rehabilitation: Literature review and challenges. *Journal of Robotics*, 2011(Article ID 759764):1–11, September 2011.

[57] K. N. Winfree, P. Stegall, and S. K. Agrawal. Design of a minimally constraining, passively supported gait training exoskeleton: ALEX II. In *2011 IEEE International Conference on Rehabilitation Robotics*, pages 1–6, July 2011.

[58] S. K. Banala, S. K. Agrawal, and J. P. Scholz. Active leg exoskeleton (ALEX) for gait rehabilitation of motor-impaired patients. In *2007 IEEE 10th International Conference on Rehabilitation Robotics*, pages 401–407, June 2007.

[59] J. F. Veneman, R. Kruidhof, E. E. G. Hekman, R. Ekkelenkamp, E. H. F. Van Asseldonk, and H. van der Kooij. Design and evaluation of the LOPES exoskeleton robot for interactive gait rehabilitation. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 15(3):379–386, September 2007.

[60] P. B. Butler, R. E. Major, and J. H. Patrick. The technique of reciprocal walking using the hip guidace orthosis (HGO) with crutches. *Prosthetics and Orthotics International*, 8:33–38, April 1984.

[61] What's HAL? - The World's First Cyborg Type Robot. `https://www.cyberdyne.jp/english/products/HAL/`. Accessed: 2017-04-09.

[62] H. Kawamoto, S. Lee, S. Kanbe, and Y. Sankai. Power assist method for HAL-3 using EMG-based feedback controller. In *IEEE International Conference on Systems, Man and Cybernetics*, volume 2, pages 1648–1653, October 2003.

[63] H. Sankai and Y. Kawamoto. EMG-based hybrid assistive leg for walking aid using feedforward controller. In *Proceedings of the International Conference on Control, Automation and Systems*, pages 190–193, 2001.

[64] K. Suzuki, G. Mito, H. Kawamoto, Y. Hasegawa, and Y. Sankai. Intention-based walking support for paraplegia patients with Robot Suit HAL. *Advanced Robotics*, 21 (12):1441–1469, April 2007.

[65] Y. Sankai. HAL: Hybrid assistive limb based on cybernics. In *Robotics Research*, pages 25–34, 2010.

[66] Y. Ohta, H. Yano, R. Suzuki, M. Yoshida, N. Kawashima, and K. Nakazawa. A two-degree-of-freedom motor-powered gait orthosis for spinal cord injury patients. *Proceedings of the Institution of Mechanical Engineers, Part H: Journal of Engineering in Medicine*, 221(6):629–639, August 2007.

[67] E. Strickland. Good-bye, Wheelchair, Hello Exoskeleton. *IEEE Spectrum*, 1:30–32, December 2011.

[68] G. J. Androwis and K. J. Nolan. Evaluation of a robotic exoskeleton for gait training

in acute stroke. *Archives of Physical Medicine and Rehabilitation*, 97(10):e115–e116, October 2016.

[69] A. Esquenazi, M. Talaty, A. Packel, and M. Saulino. The ReWalk powered exoskeleton to restore ambulatory function to individuals with thoracic-level motor-complete spinal cord injury. *American Journal of Physical Medicine & Rehabilitation*, 91(11): 911–921, November 2012.

[70] A. Esquenazi. New bipedal locomotion options for individuals with thoracic level motor complete spinal cord injury. *Journal of the Spinal Research Foundation*, 8(1): 26–28, 2013.

[71] ReWalk - More than Walking. `http://rewalk.com`. Accessed: 2017-01-28.

[72] R. J. Farris, H. A Quintero, S. A. Murray, , K. H. Ha, C. Hartigan, and M. Goldfarb. A preliminary assessment of legged mobility provided by a lower limb exoskeleton for persons with paraplegia. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 22(3):482–490, May 2014.

[73] J.-Y. Jung, W. Heo, H. Yang, and H. Park. A neural network-based gait phase classification method using sensors equipped on lower limb exoskeleton robots. *Sensors*, 15:27738–27759, 10 2015.

[74] K. Kiguchi and Y. Imada. EMG-based control for lower-limb power-assist exoskeletons. In *2009 IEEE Workshop on Robotic Intelligence in Informationally Structured Space*, pages 19–24, March 2009.

[75] M. Sartori, M. Reggiani, D. Farina, and D. G. Lloyd. EMG-driven forward-dynamic estimation of muscle force and joint moment about multiple degrees of freedom in the human lower extremity. *Gribble PL, ed. PLoS ONE*, 7:e52618, December 2012.

[76] D. G. Barth, L. Schumacher, and S. Sienko Thomas. Gait analysis and energy cost of below-knee amputees wearing six different prosthetic feet. *JPO: Journal of Prosthetics and Orthotics*, 4(2):63–75, 1992.

[77] D. H. Nielsen, D. G. Shurr, J. C. Golden, and K. Meier. Comparison of energy cost and gait efficiency during ambulation in below-knee amputees using different prosthetic feet-A preliminary report. *JPO: Journal of Prosthetics and Orthotics*, 1(1):24–31, October 1988.

[78] J.-M. Casillas, V. Dulieu, M. Cohen, I. Marcer, and J.-P. Didier. Bioenergetic comparison of a new energy-storing foot and SACH foot in traumatic below-knee vascular amputations. *Archives of Physical Medicine and Rehabilitation*, 76(1):39–44, January 1995.

[79] P. L. Ephraim, S. T. Wegener, E. J. MacKenzie, T. R. Dillingham, and L. E. Pezzin. Phantom pain, residual limb pain, and back pain in amputees: Results of a national survey. *Archives of Physical Medicine and Rehabilitation*, 86(10):1910–1919, October 2005.

[80] P. A. Struyf, C. M. van Heugten, M. W. Hitters, and R. J. Smeets. The prevalence of osteoarthritis of the intact hip and knee among traumatic leg amputees. *Archives of Physical Medicine and Rehabilitation*, 90(3):440–446, March 2009.

[81] E. D. Lemaire and F. R. Fisher. Osteoarthritis and elderly amputee gait. *Archives of Physical Medicine and Rehabilitation*, 75(10):1094–1099, October 1994.

PART I.

Perturbed Walking Experiment Design and

Inertial Compensation

CHAPTER II

AN ELABORATE DATA SET ON HUMAN GAIT AND
THE EFFECT OF MECHANICAL PERTURBATIONS

Our approach to designing a more bio-inspired controller is to use human experimental data collected from able-bodied subjects. By recording kinematic and kinetic gait data, we can design our controller to reproduce the subject's joint torque pattern, which would operate the electric motors of a prosthesis or exoskeleton. However, tuning a controller based on an average gait cycle may not provide enough insight into the human control system, especially in response to an outside disturbance. Instead, in our experiments we add mechanical perturbations to an average walking speed with the hope our controller can mimic the respective reactions of test subjects. In this chapter, we describe our experimental study in great detail, emphasizing how we made this dataset freely available to the public.

ABSTRACT

Here we share a rich gait dataset collected from fifteen subjects walking at three speeds on an instrumented treadmill. Each trial consists of 120 seconds of normal walking and 480 seconds of walking while being longitudinally perturbed during each stance phase with pseudo-random fluctuations in the speed of the treadmill belt. A total of approximately 1.5 hours of normal walking (> 5000 gait cycles) and 6 hours of perturbed walking (> 20,000 gait cycles) is included in the dataset. We provide full-body marker trajectories and ground reaction loads in addition to a presentation of processed data that includes gait events, 2D joint angles, angular rates, and joint torques along with the open source software used for the computations. The protocol is described in detail and supported with additional elaborate meta data for each trial. This data can likely be useful for validating or generating mathematical models that are capable of simulating normal periodic gait and non-periodic, perturbed gaits.

2.1 Introduction

The collection of dynamical data during human walking has a long history beginning with the first motion pictures and now with modern marker-based motion capture techniques and high fidelity ground reaction load measurements. Even though years of data on thousands of subjects now exist, this data is not widely disseminated, well organized, nor available with few or no restrictions. David Winter's published normative gait data [1] is widely used in biomechanical studies, yet it comes from few subjects and only a small number of gait cycles per subject. This small source has successfully enabled many other studies, such as powered prosthetic control design [2], but success in other research fields using large sets of data for discovery lead us to believe that more elaborate datasets may benefit the field of human motion studies. To enable such work, biomechanical data needs to be shared extensively, organized, and curated to enable future analysts.

25

There are some notable gait data sets and databases besides Winter's authoritative set that are publicly available. The International Society of Biomechanics has maintained a web page (http://isbweb.org/data) since approximately 1995 that includes data sets for download and mostly unencumbered use. For example, the kinematic and force plate measurements from several subjects from [3] is available on the site. Another website, the CGA Normative Gait Database, [4] curates and shares normative clinical gait data collected from multiple labs and these datasets have influenced other studies. For example, [5] made use of the average gait cycles from the child subjects.

[6] report on a large gait database comparison where one database contained kinematic data of 409 gait cycles of children from 1 to 7 years old but the data does not seem to be publicly available. This is unfortunately typical. [7] recognized the need for a comprehensive database for clinical gait data and created the Gaitabase. This database may contain a substantial amount of data, but it is encumbered by a complicated and restrictive license and sharing scheme. [8] provides lower body kinematic data of single gait cycles from over one hundred subjects extracted from the large KIST Human Gait Pattern Data database which may also include a substantial amount of raw data, but it is private. However, there are examples of data with less restrictions. The University of Wisconsin at LaCrosse has an easily accessible normative gait data set [9] from 25 subjects with lower extremity marker data from multiple gait cycles and force plate measurements from a single gait cycle. The CMU Graphics Lab Motion Capture Database [10] is also a good example and contains full-body marker kinematics for a fair number of trials with a small number of gait cycles during both walking and running.

More recent examples of biomechanists sharing their data alongside publications are: [11] which includes full body joint kinematics and kinetics from eleven subjects walking on an instrumented treadmill and [12] who include a larger set of data from ten subjects walking for five minutes each at three different speeds, but only a small set of lower extremity markers are present. The second is notable because they publish the data in Dryad, a

modern citable data repository. It is also worth noting purely visual data collections of gait, like the one presented in [13], which contain videos of subjects walking on a treadmill in full clothing. This database is also, unfortunately, tightly secured with an extensive release agreement for reuse.

The amount of publicly available gait data is small compared to the number of gait studies that have been performed over the years. The data that is available generally suffers from limitations such as few subjects, few gait cycles, few markers, highly clinical, no raw data, limited force plate measurements, lack of meta data, non-standard formats, and restrictive licensing. To help with this situation, we are making the data we collected for our research purposes publicly available and free of the previously mentioned deficiencies. Not only do we provide a larger set of normative gait data that has been previously available, we also include an even larger set of data in which the subject is being perturbed, something that does not currently exist.We believe both of these sets of data can serve a variety of use cases and hope that we can save time and effort for future researchers by sharing it.

But our reasons are not entirely altruistic, as governments and granting agencies are also encouraging researchers to share data with recent policy changes. For example, the European Commission [14] has outlined publicly funded data's role in innovation and the White House [15] laid out a plan for public access to publications and data in 2013. The National Science Foundation, which partially funds this work, was ahead of the White House and required all grants to include a data management plan in 2011. This work is a partial fulfillment of the grant requirements laid out in our grant's data management plan and we hope that this work can be a good model for dissemination of biomechanical data.

Our use case for the data is centered around the need for bio-inspired control systems in emerging powered prosthetics and orthotics. Ideally, a powered prosthetic would behave in such a way that the user would feel like their limb was never disabled. There are a variety of approaches to developing bio-inspired control systems, some of which aim to mimic the reactions and motion of an able-bodied person. A modern gait lab is able to collect a

27

variety of kinematic, kinetic, and physiological data from humans during gait. This data can potentially be used to drive the design of the human-mimicking controller. With a rich enough data set, one may be able to identify control mechanisms used during a human's natural gait and recovery from perturbations. We hypothesize that by forcing the human to recover from external perturbations, the resulting reactive actions can be used along with system identification techniques to expose the feedback related relationships among the human's sensors and actuators. With this in mind, we have collected data that is richer than previous gait data sets and may be rich enough for control identification. The data can also be used for verification purposes for controllers that have been designed in other manners, such as those constructed from first principles, e.g. [16].

With all of this in mind, we collected over seven and a half hours of gait data from fifteen able bodied subjects which amounts to over 25,000 gait cycles [17]. The subjects walked at three different speeds on an instrumented treadmill while we collected full-body marker locations and ground reaction loads from a pair of force plates. The final protocol for the majority of the trials included two minutes of normal walking and eight minutes of walking under the influence of pseudo-random belt speed fluctuations. The data has been organized, complete with rich meta data, and made available in the most unrestrictive form for other research uses following modern best practices in data sharing [18].

Furthermore, we include a small Apache licensed open source software library for basic gait analysis. The combination of the open data and open software allow the results presented within to be computationally reproducible and instructions are included in the associated repository [1] for reproducing the results.

2.2 Methods

In this section, we describe our experimental design beginning with descriptions of the participants and equipment. This is then followed by the protocol details and specifics on

---

[1] https://github.com/csu-hmc/perturbed-data-paper

Table I: Information about the 15 study participants in order of collection date. The subjects are divided into those that were used for the protocol pilot trials, i.e. the first three, and those used for the final protocol. The final three columns provide the trial numbers associated with each nominal treadmill speed. The measured mass is computed from the mean total vertical ground reaction force just after the calibration pose event, if possible. If the mass is reported without an accompanying standard deviation, it is the subject's self-reported mass. Additional trials found in the data set with a subject identification number 0 are trials with no subject, i.e. unloaded trials that can be used for inertial compensation purposes, and are not shown in the table.

Id	Gender	Age [yr]	Height [m]	Mass [kg]	0.8 m/s	1.2 m/s	1.6 m/s
1	male	25	1.87	101	NA	6, 7, 8	NA
11	male	22	1.85	80	9	10	11
4	male	30	1.76	74	12, 15	13	14
7	female	29	1.72	64.5 ± 0.8	16	17	18
8	male	20	1.57	74.9 ± 0.9	19	20	21
9	male	20	1.69	67 ± 2	25	26	27
5	male	23	1.73	71.2 ± 0.9	32	31	33
6	male	26	1.77	86.8 ± 0.6	40	41	42
3	female	32	1.62	54 ± 2	46	47	48
12	male	22	1.85	74.2 ± 0.5	49	50	51
13	female	21	1.70	58 ± 2	55	56	57
10	male	19	1.77	92 ± 2	61	62	63
15	male	22	1.83	80.5 ± 0.8	67	68	69
17	male	23	1.86	88.3 ± 0.8	73	74	75
16	female	28	1.69	56.2 ± 0.6	76	77	78

the perturbation design.

Participants

Fifteen able bodied subjects including four females and eleven males with an average age of 24 ± 4 years, height of 1.75 ± 0.09 m, mass of 74 ± 13 kg participated in the study. The study was approved by the Institutional Review Board of Cleveland State University (# 29904-VAN-HS) and written informed consent was obtained from all participants. The data has been anonymized with respect to the participants' identities and a unique identification number was assigned to each subject. A selection of the meta data collected for each subject is shown in Table I.

Equipment

The data were collected in the Parker Hannifin Laboratory for Human Motion and Control at Cleveland State University, using the following equipment:

- A R-Mill treadmill which has dual 6 degree of freedom force plates, independent belts for each foot, along with lateral translation and pitch rotation capabilities (Forcelink, Culemborg, Netherlands).

- A 10 Osprey camera motion capture system paired with the Cortex 3.1.1.1290 software (Motion Analysis, Santa Rosa, CA, USA).

- USB-6255 data acquisition unit (National Instruments, Austin, Texas, USA).

- Four ADXL330 Triple Axis Accelerometer Breakout boards attached to the treadmill (Sparkfun, Niwot, Colorado, USA).

- D-Flow software (versions 3.16.1 to 3.16.2) and visual display system, (Motek Medical, Amsterdam, Netherlands).

The Cortex software delivers high accuracy 3D marker trajectories from the cameras along with data from the force plates and analog sensors (e.g. EMG/Accelerometer) through a National Instruments USB-6255 data acquisition unit. D-Flow then receives streaming data from Cortex and any other digital sensors. It is also responsible for controlling the treadmill's motion (lateral, pitch, belts). D-Flow can process the data in real time and/or export data to file.

Our motion capture system's coordinate system is such that the X coordinate points to the right, the Y coordinate points upwards, and the Z coordinate follows from the right-hand-rule, (i.e. points backwards with respect to the walking direction). The camera's coordinate system is aligned to an origin point on the treadmill's surface during camera calibration. The same point is used as the origin of the ground reaction force measuring system. Figure 4 shows the layout of the equipment.

Early on, we discovered that the factory setup of the R-Mill treadmill had a vibration mode as low as 5 Hz that was detectable in the force measurements; likely due to the flexible undercarriage and pitch motion mechanism. Trials 6–8 are affected by this vibration mode. During trials 9–15 the treadmill was stabilized with wooden blocks. During the remaining trials (> 15) the treadmill was stabilized with metal supports; both with ones fabricated in-house and ones supplied by the vendor. These supports aimed to improve the stiffness and frequency response of the force plate system.

The acceleration of the treadmill base was measured during each trial by the ADXL330 accelerometers placed at the four corners of the machine. These accelerometers were intended to provide information for inertial compensation purposes when the treadmill moved laterally or in pitch, but are extraneous for trials greater than number 8 due to the treadmill being stabilized in those degrees of freedom by the aforementioned supports.
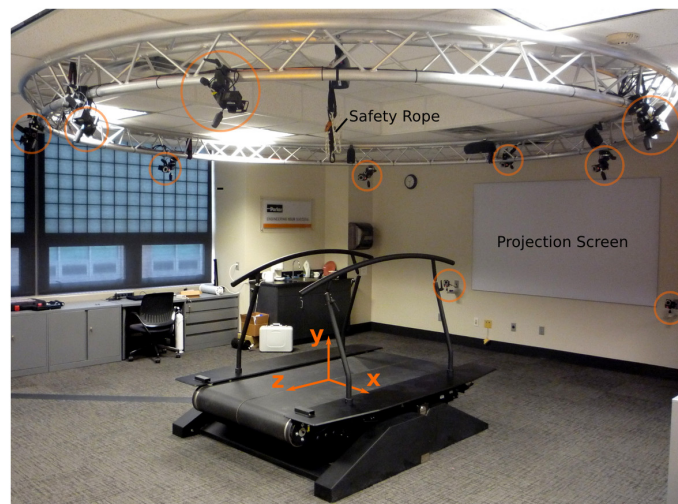
Figure 4: The treadmill with coordinate system, cameras (circled in orange), projection screen, and safety rope. The direction of travel is in the $-z$ direction.

Protocol

The experimental protocol consisted of both static measurements and walking on the treadmill for 10 minutes under unperturbed and perturbed conditions. Before a set of trials on the same day, the motion capture system was calibrated using the manufacturer's recom-

mended procedure. Before each subject's gait data were collected, the subject changed into athletic shoes, shorts, a sports bra, a baseball cap [2], and a rock climbing harness. All 47 markers were applied directly to the skin at the landmarks noted in Table II except for the heel, toe, and head markers, which were placed on the respective article of clothing. [3] Then the subject self-reported their age, gender, and mass. Finally, their height was measured by the experimentalist and four reference photographs (front, back, right, left) were taken of subject's marker locations.

After obtaining informed consent and a briefing by the experimentalist on the trial protocol, the subject followed the verbal instructions of the experimentalist and the on-screen instructions from the video display. The final protocol for a single trial was as follows:

1. The subject stepped onto the treadmill and markers were identified with Cortex.

2. The safety rope was attached loosely to the rock climbing harness such that no forces were acting on the subject during walking, but so that the harness would prevent a full fall.

3. The subject started by stepping on sides of treadmill so that feet did not touch the force plates and the force plate signals are zeroed. This corresponds to the "Force Plate Zeroing" event.

4. Once notified by the video display, the subject stood in the calibration pose: standing straight up, looking forward, arms out by their sides (approximately 45 degree abduction) and the event, "Calibration Pose", was manually recorded by the operator.

5. A countdown to the first normal walking phase was displayed. At the end of the countdown the event "First Normal Walking" was recorded and the treadmill ramped up to the specified speed and the subject was instructed to walk normally, to focus on the "endless" road on the display, and not to look at their feet.

---

[2] A cap was used to avoid having to shave participants' hair at the expense of accuracy.
[3] The sacrum and rear pelvic markers were placed on the shorts for a small number of the subjects.

Table II: Descriptions of the 47 subject markers used in this study. The "Set" column indicates whether the marker exists in the lower and/or full body marker set. The label column matches the column headers in the *mocap-xxx.txt* and/or the marker map in the *meta-xxx.yml* file

Set	#	Label	Name	Description
F	1	LHEAD	Left head	Just above the ear, in the middle.
F	2	THEAD	Top head	On top of the head, in line with the LHEAD and RHEAD.
F	3	RHEAD	Right head	Just above the ear, in the middle.
F	4	FHEAD	Forehead	Between line LHEAD/RHEAD and THEAD a bit right from center.
L/F	5	C7	C7	On the 7th cervical vertebrae.
L/F	6	T10	T10	On the 10th thoracic vertbrae.
L/F	7	SACR	Sacrum bone	On the sacral bone.
L/F	8	NAVE	Navel	On the navel.
L/F	9	XYPH	Xiphoid process	Xiphoid process of the sternum.
F	10	STRN	Sternum	On the jugular notch of the sternum.
F	11	BBAC	Scapula	On the inferior angle of the right scapula.
F	12	LSHO	Left shoulder	Left acromion.
F	13	LDELT	Left deltoid muscle	Apex of the deltoid muscle.
F	14	LLEE	Left lateral elbow	Left lateral epicondyle of the elbow.
F	15	LMEE	Left medial elbow	Left medial epicondyle of the elbow.
F	16	LFRM	Left forearm	On 2/3 on the line between the LLEE and LMW.
F	17	LMW	Left medial wrist	On styloid process radius, thumb side.
F	18	LLW	Left lateral wrist	On styloid process ulna, pinky side.
F	19	LFIN	Left fingers	Center of the hand. Caput metatarsal 3.
F	20	RSHO	Right shoulder	Right acromion.
F	21	RDELT	Right deltoid muscle	Apex of deltoid muscle.
F	22	RLEE	Right lateral elbow	Right lateral epicondyle of the elbow.
F	23	RMEE	Right medial elbow	Right medial epicondyle of the elbow.
F	24	RFRM	Right forearm	On 1/3 on the line between the RLEE and RMW.
F	25	RMW	Right medial wrist	On styloid process radius, thumb side.
F	26	RLW	Right lateral wrist	On styloid process ulna, pinky side.
F	27	RFIN	Right fingers	Center of the hand. Caput metatarsal 3.
L/F	28	LASIS	Pelvic bone left front	Left anterior superior iliac spine.
L/F	29	RASIS	Pelvic bone right front	Right anterior superior iliac spine.
L/F	30	LPSIS	Pelvic bone left back	Left posterior superio iliac spine.
L/F	31	RPSIS	Pelvic bone right back	Right posterior superior iliac spine.
L/F	32	LGTRO	Left greater trochanter of the femur	On the center of the left greater trochanter.
L/F	33	FLTHI	Left thigh	On 1/3 on the line between the LFTRO and LLEK.
L/F	34	LLEK	Left lateral epicondyle of the knee	On the lateral side of the joint axis.
L/F	35	LATI	Left anterior of the tibia	On 2/3 on the line between the LLEK and LLM.
L/F	36	LLM	Left lateral malleoulus of the ankle	The center of the heel at the same height as the toe.
L/F	37	LHEE	Left heel	Center of the heel at the same height as the toe.
L/F	38	LTOE	Left toe	Tip of big toe.
L/F	39	LMT5	Left 5th metatarsal	Caput of the 5th metatarsal bone, on joint line midfoot/toes.
L/F	40	RGTRO	Right greater trochanter of the femur	On the center of the right greater trochanter.
L/F	41	FRTHI	Right thigh	On 2/3 on the line between the RFTRO and RLEK.
L/F	42	RLEK	Right lateral epicondyle of the knee	On the lateral side of the joint axis.
L/F	43	RATI	Right anterior of the tibia	On 1/3 on the line between the RLEK and RLM.
L/F	44	RLM	Right lateral malleoulus of the ankle	The center of the heel at the same height as the toe.
L/F	45	RHEE	Right heel	Center of the heel at the same height as the toe.
L/F	46	RTOE	Right toe	Tip of big toe.
L/F	47	RMT5	Right 5th metatarsal	Caput of the 5th metatarsal bone, on joint line midfoot/toes.

6. After 1 minute of normal walking, the longitudinal perturbation phase begun and was recorded as "Longitudinal Perturbation".

7. After 8 minutes of walking under the influence of the perturbations, the second normal walking phase begun and was recorded as "Second Normal Walking".

8. After 1 minute of normal walking, a countdown was shown on the display and the treadmill decelerated to a stop.

9. The subject was instructed to step off of the force plates for 10 seconds and the "Unloaded End" event was recorded.

10. The subject could then take a rest break before each additional trial.

Perturbation Signals

As previously described, the protocol included a phase of normal walking, followed by longitudinal belt speed perturbations, and ended with a second segment of normal walking. Three pseudo-random belt speed control signals, with mean velocities of $0.8\,\mathrm{m\,s^{-1}}$, $1.2\,\mathrm{m\,s^{-1}}$ and $1.6\,\mathrm{m\,s^{-1}}$, were pre-generated with MATLAB® and Simulink® (Mathworks, Natick, Massachusetts, USA) and are available for download from Zenodo [19]. The same control signal was used for all trials at that given speed.

To create the signals, we started by generating random acceleration signals, sampled at 100 Hz, using the Simulink discrete-time Gaussian white noise block followed by a saturation block set at the maximum belt acceleration of $15\,\mathrm{m\,s^{-2}}$. The signal was then integrated to obtain belt speed and high-pass filtered with a second-order Butterworth filter to eliminate drift. One of the three mean speeds were then added to the signal and limited between $0\,\mathrm{m\,s^{-1}}$ to $3.6\,\mathrm{m\,s^{-1}}$. The cutoff frequencies of the high-pass filter, as well as the variance in the acceleration signal, were manually adjusted until acceptable standard deviations for each mean speed were obtained: $0.06\,\mathrm{m\,s^{-1}}$, $0.12\,\mathrm{m\,s^{-1}}$ and $0.21\,\mathrm{m\,s^{-1}}$ for the three speeds, respectively. These ensured that the test subjects were sufficiently perturbed at each speed

while remaining within the limits of our equipment and testing protocol. To ensure that the treadmill belts could accelerate to the desired values, the high performance mode in the D-Flow software was enabled. The MATLAB script and Simulink model produce a comma-delimited text file with the desired belt speed signals indexed by the time stamp.

Figure 5 gives an idea of the effect of the treadmill and controller dynamics by plotting the measured speed of the treadmill belts from loaded trials (76, 77, 78) against the commanded treadmill control input signal. The system introduces a delay and seems to act as a low pass filter. The standard deviations of the measured speeds do not significantly differ from those of the commanded speeds: $0.05\,\mathrm{m\,s^{-1}}$, $0.12\,\mathrm{m\,s^{-1}}$ and $0.2\,\mathrm{m\,s^{-1}}$ for the three speeds, respectively.
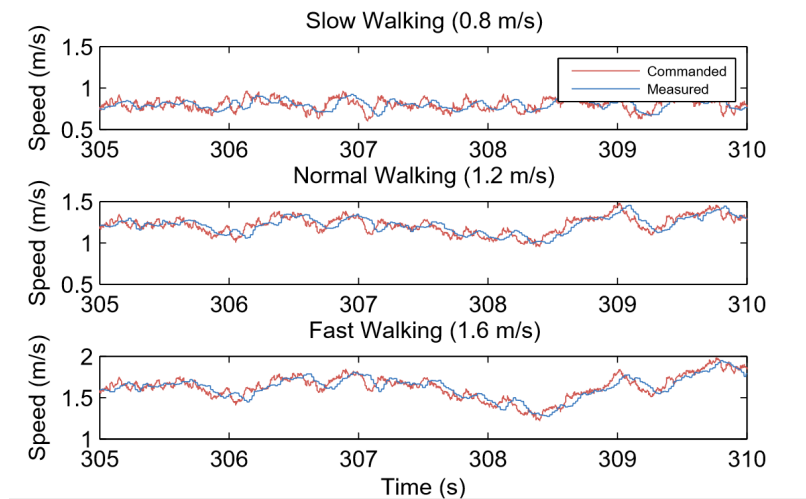
Figure 5: Commanded treadmill belt speed (red) and the recorded speed (blue) for average belt speeds of $0.8\,\mathrm{m\,s^{-1}}$, $1.2\,\mathrm{m\,s^{-1}}$ and $1.6\,\mathrm{m\,s^{-1}}$, respectively.

Figure 6 gives a frequency domain view of the effects of the treadmill dynamics. These spectral density plots were created by averaging a spectrogram of a twenty second Hamming window. For all speeds, the frequency content of the commanded and measured time series show similarity below 4 Hz and attenuation in the measured spectral density above 4 Hz.

When belt speed is not constant, the inertia of the rollers and motor will likely induce error in the force plate x axis moment, and hence, the anterior-posterior coordinate (z axis)
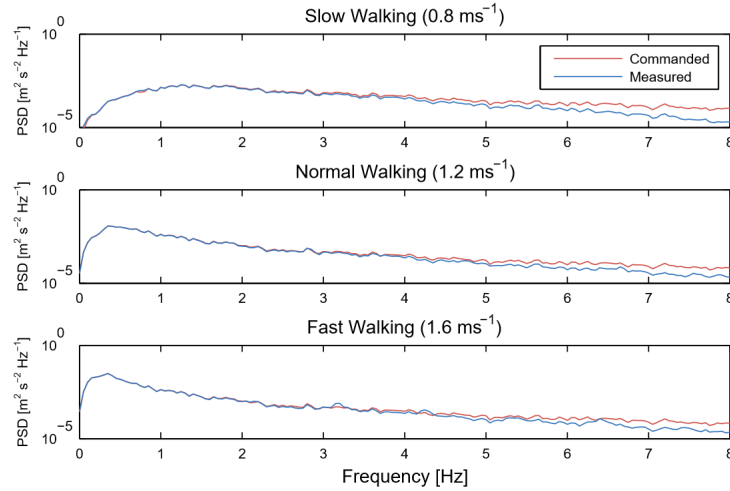
Figure 6: Power spectral density of the commanded treadmill belt speed (red) and the recorded speed (blue) for average belt speeds of $0.8\,\mathrm{m\,s^{-1}}$, $1.2\,\mathrm{m\,s^{-1}}$ and $1.6\,\mathrm{m\,s^{-1}}$, respectively.

of the center of pressure that is measured by the instrumentation in the treadmill. This error may or may not be pertinent to different analyses. If needed, this error can be partially compensated by a linear model as shown in [20], described in more detail in Part I, Chapter III. The model coefficients can be identified from the unloaded trials given in Table III. The error due to inertia is random and does not affect the averaged joint moments presented in Figure 8. Compensation should, however, be done if joint moments from individual gait cycles are of interest rather than the ensemble average.

Table III: A list of unloaded trials collected for each speed. Each loaded trial includes a compensation file listed in its meta data which matches it to these unloaded trials.

Speed	Trial Numbers
0.8 m/s	22, 30, 34, 43, 52, 58, 64, 70, 79
1.2 m/s	3, 4, 5, 23, 29, 35, 44, 53, 59, 65, 71, 80
1.6 m/s	24, 28, 36, 45, 54, 60, 66, 72, 81

In addition to the longitudinal perturbations, lateral perturbations were also prescribed for a duration of four minutes in the pilot trials 3–8. Figure 7 shows an example of the measured lateral deviation of the treadmill base. These signals were generated in a similar manner using MATLAB and Simulink in which a Gaussian white noise block was twice

integrated to obtain the lateral deviation. The signal was then high-pass filtered with a second-order Butterworth filter to eliminate drift and then saturated so that the signal remained within the 5 cm lateral range of the physical hardware. The same perturbation signal was used for each of the three trials.
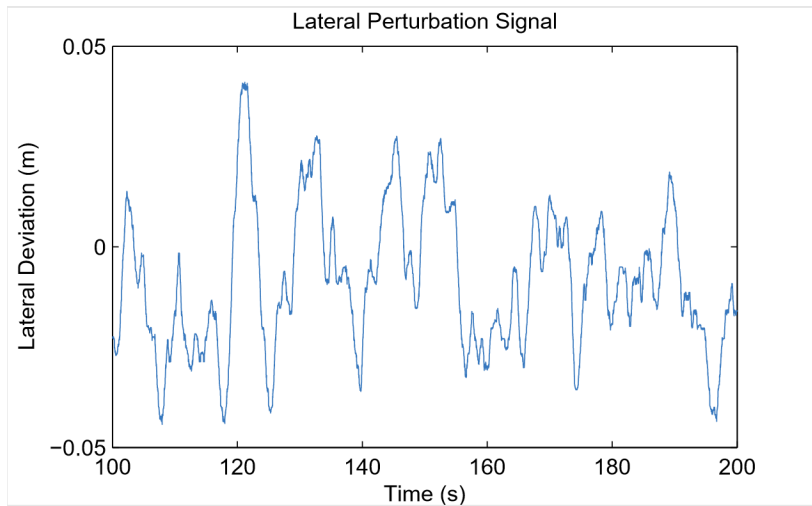
Figure 7: The measured lateral deviation of the treadmill base from trial 6.

2.3 Results

Here we present some basic results. We first provide a detailed description of the raw data followed by an overview of several computed variables that give an idea of the characteristics of both the unperturbed and perturbed gait.

mocap-xxx.txt

The output from the D-Flow mocap module is stored in a tab separated value (TSV) file named `mocap-xxx.txt` where `xxx` represents the trial id number. The file contains a number of time series. The numerical values of the time series are provided in decimal fixed point notation with 6 decimals of precision, e.g. `123456.123456`, regardless of the units. The first line of the file holds the header. The header includes time stamp column, frame number column, marker position columns, force plate force/moment columns, force

plate center of pressure columns, and other analog columns.

We make use of the full-body, 47-marker set described in [11] and presented in detail in Table II. As with all camera based motion capture systems, the markers sometimes disappear in the recording. When a marker goes missing, if the data was recorded in a D-Flow version less than 3.16.2rc4, D-Flow continues to record the last non-missing value in all three axes until the marker is visible again. In D-Flow versions greater than or equal to 3.16.2rc4, the missing markers are indicated in the TSV file as either `0.000000` or `-0.000000`. The D-Flow version must be provided in the meta data YAML file to be able to distinguish this detail.

record-xxx.txt

The record module also outputs a tab delimited ASCII text file with numerical values at six decimal digits. It includes a `Time` column which records the D-Flow system time in seconds. This time corresponds to the time recorded in the `TimeStamp` column in mocap module TSV file which is necessary for time synchronization. There are two additional columns `RightBeltSpeed` and `LeftBeltSpeed` which provide the independent belt speeds measured in meters per second by a factory installed encoder in the treadmill.

Additionally, the record module is capable of recording the time at which various preprogrammed events occur, as detected or set by D-Flow. It does this by inserting commented (#) lines in between the rows when the event occurred. The record files have several events that delineate the different phases of the protocol:

A: Force Plate Zeroing Marks the time at the beginning of the trial at which there is no load on the force plates and when the force plate voltages were zeroed.

B: Calibration Pose Marks the time at which the person is in the calibration pose.

C: First Normal Walking Marks the time when the treadmill begins Phase 1: constant belt speed.

D: Longitudinal Perturbation Marks the time when the treadmill begins Phase 2: longitudinal perturbations in the belt speed.

E: Second Normal Walking Marks the time when phase 3 starts: constant belt speed.

F: Unloaded End Marks the time at which there is no load on the force plates and the belts are stationary.

meta-xxx.yml

Each trial directory contains a meta data file in the YAML format named in the following style `meta-xxx.yml` where `xxx` is the three digit trial identification number. There are three main headings in the file: `study`, `subject`, and `trial`. The `study` section contains identifying information for the overall study, an identification number, name, and description.

The `subject` section provides key value pairs of information about the subject in that trial. Each subject has a unique identification number along with basic anthropomorphic data. Possible meta data for the subject includes: age, ankle widths, knee widths, identification number, gender, height, and mass.

The `trial` section contains the information about the particular trial. Each trial has a unique identification number along with a variety of other information, including: trial number, analog channel names, marker names and marker set, software versions, date of trial, events in the trial, associated files, hardware settings, speed of treadmill, position of treadmill platform, and any notes regarding the experiment.

Processed Data

We developed a toolkit for data processing (GaitAnalysisToolKit v0.1.2, [21]) for common gait computations and provide an example processed trial to present the nature of the data. The tool was developed in Python, is dependent on the SciPy Stack (NumPy [22], SciPy [23], matplotlib [24], Pandas [25], etc.) and Octave [26], and provides two main

classes: one to do basic gait data cleaning from D-Flow's output files, `DFlowData`, and a second to compute common gait variables of interest, `GaitData`.

The `DFlowData` class collects and stores all the raw data presented in the previous section and applies several "cleaning" operations to transform the data into a usable form. The cleaning process follows these steps:

1. Load the meta data file into a Python dictionary.

2. Load the D-Flow mocap module TSV file into Pandas `DataFrame`.

3. Relabel the column headers to more meaningful names if this is specified in the meta data.

4. Optionally identify the missing values in the mocap marker data and replace them with `numpy.nan`.

5. Optionally interpolate the missing marker values and replaces them with interpolated estimates using a variety of interpolation methods.

6. Load the D-Flow record module TSV file into a Pandas `DataFrame`.

7. Extract the events and create a dictionary mapping the event names in the meta data to the events detected in the record module file.

8. Inertially compensate the ground reaction loads based on whether the meta data indicates there was treadmill motion.

9. Merge the data from the mocap module and record module into one data frame at the maximum common constant sample rate.

Once the data is cleaned there are two methods that allow the user to extract the cleaned data: either extract sections of the data bounded by the events recorded in the `record-xxx.txt` file or save the cleaned data to disk. These operations are available as a command line application and as an application programming interface (API) in Python.
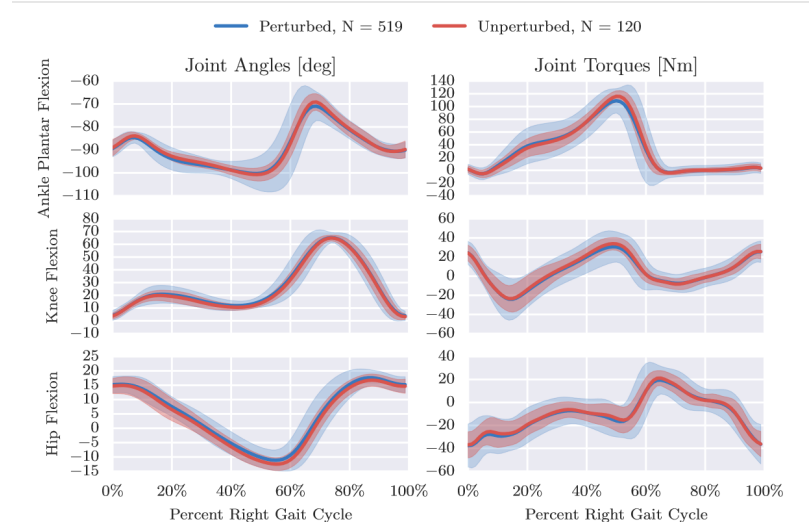
Figure 8: Right leg mean and 3σ (shaded) joint angles and torques from both unperturbed (red) and perturbed (blue) gait cycles, where N is the number of gait cycles, from trial 20. We define the nominal configuration, i.e. all joint angles equal to zero, such that the vectors from the shoulder to the hip, the hip to the knee, the knee to the ankle, and the heel to the toe are all aligned.

The `GaitData` class is then used to compute gait events (toe off and heel strike times), basic 2D inverse kinematics and dynamics based on Winter's method [1], and to store the data into a Pandas `Panel` with each gait cycle on the item axis at a specified sampling rate. This object can also be serialized to disk in HDF5 format.

This work flow was used to produce Figure 8 which compares the mean and standard deviation of sagittal plane joint angles and torques from the perturbed gait cycles to the unperturbed gait cycles computed from trial 20, where N is the number of gait cycles. This gives an idea of the more highly variable dynamics required to walk while being longitudinally perturbed.

For more insight into the difference in the unperturbed and perturbed data, Figure 9 compares the distribution of a few gait cycle statistics. One can see that the perturbed strides have a much larger variation in frequency and length and even larger variation in stride width. It is also interesting to note that the coupled nature of the system's degrees of freedom can be exploited to increase the stride width with only longitudinal perturbations, although not relatively as much as is in the other statistics.
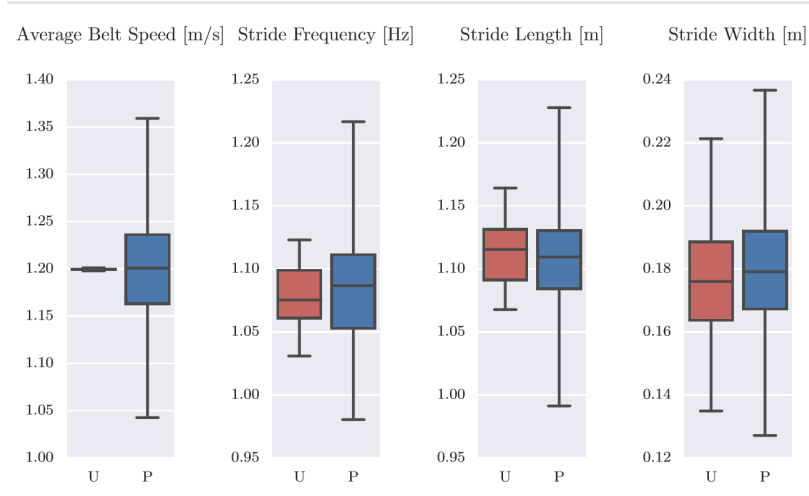
Figure 9: Box plots of the average belt speed, stride frequency, stride length, and stride width which compare 120 unperturbed (U: red) and 519 perturbed (P: blue) gait cycles. The median is given with the box bounding the first and third quartiles and the whiskers bound the range of the data.

Data Limitations

The data is provided in good faith with great attention to detail, but as with all data, there are anomalies that may affect the use and interpretation of results emanating from the data. We have selected 10 subjects (30 trials) whose data would be best to use for system identification, shown in Table IV.

Table IV: A list of the 10 subjects selected, where trial data is free from abnormalities or experimental errors

Subject Number	Trial Number	Subject Number	Trial Number
7	16, 17, 18	8	19, 20, 21
5	31, 32, 33	6	40, 41, 42
3	46, 47, 48	12	49, 50, 51
13	55, 56, 57	10	61, 62, 63
15	67, 68, 69	17	73, 74, 75

2.4 Conclusion

We have presented a rich and elaborate data set of motion and ground reaction loads from human subjects during both normal walking and when recovering from perturbations. The

raw data is provided for reuse with complete meta data. In addition to the data, we provide software that can process the data for both cleaning purposes and to produce typical sagittal plane gait variables of interest. Among other uses, we believe the dataset is ideally suited for control identification purposes. Many researchers are working on mathematical models for control in gait and this dataset provides both a way to validate these models and a source for generating them.

Data Availability

The data set [17] is available via two Zenodo data repositories. The first holds two approximately 1.2GB gzipped tar balls which contain the main data and a `README` file with a short description of the contents. The second [19] contains the Simulink generated data for the treadmill control inputs. The data is released under the Creative Commons CC0 license (http://creativecommons.org/about/cc0) following best practices for sharing scientific data.

Software Availability

The tables, figures, and the paper can be reproduced from the source repository shared on Github: https://github.com/csu-hmc/perturbed-data-paper. Along with the source code in the repository, the computations depend on version 0.1.2 of the GaitAnalysisToolKit [21] which can be downloaded from Zenodo or the Python Package Index (http://pypi.python.org).

Acknowledgments

2.5 REFERENCES

[1] D. A. Winter. *Biomechanics and Motor Control of Human Movement*. John Wiley and Sons, Inc, 2nd edition, 1990.

[2] F. Sup, A. Bohara, and M. Goldfarb. Design and control of a powered transfemoral prosthesis. *International Journal of Robotics Research*, 27(2):263–273, February 2008.

[3] C. L. Vaughan, B. L. Davis, and J. C. O'Connor. *Dynamics of Human Gait*. Human Kinetics Publishers, 1st edition, 1992.

[4] C. Kirtley. CGA Normative Gait Database, 2014. URL `http://www.clinicalgaitanalysis.com/data/`.

[5] A. J. Van den Bogert. Exotendons for assistance of human locomotion. *BioMedical Engineering OnLine*, 2(17), October 2003.

[6] V. L. Chester, M. Tingley, and E. N. Biden. Comparison of two normative paediatric gait databases. *Dynamic Medicine*, 6(8), July 2007.

[7] O. Tirosh, R. Baker, and J. McGinley. GaitaBase: Web-based repository system for gait analysis. *Computers in Biology and Medicine*, 40(2):201–207, February 2010.

[8] Y. Yun, H.-C. Kim, Sung Y. Shin, J. Lee, A. D. Deshpande, and C. Kim. Statistical method for prediction of gait kinematics with Gaussian process regression. *Journal of Biomechanics*, 47(1):186–192, January 2014.

[9] J. D. Willson and T. Kernozek. Gait data collected at University of Wisconsin-LaCrosse, 2014. URL `http://www.innsport.com/related-products/data-sets/uw-l-gait-data-set.aspx`.

[10] J. Hodgins. CMU Graphics Lab Motion Capture Database, 2015. URL `http://mocap.cs.cmu.edu`.

[11] A. J. van den Bogert, T. Geijtenbeek, O. Even-Zohar, F. Steenbrink, and E. C. Hardin. A real-time system for biomechanical analysis of human movement and muscle function. *Medical and Biological Engineering and Computing*, 154:1069–1077, October 2013. ISSN 51.

[12] Y. Wang and M. Srinivasan. Stepping in the direction of the fall: The next foot placement can be predicted from current upper body state in steady-state walking. *Biology Letters*, 10(9):20140405, September 2014.

[13] Y. Makihara, H. Mannami, A. Tsuji, M.A. Hossain, K. Sugiura, A. Mori, and Y. Yagi. The OU-ISIR gait database comprising the treadmill dataset. *IPSJ Trans. on Computer Vision and Applications*, 4:53–62, April 2012.

[14] European Commission. Scientific data: Open access to research results will boost Europe's innovation capacity, 2012. URL `http://europa.eu/rapid/press-release_IP-12-790_en.htm`.

[15] White House. Increasing access to the results of federally funded scientific research, 2013. URL `https://www.whitehouse.gov/sites/default/files/microsites/ostp/ostp_public_access_memo_2013.pdf`.

[16] H. Geyer and H. Herr. A muscle-reflex model that encodes principles of legged mechanics produces human walking dynamics and muscle activities. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 18(3):263–273, June 2010.

[17] J. K. Moore, S. K. Hnat, and A. J. van den Bogert. An elaborate data set on human gait and the effect of mechanical perturbations, 2014. URL `http://dx.doi.org/10.5281/zenodo.13030`.

[18] E. P. White, E. Baldridge, Z. T. Brym, K. J. Locey, D. J. McGlinn, and S. R. Supp. Nine simple ways to make it easier to (re)use your data. *PeerJ PrePrints*, 1:e7v2, 7 2013. ISSN 2167-9843. doi: 10.7287/peerj.preprints.7v2.

[19] S. K. Hnat, J. K. Moore, and A. J. van den Bogert. Commanded Treadmill Motions for Perturbation Experiments, March 2015. URL `http://dx.doi.org/10.5281/zenodo.16064`.

[20] S. K. Hnat and A. J. van den Bogert. Inertial compensation for belt acceleration in an instrumented treadmill. *Journal of Biomechanics*, 47(15):3758–3761, November 2014.

[21] GaitAnalysisToolKit: Version 0.1.2.

[22] The NumPy Array: A structure for efficient numerical computation.

[23] E. Jones, T. Oliphant, P. Peterson, et al. SciPy: Open source scientific tools for Python, 2001. URL `http://www.scipy.org`.

[24] J. D. Hunter. Matplotlib: A 2D graphics environment. *Computing in Science & Engineering*, 9(3):90–95, 2007.

[25] W. McKinney. Data structures for statistical computing in Python. In *Proceedings of the 9th Python in Science Conference*, pages 51–56, 2010.

[26] Octave community. GNU Octave 3.8.1, 2014. URL `www.gnu.org/software/octave/`.

CHAPTER III

INERTIAL COMPENSATION FOR BELT ACCELERATION

IN INSTRUMENTED TREADMILLS

To design a more robust controller, we developed a human walking data set that included treadmill belt speed perturbations. With this protocol, we hoped to capture some of the human reflexive control actions in response to an unexpected disturbance. However, varying the belt speed will introduce errors in the pitch moment since the force plates underneath the treadmill record inertia when the treadmill rollers are continually started and stopped. Pitch moment directly affects the calculation of joint torques. Without compensation for these inertial artifacts, the joint torques from these experiments cannot be used as a "gold standard" for designing our controller. In this chapter, we describe an inertial compensation method that reduces these errors.

ABSTRACT

Instrumented treadmills provide a convenient means for applying horizontal perturbations during gait or standing. However, varying the treadmill belt speed introduces inertial artifacts in the sagittal plane moment component of the ground reaction force. Here we present a compensation method based on a second-order dynamic model that predicts inertial pitch moment from belt acceleration. The method was tested experimentally on an unloaded treadmill at a slow belt speed with small random variations (1.20 ± 0.10 m/s) and at a faster belt speed with large random variations (2.00 ± 0.50 m/s). Inertial artifacts of up to 12 Nm (root-mean-square, RMS) and 30 Nm (peak) were observed. Coefficients of the model were calibrated on one trial and then used to predict and compensate the pitch moment of another trial with different random variations. Coefficients of determination (R^2) were 72.08% and 96.75% for the slow and fast conditions, respectively. After compensation, the root-mean-square (RMS) of the inertial artifact was reduced by 47.37% for the slow speed and 81.98% for fast speed, leaving only 1.5 and 2.1 Nm of artifact uncorrected, respectively. It was concluded that the compensation technique reduced inertial errors substantially, thereby improving the accuracy in joint moment calculations on an instrumented treadmill with varying belt speed.

3.1 Introduction

Horizontal acceleration of the ground surface is commonly used in experiments to perturb posture and gait [1, 2]. In order to determine joint moments during such tests, ground reaction forces must be measured, but this is problematic when the force plate is not rigidly attached to an inertial reference frame. In an accelerating force plate, inertial forces arise due to accelerating masses between the subject's foot and the force sensors. Compensation for these errors in a translating force plate is possible when mass properties of the plate are known and accelerations are measured [3].

Instrumented treadmills are increasingly available as tools for gait analysis, and these provide a convenient means to apply horizontal surface acceleration during posture and gait [4, 5]. It is expected, however, that acceleration of the treadmill belt will induce a sagittal plane moment in the ground reaction force due to the inertia of the belt, rollers, and motor. This moment is an artifact which will directly translate into an error in the sagittal plane joint moments determined from inverse dynamic analysis. Similar errors may occur when belt speed is interactively controlled ("self-paced") by the subject in a virtual reality environment [6]. In this paper we will quantify the inertial artifacts caused by belt speed variations and present a method to compensate for these artifacts.

3.2   Methods

Input signals for control of belt velocity were created using MATLAB®/Simulink® (Version 2014a, Mathworks, Natick, MA) as shown in Figure 10. A total of four 6-minute trials were generated, two with mean speed of 1.2 m/s and small perturbations, and two with a mean speed of 2.0 m/s and larger perturbations. A random acceleration signal was generated with discrete-time Gaussian white noise and variance in the acceleration was set to $25 \text{ m}^2/\text{s}^4$ (for the 1.2 m/s trials) and $2000 \text{ m}^2/\text{s}^4$ (for 2.0 m/s trials), respectively. The signal was clipped at the maximum belt acceleration of 15 m/s^2, integrated, and high-pass filtered (second-order, Butterworth) with a 0.2069 Hz passband edge frequency to eliminate any velocity drift. The mean velocity was added as a constant to the random signal and limited to a maximum speed of 3 m/s. Two trials were generated in this manner for each speed, using different random number seeds. The resulting belt velocity signals had a mean and standard deviation of 1.20 ± 0.10 m/s and 2.00 ± 0.50 m/s, respectively.

Experiments were performed on a split-belt instrumented treadmill (VG005-A, Motek Medical, Amsterdam, Netherlands). The belt speed signals were used to control the belt speed through software (D-Flow 3.16.2, Motek Medical). The D-Flow software was used to record ground reaction forces and moments and actual belt velocity during the four trials,

49
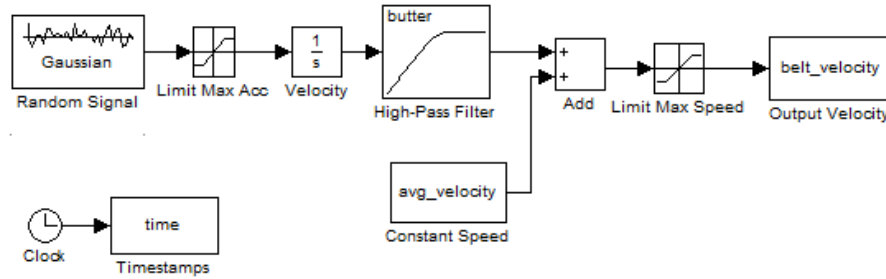
Figure 10: MATLAB® Simulink® diagram for creating random belt velocities, in which a random acceleration signal is generated with Gaussian white noise. Belt velocity is obtained by integrating the signal and filtering with a high-pass Butterworth filter to reduce integration drift

without external loads applied to the treadmill surface. The sampling rate was 100 Hz. An example of the treadmill and coordinate system is shown in Figure 11, where inertia due to the treadmill rollers will result in an artifact in the pitch moment (M_x).
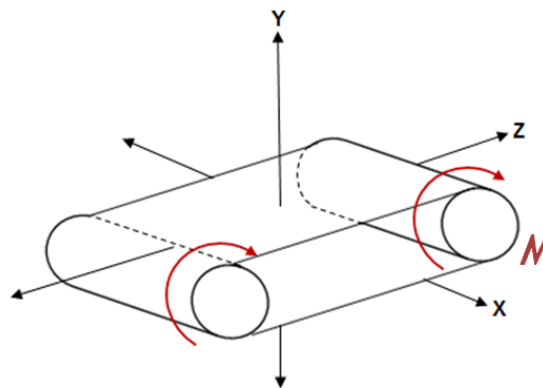
Figure 11: Schematic of the treadmill with coordinate system indicated, where stopping and starting the treadmill rollers will result in a pitch moment artifact about the X-axis

The recorded pitch moment (sagittal plane moment) and belt speed were low-pass filtered (second-order, Butterworth) with a 6 Hz cutoff frequency to simulate how signals are typically processed for inverse dynamic analysis of gait [7]. Belt acceleration was derived from the low-pass filtered belt speed by a central difference formula.

A linear, second-order discrete-time model was used to predict the pitch moment M

from belt acceleration a:

$$M_i = \theta_1 M_{i-1} + \theta_2 M_{i-2} + \theta_3 a_i + \theta_4 a_{i-1} + \theta_5 a_{i-2} \qquad (3.1)$$

One trial at each speed was used to calibrate the model. The five model coefficients θ were determined by minimizing the sum of the squared error between the predicted and measured pitch moment in the calibration trial. The minimization was performed in MAT-LAB using the `fmincon` function.

The calibrated model was used to predict the pitch moment of the other trial at the same speed, and the predicted moment was subtracted from the recorded moment as would be done when compensating human test data for inertial artifacts. The root-mean-square (RMS) of the uncompensated and compensated pitch moment were computed. The coefficient of determination (R^2) verified the predictiveness of the simulation when compared to the measured values.

The analysis was repeated with low-pass filter cutoff frequencies up to 20 Hz to determine how well the inertial compensation would perform in other movements such as sports maneuvers, where the inverse dynamic analysis requires a higher cutoff frequency.

All software and data used are available ([8], Github Repository, 2014, DOI:10.5281/zenodo.10905).

3.3   Results

The pitch moment predicted by the model was in close agreement with the measured pitch moment, as illustrated in Figure 12. The predicted moment was generated after calibrating the model using data from the other random trial at the same speed. R^2 values of 72.08% and 96.75% were obtained for the slow speed (1.2 m/s) and fast speed (2.0 m/s), respectively, and indicate that the model explains most of the pitch moment that is generated by belt speed variations.

Table V shows the RMS values of pitch moment before and after compensation. Sub-
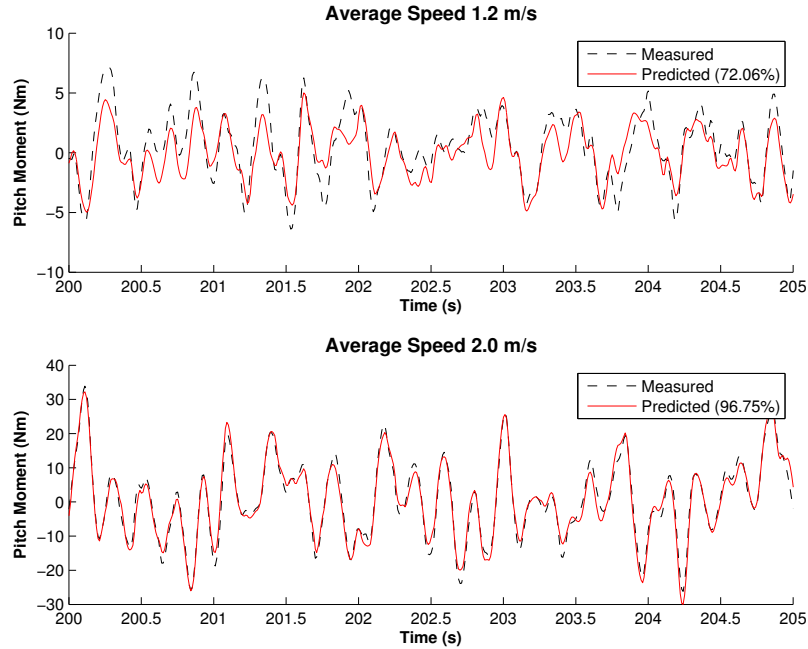
Figure 12: Measured (black dotted) and predicted (red) pitch moment for the slow (top) and fast (bottom) speeds, with the coefficient of determination (R^2) values of 72.08% and 96.75%, respectively. Only a small section of the 6-minute trial is shown.

stantial reductions in error were achieved, especially in the high speed condition.

Trial	RMS (Nm) Before Compensation	RMS (Nm) After Compensation	% Reduction
Low Speed (1.2 m/s)	2.77	1.46	47.37
High Speed (2.0 m/s)	11.51	2.07	81.98

Table V: Root-mean-square (RMS) of the pitch moment before and after compensation. Compensation was performed by subtracting the predicted pitch moment from the measured value.

Figure 13 shows the pitch moment before and after compensation for cut-off frequencies between 1 and 20 Hz. The relative error reduction is diminished at higher frequencies, but a substantial amount of compensation is still achieved.

3.4 Discussion

Accelerating the belt of an instrumented treadmill induces inertial artifacts in the sagittal plane ground reaction moment. To minimize these inertial errors, a compensation method

52
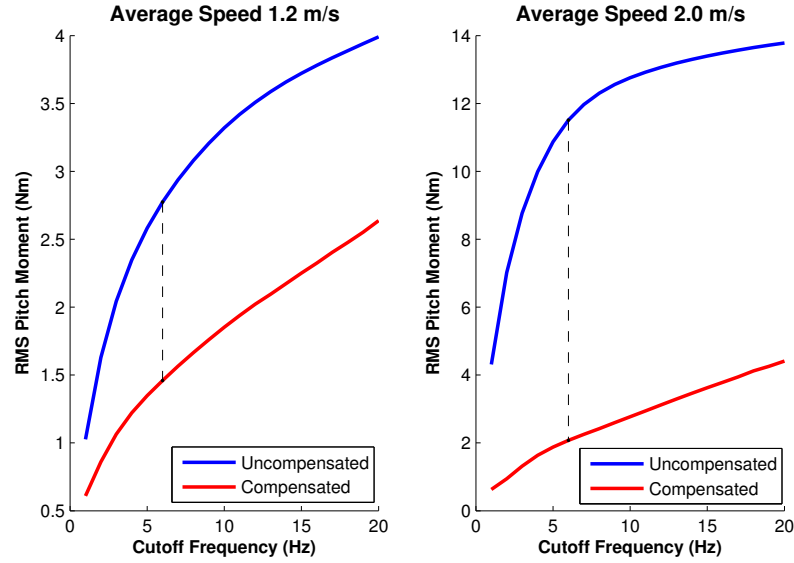
Figure 13: Root-mean-square (RMS) of the uncompensated (blue) and compensated (red) pitch moment as the cutoff frequency of the low-pass, second-order filter was increased from a range of 1-20 Hz. The dashed line indicates the results obtained at 6 Hz

was developed based on a second-order linear model. The method was tested in two experimental conditions: a slow belt speed with small variations (1.2 ± 0.10 m/s) and a fast belt speed with large variations (2.0 ± 0.50 m/s). At the 6 Hz bandwidth that is typically used in gait analysis, inertial artifacts were reduced by 47% and 82%, respectively, in these conditions. The effectiveness of the method declined somewhat at larger bandwidth (Figure 13), but a substantial fraction of the artifact could still be eliminated.

Sagittal plane moment between foot and ground is one of the inputs for inverse dynamic analysis of human motion. Inspection of the inverse dynamic equations reveals that any measurement error in this moment will result in an equal error in human joint moments. Peaks in the inertial artifacts during our tests were typically 6 Nm in the slow condition and 30 Nm in the fast condition (Fig. 12). Peak knee joint moments during human gait are about 40 Nm during slow walking [7], so these inertial artifacts are large. We were able to reduce this error by about half. Our slow condition is representative of an experiment in which gait is perturbed just enough to cause subtle adaptive responses. In more extreme perturbations, such as sudden stopping or starting of the belt, the inertial artifacts would

be larger, and as indicated by our fast tests, a much larger fraction of the artifact can be compensated in such conditions.

We calibrated the model on a trial containing similar belt variations to the trial that requires the compensation. We found, however, that the compensation on the slow trials worked almost equally well when the calibration was done on a fast trial. This suggests that the model accurately captured the dynamic properties of the system, and can be generalized to other movements without requiring recalibration. Other compensation methods using the mathematical relationships between motor, pulley, and belt inertias may also describe the dynamic properties of the system. However, these techniques require more complicated analyses and sufficient knowledge of the mechanical properties of the system. The system identification method presented here does not require such information.

At 6 Hz bandwidth, there was about 1-2 Nm of sagittal plane moment error that could not be explained or compensated by the model. Some of this may simply be the noise in the force plate system. Additionally, the model predictions were based on belt acceleration, which was estimated by numerical differentiation of belt speed data, potentially introducing error. It would be of interest to measure belt acceleration with better instrumentation, instead of using the belt speed information provided by the treadmill control system. Alternatively, the pseudorandom belt speed command signal could be used as input for the model. This would eliminate noise but requires the assumption that the belt speed control system can execute the commanded speed despite horizontal forces applied by a human subject. Finally, the linear second-order model may not be able to accurately capture the internal dynamics of the system, especially at higher bandwidth. Unmodeled dynamics may include nonlinearities due to friction and internal vibration modes. Potential improvements to the model include nonlinear system identification approaches or an auto-regressive moving-average exogenous (ARMAX) model that more effectively identifies noise in the system.

With the emerging trend of using instrumented treadmills for gait and balance control,

this method is simple and easy to implement in other instrumented treadmills, as long as they have the ability to control and measure belt velocity. The amount of inertial correction will vary depending on the properties of the treadmill and its instrumentation. Other balance perturbations, such as mediolateral translation or rotations of the walking surface, will require more sophisticated techniques to compensate for inertial effects in the moving force plate. For variations in belt speed, the compensation method presented here is capable of significantly reducing inertial artifacts, thereby allowing joint moments to be measured in experimental conditions where this was previously not possible.

Acknowledgments

3.5 REFERENCES

[1] S. Park F. B., Horak, and A. D. Kuo. Postural feedback responses scale with biomechanical constraints in human standing. *Experimental Brain Research*, 154(4):417–427, February 2004.

[2] C. L. Chen, S. Z. Lou, H. W. Wu, S. K. Wu, K. T. Yeung, and F. C. Su. Effects of the type and direction of support surface perturbation on postural responses. *Journal of Neuroengineering and Rehabilitation*, 1:50, April 2014.

[3] G. Pagnacco, A. Silva, E. Oggero, and N. Berme. Inertially compensated force plate: A means for quantifying subject's ground reaction forces in non-inertial conditions. *Biomedical Sciences Instrumentation*, pages 397–402, February 2000.

[4] T. M. Owings, M. J. Pavol, and M. D. Grabiner. Mechanisms of failed recovery fol-

lowing postural perturbations on a motorized treadmill mimic those associated with an actual forward trip. *Clinical Biomechanics*, 16(9):813–819, November 2001.

[5] P. H. Sessoms, M. Wyatt, M. Grabiner, J.-D. Collins, T. Kingsbury, N. Thesing, and K. Kaufman. Method for evoking a trip-like response using a treadmill-based perturbation during locomotion. *Journal of Biomechanics*, 47(1):277–280, January 2014.

[6] L. H. Sloot, M. M. van der Krogt, and J. Harlaar. Effects of adding a virtual reality environment to different modes of treadmill walking. *Gait and Posture*, 39(3):939–945, March 2014.

[7] A. J. van den Bogert, T. Geijtenbeek, O. Even-Zohar, F. Steenbrink, and E. C. Hardin. A real-time system for biomechanical analysis of human movement and muscle function. *Medical and Biological Engineering and Computing*, 154:1069–1077, October 2013. ISSN 51.

[8] S. K. Hnat, J. K. Moore, and A. J. van den Bogert. Pitch moment compensation. *GitHub Repository*, 2014. Available at `https://github.com/csu-hmc/pitch-moment-compensation`, version 1.0.

CHAPTER IV

COMPENSATION FOR INERTIAL AND GRAVITY EFFECTS

IN A MOVING FORCE PLATFORM

Varying the treadmill belt speed is only one type of perturbation that can be applied to human subjects during walking. Some instrumented treadmills are capable of moving the entire walking surface (pitch, yaw, and lateral motions). In these systems, the force plates are located underneath the large mass of the treadmill platform. Therefore, platform perturbations will also induce large inertial errors in the ground reaction forces recorded by the force plates. This is a known problem in biomechanics and researchers typically use kinematic data alone for research studies because of this limitation. Though our experimental data set did not include platform perturbations, we developed a separate, accelerometer-based compensation method to reduce these type of inertial errors so that ground reaction forces, and joint torques, can be more accurately measured during these testing protocols.

ABSTRACT

Force plates for human movement analysis provide accurate measurements when mounted rigidly on an inertial reference frame. Large measurement errors occur, however, when the force plate is accelerated, or tilted relative to gravity. This prohibits the use of force plates in human perturbation studies with controlled surface movements, or in conditions where the foundation is moving or not sufficiently rigid. Here we present a linear model to predict the inertial and gravitational artifacts using accelerometer signals. The model is first calibrated with data collected from random movements of the unloaded system and then used to compensate for the errors in another trial. The method was tested experimentally on an instrumented force treadmill capable of dynamic mediolateral translation and sagittal pitch. The compensation was evaluated in five experimental conditions, including platform motions induced by actuators, by motor vibration, and by human ground reaction forces. In the test that included all sources of platform motion, the root-mean-square (RMS) errors were 39.0 N and 15.3 $\mathrm{N\,m}$ *in force and moment, before compensation, and 1.6 N and 1.1* $\mathrm{N\,m}$*, after compensation. A sensitivity analysis was performed to determine the effect on estimating joint moments during human gait. Joint moment errors in hip, knee, and ankle were initially 53.80* $\mathrm{N\,m}$*, 32.69* $\mathrm{N\,m}$*, and 19.10* $\mathrm{N\,m}$*, and reduced to 1.67* $\mathrm{N\,m}$*, 1.37* $\mathrm{N\,m}$*, and 1.13* $\mathrm{N\,m}$ *with our method. It was concluded that the compensation method can reduce the inertial and gravitational artifacts to an acceptable level for human gait analysis.*

4.1 Introduction

Force plates and instrumented treadmills are commonly used to measure ground reaction forces (GRF) for clinical movement analysis, sports performance, or research on human movement. The combination of motion capture and force plate data allow for the calculation of joint moments through inverse dynamic analysis. Recently, instrumented treadmills have become equipped with actuators to translate and rotate the walking surface, for either

virtual reality applications or for testing human response to perturbations. Acceleration of the force plate creates large inertial artifacts in the GRF measurement, because a large moving mass is located between the force of interest (foot/ground interface) and the load cells. Additionally, when the frame is tilted, its gravitational mass starts contributing to the horizontal GRF signals. The problem is especially severe in an instrumented treadmill, where the moving mass includes the treadmill frame, motor, and belts. These large errors in GRF data make it impossible to perform standard inverse dynamics in these conditions because the joint moment calculations are based on inaccurate force measurements. The same problem also occurs when the force plate foundation is not sufficiently rigid, or in a moving vehicle in order to study the biomechanics of driving.

Although the problem is noteworthy in human movement analysis, it exists in any load measurement system where the force of interest and the load cells are separated by a moving mass. This includes certain applications in high-speed material testing and in force-controlled robots [1, 2]. The work presented in this paper will be applicable in those fields as well.

In principle, the inertial and gravitational forces can be estimated and compensated using rigid body dynamics. This requires knowledge of the mass, inertia matrix, acceleration, angular acceleration, angular velocity, and orientation of the frame. This has been successfully done for one-dimensional linear motion such as in materials testing or a sliding force plate [1–4]. While it is straightforward to extend this approach into a six degree of freedom (DOF) load measurement, it becomes impractical due to the requirement to estimate full 3D motion relative to an inertial reference frame, and the use of nonlinear models [5–8]. Furthermore, mass and inertial properties of the frame must be known [9]. Some of the existing methods neglect the effect of rotation and are limited to compensating for errors due to vibrations within the building or floor [10].

In this paper, we introduce a simple linear, accelerometer-based compensation method for a fully general inertial and gravitational compensation of force plate data. The linear

59

model is based on the principle that an accelerometer directly measures the inertial and gravitational force on its internal test mass. With a sufficient number of accelerometers, attached at different locations, the total inertial and gravitational artifact of all mass elements in the moving frame will be a linear combination of accelerometer signals [11]. The method will be presented and evaluated on an instrumented treadmill in various experimental conditions.

4.2 Methods

4.2.1 Compensation Method

In a 6-DOF load measurement, three-dimensional force and moment are the variables of interest. The compensation model assumes that the effect of gravity and inertia on each of the six load signals, when expressed in the local reference frame of the sensor, is a static linear function of N accelerometer signals:

$$\begin{bmatrix} F_x \\ F_y \\ F_z \\ M_x \\ M_y \\ M_z \end{bmatrix} = C \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_N \\ 1 \end{bmatrix}, \tag{4.1}$$

where C is a $6 \times (N + 1)$ matrix of model coefficients. Note that the model includes a constant term for each load variable (the last column of C) which will be used by the calibration to remove any static offsets that may be present in the load cell signals. It is not necessary to calibrate the accelerometers. Neither is it necessary to separate the accelerometer information into acceleration and gravitational effects. In fact, it is essential that such a separation is not performed because inertial artifacts arise from the combination of the two [8]. Raw accelerometer signals should be used.

60

The model must be calibrated by a system identification experiment, in which no external load is applied, and the load cells will only measure the artifacts. This is best done with random movements of the frame, to explore the entire space of potential artifacts. During this experiment, K samples of force and moment data are collected, along with accelerometer data. First, the data is arranged into matrices A (accelerometer signals) and F (force and moment data):

$$A = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1N} & 1 \\ \vdots & \vdots & \vdots & & \vdots \\ a_{K1} & a_{K2} & \cdots & a_{KN} & 1 \end{bmatrix} \tag{4.2}$$

$$F = \begin{bmatrix} F_{1x} & F_{1y} & F_{1z} & M_{1x} & M_{1y} & M_{1z} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ F_{Kx} & F_{Ky} & F_{Kz} & M_{Kx} & M_{Ky} & M_{Kz} \end{bmatrix} \tag{4.3}$$

and the model coefficients C are determined through a least-squares solution of the overdetermined system of linear equations:

$$AC^T = F \tag{4.4}$$

The calibration was coded in MATLAB® version 2016a (Mathworks, Santa Rosa, California, USA) as well as in C++. In MATLAB, QR decomposition was used to obtain the least-squares solution, which is implemented as the "backslash" operator:

$$\texttt{C = (A\textbackslash F)'} \tag{4.5}$$

In C++, model coefficients were determined through linear regression using the Shark library [12]. It was verified that the results were identical in both implementations.

4.2.2 Experimental Validation

Experiments were performed on a split-belt instrumented treadmill capable of actuated mediolateral translation (sway) along the X-axis, and sagittal pitch in the YZ-plane (V-Gait, Motekforce Link, Amsterdam, Figure 14). Each belt assembly, including motors, has a mass of about 150 kg which is located between the load cells and the walking surface. Two triaxial accelerometers (4030 2G range, Measurement Specialties) were mounted on the side of the treadmill (posterior-left and anterior-right), separated by 1.30 m in both the lateral and anterior/posterior directions (Euclidean distance of 1.83 m). Zappa [11] proved that four non-coplanar accelerometers are sufficient to uniquely determine the acceleration at each point in a rigid body. Since there are only two actuated degrees of freedom (DOF, pitch and sway) in this application, only two accelerometers are required to detect the range of movements.

Sway and pitch signals were commanded to the treadmill using D-Flow 3.24 (Motekforce Link, Amsterdam). GRFs from the right belt, and accelerometer data, were directly acquired in the data acquisition unit of an optical motion capture system (Nexus 1.8.5, Vicon). All files were exported in C3D files for further offline analysis. Sampling rate was 1000 Hz. All signals were filtered with a 2nd order low-pass Butterworth filter with a cut-off frequency of 6 Hz, which is typically used for inverse dynamic analysis of walking [13, 14]. After filtering, the first full second of data was removed to eliminate the filter startup effect.

The 6×7 model coefficient matrix C was calibrated using a 60-second unloaded trial in which the treadmill surface was randomly translated and rotated. Zero-mean Gaussian white noise signals were generated with a sampling time of 3.3 ms and RMS amplitudes of 0.707 m/s^2 and 127 °/s^2 for sway and pitch, respectively, in MATLAB Simulink®, as shown in Figure 15. The signals were twice-integrated to obtain smooth treadmill displacements with realizable accelerations. Integration drift was eliminated by processing the input signals through high-pass filters (2nd order Butterworth) with a passband edge frequency of 0.21 Hz. Upper and lower limits were imposed on the signals, thereby restricting the trans-
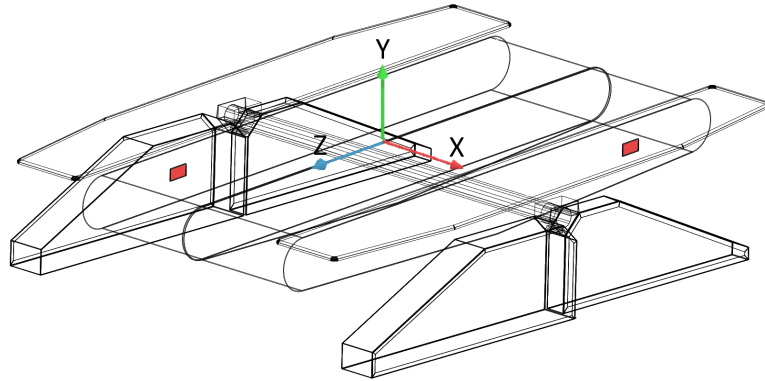
Figure 14: Line drawing of the instrumented treadmill, including the force platform coordinate system. Accelerometer placements are shown in red.

lation and rotation movements to the maximum of ± 0.05 m displacement and $\pm 10°$ pitch angle.
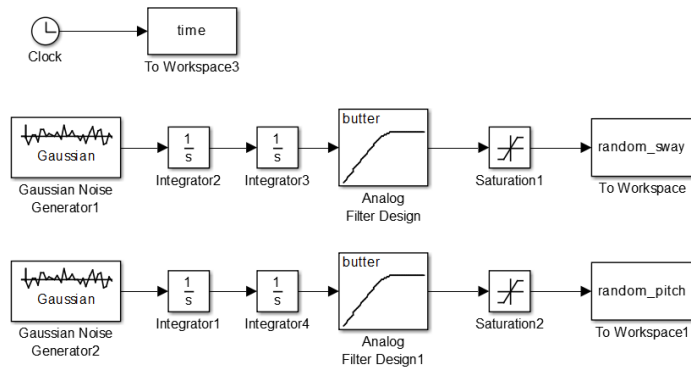
Figure 15: MATLAB® Simulink® diagram for generating random sway (mediolateral translation) and sagittal pitch (rotation) commands. Random Gaussian white noise on the acceleration signals are twice integrated to obtain position, filtered using a second-order Butterworth filter, and saturated to remain within the platform movement limits of ± 5 cm (sway) and $\pm 10°$ (pitch)

The calibrated model was evaluated in five tests where the platform was moved without load applied to the right force plate. The tests included platform motions induced by actuators, by motor vibration, and by human ground reaction force impacts. In Trial 1, the treadmill was subjected to 60 seconds of random pitch and sway movement generated by the same procedure as the calibration trial, but with a different random number seed. Trial 2 had the same random movement, but with the treadmill belts running at 1.3 m/s to evaluate

the effect of motor vibration. Trial 3 again contains the same random movement as Trial 1 and 2, but with a subject walking on the left belt at 1.3 m/s. In Trial 4, the subject walked on the left belt at 1.3 m/s, without pitch or sway motion. In this trial, the subject walked on the front half of the treadmill to represent the worst case of walking-induced frame motion. In Trial 5, the treadmill was set in a static pitch angle of $9°$. As the treadmill is not moving in this test, all artifacts are induced by gravity.

In each trial, the model was used to predict the inertial and gravitational force and moment artifacts from the accelerometer signals. The predicted artifacts were subtracted from the measured forces and moments. Since no external loads were applied to the right force plate, the resulting forces and moments would be zero if the compensation was perfect. To quantify the performance of the compensation method, we determine the RMS (root-mean-square) of the force and moment residuals. This was done on all six load variables, on all test trials, before filtering, after filtering, and after the compensation. This allowed us to quantify how much reduction in the artifacts was achieved by filtering alone, and how much additional reduction was achieved by the linear compensation model:

$$\text{RMS} = \sqrt{\frac{\sum \left(F_{\text{estimate}} - F_{\text{measured}}\right)^2}{n}} \tag{4.6}$$

where $F_{\text{estimate}} = A_{\text{measured}} C^T$ and A_{measured} are the measured accelerometer signals.

4.2.3 Sensitivity Analysis

To assess the importance of inertial and gravitational artifacts and the effectiveness of the compensation, we used human gait data that was collected on a stationary instrumented treadmill where the artifacts were negligible [15]. A standard sagittal plane inverse dynamic analysis [13] was performed three times. First, with the original "true" GRF data. Second, with inertial artifacts from Trial 3 added to the original GRF prior to performing inverse dynamics. Third, with the inertial artifacts from Trial 3, after compensation using the previously calibrated model, added to the original GRF data, prior to performing inverse

64

dynamics. Trial 3 was chosen because it is the test condition that includes all sources of platform motion, leading to the highest inertial artifact. No low-pass filtering was applied to the inertial artifacts and the accelerometer data that was used to compensate. However, the subsequent inverse dynamic analysis used a 6 Hz low-pass filter for both kinematic and force plate data [14]. Ideally, the compensated GRFs will result in joint moments more similar to the "true" joint moments.

4.3   Results

Figure 16 illustrates the performance of the compensation method on the first four seconds of Trial 3 which contains artifacts due to actuated random motions, human GRFs, and motor vibration. Before compensation, there are large artifacts, especially in F_x, F_z, M_x, and M_y. The artifacts in F_z and M_y are mostly gravitational and reflect low-frequency changes in pitch angle. The inertial contributions have higher frequencies, reflecting the sensitivity to linear and angular accelerations, rather than orientation. After predicting the artifacts from the accelerometer signals, and subtracting them from the measured loads, the remaining artifact is small.

For each trial, the effectiveness of the compensation was quantified by the RMS comparison between uncompensated and compensated force signals. Results for the 5 conditions are shown in Table VI.

The largest inertial artifacts were present in Trial 3. When residuals in all coordinate axes were combined, the overall RMS errors were 39.0 N and 15.3 N m in force and moment components, respectively, after filtering. The compensation reduced these errors to 1.6 N and 1.1 N m. The differences between the trials provide insight into the effect of the various mechanical inputs. The errors after compensation were lowest when the platform motion was entirely induced by the actuators (Trial 1), and increased due to vibrations caused by the motors that drive the belts (Trial 2). There was a slight further increase in Trial 3, when human GRFs were applied to the moving frame. In Trial 4, platform motions
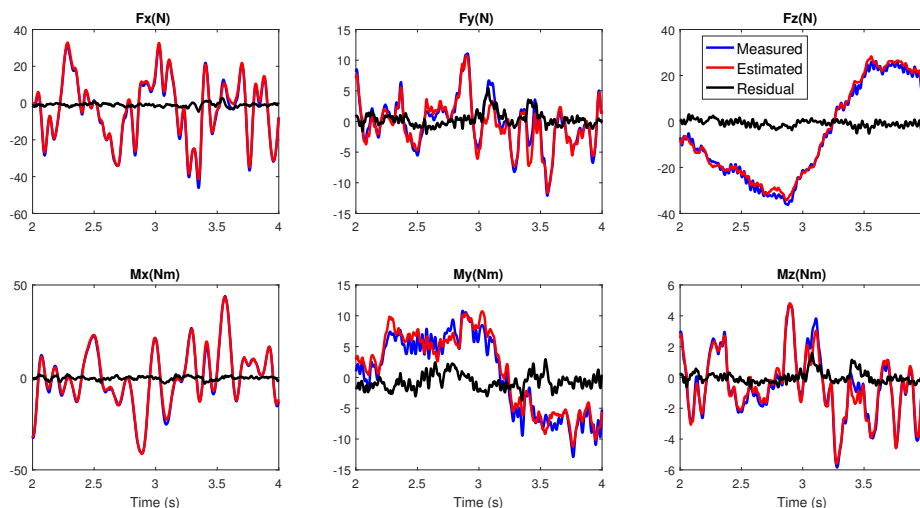
Figure 16: Force and moment data from Trial 3. The measured inertial artifact (blue) was compared to the prediction from accelerometers (red). The residuals (black) are the errors that remain after subtracting the predicted inertial artifact. All data were 6 Hz low-pass filtered before processing. Note the different scales on the vertical axes.

were solely induced by human GRFs and motor vibration. In case of a static pitch of $9°$ (Trial 5) the gravitational effects were large and consistent with the downhill component of the gravitational force on the 150 kg platform mass. After compensation, these errors were reduced to less than 1 N and 1 N m. Even in the most challenging test cases (Trials 2, 3, 4), the compensation reduced the artifacts to values below 2 N and 2 N m.

Figure 17 shows the human joint moments that would be obtained if the GRF data had been affected by the inertial and gravitational artifacts from Trial 3. The joint angles were greatly affected, mostly by the horizontal force F_z and the effect was largest in the hip joint. Before compensation, the RMS errors in hip, knee, and ankle moments were 53.80 N m, 32.69 N m, and 19.10 N m. After compensation, these errors were reduced to 1.67 N m, 1.37 N m, and 1.13 N m, respectively.

4.4 Discussion

Rotating and translating a platform equipped with load cells will introduce inertial artifacts in the force measurements. We developed a linear model to estimate these errors from
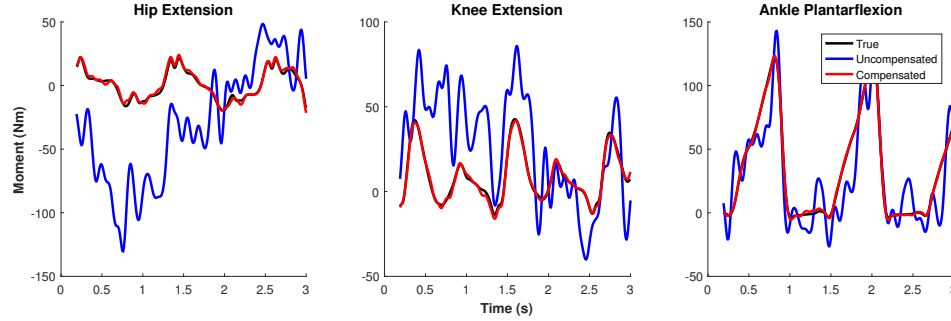
Figure 17: Hip, knee, and ankle joint moments during walking, affected by the uncompensated (blue) and compensated (red) inertial artifacts from Trial 3. The compensated result is almost identical to the "true" result from the original data containing no platform movement (black).

accelerometers so that these errors can be compensated. The method was evaluated on the test case of human gait analysis on an instrumented treadmill undergoing movement. It was found that the errors were initially too large to allow a useful inverse dynamic gait analysis. The compensation method reduced the errors to acceptable levels within the range of noise typically observed in force plate signals. This makes it possible to study the dynamics and control of human gait during mechanical perturbations. This was previously only attempted for simple platform translations or rotations [2, 3, 9]. Our method does not require measuring the kinematics of the moving frame relative to an inertial reference frame and does not require knowledge of the mass properties of the frame, nor exact locations of the accelerometers. Furthermore, our residual forces and moments were far lower than those obtained on an instrumented treadmill with the existing methods [9].

A similar accelerometer-based method was described for an application where load cells were used to determine the location, where a computer screen was touched [16]. In our application, the mass of the moving frame was, however, much larger. With a mass of 150 kg, an acceleration of only $0.01 \, \mathrm{m\,s^{-2}}$ ($1 \, \mathrm{m\,g}$) is sufficient to induce an inertial artifact of 1.5 N. Reducing the artifact to this level required accelerometers with sufficient accuracy and resolution.

Tests were performed on a system with two actuated DOFs (pitch and sway), but the compensation method is completely general and can compensate for artifacts due to full

6-DOF motions in all six load components, using any number of accelerometer signals. To reliably detect inertial artifacts during arbitrary 3D translations and rotations, more than two triaxial accelerometers should be used. With only two accelerometers, an angular velocity or acceleration about an axis through the two accelerometers cannot be detected. Zappa's method [11] used four non-coplanar accelerometers, but did not show that this was necessary. It may be possible to use only three accelerometers, or a coplanar accelerometer configuration. Because our method uses a least-squares approach, any number of accelerometers can be used, and more accelerometers will give better results. The number of accelerometers should be selected based on the DOFs of the particular system. Including more accelerometers will more effectively reduce noise and non-rigid effects, though these additions will eventually have diminishing returns.

Another attractive feature of the approach is that the calibration matrix automatically includes the linear transformations that are needed to predict forces and moments in the force plate coordinate system. Therefore, the accelerometers can be randomly oriented and measure data in their own coordinate system.

The compensation method is based on the assumption that the moving platform is a rigid body. Inertial artifacts associated with deformation or vibration within the frame will only be partially compensated. This explains why our results were slightly worse in the trials where internal vibration was introduced from the belt motors or human impacts. However, it may be possible to use a larger array of accelerometers to detect and compensate for non-rigid modes. Another possibility is to use and calibrate a dynamic model that can simulate vibrations. Treadmill frames have internal moving parts (motor and rollers) which will introduce additional artifacts if their speed is not constant. This artifact only exists in the sagittal plane moment M_x. A dynamic, 1-DOF compensation model was successfully developed [17] and can be added on to the techniques presented here.

During all tests, signals were low-pass filtered at 6 Hz which is typical for human gait analysis. When higher bandwidth is needed, the remaining error will be larger because

more of the non-rigid motion effect will be included. Therefore, the performance of the compensation method must be tested for a specific application and bandwidth requirement. It is also important to note that the compensation method was not evaluated in the case where a test subject walked on both treadmill belts, though the method could still be applied.

In one of the test conditions (Trial 4), the frame was not actively moved, but all movements were due to the motors and human GRF. The pitch moment (M_x) artifact was about 10 N m, after filtering, due to a small rotational vibration about the pitch axis. Such a vibration is expected based on the structure of the machine (Figure 1). The frequency of this vibration was low enough that the artifact was not sufficiently attenuated by the 6 Hz low pass filter alone. However, the accelerometer-based compensation reduced the error to 1.04 N m which is acceptable for gait analysis. Furthermore, as our compensation method consists of a single matrix multiplication (and vector subtraction), it can be applied in real-time in combination with other real-time biomechanical models [14].

Traditionally, force measuring systems are designed with a natural frequency that is well above the frequency of interest. Our results demonstrate that this requirement can be relaxed, because low-frequency errors can be partially removed by inertial compensation. Similarly, artifacts from low frequency building vibrations, or from doing force measurement in moving vehicles, could be compensated. This reduces the need to place force measuring systems in basements or on a separate mechanically isolated foundation. Because accelerometers are inexpensive, and the compensation method is easily implemented, it may be wise to include inertial compensation as a standard feature on force measuring systems that are sensitive to inertial artifacts.

Load cells measure forces and moments in the local (moving) reference frame, and the compensation is carried out entirely in that reference frame. When load measurement is combined with motion capture performed in an inertial reference frame, a coordinate transformation is needed to bring the (compensated) load data into the inertial reference

frame. This transformation requires knowledge of the position and orientation of the moving frame. In the actuated treadmill, the position and orientation of the frame is known from the encoders in the position control system. Alternatively, optical motion capture can be used, with three or more motion capture markers placed on the treadmill frame [18].

4.5   Conclusion

The proposed method is capable of reducing inertial and gravitational artifacts in data obtained from a moving instrumented treadmill. For gait applications, the remaining error is small enough to allow accurate joint torque calculations through inverse dynamic analysis.

Test condition	Variable	RMS(raw)	RMS(filtered)	RMS(compensated)
Trial 1: random pitch and sway	F_x (N)	35.32	16.13	0.57
	F_y (N)	9.99	3.91	0.76
	F_z (N)	69.62	65.69	0.89
	M_x (N m)	18.54	13.18	0.76
	M_y (N m)	24.82	19.07	0.64
	M_z (N m)	4.50	2.04	0.23
Trial 2: random pitch and sway, 1.3 m/s belt speed	F_x (N)	33.17	16.02	1.28
	F_y (N)	11.61	3.91	1.17
	F_z (N)	69.08	65.72	1.52
	M_x (N m)	18.75	13.24	1.16
	M_y (N m)	24.54	19.04	1.25
	M_z (N m)	4.49	2.05	0.35
Trial 3: random pitch and sway, 1.3 m/s belt speed, human walking	F_x (N)	33.69	16.16	1.51
	F_y (N)	20.25	5.19	1.22
	F_z (N)	71.45	65.41	1.88
	M_x (N m)	26.11	18.42	1.24
	M_y (N m)	26.56	18.91	1.46
	M_z (N m)	6.89	2.43	0.38
Trial 4: 1.3 m/s belt speed, human walking	F_x (N)	12.29	1.31	1.44
	F_y (N)	13.94	2.75	1.06
	F_z (N)	17.89	1.53	1.63
	M_x (N m)	18.68	9.65	1.04
	M_y (N m)	11.68	1.28	1.33
	M_z (N m)	4.43	1.07	0.32
Trial 5: 9° static pitch	F_x (N)	1.65	1.44	0.42
	F_y (N)	19.46	19.44	0.96
	F_z (N)	227.33	227.33	0.66
	M_x (N m)	24.87	24.86	0.39
	M_y (N m)	65.08	65.07	0.38
	M_z (N m)	6.20	6.19	0.33

Table VI: Root-mean-square (RMS) values of force and moment data from the unloaded force plate during the test trials. See text for details.

4.6 REFERENCES

[1] J. P. Hessling. Models of dynamic measurement error variations of material testing machines. *Mechanical Systems and Signal Processing*, 23(8):2510–2518, September 2009.

[2] M. J. Dixon. Development of a load-cell compensation system. *Experimental Mechanics*, 31:21–24, March 1991.

[3] G. Pagnacco, A. Silva, E. Oggero, and N. Berme. Inertially compensated force plate: A means for quantifying subject's ground reaction forces in non-inertial conditions. *Biomedical Sciences Instrumentation*, pages 397–402, February 2000.

[4] F. Yang and Y.-C. Pai. Correction of the inertial effect resulting from a plate moving under low-friction conditions. *Journal of Biomechanics*, 40:2723–2730, February 2006.

[5] N. Berme and H. C. Guler. Force and/or motion measurement system having inertial compensation and method thereof, 2012. US Patent 8,315,823.

[6] N. Berme and H. C. Guler. Force measurement system having inertial compensation, 2012. US Patent 8,315,822.

[7] Z.-C. Hou, Y.-N. Lu, Y.-X. Lao, and D. Liu. A new trifilar pendulum approach to identify all inertia parameters of a rigid body or assembly. *Mechanism and Machine Theory*, 44(6):1270–1280, June 2009.

[8] A. J. van den Bogert, L. Read, and B. M. Nigg. A method for inverse dynamic analysis using accelerometry. *Journal of Biomechanics*, 29(7):949–954, July 1996.

[9] R. Preuss and J. Fung. A simple method to estimate force plate inertial components in a moving surface. *Journal of Biomechanics*, 37(8):1177–1180, August 2004.

[10] G. Boschetti, R. Caracciolo, D. Richiedei, and A. Trevisani. Model-based dynamic compensation of load cell response in weighing machines affected by environmental vibrations. *Mechanical Systems and Signal Processing*, 34(1–2):116–130, January 2013.

[11] B. Zappa, G. Legnani, A. J. van den Bogert, and R. Adamini. On the number and placement of accelerometers for angular velocity and acceleration determination. *Journal of Dynamic Systems, Measurement, and Control*, 123:552–554, March 2001.

[12] C. Igel, V. Heidrich-Meisner, and T. Glasmachers. Shark 2.1. *Journal of Machine Learning Research*, 9:993–996, 2008.

[13] D. A. Winter. *Biomechanics and Motor Control of Human Movement*. John Wiley and Sons, New York, 1990.

[14] A. J. van den Bogert, T. Geijtenbeek, O. Even-Zohar, F. Steenbrink, and E. C. Hardin. A real-time system for biomechanical analysis of human movement and muscle function. *Medical and Biological Engineering and Computing*, 154:1069–1077, October 2013. ISSN 51.

[15] J. K. Moore, S. K. Hnat, and A. J. van den Bogert. An elaborate data set on human gait and the effect of mechanical perturbations. *PeerJ*, 3:e918, April 2015.

[16] J. B. Roberts. Force measurement system correcting for inertial interference, 2006. US Patent 7,102,621.

[17] S. K. Hnat and A. J. van den Bogert. Inertial compensation for belt acceleration in an instrumented treadmill. *Journal of Biomechanics*, 47(15):3758–3761, November 2014.

[18] J. H. Challis. A procedure for determining rigid body transformation parameters. *Journal of Biomechanics*, 28(6):733–737, June 1995.

PART II.

Designing the Virtual-Muscle Reflex

Controller

CHAPTER V

DEVELOPMENT OF A MUSCLE MODEL FOR REAL-TIME APPLICATIONS

The first step in designing the VMR controller was to model virtual muscles that could operate in real-time on a P/O device. A real-time implicit formulation of a Hill-type muscle model had already been created, but was fully integrated within a simulation of musculoskeletal dynamics [1]. In this chapter, we separate the muscle model from the methods described in previous literature, using joint angles from our experimental data to drive the muscle dynamics. We describe our tests to check if the muscle model behaved realistically in response to both step and sinusoidal neural input commands. We also ensured the model was fast enough to be simulated in real-time. To further reduce simulation time for our optimizations, we reformulated the Rosenbrock solver into a closed form solution of the linear system of equations, which had not been previously done.

Conference Abstracts:

1. S. K. Hnat and A. J. van den Bogert. A Real-Time Virtual Muscle Controller for Powered Prostheses. American Society of Biomechanics Midwest Meeting, The University of Akron, Akron, Ohio USA, February 2015.

2. S. K. Hnat and A. J. van den Bogert. Real-Time Virtual Muscle Control for Prostheses and Exoskeletons. 25th Congress of the International Society of Biomechanics, The Scottish Exhibition and Conference Center, Glasgow, Scotland, July 2015.

3. S. K. Hnat and A. J. van den Bogert. A Real-Time Virtual Muscle System for Prosthesis Control. Dynamic Walking, The Ohio State University, Columbus, Ohio USA, July 2015.

---

[1] Van den Bogert AJ, Blana D, Heinrich D. Implicit methods for efficient musculoskeletal simulation and optimal control. Procedia IUTAM. 2011;2(2011):297-316. doi:10.1016/j.piutam.2011.04.027

ABSTRACT

Phase-based impedance controllers are the current standard for controlling a powered orthosis. We hypothesize that the spring-damper characteristics of an impedance controller can be achieved with biologically-based virtual muscles. Here, we generate the torque signals for the electric motor of a prosthesis using a muscle controller. We reformulate the implicit fixed-step Rosenbrock solver to achieve even faster computation times. To test the muscle model in simulation, we used a planar leg consisting of 6 different muscles with both monoarticular and biarticular features to control the knee and ankle joints. First, we use joint angles of 0° and a step-input as the neural control signal and observe the resulting torque directions in the knee and ankle. Then, we drive the muscle dynamics using 30 seconds of walking data for the joint angles and sinusoidal activation signals, where antagonist muscle pairs are completely out of phase to allow for maximum joint torque production. Our results suggest the model behaves appropriately in response to the prescribed sensor inputs, and can also perform all calculations using a time step of 0.18 ms.

5.1 Introduction

Prosthetic and orthotic (P/O) devices aim to improve the quality of life for persons with a spinal cord injury (SCI) or amputation. Passive P/O models, likened to a spring-damper, are capable of restoring some features of able-bodied walking, though these devices are incapable of adapting to outside perturbations [1]. As an alternative, actuated prostheses can respond to environmental changes using feedback control. Electrically-actuated prostheses are an emerging trend due to improvements to the physical hardware, including battery life and microcontroller processors [2–4].

Though various control strategies have been designed and implemented for powered P/O devices, phase-based impedance controllers are the current standard [5, 6]. Through

this approach, a joint torque signal is generated in order to drive the electric motors of the device. Joint torque is calculated as a function of the joint angle and joint velocity, similar to a spring-damper system. Instead of one spring-damper, the control is split throughout finite-states, with separate stiffness and damping parameters for each phase of the gait cycle.

In place of an impedance controller, the same spring-damper characteristics can be achieved through the nonlinear viscoelastic and dynamics properties of muscles. Even without active muscle control, the muscle tendon and surrounding tissues behave similarly to springs [7], while the water content of muscle can be described as a damper [8–10]. Instead of physically modeling the muscles on the hardware, a simulated, virtual representation of muscles can actuate the electric motors of a P/O device. Joint torque can be generated by multiplying the simulated muscle force by the moment arm, or the distance between the muscle attachment point and the joint (motor). In simulation, virtual muscles have been shown to greatly increase stability during walking when compared to direct torque-driven control [11]. For practical applications on hardware, muscle dynamics must be simulated at least as fast as real-time. A generalized overview of this strategy is outlined in Figure 18.
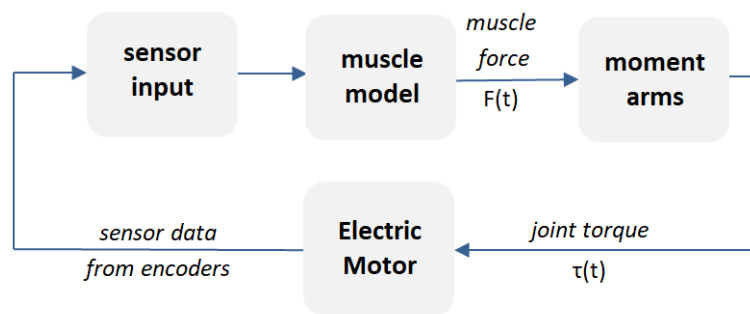
Figure 18: Generalized schematic of a muscle model for operating an electric motor. The muscle model receives sensor inputs from hardware encoders to estimate a virtual muscle force which is multiplied by a moment arm to obtain joint torque

Musclulotendon models are divided into two separate categories: cross-bridge models and Hill-type models. Though cross-bridge models describe the biological function and

structure of muscles [12–14], the high dimensionality renders computation time both impractical and unfeasible for real-time use in prosthetic devices [9]. Other methods more accurately describe the biological properties of cross-bridge models, though they are equally demanding, such as actin fiber formation [15], modeling of residual forces in cross-bridge models [16], and cytoskeletal cross-link estimation [17].

Alternatively, Hill-type muscle models are widely used in muscle-driven simulations of human movement [18, 19]. The primary feature of Hill-type models centers around a contractile element (CE) which generates force due to the myosin and actin cross-bridge at the sacromere level of muscles. The model also uses series and parallel elasticity to represent the muscle tendon and its surrounding tissue, (details follow in Section 5.2.1). Different from other cross-bridge models, the complexity of muscle contraction is simplified by assuming general force-length and force-velocity relationships exhibited by muscles [20], which were estimated through experiments conducted by Hill [21].

Real-time simulation of Hill-type muscle dynamics is challenging because the differential equations that describe muscle dynamics are nonlinear and numerically stiff, containing near-singularities. Fixed step solvers can become unstable and variable step solvers will be slow due to small time steps. Recently, a simulation of muscle dynamics using a Hill-type muscle model can be achieved in real-time through a fixed-step implicit Rosenbrock solver, described by Van den Bogert et al. [22]. In their study, the muscle model operated in concert with a simulation of multibody dynamics.

Here, we consider the possibility of actuating a powered prosthetic system with torques generated by a real-time simulation of muscle dynamics. We isolate this implicit muscle model [22] and further improve the simulation time by reformulating the Rosenbrock fixed-step solver as an analytic, closed form solution of the system of linear equations. This study has two purposes: 1) to verify the muscle model responds appropriately to sensor inputs and 2) to assess the accuracy and computation time of the model.

5.2 Methods

In this section, we detail the muscle model (5.2.1) and the Rosenbrock solver (5.2.2). We also describe the simulation tests for verifying the model behavior and the speed/accuracy tests (5.2.3).

5.2.1 Implicit Mathematical Model of Hill-Type Muscle Dynamics

The mathematical relationships describing muscle contraction and muscle activation are formulated from the contributions of each element of a Hill-type muscle model: 1) a contractile element (CE), for muscle contaction dynamics, 2) a parallel elastic element (PEE), representing the surrounding muscle tissue, 3) a series elastic element (SEE), representing the muscle's tendon, and 4) a small amount of viscous damping in parallel to the contractile element. Figure 19 illustrates the Hill muscle model, with the main elements highlighted.
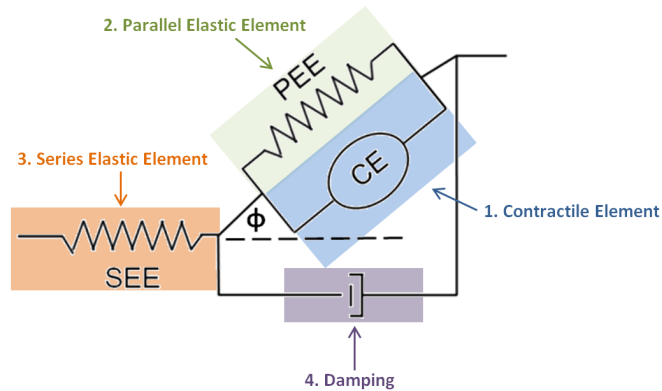
Figure 19: Typical Hill-type muscle model, consisting of a: 1. Contractile element (CE), 2. Parallel elastic element (PEE), 3. Series elastic element (SEE), pennation angle (ϕ), and 4. Viscous damping in parallel to the CE. The indicated components each contribute to the output force of the muscle, which acts through the SEE

Hill models assume muscle fiber geometry as perfectly parallel fibers of equal length and constant muscle volume [9]. The attachment angle of the fiber to the tendon is defined by the pennation angle (ϕ), which we assume to be a constant of $0°$. For faster computation time, all calculations are normalized to the maximum isometric muscle force, F_{max} and

optimal length of the contractile element, $L_{CE,opt}$.

In the following sections, we describe the individual force contributions of each element in the Hill-Model and how they are used to estimate the total muscle force through two differential equations representing muscle activation and contraction. All relationships used in our model were derived from both [18] and [22].

Contractile Element (CE)

Figure 20 demonstrates the standard force-length (left) and force-velocity (right) relationships of the CE. Each muscle produces a maximum force at its optimal fiber length, with decreasing force as the joint moves away from this position. The force-velocity relationship dictates that the muscle force decreases while shortening velocity increases.
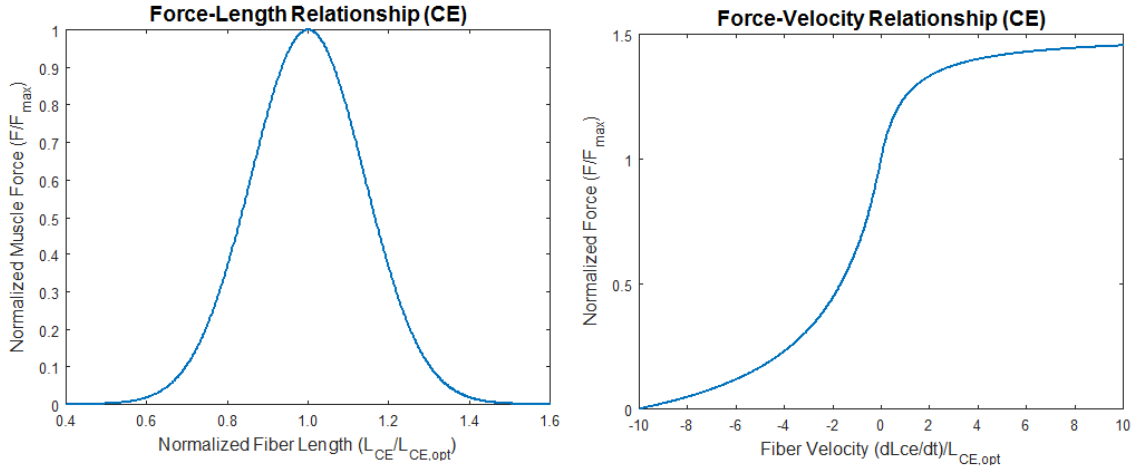
Figure 20: Standard force-length (left) and force-velocity (right) of the contactile element (CE) in Hill-type muscle models, where negative and positive velocities represent shortening and lengthening, respectively. The example shown uses a muscle with a width parameter (W) of 0.63 and optimum fiber length ($L_{CE,opt}$) of 0.1 m.

The length dependence is modeled as:

$$f_{FL_{CE}} = exp\left(\frac{\frac{L_{CE}}{L_{CE,opt}} - 1}{W}\right)^2 \tag{5.1}$$

where W is a width parameter for the contractile element's force-length relationship and

L_{CE} is the length of the contractile element.

The contribution of the force-velocity relationship (f_{FV}) depends on the direction of velocity. If the velocity of the contractile element ($\dot{L}_{CE}$) is decreasing, then the muscle experiences *concentric* contraction. With decreasing velocity, the force contribution of f_{FV} becomes:

$$f_{FV} = \frac{V_{max} + \dot{L}_{CE}}{V_{max} - A} \tag{5.2}$$

where V_{max} is the maximum contraction velocity of the muscle and A is a constant variable of 0.25 used in Hill-muscle models [23].

However, if the velocity is increasing, then the muscle undergoes *eccentric* contraction. Muscles eccentrically contract when they are producing force while lengthening. With increasing velocity, the force contribution of the f_{FV} becomes:

$$f_{FV} = \frac{G_{max} + c}{\dot{L}_{CE} + c} \tag{5.3}$$

where G_{max} is the maximum eccentric force of the muscle, and c is a continuity parameter for the eccentric force-velocity relationship at $\dot{L}_{CE} = 0$, derived from $\frac{V_{max}}{V_{max} - A} = \frac{G_{max} + c}{c}$.

Finally, using the length dependence (5.1) and either concentric (5.2) or eccentric (5.3) velocity, the force in the CE is:

$$F_{CE} = a \cdot F_{max} \cdot f_{FL_{CE}} \cdot f_{FV} \tag{5.4}$$

Series and Parallel Elastic Elements (SEE/PEE) and Viscous Damping

When the CE is not activated, the spring-like behavior of the PEE and SEE ensures passive dynamics of muscles. In Hill-muscle models, both the SEE and PEE are modeled

as nonlinear springs, in which force is not produced until the length of the element is longer than its slack-length. The forces of these elements, F_{PEE} and F_{SEE}, are represented by two sets of equations depending if they are undergoing compression or tension.

While stretching (positive deformation), the elongation of the PEE is positive, and the F_{PEE} can be calculated by multiplying the stiffness k_{PEE} by its squared length [18], which is L_{CE} subtracted by the PEE slack-length:

$$F_{PEE} = k_{PEE} \left(L_{CE} - PEE_{slack} \right)^2 \tag{5.5}$$

Similarly, force in the F_{SEE} can be calculated by multiplying the stiffness k_{SEE} and the squared length of SEE, which is the length of the muscle L_m subtracted by L_{CE} and the slack-length of the SEE:

$$F_{SEE} = k_{SEE} \left(L_m - L_{CE} L_{CE,opt} - SEE_{slack} \right)^2 \tag{5.6}$$

In compression (negative deformation), the PEE and SEE can be described by a linear equation with separate stiffness terms from those used in the quadratic equation for stretch. We designate these stiffnesses with the subscript l, representing *linear*. The F_{PEE} is compression becomes:

$$F_{PEE} = k_{PEE,l} \left(L_{CE} - PEE_{slack} \right) \tag{5.7}$$

Similarly, the F_{SEE} in compression is described by:

$$F_{SEE} = k_{SEE,l} \left(L_m - L_{CE} L_{CE,opt} - SEE_{slack} \right) \tag{5.8}$$

Numerical instability occurs when the derivative of the muscle force with respect to $\dot{L}_{CE}$ is zero. To prevent this, a small amount of viscous damping (F_{damp}) is added in parallel to the contractile element. The force contribution due to damping (F_D) is calculated by:

$$F_D = \beta \dot{L}_{CE} \tag{5.9}$$

where β is the damping coefficient and $\dot{L}_{CE}$ is the velocity of the contractile element.

Using this muscle architecture, activation and contraction dynamics can be represented as a set of two differential equations, where the state variables are the length of the contractile element (L_{CE}) and the muscle activation (a).

Activation dynamics is driven by neural excitation signals (u) from the brain or spinal cord, as described in [18]:

$$\dot{a} = (u - a)\left(\left(\frac{u}{T_{act}} - \frac{1 - u}{T_{deact}}\right)\right) \tag{5.10}$$

where T_{act} and T_{deact} are the the time constants of muscle activation and deactivation, respectively.

Subsequently, the overall force of the muscle will act through its tendon. Force in the SEE is equal to the individual force contributions of the other elements of the Hill-type muscle model. The contraction dynamics, rearranged for the force (F_{SEE}), is represented by the differential equation, as described in [22]:

$$F_{SEE} = a \cdot F_{max} \cdot f_{FL}(L_{CE}) \cdot f_{FV}(\dot{L}_{CE}) + F_{PEE} + F_D \tag{5.11}$$

where F_{max} is the maximum isometric force of the muscle, f_{FL} and f_{FV} are the force-length and force-velocity relationships of the CE, respectively, and F_{PEE} and F_D are the

force contributions due to the PEE and damping element. Joint torque can be obtained by multiplying the force of the muscle by its moment arm (d), or tendon attachment point:

$$\tau = \sum_i d_i F_{SEE,i} \qquad (5.12)$$

A description of the muscle parameters used amongst all elements of the Hill-type muscle is shown in Table VII. The nominal properties of these muscles were obtained from previous literature [23], where the methods and studies used to obtain all muscle parameters are detailed.

Table VII: Muscle parameters used in the three-element Hill-model which simulate muscle contraction and activation dynamics

Parameter	Description
F_{max}	Maximal isometric force of CE (N)
$L_{ce,opt}$	Optimal length of CE (m)
$Width$	Width of CE force-length relationship (m)
PEE_{slack}	Slack-length of the PEE, relative to $L_{ce,opt}$ (m)
SEE_{slack}	Slack length of the SEE (m)
L_0	Muscle-tendon length when all DOFs are zero (m)
d_{hip}	hip moment arm (m)
d_{knee}	knee moment arm (m)
d_{ankle}	ankle moment (m)
k_{PEE}	Stiffness parameter of PEE, in Fmax/Lceopt^2
u_{max}	strain of SEE at F_{max} load
V_{max}	Maximum contraction velocity in Lceopt/s
T_{act}	activation time constant
T_{deact}	deactivation time constant
G_{max}	maximum eccentric force
A	Hill's A parameter, relative to F_{max}

5.2.2 Real-Time Simulation of Muscle Dynamics

For use in hardware, the muscle model must perform all calculations at least as fast as real-time. The differential equations for muscle dynamics are highly nonlinear, allowing for singularities which prevent real-time simulations. A fixed-step, implicit formulation of musculoskeletal dynamics has been developed that can produce accurate results in real-

time [22]. Muscle activation (5.10) and contraction (5.11) relationships can be combined into a single, implicit state equation:

$$f(x, \dot{x}, u) = 0 \tag{5.13}$$

with state variables $x_n = \begin{bmatrix} L_{CE,1}...L_{CE,n} & a_1...a_n \end{bmatrix}^T$ and state derivatives $\dot{x}_n = \begin{bmatrix} \dot{L}_{CE,1}...\dot{L}_{CE,n} & \dot{a}_1...\dot{a}_n \end{bmatrix}^T$, where n is the number of muscles, L_{CE} is the length of the contractile element, and a is the muscle activation signal. The control inputs, u, are the neural excitation control signals generated by a neural controller. This implicit formulation of forward muscle dynamics can be solved using a fixed-step, linear, first-order Rosenbrock implicit differential equation solver [22], where h is the step-size:

$$\left(\left(\frac{\partial f}{\partial x} \right) + \frac{1}{h} \frac{\partial f}{\partial \dot{x}} \right) \Delta x = \left(\frac{\partial f}{\partial \dot{x}} \dot{x}_n - f(x_n, \dot{x}_n, u_n) \right) - \frac{\partial f}{\partial u} (u_{n+1} - u_n) \tag{5.14}$$

$$x_{n+1} = x_n + \Delta x$$

$$\dot{x}_{n+1} = \Delta x / h$$

Based on the muscle dynamics equations, the relationships of 5.14 can be rewritten as

follows, where f_1 and f_2 represent muscle contraction and muscle activation, respectively:

$$f = \begin{bmatrix} f_1 \\ f_2 \end{bmatrix} \tag{5.15a}$$

$$\frac{\partial f}{\partial x} = \begin{bmatrix} \dfrac{\partial f_1}{\partial L_{CE}} & \dfrac{\partial f_1}{\partial a} \\ 0 & \dfrac{\partial f_2}{\partial a} \end{bmatrix} \tag{5.15b}$$

$$\frac{\partial f}{\partial \dot{x}} = \begin{bmatrix} \dfrac{\partial f_1}{\partial \dot{L}_{CE}} & 0 \\ 0 & \dfrac{\partial f_2}{\partial \dot{a}} \end{bmatrix} \tag{5.15c}$$

$$\frac{\partial f}{\partial u} = \begin{bmatrix} 0 & \dfrac{\partial f_2}{\partial u} \end{bmatrix}^T \tag{5.15d}$$

For brevity, the partial-derivatives of each force contribution required by the Rosenbrock solver have been omitted. In order to be used in low-level programming languages, the Rosenbrock solver (5.14) can be reformulated as a closed form solution of the system of linear equations:

$$\begin{bmatrix} A & B \\ C & D \end{bmatrix} \Delta x = \begin{bmatrix} E \\ F \end{bmatrix}$$

86

Substituting the relationships 5.15b - 5.15d into the linear form, we obtain:

$$A = \frac{\partial f_1}{\partial L_{CE}} + \frac{1}{h}\frac{\partial f_1}{\partial \dot{L}_{CE}} \quad D = \frac{\partial f_2}{\partial a} + \frac{1}{h}\frac{\partial f_2}{\partial \dot{a}}$$

$$B = \frac{\partial f_1}{\partial a} \qquad\qquad E = \frac{\partial f_1}{\partial \dot{L}_{CE}}\dot{L}_{CE} - f_1$$

$$C = 0 \qquad\qquad F = \frac{\partial f_2}{\partial \dot{a}}\dot{a} - f_2 - \frac{\partial f_2}{\partial u}(u_{n+1} - u_n)$$

Therefore, the change in state between each time-step can be simulated by a set of independent linear equations:

$$x = \begin{bmatrix} \Delta L_{CE} \\ \Delta a \end{bmatrix}$$

$$\Delta L_{CE} = (E - B\Delta x_a) \tag{5.16}$$

$$\Delta a = F/D$$

5.2.3 Muscle Model Simulation Tests

In this study, we used a planar model of the lower leg with six muscles to control the knee in ankle, three of which were monoarticular groups (*Vasti*, *Soleus*, *Tibialis anterior*) and three biarticular groups (*Rectus femoris*, *Hamstrings*, *Gastrocnemius*), as shown in Figure 21. The inputs to the system were musculotendon length (L_m) derived from joint angles (q) at the hip, knee, and ankle, and the control signals (u) for the six muscles.

Muscle lengths (L_m) are calculated from the muscle length when all angles are zero (L_0) and the moment arms (d) multiplied by the joint angles (θ) for the hip, knee, and ankle:

$$L_m = L_0 - d_{Hip}\theta_{hip} - d_{Knee}\theta_{knee} - d_{Ankle}\theta_{ankle} \tag{5.17}$$
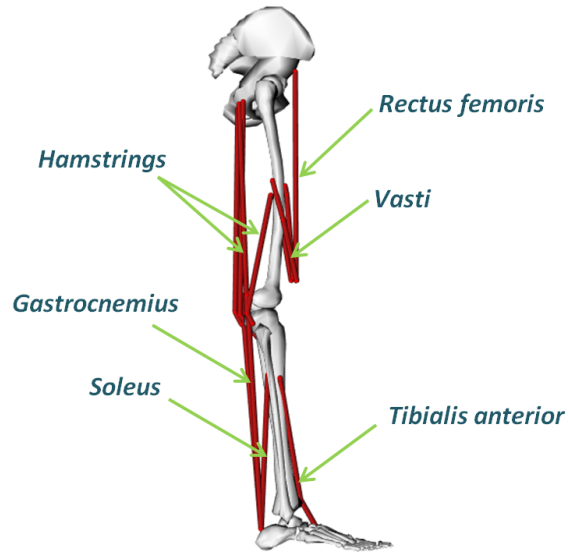
Figure 21: Virtual muscles used in the study, three monoarticular groups (*Vasti*, *Soleus*, *Tibialis Anterior*) and three biarticular groups (*Rectus Femoris*, *Hamstrings*, *Gastrocnemius* to control the knee and ankle.

where the moment arm lengths were obtained from [23]. In this study, we either manually set these joint angles, or use joint angles calculated from human walking experiments [24]. In a P/O device, these joint angles would be obtained through encoders.

Two separate test cases were performed to verify that the muscle model produces reasonable muscle forces and joint torques. Instead of modeling a neural controller to drive muscle activation, we use step and sinusoidal commands for these verifications.

Model Evaluation. The first test evaluated the feasibility of the model, where joint angles were held constant at $0°$ while a step-control fully actuated each muscle at different times throughout the simulation (Figure 22). Observing the corresponding force in each muscle and torque in each joint indicates if the model behaves appropriately when certain muscles of the leg are activated. For instance, activating a flexor muscle will produce flexion torque, while activating an extensor muscle will result in an extension torque. For biarticular muscles, this torque will affect every joint in the muscle path.

Speed and Accuracy. A second test condition evaluated the performance of the model and Rosenbrock solver by calculating the accuracy of the model and the computation time. To exercise the model through all possible states and inputs that occur during human gait, joint angle time histories $q(t)$ for 30 seconds of normal walking and 1 Hz sinusoidal test signals for the muscle excitations $u(t)$ were used in the simulation (for the experimental data, refer to Part I, Chapter II).

The flexors and extensors were activated out of phase to achieve the maximum torque which could ever be produced by the model (Figure 22). Simulations were performed in MATLAB® (Mathworks, Santa Rosa, California, USA) using fixed time steps (h) ranging from 0.08 to 16.0 ms, where the step sizes were increased until the muscle model became unstable and did not produce a muscle force.

To quantify numerical simulation errors, we first estimated the joint torques using a small time step outside of real-time capabilities (0.001 ms) to use as a benchmark or "true" result. Next, we increase the step size and calculate the root-mean-square (RMS) error as a percentage of the maximum peak joint torque obtained in the current time step, when compared to the result from the benchmark time step. This was performed for all subsequent step sizes. Computation times were measured in all tests, using MATLAB's internal stopwatch through the `tic` and `toc` commands.
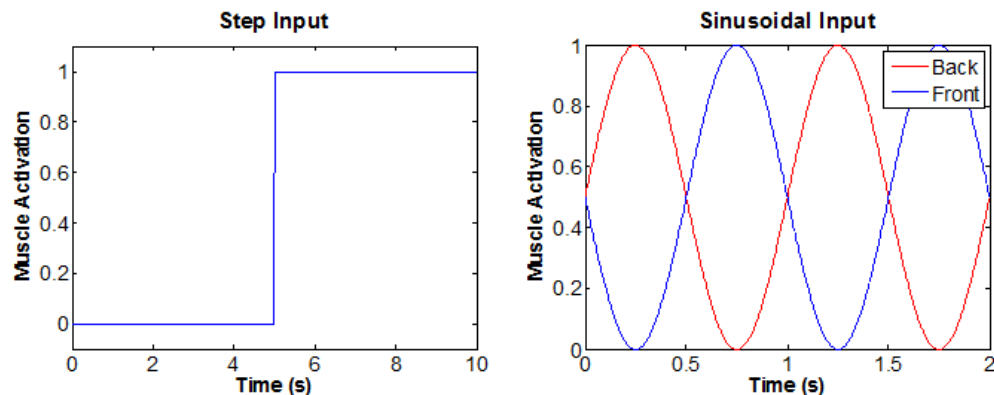
Figure 22: Step-input signal for the muscle model validation (left) and sinusoidal input signals with 1 Hz frequency (right) for evaluating the speed and accuracy of the model.

Initially, the muscle model was programmed in MATLAB when the speed and accuracy tests were performed. However, for even faster computation times, the muscle model was coded in the C-programming language and compiled as a MEX function to be run in the MATLAB pipeline. The output torque and muscle models of both the MATLAB and C-MEX functions were compared to ensure the models were equivalent.

5.3 Results

Two test cases evaluated the performance of the musculotendon model. First, the results of the predicted muscle forces and joint torques produced by the model given a step-input neural excitation command are shown in Section 5.3.1. Subsequently, the results of the speed and accuracy tests are shown in Section 5.3.2, where 30 seconds of joint angle time histories and sinusoidal muscle excitations represented the most extreme scenario for the muscle model to perform.

5.3.1 Model Evaluation

The muscle forces and joint torques obtained through the model responded appropriately to the prescribed muscle excitations. Figure 23 demonstrates the first test case where joint angles were held constant at 0° while a step-control fully actuated each muscle at random times throughout the simulation. Predicted torques in the knee (blue) and ankle (green) correspond as expected to the muscle activations in the thigh and shank. Note that negative torque corresponds to flexion and plantarflexion. The torque directions in the knee and ankle are influenced by muscle activations in the flexors and extensors. Additionally, the biarticular properties of muscles are represented when the *Gastrocnemius* (cyan) is activated, in which both the knee and ankle torques are affected. The smaller force contribution of the *Tibialis anterior* (yellow), when compared to the magnitude of the *Soleus* (pink), corresponds to the respective F_{max} values of each muscle.
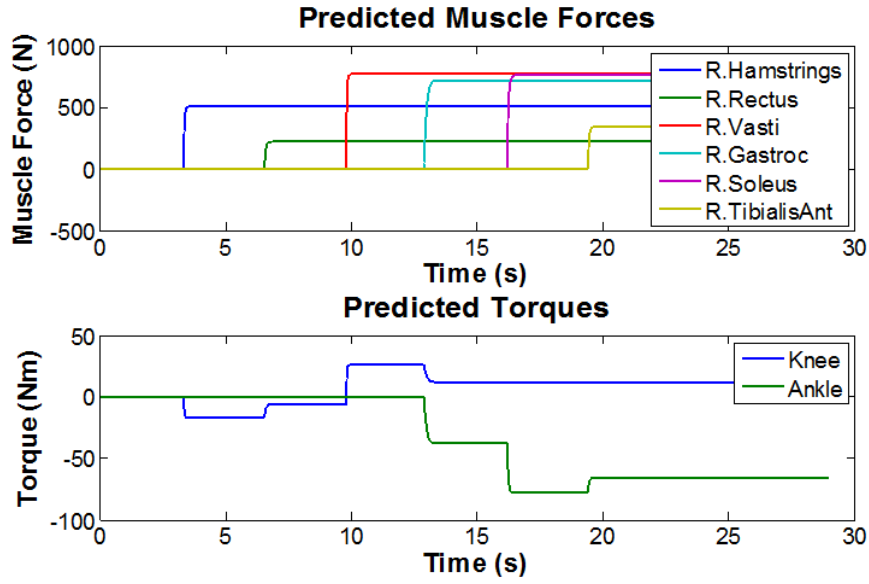
Figure 23: Predicted muscle forces (top) and torques (bottom) with constant joint angles $q(t)$ held at $0°$ and step-control signal u randomly applied throughout the simulation.

5.3.2 Speed and Accuracy

The speed and accuracy of the muscle model was evaluated using joint angles from walking data and sinusoidal input signals. As expected, the results depended on the integrator step size, as shown in Figure 24. RMS error is quantified by comparing the difference between the joint torques of the smallest step-size, or the "true" result, in comparison to the joint torques obtained in the larger step sizes. On the left, the RMS error is plotted as a function of the step size, while on the right, this error is plotted as a function of the simulation time.

The relationship was linear because the Rosenbrock is a first-order method. In MATLAB, real-time is achieved (simulation time < 30 s) when the step size was 0.18 ms or larger, with simulation errors below 1%, indicated by the red line. Though larger time steps still produce acceptably accurate results, the model became unstable for time steps larger than 16 ms.

When reprogrammed in a C-MEX function, the computation time decreased by a factor of over 100, thereby allowing for faster convergence in optimizations (used in Part II, Chapter VI and VII). For example, using a small time-step of 0.001 ms, we can simulate 8
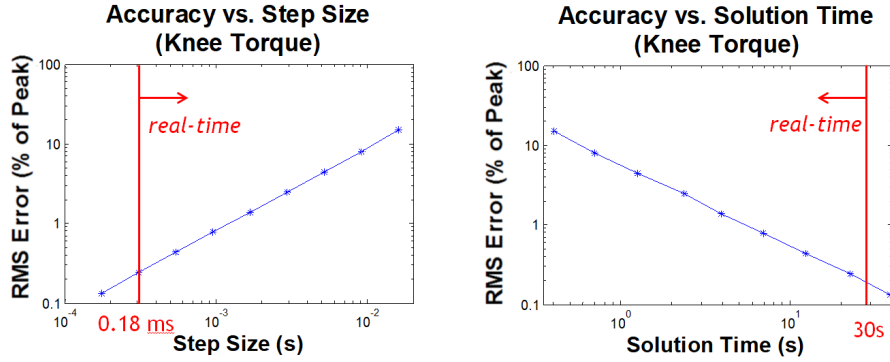
Figure 24: Accuracy is effected by the integrator step size (left) and computation speed (right). Root-mean-square (RMS) error was quantified as a percentage of the maximum torques in the knee and ankle, in which the "true" result at the smallest step size is used as a benchmark. The error is the difference between the maximum joint torque of the larger time steps in comparison to the smallest time step. Real-time is indicated by the red line, which is faster than 30 seconds, with a step-size of 0.18 ms. These results indicate that torques can be obtained in a simulation of muscle dynamics with accurate results and minimal computation time.

minutes of walking data well within real-time.

5.4 Discussion

Phased-based impedance controllers are the standard controller used in powered P/O devices. However, the spring-damper characteristics of an impedance controller could be replaced by a virtual muscle model. In this study, we used a real-time implicit formulation of muscle dynamics [22] to generate the joint torque signals that would be used to control the electric motors of a prosthesis or exoskeleton. We reformulated the fixed-step Rosenbrock solver to further improve the computation speed. Afterwards, we performed two feasibility tests: 1) a simple validation to ensure the muscle model behaved appropriately in response to joint angle and neural inputs, and 2) to test the speed and accuracy of the model.

Our results suggest the model behaves appropriately in response to the prescribed neural step-inputs. For instance, activating the *Vasti* results in an extension torque in the knee. A small force contribution of the *Tibialis anterior*, when compared to the magnitude of the *Soleus*, correspond to the respective F_{max} of each muscle. Similarly, the biarticular

Gastrocnemius produces ankle plantarflexion and knee flexion when activated.

We are able to perform muscle simulations well within real-time. In MATLAB, this is achieved at step sizes as small as 0.18 ms with simulation errors below 1%. When reprogrammed in C-code, a step size of 0.001 ms can be used. It is important to note that we defined accuracy by comparing each time-step to the results of an extremely small time-step. Accuracy does not imply our implicit Hill-model can better explain the physical mechanics of human muscles when compared to other methods. For our application, Hill-type estimations may be accurate enough to generate a torque signal for a prosthesis.

Since a muscle model is its own system, it can be used in conjunction with another controller that drives the muscle dynamics. An advantage of muscle models are the additional stability they can provide, which prohibit erratic movements in response to infeasible torque signals. Even while deactivated, the passive properties within a muscle can limit unrealistic movements. It can be suggested that adding a virtual muscle model to any controller might improve its overall stability. Virtual muscles, however, cannot replace safety features already present on P/O devices, such as fall recovery.

The simple formulation of muscle dynamics allows for the equations to be directly implemented in the low-level programming environment typically used in the microcontrollers. An intuitive next step is to design a neural controller for the muscle model, such as reflex control or central-pattern-generators (CPGs). The inputs to the muscle model are conducive to such a controller, since an activation signal sent from the brain, or spinal cord, initiates muscle contraction. Future studies could involve the development of such a controller that can naturally drive the muscle dynamics of our model.

5.5 REFERENCES

[1] M. R. Tucker, J. Olivier, A. Pagel, H. Bleuler, M. Bouri, O. Lambercy, J. del R. Millán, R. Riener, H. Vallery, and R. Gassert. Control strategies for active lower extremity

prosthetics and orthotics: A review. *Journal of NeuroEngineering and Rehabilitation*, 12(1):1, December 2015.

[2] J. Yu, M. Tan, J. Chen, and J. Zhang. A survey on CPG-inspired control models and system implementation. *IEEE Transactions on Neural Networks and Learning Systems*, 25(3):441–456, September 2013.

[3] K. Nakada, T. Asai, and Y. Amemiya. An analog CMOS central pattern generator for interlimb coordination in quadruped locomotion. *IEEE Transactions on Neural Networks*, 14(5):1356–1365, November 2003.

[4] C. Torres-Huitzil and B. Girau. Implementation of central pattern generator in an FPGA-based embedded system. In *International Conference on Artificial Neural Networks*, pages 179–187, November 2008.

[5] F. Sup, A. Bohara, and M. Goldfarb. Design and control of a powered transfemoral prosthesis. *International Journal of Robotics Research*, 27(2):263–273, February 2008.

[6] B. E. Lawson, B. Ruhe, A. Shultz, and M. Goldfarb. A powered prosthetic intervention for bilateral transfemoral amputees. *IEEE Transactions on Biomedical Engineering*, 62(4):1042–1050, April 2015.

[7] S. L. Lindstedt, T. E. Reich, P. Keim, and P. C. LaStayo. Do muscles function as adaptable locomotor springs? *Journal of Experimental Biology*, 205(15):2211–2216, August 2002.

[8] N. Hogan. The mechanics of multi-joint posture and movement control. *Biological Cybernetics*, 52(5):315–331, September 1985.

[9] M. Millard, T. Uchida, A. Seth, and S. L. Delp. Flexing computational muscle: Mod-

eling and simulation of musculotendon dynamics. *ASME Journal of Biomechanical Engineering*, 135(2):0210051–02100511, February 2013.

[10] E. Vinnars, J. Bergstöm, and P. Fürst. Influence of the postoperative state on the intracellular free amino acids in human muscle tissue. *Annals of Surgery*, 182(6): 665–671, December 1975.

[11] K. G. Gerritsen, A. J. van den Bogert, M. Hulliger, and R. F. Nernicke. Intrinsic muscle properties facilitate locomotor control - A computer simulation study. *Motor Control*, 2(3):206–220, July 1998.

[12] E. Eisenberg, T. L. Hill, and Y. Chen. Cross-bridge model of muscle contraction. quantitative analysis. *Biophysical Journal*, 29(2):195–227, February 1980.

[13] G. I. Zahalak and S. Ma. Muscle activation and contraction: Constitutive relations based directly on cross-bridge kinetics. *ASME Journal of Biomechanical Engineering*, 112:52–62, February 1990.

[14] J. C. Haselgrove and H. E. Huxley. X-ray evidence for radial cross-bridge movement and for the sliding filament model in actively contracting skeletal muscle. *Journal of Molecular Biology*, 77(4):549–568, July 1973.

[15] S. Walcott and S. X. Sun. A mechanical model of actin stress fiber formation and substrate elasticity sensing in adherent cells. *Proceedings of the National Academy of Sciences*, 107(17):7757–7762, April 2010.

[16] S. Walcott and W. Herzog. Modeling residual force enhancement with generic cross-bridge models. *Mathematical Biosciences*, 216(2):172–186, December 2008.

[17] S. X. Sun, S. Walcott, and C. W. Wolgemuth. Cytoskeletal cross-linking and bundling in motor-independent contraction. *Current Biology*, 20(15):R649–R654, August 2010.

[18] F. E. Zajac. Muscle and tendon: properties, models, scaling, and application to biomechanics and motor control. *Critical Reviews in Biomedical Engineering*, 17(4):359–411, February 1989.

[19] J. M. Winters and L. Stark. Muscle models: What is gained and what is lost by varying model complexity. *Biological Cybernetics*, 55(6):403–420, March 1987.

[20] J. M. Winters and S. L.-Y. Woo. *Multiple Muscle Systems: Biomechanics and Movement Organization*. Springer-Verlag, New York, 1990.

[21] A. V. Hill. The heat of shortening and the dynamic constants of muscle. *Proceedings of the Royal Society of London B: Biological Sciences*, 126(843):136–195, October 1938.

[22] A. J. van den Bogert, D. Blana, and D. Heinrich. Implicit methods for efficient musculoskeletal simulation and optimal control. *Procedia IUTAM*, 2:297–316, January 2011.

[23] K. G. M. Gerritsen, W. Nachbauer, and A. J. van den Bogert. Computer simulation of landing movement in downhill skiing: Anterior cruciate ligament injuries. *Journal of Biomechanics*, 29(7):845–854, July 1996.

[24] J. K. Moore, S. K. Hnat, and A. J. van den Bogert. An elaborate data set on human gait and the effect of mechanical perturbations. *PeerJ*, 3:e1248, 4 2015.

CHAPTER VI

REAL-TIME VIRTUAL MUSCLE REFLEX CONTROL AND PHASE-BASED

IMPEDANCE CONTROL IN DESCRIBING PERTURBED WALKING

Reflex controllers have been previously used in simulation, but were tuned without information from human walking experiments. When first designing and testing the VMR controller, we started with the ankle joint only. The ankle is simpler joint to control than the knee and hip and would serve as a preliminary feasibility test for our approach. We then compared our VMR controller to a four-phase impedance controller typically used in powered P/O devices. Here, we describe our study of the VMR and impedance controller for a powered ankle joint.

Intended Publication:

1. S. K. Hnat and A. J. van den Bogert. Real-Time Virtual Muscle Reflex Control and Phase-Based Impedance Control in Describing Perturbed Walking. In preparation. *In preparation.*

Conference Abstracts:

1. S. K. Hnat and A. J. van den Bogert. Virtual Muscles and Reflex Controllers are Capable of Describing Human Gait and Responses to Perturbation. Dynamic Walking, Camp Ohiyesa, Holly, Michigan, USA, June 2016.

2. S. K. Hnat and A. J. van den Bogert. Are Virtual Muscles and Reflex Control Capable of Describing Variations within the Human Gait Cycle? American Society of Biomechanics Midwest Meeting, Grand Valley State University, Grand Rapids, Michigan USA, February 2017.

ABSTRACT

A biologically-inspired actuation system, including muscles, spinal reflexes, and vestibular feedback, may be capable of achieving more natural gait mechanics in powered prostheses or exoskeletons. In this study, we developed a Virtual Muscle Reflex (VMR) system to control ankle torque, and tuned it using data from human responses to anteroposterior mechanical perturbations at three walking speeds. The system consists of three Hill-Type muscles, simulated in real time, and uses feedback from ground reaction forces and from stretch sensors on the virtual muscle fibers. Controller gains, muscle properties, and reflex and vestibular time delays were optimized using Covariance Matrix Adaptation (CMA) to minimize the difference between the VMR torque output and the torque measured from the experiment. We repeated the procedure using a conventional finite-state impedance controller. For both controllers, the coefficient of determination (R^2) and root-mean-square (RMS) error was calculated as a function of the gait cycle. For the impedance controller, mean RMS and maximum R^2 throughout all trials was 16.38 Nm and 0.52, respectively, and for the VMR controller, these values were 11.93 Nm and 0.76, respectively. With lower RMS error and higher correlation coefficients, it was concluded that the VMR system can better reproduce the human responses to perturbations than the impedance controller at the ankle.

6.1 Introduction

Phase-based impedance controllers are the current standard for powered prosthetic and orthotic devices [1–4], in which the joint torque is calculated as a function of the joint angle and joint velocity. However, discrete switching between controller phases can lead to unnatural gait mechanics. Additionally, these type of impedance controllers are linear and may not be able to capture the nonlinear nature of human walking, nor are they tuned using information from human walking experiments. Recently, optimization techniques

have been used to determine the parameters of both a constant and time-varying impedance controller [5] and a lower-limb prosthetic with energy regeneration [6] using human gait data.

An intuitive approach for designing smoother P/O controllers is to model the human biological control system. Yet, the various strategies of the brain and central nervous system (CNS) for controlling human motion is a challenging area of ongoing research and is difficult to achieve in real-time applications [7]. A simpler strategy is to design the controller using virtual muscles and spinal reflexes.

It has been shown through simulation that muscle properties contribute substantially to stability during walking when compared to direct torque-driven control [8]. The same spring-damper properties of an impedance controller can be achieved in a virtual muscle model. The passive, spring-like properties of muscles prohibit unnatural behaviors arising from unstable torque commands. Muscle dynamics can be replicated by the standard 3-element Hill model, and it was recently shown that real-time muscle simulation, as required for use in P/O devices, is possible [9]. Joint torque can then be obtained by multiplying the muscle force by the distance between the attachment point of the muscle and the joint.

As a controller for these virtual muscles, spinal reflexes are responsible for stabilizing gait, responding to perturbations, and assisting in balance correction [10]. The activation signal generated by reflexes is sent to the motor neurons in the muscle to produce movement through muscle contractions [11, 12]. Feedback signals are then sent back to the spinal cord via afferent neurons in the muscle spindles, Golgi tendon organs, and other receptors on the skin which stimulate spinal reflexes [13].

An autonomous, neuromuscular reflex controller has been shown to produce stable walking in simulated models [14, 15]. Though the simulated model can walk at various speeds and slopes, it lacks several features of human walking, including stabilization from the *Tibialis anterior* during stance phase. Reflex control has also been tested in the BioM actuated ankle joint [16] and an actuated knee and ankle prosthesis [17]. The model uses

positive-feedback from the muscle (force or length) and proportional-derivative (PD) control for maintaining postural stability of the trunk. Though positive feedback is usually avoided in classical control design, it has been shown that humans use positive feedback as a stabilizing mechanism during walking [18, 19]. Recently, a reflex controller was evaluated on an exoskeleton with an actuated ankle, resulting in a reduction of net metabolic cost during rehabilitation exercises [20].

It has been suggested that reflex controllers can more readily adapt to environmental disturbances than the impedance controllers used in many powered P/O devices [17]. Since reflex models solely rely on muscle force feedback, any disturbance in the environment (sloped terrain or external perturbations) will be sensed by reflexes due to the reactive change in muscle forces or lengths [15].

Though these reflex controllers produce natural gait in both simulated environments and commercialized hardware, current models have not been tuned or validated using human walking data obtained through experiments. Optimization techniques can determine the parameters which minimize the error between the joint torque exhibited by the subject and the torque generated from the model [21, 22].

In this study, we investigate the possibility of using a Virtual Muscle Reflex (VMR) system in order to determine if muscle reflexes are capable of reproducing able-bodied walking exhibited by test subjects. In addition, by applying anteroposterior mechanical perturbations, the respective reactive actions of the test subject can be used to define the nature of the controller. This will demonstrate the controller's ability in replicating the variations in human joint moments, within and between gait cycles, when subjected to random perturbations. We then compare the performance of the VMR system to a typical phase-based impedance controller, tuned using the same optimization technique and experimental dataset.

6.2 Methods

In this section, we detail the experimental protocol (6.2.1), the muscle model (6.2.2), the neuromuscular reflex controller (6.2.3), the impedance controller (6.2.4), and the optimization method for tuning the VMR system (6.2.5). The parameters of both the impedance controller and the VMR controller are obtained through an optimization, in which the difference between the experimental joint torque and the joint torque from the impedance or VMR model is minimized.

6.2.1 Experimental Protocol

Experiments were conducted on a dual-belt, instrumented R-mill treadmill, capable of measuring ground reaction forces (GRFs) and moving the platform in two degrees-of-freedom: sagittal plane rotation and lateral translation (Forcelink, Culemborg, Netherlands). Antero-posterior perturbation was applied by varying the belt speed through D-Flow software, versions 3.16.1 to 3.16.2 (Motek Medical, Amsterdam, Netherlands). We used a 10-camera Osprey motion capture system, along with Cortex software, version 3.1.1.1290 (Motion Analysis, Santa Rosa, CA, USA). Joint positions were measured from motion capture using a full-body marker set of 47 markers [23].

Walking data from 10 participants (3 females and 7 males, age: 24 ± 4 years, height: 1.74 ± 0.09 m, mass: 74.4 ± 12 kg) was used in the study [23]. The test subjects walked for 8 minutes on an instrumented treadmill at 0.8, 1.2, and 1.6 m/s and were perturbed using random belt acceleration signals generated by discrete-time Gaussian white noise. The variance of the signal was adjusted until the perturbations were within 10% of the mean walking speed. For more details concerning the testing protocol, refer to Part I, Chapter II and [23].

Joint angles and torques can be calculated through standard inverse 2D analysis using joint positions from motion capture and the measured ground reaction forces (GRFs) [24]. Before performing the inverse dynamics, the sagittal pitch moment was first compensated

101

for the inertial artifacts caused by the rotation of the treadmill rollers (refer to Part I, Chapter III and [25]). Joint angles from the experiment are used as inputs to both the impedance and VMR controllers, while the joint torque is used as a benchmark for tuning the controller parameters.

6.2.2 Muscle Model

To simulate the VMR controller, we used a planar model of the lower leg with three muscle groups that produce the majority of ankle dorsiflexion and plantarflexion in the sagittal plane: *Gastrocnemius*, *Soleus*, and *Tibialis anterior* (shown in Figure 25). This represents a lower-limb prosthesis with an actuated ankle joint.
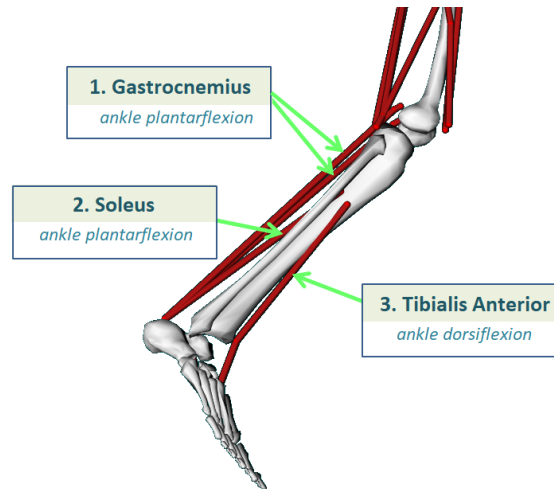
Figure 25: Muscles used in the VMR model, in which muscle pairs are capable of producing flexion and extension at the ankle joint. These muscles include: *Gastrocnemius*, *Soleus*, and *Tibialis anterior*.

The contraction and activation of each muscle was described by a typical 3-element Hill muscle model, including a contractile element (CE), based on standard force-length and force-velocity relationships, and a series and parallel elastic element (SEE/PEE), modeled as nonlinear springs. For numerical stability in muscle-driven simulations, a small amount of viscous damping can be added in parallel to the contractile element. Biologically, a damping term reflects the high water content of muscle tissue [26–28]. The nominal

properties of these muscles were obtained from previous literature [29]. Our sign conventions are positive torque in dorsiflexion and negative torque in plantarflexion.

Inputs to the muscle model are the neurological excitation signals from the reflex controller (u) and the joint angles of the knee and ankle (θ). In a prosthetic device, these joint angles would be provided by encoders at the joints; however, in our tuning process, joint angles were obtained from the experimental data. The joint angles determine the muscle length, L_m:

$$L_m = L_0 - d\theta, \tag{6.1}$$

where d is the moment arm of the muscle, assumed to be constant, and L_0 is the length of the muscle when the joint angle is zero. The force produced by each muscle is solved in real-time from a set of two differential equations for activation and contraction dynamics. For more details regarding the structure of the muscle model, refer to Part II, Chapter V.

The force in each muscle is then multiplied by its moment arm to obtain the joint torque:

$$\tau = \sum_{i=1}^{3} d_i F_i \tag{6.2}$$

Muscle force and muscle length are also used as feedback terms for the reflex controller. The respective inputs and outputs of the muscle model are shown in Figure 26.

For use in powered prosthetics or orthotics, the muscle model must perform all calculations at least as fast as real-time. Recently, a fixed-step, implicit formulation of musculoskeletal dynamics was developed [9, 30, 31] to eliminate the mathematical singularities that can lengthen computation time.

Using this method, the muscle activation and contraction relationships can be combined into a single, implicit state equation: $f(x, \dot{x}, u) = 0$, with state variables $x_n = \begin{bmatrix} L_{CE,1}...L_{CE,n} & a_1...a_n \end{bmatrix}^T$ and state derivatives $\dot{x_n} = \begin{bmatrix} \dot{L}_{CE,1}...\dot{L}_{CE,n} & \dot{a}_1...\dot{a}_n \end{bmatrix}^T$, where n is the number of muscles. This implicit formulation of forward muscle dynamics can be simulated in MATLAB® (Mathworks, Santa Rosa, California, USA) using a fixed-
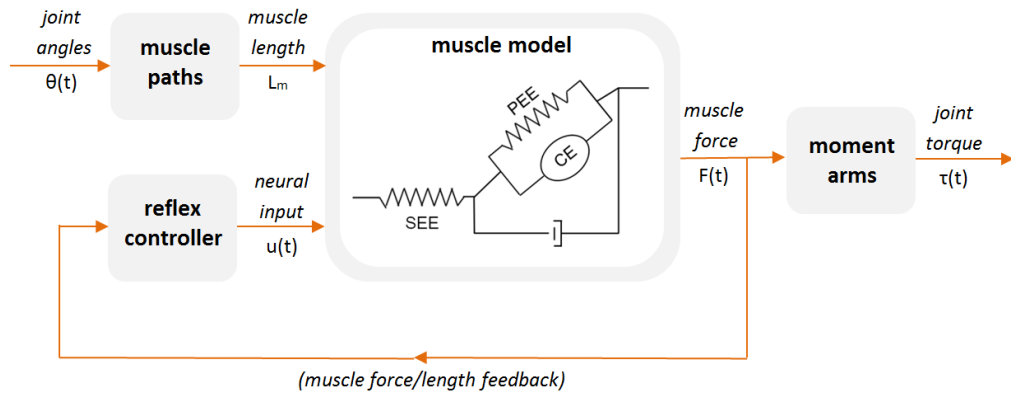
Figure 26: Virtual Muscle Reflex (VMR) controller, where the inputs to the model are the muscle length L_m (determined from joint angles from either an encoder or from walking experiments), and the neural excitation signals generated by the reflex controller $u(t)$. The output is the muscle force which can be multiplied by the moment arm to obtain the joint torque, $\tau(t)$. Muscle force and length are used as feedback for the reflex controller.

step, linear, first-order Rosenbrock solver. For more details of this solver, refer to Part II, Chapter V.

6.2.3 Reflex Controller

The autonomous, positive force feedback reflex controller of Geyer and Herr was adapted for use in the VMR system [14]. Though positive feedback is typically avoided in most classical control applications, studies in both animals and humans suggest that positive force feedback increases stability in walking and running [32, 33]. Positive force feedback is a stabilizing mechanism in response to the activity of other muscles [18, 19]. Physiologically, this force and length feedback represent the Golgi tendon organs and muscle spindles, which provide both muscle length and velocity feedback to the central nervous system [34].

Modifications to Geyer's controller include the addition of vestibular controls during the stance phase of each muscle, in which the time-delayed, angular acceleration of the trunk is multiplied by a gain of the general form: $k_m \ddot{\theta}(\Delta t)$. Since anteroposterior perturbations also accelerate the trunk, the vestibular system would sense the change in acceleration

and correct for the disturbance by extending the step-length of the swing leg to maintain balance [35]. The reflex controller has two gait phases for each muscle, *stance* and *swing*. We determine the phase based on the magnitude of the vertical GRF, where the transition threshold is 10 N. In all controller equations, the abbreviations st and sw denote stance and swing phases, respectively.

In the next section, we describe our modified reflex equations and how we aim to reproduce each muscle's function throughout a normal gait cycle.

Soleus and Gastrocnemius

The *Soleus* is a monoarticular muscle crossing the ankle joint. As a primary plantarflexor of the foot, the *Soleus* provides large push-off forces during the end of stance phase. The *Gastrocnemius* serves the same function as the *Soleus* at the ankle, though it crosses both the ankle and knee joints. Despite its biarticular property, the control strategy is similar.

Because the *Soleus* and *Gastrocnemius* are controlled by the same equations, we use the label m to correspond to the control of either muscle. In stance, each muscle is controlled by positive force feedback:

$$u_{m,st} = S_{0,m} + G_m F_m(t - \Delta t_{m,R}) + k_m \ddot{\theta}(t - \Delta t_{m,V}) \tag{6.3}$$

in which a gain (G_m) is multiplied by the force of the muscle (F_m). This force is subject to time delay ($t - \Delta t_{m,R}$), where $t_{m,R}$ represents the feedback delay between the muscle and the spinal cord. Included in the stance equation are the pre-stimulation term ($S_{0,m}$) and vestibular feedback gain (k_m). The pre-stimulation term ensures the muscle is always slightly activated. The vestibular gain is multiplied by the trunk's angular acceleration ($\ddot{\theta}$), subject to time delay ($t - \Delta t_{m,V}$). Vestibular delay $t_{m,V}$ represents the signal delay between the muscle and the cerebellum.

However, the *Soleus* and *Gastrocnemius* are largely inactive in the swing phase, reducing its control signal to pre-stimulation alone: $u_{m,sw} = S_{0,m}$.

Tibialis Anterior

A monoarticular muscle, the primary function of the *Tibialis anterior* (TA, hereafter) is to dorsiflex the foot and orient the toes upwards during the swing phase. Without this action, the foot would drop and drag during swing. In stance, the *TA* is also active since muscle co-contraction stabilizes joints during both static and dynamic tasks [36–38].

To capture ankle dorsiflexion during swing, the control signal of the *TA* is obtained through feedback from its fiber length:

$$u_{TA,sw} = S_{0,TA} + G_{TA,sw}[L_{CE,TA}(t - \Delta t_{TA}) - l_{off,TA}] \tag{6.4}$$

where $G_{TA,sw}$ is the swing gain, $L_{CE,TA}$ is the length of the muscle's contractile element, $S_{0,TA}$ is the pre-stimulation term, and $l_{off,TA}$ is the length offset of the muscle. The length feedback is subjected to the time delay between the muscle and spinal cord $(t - \Delta t_{TA})$.

In stance, the *TA* is also active, though inhibited by a negative feedback gain, G_{SOLTA}, multiplied by the force of the *Soleus*. This allows some stabilizing, antagonistic forces while ensuring the *Soleus* can provide the necessary push-off torque at the end of stance without being resisted by the *TA*. The stance-phase control signal becomes:

$$\begin{aligned} u_{TA,st} = S_{0,TA} + G_{TA,st}[L_{CE,TA}(t - \Delta t_{TA}) - l_{off,TA}] \\ -G_{SOLTA}F_{SOL} + k_{TA}\ddot{\theta}(t - \Delta t_{TA,V}) \end{aligned} \tag{6.5}$$

where $G_{TA,st}$ is the stance-phase gain and $k_{TA}\ddot{\theta}(t - \Delta t_{TA,V})$ is the vestibular feedback gain, subjected to a time delay between the muscle and cerebellum.

The splitting of *TA* length feedback control gains ($G_{TA,st}$ and $G_{TA,sw}$) is to prevent the tendency of the optimization to prioritize either the swing or stance phase, which would occur with a single control gain. With two gains, the *TA* can provide both stability in the stance phase and also dorsiflex during swing. In Geyer and Herr's model [14], the *TA* had one gain, which may prohibit its stabilizing capability during the stance phase.

106

6.2.4 Impedance Controller

Impedance control describes the relationship between input flow and output effort [26]. In a prosthetic device, the input is defined as the angular velocity, while the output is the torque. Stiffness and damping terms approximately represent the passive, spring-like properties of the muscle tendon and connective tissues [27].

We used the impedance controller described by Sup et al. [2], in which the controller cycles through four finite states (phases) and separate stiffness and damping parameters are used for each of the phases. Joint torque in phase i is generated as a function of the joint angle and velocity, using stiffness k_i, damping b_i, and equilibrium position of the virtual spring, θ_i:

$$\tau = -k_i(\theta - \theta_i) - b_i\dot{\theta} \tag{6.6}$$

However, a single spring-damper term would not provide the necessary active control during an entire gait cycle. Instead, impedance controllers typically separate gait into early stance, late stance, early swing, and late swing. The transition criteria for switching to the four phases is defined as follows:

- *Phase 1*: heel-strike

- *Phase 2*: ankle angle reaches a threshold for pre-swing

- *Phase 3*: ankle dorsiflexes after push-off

- *Phase 4*: the knee velocity changes direction to slow the foot for the next heel-strike

With four phase, the impedance controller has separate parameters, k, θ, and b, for each phase, totaling 12 parameters.

6.2.5 Optimization

We optimized controller parameters in MATLAB using a Covariance Matrix Adaption (CMA) algorithm [22] and a multi-objective cost function (6.7). CMA was selected because it is a fast, population-based algorithm and shooting methods must be used when the control is discontinuous. The tracking term of the cost function is the norm of the difference between the ankle torque τ_{vmr} produced by the candidate controller and the ankle torque τ_{exp} measured during the 8 minutes of perturbed walking ($N = 48,000$ samples). The effort term was modeled as the mean of the squared muscle activation (a), which promotes load-sharing between all three muscles (m). Weight terms W_1 and W_2 are selected to prioritize either the tracking or effort:

$$C = W_1 \sqrt{\frac{1}{N} \sum_{i=1}^{N} (\tau_{exp} - \tau_{vmr})^2} + W_2 \sqrt{\frac{1}{N} \sum_{i=1}^{N} a_m^2} \qquad (6.7)$$

For the VMR controller, we solve for 27 total parameters, including 12 control parameters (reflex and vestibular contributions), and an additional 5 muscle properties for each muscle. Muscle parameters were selected by running 10 optimizations on one representative walking trial. In each of these sets, one additional muscle property was included as an optimization parameter in addition to the 12 control parameters. The average RMS error between the experimental torque and torque from the model was calculated from the 10 optimizations. If the addition of the particular muscle parameter reduced this RMS tracking error, it was included as a parameter in the optimization. The selected muscle properties include: slack length of the SEE, slack length of the PEE, and the strain of the SEE at maximum isometric muscle force (u_{max}). Note that the time delays for both spinal reflexes and vestibular feedback were also used as optimization parameters. A summary of all optimization parameters is presented in Table VIII.

For the impedance controller, the cost function (6.7) has only the first term (tracking). We solve for 13 control parameters: stiffness (k_i), damping (b_i), and equilibrium position

Table VIII: Control parameters included in the optimization of the Virtual Muscle Reflex (VMR) controller on the ankle joint. Note that the muscle properties for each muscle were optimized.

Control Parameters for Optimization	
Reflex Control	$S_{0,SOL}$, $S_{0,TA}$, $S_{0,GAS}$, G_{SOL}, $G_{TA,sw}$, $G_{TA,st}$ G_{GAS}, $l_{off,TA}$, G_{SOLTA}
Vestibular Control	k_{SOL}, k_{TA}, k_{GAS}
Muscle Properties	SEE slacklength, PEE slacklength, u_{max}, reflex time delay, vestibular time delay

(θ_i) for each of the 4 gait phases, and also for the ankle angle threshold that determined the pre-swing transition.

The initial guess for the VMR optimization are the original reflex control parameters reported by Geyer and Herr [14] and muscle property values previously reported in literature [29]. The initial guess for the reflex time delays were approximately 30 ms for the muscles and 80 ms for the vestibular time delays. Similarly, the initial guess for the impedance control parameters are those outlined by Sup et al. [2]. Upper and lower bounds were determined by double and half of the initial guess. The number of generations and population size of the CMA controller was adjusted until the model converged and showed no further minimization of the cost function.

The performance of both the impedance and VMR controllers was quantified by separating each of the gait cycles into 100 different points. The coefficient of determination (R^2) and root-mean-square (RMS) error was calculated for all gait cycles, at each of the 100 points, to correlate how well the controller describes that instance of the gait cycle when compared to the torque exhibited by the test subject at the same time, across all gait cycles.

R^2 is calculated by dividing the sum of the residuals by the sum of squares, for each point in the gait cycle (i) where N is the number of gait cycles:

$$R_i^2 = 1 - \left[\frac{\sum_{i=1}^{N}(\tau_{exp,i} - \tau_{vmr,i})^2}{\sum_{i=1}^{N}(\tau_{exp,i} - \bar{\tau}_{exp,i})^2} \right] \tag{6.8}$$

The RMSE is calculated as the difference between τ_{exp} and τ_{vmr} at each of the 100

points, across all gait cycles:

$$RMSE_i = \sqrt{\frac{1}{N} \sum_{i=1}^{N} (\tau_{exp,i} - \tau_{vmr,i})^2} \qquad (6.9)$$

Lastly, we calculate the mean, standard deviation, maximum value, and minimum value for every parameter in the VMR controller and the impedance controller. After performing the optimizations, we will obtain 30 separate VMR controllers and 30 impedance controllers. By reviewing the range within their optimized parameters, we can observe the trends across subjects or speeds and detect areas which may require additional modifications for future studies.

6.3 Results

Figure 27 shows a representative result from one test subject (female, age = 21, mass = 58 kg, speed = 0.8 m/s), comparing ankle torques from the experiment (black), impedance controller (red), and VMR controller (blue-dashed). Negative ankle torque corresponds to plantarflexion and the peak torque denotes the timing before push-off. Only a few gait cycles of the 8-minute trial are shown. Both controllers produce variations in peak moment between gait cycles and the predicted joint torque reliably matches the timing of ankle push-off. In general, the impedance controller has more difficulty tracking during heel-strike, while the VMR controller can produce more torque than required during push-off.

The R^2 (top) and RMS error (bottom) are also shown in Figure 27, in which 60% of the gait cycle corresponds to push-off. Note that R^2 can be negative if the fit is worse than a constant, and higher R^2 indicates better correlation. For this test subject, results indicate that both controllers perform best at the end of stance and worst during the swing phase. The RMS error is also the largest at this time. Though the performance of the controllers vary between test subjects and walking speeds, this behavior is consistent for all trials.

Table IX summarizes the mean RMS error and maximum R^2 for both controllers for all
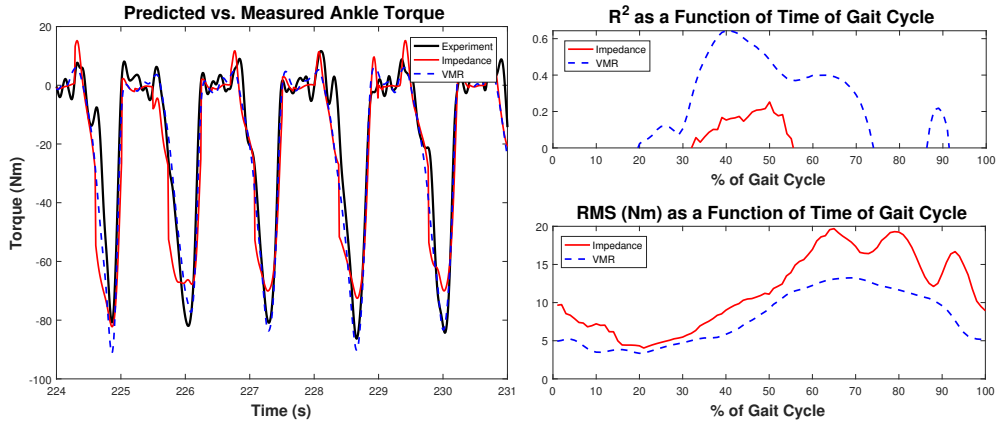
Figure 27: *Left*. Comparison of ankle torque measured from the experiment (black) and generated by the impedance controller (red) and the VMR controller (blue-dashed). Negative torque corresponds to ankle plantarflexion and the peak torque is the moment of push-off. *Right*. Correlation coefficient (top) and root-mean square error (bottom) as functions of the gait cycle, in which each gait cycle was split into 100 different data points. Better controllers have high correlation coefficients R^2 and lower root-mean-square (RMS) errors.

subjects and speeds. The VMR controller performs with less RMS error than the impedance controller in all experimental trials. Similarly, the R^2 for the VMR controller is higher than the impedance controller in 90% of the trials. Regardless, the RMS error is still lower or equivalent in these trials. Results indicate that the RMS error increases with faster walking speeds for both controllers.

Table X summarizes the results of the 30 optimized VMR controllers (10 subjects at three walking speeds), designated by the associated muscle. The mean, standard deviation, maximum, and minimum was calculated for all parameters. Parameters that experience the largest variation across test subjects are the vestibular feedback gains (k_m). A minimum value of 0.00 was obtained in a few trials, indicating the optimization selected a value at the lower bound for a particular test subject and speed. This tendency suggests that the mechanism by which the cerebellum responds to trunk acceleration for each muscle is not entirely known and may either inhibit or assist depending on the test subject's control strategy. Therefore, we should consider allowing this vestibular feedback to be either negative

Table IX: Summarized results of the optimized impedance and VMR controller. See text for details.

Trial		Mean RMS (Nm)		Max R^2	
Subject	Speed (m/s)	Impedance	VMR	Impedance	VMR
7	0.8	9.5	7.2	0.33	0.74
	1.2	9.84	9.25	0.60	0.78
	1.6	13.41	13.10	0.59	0.70
12	0.8	10.71	8.33	0.62	0.83
	1.2	10.70	10.41	0.73	0.86
	1.6	12.95	11.07	0.79	0.84
8	0.8	10.73	6.75	0.05	0.82
	1.2	14.94	9.98	0.49	0.82
	1.6	19.53	15.34	0.72	0.67
13	0.8	11.49	7.89	0.25	0.64
	1.2	13.91	8.72	0.19	0.69
	1.6	19.39	16.83	0.24	0.41
10	0.8	11.26	7.57	0.31	0.82
	1.2	10.72	8.91	0.68	0.84
	1.6	17.69	12.09	0.68	0.79
5	0.8	15.23	12.94	0.58	0.81
	1.2	20.26	14.67	0.53	0.75
	1.6	26.36	20.14	0.52	0.64
6	0.8	17.90	11.02	0.24	0.83
	1.2	19.86	15.23	0.61	0.79
	1.6	27.02	19.22	0.35	0.71
15	0.8	11.22	8.47	0.78	0.78
	1.2	13.14	13.13	0.86	0.75
	1.6	15.12	14.89	0.87	0.78
3	0.8	18.25	9.22	0.01	0.70
	1.2	19.5	10.04	0.54	0.85
	1.6	22.64	15.07	0.64	0.60
17	0.8	16.46	9.86	0.68	0.87
	1.2	20.89	13.32	0.66	0.85
	1.6	30.67	18.37	0.40	0.79

or positive. Additionally, the controller gains (G_m) also vary between subjects, though this can again be attributed to the test subject, where some individuals require more muscle activation to respond to the perturbations.

Table XI summarizes the results of the 30 optimized impedance controllers (10 subjects at 3 walking speeds), separated by gait phase. The mean, standard deviation, maximum, and minimum was calculated for all parameters. Note that many damping parameters are 0, as outlined in [2], since the *Soleus* and *Gastrocnemius* behave more similarly to a stiff

Table X: Statistical summary of optimized virtual muscle reflex (VMR) variables, including mean, standard deviation (std), minimum value (min), and maximum value (max). Parameters are separated by muscle: *Soleus*, *Tibialis anterior*, and *Gastrocnemius*.

Muscle	Variable	Mean	Std	Min	Max
Soleus	$s_{0,SOL}$	0.12	0.05	0.05	0.19
	G_{SOL}	1.58	0.60	0.55	2.20
	k_{SOL}	0.13	0.21	0.00	0.88
Tibialis anterior	$s_{0,TA}$	0.13	0.06	0.06	0.2
	$G_{TA,st}$	1.14	0.45	0.5	2.00
	$G_{TA,sw}$	0.80	0.37	0.35	1.40
	G_{SOLTA}	0.41	0.13	0.14	0.56
	$l_{off,TA}$	0.92	0.31	0.37	1.42
	k_{TA}	0.28	0.79	0.00	2.98
Gastrocnemius	$s_{0,GAS}$	0.08	0.04	0.05	0.17
	G_{GAS}	1.63	0.40	0.55	2.00
	k_{GAS}	1.28	0.91	0.00	2.84

spring during these gait phases. The largest variations in the parameters occur in the joint reference angles, discussed more thoroughly in Section 6.4.

Table XI: Optimized parameters of the impedance controller for the ankle, separated by gait phase. The mean, standard deviation (std), minimum (min), and maximum (max) of each parameter were calculated from each of the 30 walking trials.

Phase	Variable	Mean	Std	Min	Max
Mode 1: Heel-Strike	k_1	4.11	1.23	1.75	7.00
	b_1	0.00	0.00	0.00	0.00
	θ_{eq1}	-0.98	1.59	-2.2	2.76
Mode 2: Pre-swing	k_2	4.23	0.89	2.44	6.36
	b_2	0.00	0.00	0.00	0.00
	θ_{eq2}	-6.88	3.75	-9.00	8.13
Mode 3: Push-Off	k_3	0.31	0.14	0.15	0.6
	b_3	0.02	0.01	0.01	0.04
	θ_{eq3}	0.00	0.00	0.00	0.00
Mode 4: Late Swing	k_4	0.17	0.06	0.15	0.44
	b_4	0.00	0.00	0.00	0.00
	θ_{eq4}	0.00	0.00	0.00	0.00

6.4 Discussion

We presented a joint torque controller with virtual muscles, reflexes, and vestibular control. Though reflex controllers have been shown to produce able-bodied walking in both simulation and hardware [14, 15, 17], these controllers are manually tuned until stable walking is achieved. We conducted experiments to measure the ability of reflex controllers in replicating able-bodied walking and human control actions. Test subjects walked for 8 minutes under the effect of anteroposterior perturbations by randomly adjusting the velocity of the treadmill belts. We obtained the parameters of the VMR controller using a CMA optimization that minimized the difference between the torque calculated from the experiment and the torque generated from the VMR model. To compare to the current state-of-the-art P/O controller, we then repeated the procedure using a typical finite-state impedance controller with four walking phases. The performance of both controllers was quantified using the RMS error and R^2 as a function of the gait cycle.

Our results indicate that the VMR controller has less RMS error for all trials, and higher correlation coefficients in 90% of the trials. For 8 minutes of walking data, these statistics were calculated from approximately 450 gait cycles. Therefore, any R^2 higher than 0.5 indicates a high level of statistical significance at that instant of the gait cycle. In a few trials, maximum R^2 values exceeded 0.8, indicating that the controller can reliably replicate that portion of the gait cycle. Generally, both controllers show the best agreement with human responses during ankle push-off, and the worst agreement at heel-strike. Both controllers match the timing of push-off and show variations in response to the perturbations. Though these controllers do not always match the peak torque for every gait cycle, the variations suggest that they are replicating some of the reactions of the test subject. Both controllers have more difficulty tracking the data when the walking speed is increased.

The impedance controller exhibits discontinuity in the transitions between the four gait phases, similar to the observable movement of powered orthoses using similar control [4]. Because test subjects are perturbed, the thresholds for transitioning between the phases

of gait may never be reached as each gait cycle has different random perturbations. For instance, if the ankle angle is not large enough to begin the pre-swing phase (Phase 2), then the controller will remain in the heelstrike phase (Phase 1) for that entire gait cycle. A more reliable method of switching between the gait phases will likely produce better results; however, our goal was to compare the VMR controller to the current state-of-the-art impedance controller, and not to make improvements to that control. The optimized impedance controller parameters were comparable to those previously described by Sup et al. [2].

Lower correlation coefficients during the stance phase and heelstrike do not suggest that the controllers perform poorly at those times. As the statistics were calculated from approximately 450 gait cycles, any noise in the data, or inconsistencies exhibited by the test subject, could prohibit the controllers from matching this portion of the gait cycle. Alternatively, the perturbations may not be large enough to elicit stretch reflexes in the lower-leg during the first part of the stance phase [39], but the plantarflexion of the ankle during push-off is large enough to trigger the stretch reflexes. This could also apply to the vestibular feedback terms, in that the trunk is not accelerated enough to cause a noteworthy balance correction. However, our addition of vestibular feedback on the trunk acceleration demonstrated a slight improvement during the stance phase and resulted in better tracking.

Splitting the gain term in the *TA* reflex equations also showed an improvement over the original controller [14]. With only one gain, the optimization would prioritize the swing phase and reduce the gain to prevent the *TA* from acting against the *Soleus*, though this lower gain limited dorsiflexion before heel-strike. However, with a separate gain in the swing and stance phases, the TA was able to more consistently provide the necessary dorsiflexion.

Additionally, since the perturbations were continuously applied throughout the trial, our method is incapable of separately determining the controller's performance for normal walking motions or for the test subject's reactions to the perturbations. The controller may

be responding to inconsistencies within normal gait cycles, as well as correcting for some perturbations. However, with low RMS error and high R^2 across the same point for all gait cycles, it can be suggested that the VMR controller achieves able-bodied walking in simulation and can react to everyday perturbations encountered during walking.

Further study includes expanding the analysis to include more joints, muscles, and reflexes in the leg (hip and knee) and repeating the optimization and statistical analysis. For more natural control, the addition of a flexible or articulating toe joint could also be considered. For a more complete assessment, the controller performance should be tested in physical hardware. However, our study suggests that reflex controllers are capable of explaining human control actions during perturbed gait, and can reproduce normal walking motions more effectively than the current control strategies used to actuate an ankle joint.

Acknowledgments

6.5 REFERENCES

[1] B. E. Lawson, B. Ruhe, A. Shultz, and M. Goldfarb. A powered prosthetic intervention for bilateral transfemoral amputees. *IEEE Transactions on Biomedical Engineering*, 62(4):1042–1050, April 2015.

[2] F. Sup, A. Bohara, and M. Goldfarb. Design and control of a powered transfemoral prosthesis. *International Journal of Robotics Research*, 27(2):263–273, February 2008.

[3] F. C. Sup. *A powered self-contained knee and ankle prosthesis for near normal gait in transfemoral amputees*. PhD thesis, Vanderbilt University, 2009.

[4] R. J. Farris, H. A Quintero, S. A. Murray, , K. H. Ha, C. Hartigan, and M. Goldfarb. A

preliminary assessment of legged mobility provided by a lower limb exoskeleton for persons with paraplegia. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 22(3):482–490, May 2014.

[5] G. Khademi, H. Mohammadi, H. Richter, and D. Simon. Optimal mixed tracking/impedance control with application to transfemoral prostheses with energy regeneration. *IEEE Transactions on Biomedical Engineering*, 65(4):894–910, April 2018.

[6] H. Warner. Optimal design and control of a lower-limb prosthesis with energy regeneration. Master's thesis, Cleveland State University, 2015.

[7] M. R. Tucker, J. Olivier, A. Pagel, H. Bleuler, M. Bouri, O. Lambercy, J. del R. Millán, R. Riener, H. Vallery, and R. Gassert. Control strategies for active lower extremity prosthetics and orthotics: A review. *Journal of NeuroEngineering and Rehabilitation*, 12(1):1, December 2015.

[8] K. G. Gerritsen, A. J. van den Bogert, M. Hulliger, and R. F. Nernicke. Intrinsic muscle properties facilitate locomotor control - A computer simulation study. *Motor Control*, 2(3):206–220, July 1998.

[9] A. J. van den Bogert, D. Blana, and D. Heinrich. Implicit methods for efficient musculoskeletal simulation and optimal control. *Procedia IUTAM*, 2:297–316, January 2011.

[10] C. Capaday. The special nature of human walking and its neural control. *TRENDS in Neurosciences*, 25(7):370–376, July 2002.

[11] J. L. Pons, J. C. Moreno, D. Torricelli, and J. S. Taylor. Principles of human locomotion: A review. In *35th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, pages 6941–6944, July 2013.

[12] J. B. Nielsen and T. Sinkjaer. Afferent feedback in the control of human gait. *Journal of Electromyography and Kinesiology*, 12(3):213–217, June 2002.

[13] A. Prochazka and D. Gillard. Sensory control of locomotion. In *Proceedings of the 1997 American Control Conference*, volume 5, pages 2846–2850, 1997.

[14] H. Geyer and H. Herr. A muscle-reflex model that encodes principles of legged mechanics produces human walking dynamics and muscle activities. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 18(3):263–273, June 2010.

[15] S. Song and H. Geyer. A neural circuitry that emphasizes spinal feedback generates diverse behaviours of human locomotion. *The Journal of Physiology*, 593(16):3493–3511, August 2015.

[16] M. F. Eilenberg, H. Geyer, and H. Herr. Control of a powered ankle-foot prosthesis based on a neuromuscular model. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 18(2):164–173, April 2010. doi: 10.1109/TNSRE.2009. 2039620.

[17] N. Thatte and H. Geyer. Toward balance recovery with leg prostheses using neuromuscular model control. *IEEE Transactions on Biomedical Engineering*, 63(4): 904–913, May 2016.

[18] A. and H. Geyer. The 3-D spring-mass model reveals a time-based deadbeat control for highly robust running and steering in uncertain environments. *IEEE Transactions on Robotics*, 29(5):1114–1124, May 2013.

[19] H. Geyer, A. Seyfarth, and R. Blickhan. Positive force feedback in bouncing gaits? *Proceedings of the Royal Society of London B: Biological Sciences*, 270(1529):2173–2183, October 2003.

[20] F. Dzeladini, A. R. Wu, D. Renjewski, A. Arami, E. Burdet, E. van Asseldonk, H. van der Kooij, and A. J. Ijspeert. Effects of a neuromuscular controller on a powered ankle exoskeleton during human walking. In *2016 6th IEEE International Conference on Biomedical Robotics and Biomechatronics (BioRob)*, pages 617–622, June 2016.

[21] A. Ostermeier, A. Gawelczyk, and N. Hansen. A derandomized approach to self-adaptation of evolution strategies. *Evolutionary Computation*, 2(4):369–380, 1994.

[22] N. Hansen and A. Ostermeier. Completely derandomized self-adaptation in evolution strategies. *Evolutionary Computation*, 9(2):159–195, June 2001.

[23] J. K. Moore, S. K. Hnat, and A. J. van den Bogert. An elaborate data set on human gait and the effect of mechanical perturbations. *PeerJ*, 3:e1248, 4 2015.

[24] D. A. Winter. *Biomechanics and Motor Control of Human Movement*. John Wiley and Sons, New York, 1990.

[25] S. K. Hnat and A. J. van den Bogert. Inertial compensation for belt acceleration in an instrumented treadmill. *Journal of Biomechanics*, 47(15):3758–3761, November 2014.

[26] N. Hogan. The mechanics of multi-joint posture and movement control. *Biological Cybernetics*, 52(5):315–331, September 1985.

[27] M. Millard, T. Uchida, A. Seth, and S. L. Delp. Flexing computational muscle: Modeling and simulation of musculotendon dynamics. *ASME Journal of Biomechanical Engineering*, 135(2):0210051–02100511, February 2013.

[28] E. Vinnars, J. Bergstöm, and P. Fürst. Influence of the postoperative state on the intracellular free amino acids in human muscle tissue. *Annals of Surgery*, 182(6): 665–671, December 1975.

[29] K. G. M. Gerritsen, W. Nachbauer, and A. J. van den Bogert. Computer simulation of landing movement in downhill skiing: Anterior cruciate ligament injuries. *Journal of Biomechanics*, 29(7):845–854, July 1996.

[30] M. Roche. Rosenbrock methods for differential algebraic equations. *Numerische Mathematik*, 52(1):45–63, January 1987.

[31] H. H. Rosenbrock. Some general implicit processes for the numerical solution of differential equations. *The Computer Journal*, 5(4):329–330, January 1963.

[32] R. M. N. Alexander. Energetics and optimization of human walking and running: The 2000 Raymond Pearl memorial lecture. *Journal of Human Biology*, 14(5):641–648, September 2002.

[33] M. J. Grey, J. B. Nielsen, N. Mazzaro, and T. Sinkjær. Positive force feedback in human walking. *The Journal of Physiology*, 581(Pt 1):99–105, March 2007.

[34] E. R. Kandel, J. H. Schwartz, T. M Jessel, and S. A. Siegelbaum. *Principles of Neural Science*. McGraw-Hill, New York, 2000.

[35] S. C. Kim, J. Y. Kim, H. N. Lee, H. H. Lee, J. H. Kwon, N. B. Kim, M. J. Kim, J. H. Hwang, and G. C. Han. A quantitative analysis of gait patterns in vestibular neuritis patients using gyroscope sensor and a continuous walking protocol. *Journal of NeuroEngineering and Rehabilitation*, 11(1):58, April 2014.

[36] E. Todorov. Direct cortical control of muscle activation in voluntary arm movements: A model. *Nature Neuroscience*, 3:391–398, April 2000.

[37] S. Hirokawa, M. Solomonow, Z. Luo, Y. Lu, and R. D'Ambrosia. Muscular co-contraction and control of knee stability. *Journal of Electromyography and Kinesiology*, 1(3):199–208, September 1991.

[38] S. Heitmann, N. Ferns, and M. Breakspear. Muscle co-contraction modulates damping and joint stability in a three-link biomechanical limb. *Front Neurorobot*, 5:5, January 2011.

[39] L. H. Sloot, J. C. Van den Noort, M. M. Van der Krogt, S. Bruijn, and J. Harlaar. Can treadmill perturbations evoke stretch reflexes in the calf muscles? *PLoS ONE*, 10(12): e0144815, December 2015.

CHAPTER VII

REAL-TIME VIRTUAL MUSCLE REFLEX CONTROL AND PHASE-BASED
IMPEDANCE CONTROL FOR A POWERED EXOSKELETON

After our initial simulations of the VMR controller on the ankle joint, we then expanded the model to include both the knee and hip joints and repeated the procedure from Part II, Chapter VI. Here, we place particular emphasis on the full VMR controller and detail the modifications made to Geyer's reflex controller in order to better represent our experimental data. We then compared an impedance controller to the VMR system. Because some methods are identical to our previous study (Part II, Chapter VI), we abridge them here for brevity, especially the details concerning the perturbation data and muscle model.

ABSTRACT

A biologically-inspired actuation system, including muscles, spinal reflexes, and vestibular feedback, may be capable of achieving more natural gait mechanics in powered prostheses or exoskeletons. In this study, we developed a Virtual Muscle Reflex (VMR) system to control ankle torque and tuned it using data from human responses to anteroposterior mechanical perturbations at three walking speeds. The system consists of seven Hill-Type muscles, simulated in real time, and uses feedback from ground reaction forces and from stretch sensors on the virtual muscle fibers. Controller gains, muscle properties, and reflex and vestibular time delays were optimized using Covariance Matrix Adaptation (CMA) to minimize the difference between the VMR torque output and the torque measured from the experiment. We repeated the procedure using a conventional finite-state impedance controller. For both controllers, the coefficient of determination (R^2) and root-mean-square (RMS) error was calculated as a function of the gait cycle. The mean RMS error of the hip, knee, and ankle joint were 16.52 Nm, 15.93 Nm, and 14.51 Nm for the impedance controller, and 16.11 Nm, 15.91 Nm, and 13.44 Nm for the VMR controller. The maximum R^2 for each joint was 0.11, 0.15, and 0.60 for the impedance controller, and 0.17, 0.50, and 0.17 for the VMR controller. With similar RMS error, it was concluded that the VMR controller can reproduce characteristics of human walking in response to perturbations as effectively as an impedance controller.

7.1 Introduction

Phase-based impedance controllers are the current standard for powered prosthetic and orthotic devices [1–4], in which the joint torque is calculated as a function of the joint angle and joint velocity. However, discrete switching between controller phases can lead to unnatural gait mechanics. Additionally, these type of impedance controllers are linear and may not be able to capture the nonlinear nature of human walking, nor are they tuned

123

using information from human walking experiments. Recently, optimization techniques have been used to determine the parameters of both a constant and time-varying impedance controller [5] and a lower-limb prosthetic with energy regeneration [6] using human gait data.

An intuitive approach for designing smoother P/O controllers is to model the human biological control system. Yet, the various strategies of the brain and central nervous system (CNS) for controlling human motion is a challenging area of ongoing research and is difficult to achieve in real-time applications [7]. A simpler strategy is to design the controller using virtual muscles and spinal reflexes.

It has been shown through simulation that muscle properties contribute substantially to stability during walking when compared to direct torque-driven control [8]. The same spring-damper properties of an impedance controller can be achieved in a virtual muscle model and the passive, spring-like properties of muscles prohibit unnatural behaviors arising from unstable torque commands. Muscle dynamics can be replicated by the standard 3-element Hill model, and it was recently shown that real-time muscle simulation, as required for use in P/O devices, is possible [9]. Joint torque can then be obtained by multiplying the muscle force by the distance between the attachment point of the muscle and the joint.

As a controller for these virtual muscles, spinal reflexes are responsible for stabilizing gait, responding to perturbations, and assisting in balance correction [10]. The activation signal generated by reflexes is sent to the motor neurons in the muscle to produce movement through muscle contractions [11, 12]. Feedback signals are then sent back to the spinal cord via afferent neurons in the muscle spindles, Golgi tendon organs, and other receptors on the skin which stimulate spinal reflexes [13].

An autonomous, neuromuscular reflex controller has been shown to produce stable walking in simulated models [14, 15]. Though the simulated model can walk at various speeds and slopes, it differs from actual human walking mechanics, especially large verti-

cal ground reaction force (GRF) at heel-strike due to increased hip extension during stance. Reflex control has been implemented in the BioM actuated ankle joint [16] and an actuated knee and ankle prostheis [17]. The model uses positive-feedback from the muscle (force or length) and proportional-derivative (PD) control for maintaining postural stability of the trunk. Though positive feedback is usually avoided in classical control design, it has been shown that humans use positive feedback as a stabilizing mechanism during walking [18, 19]. Recently, a reflex controller was evaluated on an exoskeleton with an actuated ankle, resulting in a reduction of net metabolic cost during rehabilitation exercises [20].

It has been suggested that reflex controllers can more readily adapt to environmental disturbances than the impedance controllers used in many powered P/O devices [17]. Since reflex models solely rely on muscle force feedback, any disturbance in the environment (sloped terrain or external perturbations) will be sensed by reflexes due to the reactive change in muscle forces or lengths [15].

Though these reflex controllers produce natural gait in both simulated environments [14] and commercialized hardware, current models have not been tuned or validated using human walking data obtained through experiments. Optimization techniques can determine the parameters which minimize the error between the joint torque exhibited by the subject and the torque generated from the model [21, 22].

In this study, we investigate the possibility of using a Virtual Muscle Reflex (VMR) system in order to determine if muscle reflexes are capable of reproducing able-bodied walking exhibited by test subjects. In addition, by applying anteroposterior mechanical perturbations, the respective reactive actions of the test subject can be used to define the nature of the controller. This will demonstrate the controller's ability in replicating the variations in human joint moments within and between gait cycles when subjected to random perturbations. We then compare the performance of the VMR system to a typical phase-based impedance controller tuned using the same optimization technique and experimental dataset.

7.2 Methods

Walking data from 10 participants (3 females and 7 males, age: 24 ± 4 years, height: 1.74 ± 0.09 m, mass: 74.4 ± 12 kg) was used in the study [23]. The test subjects walked for 8 minutes on an instrumented treadmill at 0.8, 1.2, and 1.6 m/s and were perturbed using random belt acceleration signals generated from discrete-time Gaussian white noise. The variance of the signal was adjusted until the perturbations were within 10% of the mean walking speed. Joint angles and torques were calculated through standard inverse 2D analysis [24] using joint positions from motion capture and the measured GRFs, after inertial compensation [25]. For more details concerning the experimental protocol, reference Part I, Chapter II and [26].

In this study, we used a planar model of the lower leg with 7 muscles in the sagittal plane to represent a P/O device with an actuated ankle, knee, and hip joint. Each joint must be capable of moving in both directions (flexion and extension). In Figure 28, we show the muscle pairs capable of performing these actions at each joint: *Gastrocnemius*, *Soleus*, *Tibialis anterior*, *Vasti*, *Hamstrings*, *Gluteals*, and *Iliopsoas*.

Muscle contraction and activation can be described by a 3-element Hill muscle model, including a contractile element (CE), based on standard force-length and force-velocity relationships, series and parallel elastic element (SEE/PEE), modeled as nonlinear springs, and a small amount of viscous damping for numerical stability. The nominal properties of these muscles were obtained from previous literature [27]. Inputs to the muscle model include the muscle length (derived from the joint angles) and neural excitation signals from the controller. Muscle contraction and activation dynamics were formulated as a set of first-order implicit differential equations (IDE) and were simulated in MATLAB® (Mathworks, Santa Rosa, California) using a first-order, implicit Rosenbrock solver [9]. The output ankle torque was obtained by multiplying the force generated by each muscle with the moment arms. For more details regarding the muscle model used to generate joint torque, refer to Part II, Chapter VI, Section 6.2.2 and Figure 26.
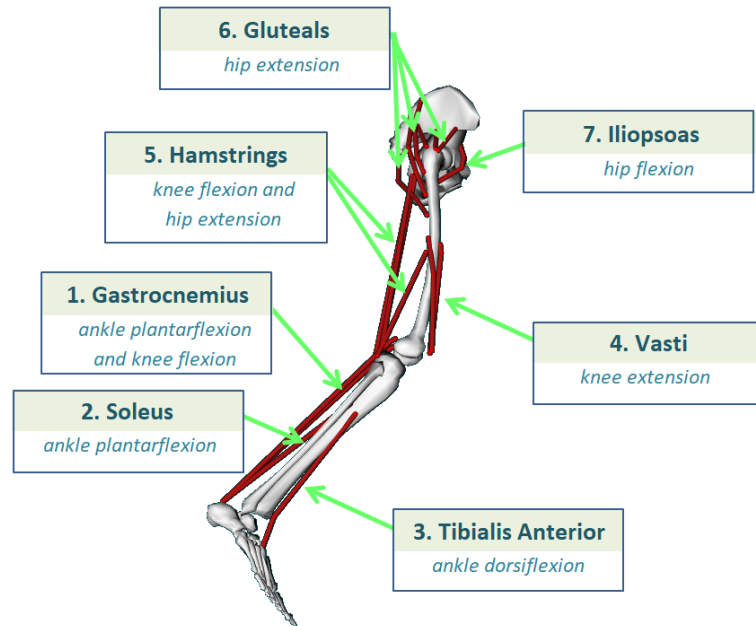
Figure 28: Muscles used in the Virtual Muscle Reflex (VMR) model, in which muscle pairs are capable of producing flexion and extension at the hip, knee and ankle joints. These muscles include: *Gastrocnemius*, *Soleus*, *Tibialis anterior*, *Vasti*, *Hamstrings*, *Gluteals*, and *Iliopsoas*.

7.2.1   Virtual Muscle Reflex (VMR) Controller

Here, we describe our modified reflex equations based on Geyer and Herr's model [14]. We detail how our modifications aim to capture each muscle's function throughout a normal gait cycle to better represent the joint torques measured from our experiments. The reflex controller has two gait phases for each muscle, *stance* and *swing*. We determine the phase based on the magnitude of the vertical GRF, where the transition threshold is 10 N. In all controller equations, the abbreviations st and sw denote stance and swing phases, respectively.

Physiologically, force and length feedback represent the Golgi tendon organs and muscle spindles, which provide both muscle length and velocity feedback to the central nervous system [28]. Trunk PD control can be correlated to the actions of the vestibular system, while double-support feedback terms are based on sensory information from the feet using GRF feedback. Any joint position control cannot be compared to an actual physiological function. These terms were added to further control specific portions of the gait cycle,

especially swing leg inhibition (details to follow).

Soleus, Gastrocnemius, and Tibialis Anterior

Reflex equations for the *Gastrocnemius* and *Soleus* during the stance phase can be described as a function of the muscle's pre-stimulation ($S_{0,m}$) and a time-delayed force (F_m) multiplied by a gain (G_m), and vestibular feedback gain (k_m) on trunk acceleration ($\ddot{\theta}$), in order to produce powerful plantarflexion during push-off:

$$u_{m,st} = S_{0,m} + G_m F_m(t - \Delta t_{m,R}) + k_m \ddot{\theta}(t - \Delta t_{m,V}) \tag{7.1}$$

Note the two separate time delays for reflexes ($t - \Delta t_{m,R}$) and vestibular feedback ($t - \Delta t_{m,V}$). In the swing phase, these muscles are controlled by the pre-stimulation alone, $u(t, sw) = S_m$, since they do not actively contribute during swing.

For the *Tibialis anterior* (TA) during stance, the control signal can be described as:

$$u_{TA,st} = S_{0,TA} + G_{TA,st}[L_{CE,TA}(t - \Delta t_{TA}) - l_{off,TA}]$$
$$-G_{SOLTA}F_{SOL} + k_{TA}\ddot{\theta}(t - \Delta t_{TA,V}) \tag{7.2}$$

where $G_{TA,st}$ is the stance gain, $L_{CE,TA}$ is the length of the muscle's contractile element, $S_{0,TA}$ is the pre-stimulation term, $l_{off,TA}$ is the length offset of the muscle. In single-support stance phase, antagonist muscle pairs exhibit co-contraction for stabilization [29–31]. To dissuade the *TA* from limiting plantarflexion during push-off, a negative feedback gain, G_{SOLTA}, is multiplied by the force of the *Soleus*.

In swing phase, the inhibition from the *Soleus* and vestibular feedback terms k_{TA} are removed from the control.

For a more thorough description of ankle reflexes and their muscle function, refer to Part II, Chapter VI, Section 6.2.3.

Vasti

The *Vasti* group, part of the quadriceps, are a monoarticular muscle group comprised of three muscles (*Vastus lateralis/medialis/intermedialis*). During the stance phase, the *Vasti* stabilize the leg in single-support, and provide some assistance when extending the knee during swing phase for proper foot placement during heel-strike.

In swing phase, the *Vasti* control equation is the muscle pre-stimulation ($S_{0,VAS}$) and a time-delayed force (F_{VAS}) multiplied by a gain ($G_{VAS,sw}$):

$$u_{VAS,sw} = S_{0,VAS} + G_{VAS,sw}F_{VAS}(t - \Delta t_{VAS,R}) \tag{7.3}$$

The stance-control equation can be represented by four key features: (1) force feedback, (2) preventing the knee from hyperextending, (3) turning off the muscle during the double-support phase, and (4) vestibular feedback. Due to the increasing complexity of the reflex control equations, we describe each stance-phase term separately, for clarity:

(1) Force feedback: a pre-stimulation term ($S_{0,VAS,st}$) and a gain ($G_{VAS,st}$) multiplied by the force of the muscle (F_{VAS}) subject to time delay ($t - \Delta t_{m,R}$), where $t_{VAS,R}$ represents the feedback delay between the muscle and the spinal cord:

$$u_{VAS,st} = S_{0,VAS} + G_{VAS,st}F_{VAS}(t - \Delta t_{VAS,R}) \tag{7.4a}$$

(2) Hyperextension inhibition: when the knee angle (ϕ) reaches a threshold (ϕ_{off}, approximately 20° flexion), and the knee velocity ($\dot{\phi}$) is positive (extension), the Vasti are inhibited by negative feedback gain, k_ϕ. This is to prevent the knee from hyperextending during stance:

$$- k_\phi[(\phi - \phi_{off})](\phi > \phi_{off})(\dot{\phi} > 0) \tag{7.4b}$$

(3) Double support inhibition: near the end of stance, before push-off, the opposite leg heel-strikes and both legs are in contact with the ground (double-support). The *Vasti* stop activating in double-support to allow knee flexion prior to swing. The inhibition term is a body-weight gain ($k_{bw,VAS}$) multiplied by the vertical GRF in the other leg (GRF_{contra}), during double-support only (D_{sup}, either 0 or 1):

$$- k_{bw,VAS}GRF_{contra}D_{sup} \tag{7.4c}$$

(4) Vestibular Feedback: a vestibular feedback gain (k_{VAS}) on trunk acceleration ($\ddot{\theta}$), subjected to a time delay between the muscle and cerebellum ($t - \Delta t_{VAS,V}$):

$$+ k_{VAS}\ddot{\theta}(t - \Delta t_{VAS,V}) \tag{7.4d}$$

Note that any experimental data used as feedback terms in these equations are delayed with respect to the reflex time delay $\Delta t_{VAS,R}$. With the *Vasti*, these parameters include ϕ, $\dot{\phi}$, and GRF_{contra}.

The splitting of *Vasti* force feedback control gains ($G_{VAS,st}$ and $G_{VAS,sw}$) is to prevent the tendency of the optimization to prioritize either the swing or stance phase, which would occur with a single control gain. Unlike Geyer and Herr's original controller [14], we added a force feedback term during the swing phase. Though inertia and gravity drive the majority of knee extension in swing, this additional force feedback provides the knee extension torque as observed in our test subjects. With two force feedback gains, the *Vasti* can provide both stability in the stance phase and provide some additional force during swing.

Hamstrings

The *Hamstrings* are a biartciular muscle group containing four separate muscles (*Biceps femoris short/long head, Semitendinosus, Semimembranosus*), which cross both the knee

and hip joints. In general, the *Hamstrings* flex the knee, extend the hip, and help stabilize the trunk. In stance phase, the primary function of the *Hamstrings* is to provide extension during early stance to pull the supported leg underneath the torso. They also flex the knee prior to push-off. In swing, the *Hamstrings* become active prior to heel-strike to slow the forward momentum of the leg.

The stance-control equation can be represented by four key features: (1) force feedback, (2) GRF feedback, (3) increased control during early stance, and (4) vestibular feedback. We describe each stance-phase term separately, for clarity:

(1) Trunk PD Control: as a hip extensor, the *Hamstrings* are used to stabilize the upper body. This action is captured through a pre-stimulation term ($S_{0,HAM}$) with proportional gain (k_p) on trunk forward lean (θ) and derivative gain (k_d) on trunk forward velocity ($\dot{\theta}$). A trunk forward lean angle of $10°$ is used as the position reference, θ_{ref}:

$$u_{HAM,st} = S_{0,HAM} + k_p[(\theta - \theta_{ref})] + k_d\dot{\theta} \tag{7.5a}$$

(2) Body-Weight Sensing: at heel-strike, the hip extensors activate in order to draw the stance leg underneath the torso. An activation term ensures the *Hamstrings* produce the necessary extension torque in stance. To represent this action, a body-weight gain ($k_{bw,HAM}$) is multiplied by the GRF of its own leg (GRF_{ipsi}):

$$+ k_{bw,HAM}GRF_{ipsi} \tag{7.5b}$$

131

(3) Early Stance Control: the body-weight term alone is not sufficient to maintain the required muscle activation during single-support. Since body-weight is a large signal, the $k_{bw,HAM}$ gain must be small to prevent too large of an extension torque at heelstrike. To separate early-stance from late-stance, we add an activation term based on the hip angle position, which allows the *Hamstrings* to remain active. A hip position gain ($k_{\gamma,st}$) is multiplied by the hip angle (γ) based on a reference hip angle (γ_{off}):

$$+ k_{\gamma,st}[(\gamma - \gamma_{off})](\gamma > \gamma_{off}) \tag{7.5c}$$

(4) Vestibular Feedback: a vestibular feedback gain (k_{HAM}) on trunk acceleration ($\ddot{\theta}$), subjected to a time delay between the muscle and cerebellum ($t - \Delta t_{HAM,V}$):

$$+ k_{HAM}\ddot{\theta}(t - \Delta t_{HAM,V}) \tag{7.5d}$$

In the swing phase, the PD-controller and double-support terms are removed from the controller. The control signal simplifies to force feedback, as previously described. Because the *Hamstrings* must slow the swing leg in preparation for heel-strike, we separate the early-swing and late-swing phases by introducing a proportional gain, γ_{sw}, based on the hip angular position:

$$u_{HAM,sw} = S_{0,HAM} + G_{HAM}F_{HAM}(t - \Delta t_{HAM,R}) + k_{\gamma,sw}[(\gamma - \gamma_{off})](\gamma > \gamma_{off}) \tag{7.6}$$

Geyer and Herr's original controller does not include a hip position control, discussed more thoroughly in Section 7.4.

Note that any experimental data used as feedback terms in these equations are delayed with respect to the reflex time delay, $\Delta t_{HAM,R}$. With the *Hamstrings*, these parameters include θ, $\dot{\theta}$, GRF_{ipsi}, γ, and F_{HAM}.

Gluteals

Powerful hip extension is possible due to the *Gluteals*, a network of strong muscles crossing the hip joint (*Gluteus maximus*, *Gluteus medius*, and *Gluteus minimus*). These muscles also stabilize the hip joint, pelvis, and trunk. During the stance phase, the *Gluteals* draw the stance leg under the torso, while in swing phase, they mitigate the forward momentum of the leg.

This stance-control equation can be represented by six key features: (1) force feedback, (2) trunk PD control, (3) body-weight sensing, (4) double-support excitation, (5) early stance control, and (6) vestibular feedback. Because this controller shares many features with the *Hamstrings* described in 7.5, we present the stance-phase equation for the *Gluteals* in its entirety,

$$u_{GLU,st} = S_{0,GLU} + G_{GLU,st}F_{GLU}(t - \Delta t_{GLU,R}) + (k_{p,h}[(\theta - \theta_{ref})] + k_{d,h}\dot{\theta})$$

$$+k_{bw,GLU}GRF_{ipsi} - \Delta SD_{sup} + k_{\gamma,st}[(\gamma - \gamma_{off})](\gamma > \gamma_{off}) + k_{GLU}\ddot{\theta}(t - \Delta t_{GLU,V})$$

$$(7.7)$$

where ΔS is a constant term added to the control during double-support phase at push-off, or when D_{sup} is 1. This inhibition prevents the *Gluteals* from producing large forces at the end of stance, prior to push-off, when the hip should begin swinging forward. Note the separate proportional and derivative gain for trunk stability, $k_{p,h}$ and $k_{d,h}$, respectively.

In swing, the control equation reduces to force feedback and hip position control. The force feedback gain is split between the swing and stance phases:

$$u_{GLU,sw} = S_{0,GLU} + G_{GLU,sw}F_{GLU}(t - \Delta t_{GLU,R}) + 4.0k_{\gamma,sw}[(\gamma - \gamma_{off})](\gamma > \gamma_{off}) \quad (7.8)$$

Though Geyer and Herr's original controller [14] did not include a force feedback term

133

during the stance phase, this addition helped provide the extension torques as exhibited by our test subjects during this gait phase. Also, the hip position gain is scaled by a factor of four to increase the effect of this term over other muscles sharing the feature. As with the other equations, any feedback term must be delayed with respect to the reflex time delay, $\Delta t_{GLU,r}$. For the *Gluteals*, these terms include: F_{GLU}, θ, $\dot{\theta}$, GRF_{ipsi}, and γ.

Iliopsoas

The hip flexor, the *Iliopsoas*, is a combination of two muscles, *Iliacus* and *Psoas major*. Though these two muscles have different functions in 3-D gait analysis, their tendons both insert onto the lesser trochanter of the femur. For planar walking motions, their control action is the same. In swing, the *Iliopsoas* propels the leg forward, while in stance, it acts as a stabilizer of the leg and trunk.

The stance equation consists of 6 different features: 1) length feedback, 2) PD control of the trunk, 3) body-weight sensing, 4) double-support excitation, 5) hip position control, and 6) vestibular feedback. Because the *Iliopsoas* controller shares its features with other muscles, we present the stance phase in its entirety:

$$
\begin{aligned}
u_{ILI,st} = S_{0,ILI} + G_{ILI,st}[L_{CE,ILI}(t - \Delta t_{ILI}) - l_{off,ILI}] + (k_p[(\theta - \theta_{ref})] + k_d\dot{\theta}) \\
-k_{bw,ILI}GRF_{ipsi} + \Delta SD_{sup} - k_{\gamma,st}[(\gamma - \gamma_{off})](\gamma > \gamma_{off}) + k_{ILI}\ddot{\theta}(t - \Delta t_{ILI,V})
\end{aligned}
\tag{7.9}
$$

where $l_{off,ILI}$ is the length offset. Note the sign changes in the controller features, particularly the GRF sensing ($k_{bw,ILI}GRF_{ipsi}$) and double-support terms (ΔSD_{sup}). During stance, the *Iliopsoas* is inhibited as to not interfere with extension torque, though the muscle is excited at late stance double-support to begin swinging the leg forward. Additionally, the hip position control suppresses the muscle in order to inhibit excessive co-contraction during early stance.

In swing phase, the control signal contains a term unique to the *Iliopsoas*, in which

a forward trunk lean gain (k_{lean}) is multiplied by the trunk angle at the point of push-off (PPO). Intuitively, when a person is falling forward, the instinct is to quickly swing the leg forward, represented by this additional term.

Also, an inhibition from the *Hamstrings* is included, where gain G_{HAMILI} is multiplied by the length of the *Hamstrings* ($L_{CE,HAM}$) subject to a length offset, $l_{off,HAM}$. The inhibition suppresses the momentum of the swing-leg. However, the hip position control is a more effective method of mitigating the leg's forward velocity. Other than these features, the swing controller for the *Iliopsoas* shares strategies previously described for the other muscles and is presented here in its entirety:

$$u_{ILI,sw} = S_{0,ILI} + G_{ILI,sw}[L_{CE,ILI}(t - \Delta t_{ILI}) - l_{off,ILI}] + k_{lean}[(\theta_{PPO} - \theta_{ref})]$$
$$-G_{HAMILI}[(L_{CE,HAM} - l_{off,HAM}] - k_{\gamma,sw}(\gamma - \gamma_{off})(\gamma > \gamma_{off})$$

$$(7.10)$$

Similar to the other muscle relationships, we separated the feedback gain between the swing and stance phase. This prohibited the optimization from favoring one phase. Note any parameter used as feedback terms must be delayed with respect to the reflex time delay, $\Delta t_{ILI,r}$. For the *Iliopsoas*, these terms include: $L_{CE,ILI}$, θ, $\dot{\theta}$, GRF_{ipsi}, γ, θ_{PPO}, and $L_{CE,HAM}$.

7.2.2 Impedance Controller

Impedance control describes the relationship between input flow and output effort [32]. In a prosthetic device, the input is defined as the angular velocity, while the output is the torque. Stiffness and damping terms approximately represent the passive, spring-like properties of the muscle tendon and connective tissues [33].

We used the impedance controller described by Sup et al. [2], in which the controller cycles through four finite states (phases), where separate stiffness and damping parameters

135

are used for each of the phases . Joint torque in phase i is generated as a function of the joint angle and velocity, using stiffness k_i, damping b_i, and equilibrium position of the virtual spring, θ_i:

$$\tau = -k_i(\theta - \theta_i) - b_i\dot{\theta} \tag{7.11}$$

However, a single spring-damper term would not provide the necessary active control during an entire gait cycle. Instead, impedance controllers typically separate gait into early stance, late stance, early swing, and late swing. The transition criteria for switching to the four phases is defined as follows:

- *Phase 1*: heel-strike

- *Phase 2*: ankle angle reaches a threshold for pre-swing

- *Phase 3*: ankle dorsiflexes after push-off

- *Phase 4*: the knee velocity changes direction to slow the foot for the next heel-strike

7.2.3 Optimization

We optimized controller parameters in MATLAB using a Covariance Matrix Adaptation (CMA) algorithm [22] and a multi-objective cost function (7.12). CMA was selected because it is a fast, population-based algorithm and shooting methods must be used when the control is discontinuous. The tracking term of the cost function is the norm of the difference between the joint torque $\tau_{j,vmr}$ produced by the candidate controller and the joint torque $\tau_{j,exp}$ measured during the 7 minutes of perturbed walking ($N = 42,000$ samples). We reserve the remaining 1 minute of data for a cross-validation analysis. Note that the tracking term is the sum of the squared root-mean-square (RMS) errors for each joint. Subsequently, the effort term was modeled as the mean of the squared muscle activation (a), which promotes load-sharing between all muscles (m). Weight terms W_1 and W_2 are

136

selected to prioritize either the tracking or effort:

$$C = W_1 \sqrt{\frac{1}{N} \sum_{i=1}^{N} (\tau_{exp} - \tau_{vmr})^2} + W_2 \sqrt{\frac{1}{N} \sum_{i=1}^{N} a_m^2} \tag{7.12}$$

For the VMR controller, we solve for 81 total parameters, including 46 control parameters (reflex and vestibular contributions), and an additional five muscle properties for each muscle. The selected muscle properties include: slack length of the SEE, slack length of the PEE, and the strain of the SEE at maximum isometric muscle force (u_{max}). Note that the time delays for both spinal reflexes and vestibular feedback were also used as optimization parameters. For details regarding the choice of muscle parameters used as variables in the optimization, refer to Part II, Chapter VI, Section 6.2.5. A summary of all optimization parameters is presented in Table XII.

Table XII: Parameters to be optimized for the Virtual Muscle Reflex (VMR) controller for the hip, knee, and ankle. Note that the muscle properties for each muscle were also optimized.

Control Parameters for Optimization	
Pre-Stimulations	$S_{0,SOL}$, $S_{0,TA}$, $S_{0,GAS}$, $S_{0,VAS}$, $S_{0,HAM}$, $S_{0,GLU}$, $S_{0,ILI}$
Feedback gains	G_{SOL}, $G_{TA,st}$, $G_{TA,sw}$, G_{SOLTA}, G_{GAS}, $G_{VAS,st}$, $G_{VAS,sw}$, G_{HAM}, ΔS, $G_{GLU,st}$, $G_{GLU,sw}$, $G_{ILI,st}$, $G_{ILI,sw}$, G_{HAMILI}
Length offsets	$l_{off,TA}$, $l_{off,ILI}$, $l_{off,HAM}$
Trunk PD Terms	k_p, k_d, θ_{ref}, $k_{p,h}$, $k_{d,h}$, k_{lean}
Knee/Hip Position Control	k_ϕ, ϕ_{off}, $k_{\gamma,st}$, $k_{\gamma,sw}$, γ_{off}
Body Weight Gains	$k_{bw,VAS}$, $k_{bw,HAM}$, $k_{bw,GLU}$, $k_{bw,ILI}$
Vestibular Control	k_{SOL}, k_{TA}, k_{GAS}, k_{VAS}, k_{HAM}, k_{GLU}, k_{ILI}
Muscle Properties	SEE slacklength, PEE slacklength, u_{max}, reflex time delay, vestibular time delay

For the impedance controller, the cost function (7.12) has only the first term (tracking). We solve for 37 control parameters: stiffness (k_i), damping (b_i), and equilibrium position (θ_i) for each of the joints in all four gait phases, and also for the ankle angle threshold that determined the pre-swing transition.

The initial guess for the VMR optimization are the original reflex control parameters

reported by Geyer and Herr [14], some manual tuning to account for our added controller terms, and muscle property values previously reported in literature [27]. The initial guess for the reflex time delays were approximately 30 ms for distal muscles (*Gastrocnemius*, *Tibialis anterior*, *Soleus*), 20 ms for the *Vasti*, and 10 ms for the most proximal muscles (*Hamstrings*, *Gluteals*, *Iliopsoas*). We used 80 ms for the vestibular time delay. Similarly, the initial guess for the impedance control parameters are those outlined by Sup et al. [2], and manual tuning of the hip. The number of generations and population size of the CMA controller were adjusted until the models converged and showed no further minimization of the cost function.

Upper and lower bounds were determined by double and half of the initial guess, respectively, though vestibular feedback gains (k_m) and trunk angle reference positions were allowed to be positive or negative. Because our dataset includes perturbations, some test subjects may not reach the reference position criteria, which would prevent these terms from triggering. Similarly, the mechanism by which the cerebellum responds to trunk acceleration for each muscle is not entirely known. Therefore, we allow this vestibular feedback to be either negative or positive in response to the perturbations.

7.2.4 Statistical Analysis

The performance of both the impedance and VMR controllers was quantified by separating each of the gait cycles into 100 different points. The coefficient of determination (R^2) and RMS error was calculated for all gait cycles, at each of the 100 points, to correlate how well the controller describes that instance of the gait cycle when compared to the torque exhibited by the test subject at the same time, across all gait cycles.

R^2 is calculated as follows, by dividing the sum of the residuals by the sum of squares, for each point in the gait cycle, i:

$$R_i^2 = 1 - \left[\frac{\sum_{i=1}^{N} \left(\tau_{exp,i} - \tau_{vmr,i} \right)^2}{\sum_{i=1}^{N} \left(\tau_{exp,i} - \bar{\tau}_{exp,i} \right)^2} \right] \tag{7.13}$$

The RMS error is calculated as the difference between τ_{exp} and τ_{vmr} at each of the 100 points, across all gait cycles:

$$RMSE_i = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(\tau_{exp,i} - \tau_{vmr,i})^2} \tag{7.14}$$

Note that the torque terms are the sum of the squared contributions of each joint.

Lastly, we calculate the mean, standard deviation, maximum value, and minimum value for every parameter in the VMR controller and the impedance controller. After performing the optimizations, we will obtain 30 separate VMR controllers and 30 impedance controllers. By reviewing the range within their optimized parameters, we can observe the trends across subjects or speeds and detect areas which may require additional modifications for future studies.

7.2.5 Cross-Validations

Since the VMR system contains many parameters, we have a higher potential to over-fit the measured torque signals. Therefore, we reserve the remaining 1 minute of walking data ($N = 6,000$ samples) for a cross-validation analysis. Each trial has its own optimized controller that we apply to the remaining walking data of its own trial. Then, we calculate the RMS error for each joint torque obtained from the 7 minutes of walking data and the 1 minute of walking data. If the controller is not over-fit, a linear function should describe the relationship between the two torque values.

7.3 Results

In this section, we compare the performance of the VMR and impedance controllers, including the RMS error and R^2 of each joint in both controllers (7.3.1 and Appendix A). We also present the statistical summary of the optimized VMR controller parameters and impedance controller parameters (7.3.2) and the results of the VMR cross-validations (7.3.3).

7.3.1 VMR and Impedance Control Comparison

Figure 29 shows a representative result from one test subject (male, age = 26, mass = 86.8 kg, speed = 0.8 m/s). On the left, we compare hip, knee, and ankle torques from the experiment (black), VMR controller (red-dashed), and impedance controller (teal). The timing of push-off is indicated by a vertical black line. Only a few gait cycles of the 8-minute trial are shown. On the right, we compare the RMS error as a function of the gait cycle for all joints. Push-off time occurs at 60% of the gait cycle, with swing and stance phases indicated.
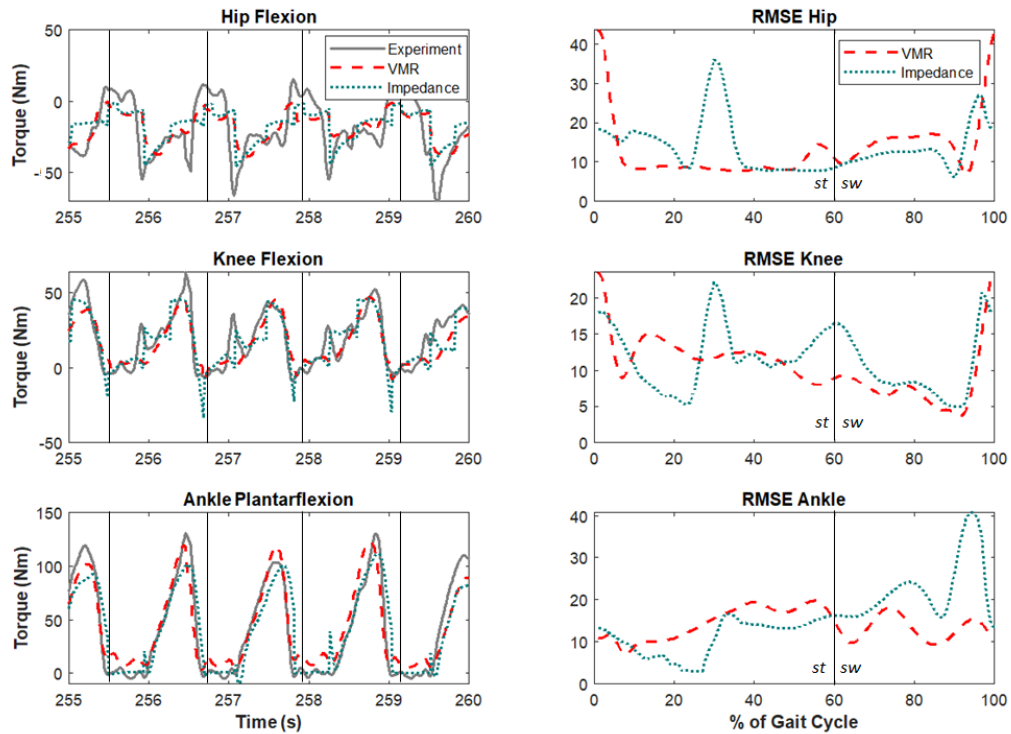
Figure 29: Torque comparison (left) of the VMR (red-dashed) and impedance controller (teal) compared to the experimental torque (black) for the hip (top), knee (middle), and ankle (bottom). The root-mean-square error (RMS) error is plotted as a function of the gait cycle (right) for each joint.

Generally, both controllers follow the trajectory for each joint torque. For this particular trial, we summarize the main similarities and differences:

- **Ankle**: Both controllers match the the timing of pre-swing prior during push-off and show variations in the peak ankle torque in response to perturbations. The impedance

140

controller demonstrates the largest RMS error at the end of the swing phase, at heel-strike, while the VMR controller's largest tracking errors occur during the stance phase.

- **Knee**: Both controllers match the knee torque prior to push-off, though the impedance controller shows large errors before the stance-to-swing transition. The VMR and impedance controller do not reach the knee flexion magnitude during swing, though they follow a similar trend in comparison to the experimental data. Additionally, the impedance controller produces sudden torque bursts in stance phase, likely due to mode transition. The highest RMS errors for the VMR controller occur at heel-strike.

- **Hip**: Neither the VMR nor impedance controller achieves the peak magnitudes of hip extension or flexion, particularly at the time of push-off. The highest RMS error for the VMR controller occurs at heel-strike, while the impedance controller struggles to match the timing of extension during stance phase.

Figure 30 shows the mean RMS error (left) and maximum correlation coefficient, R^2 (right), for all trials at the 1.2 m/s walking speed. Green and purple designate the VMR and impedance controller, respectively, with the contribution of each joint separated. The performance of both controllers varied per trial and across joints. In Trial 20, the total RMS error for the combined joints is 54.9 Nm for the impedance controller and 36.5 Nm for the VMR controller. However, the controllers demonstrate the opposite performance in Trial 68. Comparable RMS errors are observed in Trials 17, 31, 62, and 74. Concerning R^2, the impedance controller consistently achieves higher correlation coefficients for the knee and hip, while the VMR controller better represents the ankle. The RMS error and R^2 for all of the 30 trials, including the 0.8 m/s and 1.2 m/s walking speeds, can be referenced in Appendix A at the end of this dissertation.

Across all trials, the mean RMS error and maximum R^2 are summarized in XIII for
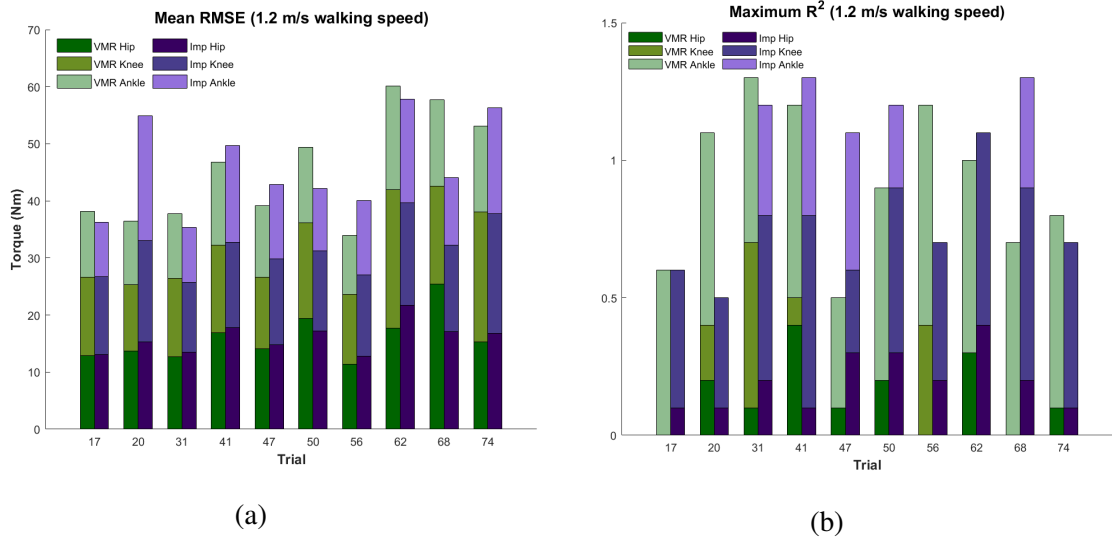
Figure 30: Mean RMS error (left) and maximum correlation coefficient, R^2 (right), as functions of the gait cycle, in which each gait cycle was split into 100 different data points. These results are obtained from 10 trials of the normal walking speed (1.2 m/s). The values for the VMR controller (green) and impedance controller (purple) are compared, with the contribution from each joint indicated. Better controllers have high correlation coefficients R^2 and lower RMS errors.

each joint. For the impedance controller, the RMS error and R^2 across all joints and all trials is 15.65 Nm and 0.28, respectively, and for the VMR controller, these values are 15.15 Nm and 0.29. The impedance controller shows higher maximum correlations in the knee and hip over the VMR controller.

Table XIII: Mean RMS error and maximum R^2 across all walking trials for each joint

	Mean RMS Error			Maximum R2		
	Hip	*Knee*	*Ankle*	*Hip*	*Knee*	*Ankle*
VMR	16.11	15.91	13.44	0.11	0.15	0.60
Impedance	16.52	15.93	14.51	0.17	0.5	0.17

7.3.2 Optimized VMR and Impedance Controller Parameters

Table XII summarizes the results of the 30 optimized VMR controllers (10 subjects at three walking speeds), designated by the associated muscle. The mean, standard deviation, maximum, and minimum was calculated for all parameters. Note that parameters in bold are shared amongst many muscles.

Pre-stimulation terms ($S_{0,m}$) and body-weight sensing terms ($k_{bw,m}$) are intentionally smaller than two-significant digits. A pre-stimulation term ensures the muscle is always slightly activated, but is not required to significantly contribute to the net control signal. Body-weight gains are small since large vertical GRFs contribute most substantially to the controller equation. An increase in these gains increases the risk of the muscle saturating at its activation limits (0 or 1, for minimum and maximum activation, respectively).

Parameters which experience the largest variation across test subjects and speeds are the vestibular feedback gains (k_m) and trunk angle thresholds. This can be attributed to the bounds of these parameters, which could be either positive or negative. For instance, some test subjects were observed to either lean the trunk forward or backward throughout the entirety of the trial. Similarly, the PD controller parameters also demonstrated large variations, likely due to these differences in trunk angles across subjects. The sign of the vestibular gains vary across subjects, where the optimization selected positive or negative values depending on the subject or trial.

Other parameters with noteworthy variation are the feedback gains for the muscles, discussed more thoroughly in Section 7.4.

Table XV summarizes the results of the 30 optimized impedance controllers (10 subjects at 3 walking speeds), separated by joint and gait phase. The mean, standard deviation, maximum, and minimum was calculated for all parameters. Note that many damping parameters are 0, as outlined in [2], since the respective muscle behaves more similarly to a stiff spring during these gait phases. The largest variations in the parameters occur in the joint reference angles, discussed more thoroughly in Section 7.4.

7.3.3 VMR Cross-Validations

Results from the cross-validation analysis are shown in Figure 31, where the RMS errors from the 7-minutes of walking are plotted against those obtained from the 1-minute dataset. Since a linear function is able to estimate the hip (left), knee (middle), and ankle (right) joint

torques, our VMR controller did not over-fit the data, despite the number of optimization parameters.
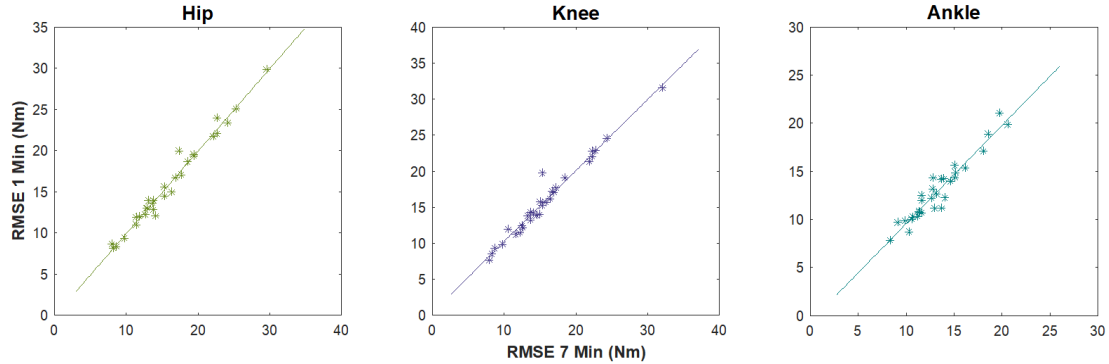

Figure 31: Cross-validation results, in which the controller identified by the first 7-minutes of walking data was applied to the remaining 1-minute of the trial. The root-mean-square (RMS) errors of the hip (left), knee (middle), and ankle (right) was calculated from each section of data, and plotted against each other. It can be shown that a linear function predicts the values

7.4 Discussion

We presented a joint torque controller with virtual muscles, reflexes, and vestibular control for the hip, knee, and ankle joints. Though reflex controllers have been shown to produce able-bodied walking in both simulation and hardware [14, 15, 17], these controllers are manually tuned until stable walking is achieved. We conducted experiments to measure the ability of reflex controllers in replicating able-bodied walking and human control actions. Test subjects walked for 8 minutes under the effect of anteroposterior perturbations by randomly adjusting the velocity of the treadmill belts. We obtained the parameters of the VMR controller using a CMA optimization that minimized the difference between the torque calculated from the experiment and the torque generated from the VMR model. To compare to the current state-of-the-art P/O controller, we then repeated the procedure using a typical finite-state impedance controller with four walking phases. The performance of both controllers was quantified using the RMS error and R^2 as a function of the gait cycle. We calculated the mean, standard deviation, maximum, and minimum value of all

controller parameters to observe trends across subjects and speeds.

Our results indicate that the VMR controller and impedance controller perform similarly in terms of mean tracking error. Both controllers have more difficulty tracking when the walking speed is increased. Additionally, neither controller can reach the maximum flexion or extension torques in any joint, though generally follow the joint trajectory patterns. The criteria for switching between stance and swing phase was 10 N of vertical GRF. This may be different across multiple test subjects and may be considered as an optimization variable to reduce the large RMS errors observed around heel-strike.

For 7 minutes of walking data, these statistics were calculated from approximately 400 gait cycles. Therefore, any R^2 higher than 0.5 indicates a high level of statistical significance at that instant of the gait cycle. For the impedance controller, the highest correlations occur in the knee and hip, while the ankle is better reproduced by the VMR controller. However, the optimizations minimized RMS error (tracking error) and did not consider R^2. Low correlation coefficients do not suggest that the controllers perform poorly, but only indicate that they do not statistically match portions of the gait cycle from 400 walking cycles. Similarly, controllers may have a high RMS error due to one poorly reproduced portion of the gait cycle, though tracking error is more acceptable throughout the remainder of the cycle.

The impedance controller exhibits discontinuity in the transitions between the four gait phases. Because test subjects are perturbed, the thresholds for transitioning between the phases of gait may never be reached as each gait cycle has different random perturbations. For instance, if the ankle angle is not large enough to begin the pre-swing phase (Phase 2), then the controller will remain in the heel-strike phase (Phase 1) for the entire gait cycle. Often, the reference angles would reach the optimization limits, suggesting that the optimal reference angle may not exist for this gait phase for a particular joint using perturbed data. A more reliable method of switching between the gait phases will likely produce better results.

It is important to note that, since the VMR controller contains biarticular muscles, the system is highly coupled. Though the cost function included a squared activation term to promote load sharing between muscles, in several trials, the optimization would turn off redundant muscles, particularly the *Hamstrings* or *Gastrocnemius*. In addition, since we optimized muscle parameters, the joint could be "turned off" by creating too long of a series slack-length for every muscle, rendering the joint completely passive. This can occur when a constant $0 \, \mathrm{N \, m}$ in one joint may produce less overall tracking error than attempting to fit all three joint torques. We observed joint passivity in 20% of the trials. Because of this capability, our feedback gains varied across subjects. For instance, the *Gluteals* may be required to produce more force in the absence of the *Hamstrings*. The RMS error for the inactive joint in the VMR controller will be higher than the impedance controller, of which none of the joints are inactive. Similarly, passive joints will result in poor R^2 errors. Because the hip and knee could become passive, the better correlation achieved by the impedance controller can be explained by passive joints in some optimized VMR controllers.

Small deviations in any VMR control parameter resulted in observable changes throughout the entire system. As an example, increasing the *Soleus* force feedback gain (G_{SOL}) requires less assistance from the *Gastrocnemius*, which would produce less knee flexion torque, which promotes larger knee flexion from the *Hamstrings*, and thereby less muscle activity in the *Gluteals*. Due to this cascading effect, we elected to include all controller variables in the optimizations. Yet, some parameters may not have as great of an impact on the overall control for each muscle, such as muscle pre-stimulations. A sensitivity analysis for VMR controller parameters is beyond the scope of this work, though may be considered for reducing optimization computation time and less of a challenge in obtaining a global optimum.

For method continuity, we also optimized the impedance controller with a combined RMS error term for all joints. However, the joints are not coupled in an impedance con-

troller and the results could improve if each joint was optimized separately. Likewise, it can be suggested that the VMR controller's accuracy would also improve if optimized for a single joint, since the option of shutting off redundant muscles would not exist. In our previous work with an actuated ankle joint, both controllers demonstrate lower RMS error and higher correlations R^2 over these results when all joints are simultaneously optimized (refer to Part II, Chapter VI). In theory, the electric motors of a powered P/O device could be programmed with different muscle properties and controllers, thereby allowing each joint to be independently optimized, similar to the impedance controller. An additional study for each individual joint may provide clearer insights regarding the capabilities of the VMR system and help justify the complexity of the control.

Another option would be to determine the muscle parameters separately from the controller. Nominal muscle properties were obtained from previous literature [27]. Since muscle properties differ across test subjects, we included them as optimization parameters, though this allowed the optimization to turn off muscles or joints. Identifying the plant (muscle model) separately from the VMR controller may eliminate this possibility and also better represent biological function. A few techniques that can determine muscle properties have been described in [27], including isometric tests, a study of sarcomere lengths, and physiological cross-sectional area. However, a method to estimate our test subject's muscle properties is outside of the scope of this study, but could be considered for a future investigation.

The original reflex controller designed by Geyer and Herr [14] was tested in a simulated walking model. Though their original controller equations were generally based on physiological attributes of muscles and reflexes, the resulting muscle forces and joint torque equations could not fit our experimental data, especially at the hip joint. In Geyer and Herr's model, the hip extensors produce a notably large torque at heel-strike and early stance, nor do they activate to inhibit the swing leg. This results in extremely large vertical GRFs at the next heel-strike in their simulation. The majority of their model's forward mo-

147

mentum can be attributed to the propulsion generated by these extensors. In order to fit our data, we were required to significantly alter these controller equations. By creating a hip position term, we regulated the extensor force production during stance and also slowed the swing leg prior to heel-strike. Geyer and Herr's model produces little extension torque at the knee; our additions to the *Vasti* equations resulted in an improved fit prior to push-off. Without these modifications, it was not possible for the VMR controller to reproduce the torque trajectories at these joints.

Geyer and Herr also did not include a *Rectus femoris* muscle, a biarticular muscle which produces hip flexion and knee extension. The addition of this muscle could improve the VMR controller for tracking the torque data at the hip and knee joints. Both controllers have difficulty reaching the magnitude of hip flexion torque and adding the *Rectus femoris* to the VMR system could provide some additional force to help reach these peak torques. However, the *Rectus femoris* is another redundant muscle and may introduce additional challenges for the optimization and increase the total controller parameters.

Our results suggest that a VMR controller is capable of explaining some human control actions during perturbed gait and can reproduce normal walking motions at least as well as the current control strategies used in current P/O devices. For a more complete assessment, the VMR should be tested in physical hardware and again compared to the performance of an impedance controller.

Table XIV: Statistical summary of optimized virtual muscle reflex (VMR) parameters (Par), including mean, standard deviation (std), maximum value, and minimum value. Parameters are separated by muscle: *Soleus* (Sol.), *Tibialis anterior* (T.A.), *Vasti* (Vas.), *Hamstrings* (Ham.), *Glutes* (Glu.), *Iliopsoas* (Ili.), and *Gastrocnemius* (Gas.). Parameters in bold are shared across multiple muscles.

Reflex Control Parameters																	
Muscle	Par.	Mean	Std	Min	Max	Muscle	Par.	Mean	Std	Min	Max	Muscle	Par.	Mean	Std	Min	Max
Sol.	$S_{0,SOL}$*	0.01	0.01	0.01	0.02	Ham.	$s_{0,HAM}$*	0.01	0.00	0.01	0.02	Ilio.	$s_{0,IL}$*	0.01	0.00	0.01	0.02
	G_{SOL}	0.88	0.20	0.52	1.31		G_{HAM}	0.83	0.22	0.45	1.23		$G_{IL,st}$	0.01	0.00	0.01	0.02
	k_{SOL}	-0.03	0.07	-0.20	0.05		$\boldsymbol{k_p}$	2.65	0.71	0.96	3.68		$G_{IL,sw}$	0.43	0.14	0.19	0.70
T.A.	$s_{0,TA}$*	0.01	0.00	0.01	0.02		$\boldsymbol{k_d}$	0.40	0.10	0.18	0.50		$l_{off,IL}$	0.74	0.16	0.36	1.05
	$G_{TA,st}$	1.20	0.33	0.51	1.83		$\boldsymbol{\theta_{ref}}$	0.18	0.23	-0.29	0.34		G_{HAMIL}	0.31	0.11	0.19	0.60
	$G_{TA,sw}$	0.86	0.21	0.43	1.21		$k_{bw,HAM}$*	0.00	0.00	0.00	0.00		$l_{off,HAM}$	0.95	0.30	0.47	1.50
	G_{SOLTA}	0.36	0.11	0.16	0.55		k_{HAM}	0.00	0.01	-0.02	0.02		k_{lean}	0.50	1.24	-2.26	2.22
	$l_{off,TA}$	0.94	0.17	0.59	1.26	Glu.	$S_{0,GLU}$	0.01	0.01	0.00	0.02		$\boldsymbol{k_{\gamma,sw}}$	0.18	0.03	0.11	0.20
	k_{TA}	0.01	0.08	-0.16	0.14		$G_{GLU,st}$	0.45	0.15	0.20	0.72		$\boldsymbol{k_{\gamma,st}}$	6.57	3.12	5.00	16.34
Vas.	$S_{0,VAS}$	0.10	0.03	0.05	0.16		$G_{GLU,sw}$	0.41	0.15	0.15	0.60		$\boldsymbol{\gamma_{off}}$	0.03	0.01	0.01	0.04
	$G_{VAS,st}$	1.22	0.46	0.56	2.09		$\boldsymbol{\Delta S}$	0.33	0.09	0.18	0.49		$k_{bw,IL}$*	0.00	0.00	0.00	0.00
	$G_{VAS,sw}$	0.13	0.03	0.08	0.19		$\boldsymbol{kp_{hip}}$	1.06	0.31	0.39	1.36		k_{IL}*	0.00	0.01	-0.01	0.02
	k_ϕ	4.63	1.36	2.57	7.90		$\boldsymbol{kd_{hip}}$	1.14	0.33	0.54	1.74	Gas.	$S_{0,GAS}$*	0.01	0.00	0.01	0.02
	ϕ_{off}	-0.21	0.05	-0.31	-0.12		$k_{bw,GLU}$*	0.00	0.00	0.00	0.00		G_{GAS}	1.18	0.36	0.40	1.60
	$k_{bw,VAS}$*	0.00	0.00	0.00	0.00		k_{GLU}	-0.01	0.01	-0.02	0.02		k_{GAS}	-0.04	0.09	-0.20	0.11
	k_{VAS}	0.10	0.04	0.05	0.19		*values are smaller than two significant digits. Parameters in bold are shared with other muscles										

Table XV: Optimized parameters of the impedance controller for the hip, knee, and ankle, separated by gait phase and joint. The mean, standard deviation, minimum, and maximum of each parameter were calculated from each of the 30 walking trials.

Phase	Joint	Variable	Mean	Std	Min	Max
Mode 1: Heel-Strike	ankle	k_{a1}	3.5	1.15	1.75	6.19
		b_{a1}	0.0	0.0	0.0	0.0
		θ_{eq1}	-3.39	2.04	-8.8	-2.2
	knee	k_{k1}	2.49	0.73	0.75	3.0
		k_{b1}	0.0	0.0	0.0	0.0
		ϕ_{eq1}	11.51	3.38	5.00	20.00
	hip	h_{k1}	1.28	0.41	0.60	2.0
		h_{b1}	0.0	0.0	0.0	0.0
		γ_{eq1}	-11.08	6.16	-20.0	-5.0
Mode 2: Pre-Swing	ankle	k_{a2}	3.32	0.77	1.32	4.58
		b_{a2}	0.0	0.0	0.0	0.0
		θ_{eq2}	-13.37	6.00	-36.0	-9.0
	knee	k_{k2}	1.81	0.19	1.75	2.76
		b_{k2}	0.03	0.0	0.03	0.03
		ϕ_{eq2}	29.15	2.94	23.17	32.0
	hip	k_{h2}	0.36	0.14	0.13	0.50
		b_{h2}	0.0	0.0	0.0	0.0
		γ_{eq2}	-29.2	3.32	-30.0	-11.68
Mode 3: Push-Off	ankle	k_{a3}	0.29	0.12	0.15	0.59
		b_{a3}	0.01	0.0	0.01	0.02
		θ_{eq3}	0.0	0.0	0.0	0.0
	knee	k_{k3}	0.0	0.0	0.0	0.0
		b_{k3}	0.02	0.0	0.0	0.0
		ϕ_{eq3}	61.09	23.29	25.0	100.0
	hip	k_{h3}	0.0	0.0	0.0	0.0
		b_{h3}	0.03	0.0	0.01	0.03
		γ_{eq3}	5.73	1.91	2.57	9.22
Mode 4: Late Swing	ankle	k_{a4}	0.16	0.04	0.01	0.02
		b_{a4}	0.0	0.0	0.0	0.0
		θ_{eq4}	0.0	0.0	0.0	0.0
	knee	k_{k4}	0.0	0.0	0.0	0.0
		b_{k4}	0.02	0.0	0.01	0.02
		ϕ_{eq4}	56.14	18.83	23.22	89.95
	hip	k_{h4}	0.87	0.38	0.5	1.62
		b_{h4}	0.03	0.0	0.01	0.03
		γ_{eq4}	8.33	2.86	2.5	10.5

Acknowledgments

7.5 REFERENCES

[1] B. E. Lawson, B. Ruhe, A. Shultz, and M. Goldfarb. A powered prosthetic intervention for bilateral transfemoral amputees. *IEEE Transactions on Biomedical Engineering*, 62(4):1042–1050, April 2015.

[2] F. Sup, A. Bohara, and M. Goldfarb. Design and control of a powered transfemoral prosthesis. *International Journal of Robotics Research*, 27(2):263–273, February 2008.

[3] F. C. Sup. *A powered self-contained knee and ankle prosthesis for near normal gait in transfemoral amputees*. PhD thesis, Vanderbilt University, 2009.

[4] R. J. Farris, H. A Quintero, S. A. Murray, , K. H. Ha, C. Hartigan, and M. Goldfarb. A preliminary assessment of legged mobility provided by a lower limb exoskeleton for persons with paraplegia. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 22(3):482–490, May 2014.

[5] G. Khademi, H. Mohammadi, H. Richter, and D. Simon. Optimal mixed tracking/impedance control with application to transfemoral prostheses with energy regeneration. *IEEE Transactions on Biomedical Engineering*, 65(4):894–910, April 2018.

[6] H. Warner. Optimal design and control of a lower-limb prosthesis with energy regeneration. Master's thesis, Cleveland State University, 2015.

[7] M. R. Tucker, J. Olivier, A. Pagel, H. Bleuler, M. Bouri, O. Lambercy, J. del R. Millán, R. Riener, H. Vallery, and R. Gassert. Control strategies for active lower extremity prosthetics and orthotics: A review. *Journal of NeuroEngineering and Rehabilitation*, 12(1):1, December 2015.

[8] K. G. Gerritsen, A. J. van den Bogert, M. Hulliger, and R. F. Nernicke. Intrinsic muscle properties facilitate locomotor control - A computer simulation study. *Motor Control*, 2(3):206–220, July 1998.

[9] A. J. van den Bogert, D. Blana, and D. Heinrich. Implicit methods for efficient musculoskeletal simulation and optimal control. *Procedia IUTAM*, 2:297–316, January 2011.

[10] C. Capaday. The special nature of human walking and its neural control. *TRENDS in Neurosciences*, 25(7):370–376, July 2002.

[11] J. L. Pons, J. C. Moreno, D. Torricelli, and J. S. Taylor. Principles of human locomotion: A review. In *35th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, pages 6941–6944, July 2013.

[12] J. B. Nielsen and T. Sinkjaer. Afferent feedback in the control of human gait. *Journal of Electromyography and Kinesiology*, 12(3):213–217, June 2002.

[13] A. Prochazka and D. Gillard. Sensory control of locomotion. In *Proceedings of the 1997 American Control Conference*, volume 5, pages 2846–2850, 1997.

[14] H. Geyer and H. Herr. A muscle-reflex model that encodes principles of legged mechanics produces human walking dynamics and muscle activities. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 18(3):263–273, June 2010.

[15] S. Song and H. Geyer. A neural circuitry that emphasizes spinal feedback generates diverse behaviours of human locomotion. *The Journal of Physiology*, 593(16):3493–3511, August 2015.

[16] M. F. Eilenberg, H. Geyer, and H. Herr. Control of a powered ankle-foot prosthesis based on a neuromuscular model. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 18(2):164–173, April 2010. doi: 10.1109/TNSRE.2009. 2039620.

[17] N. Thatte and H. Geyer. Toward balance recovery with leg prostheses using neuromuscular model control. *IEEE Transactions on Biomedical Engineering*, 63(4): 904–913, May 2016.

[18] A. and H. Geyer. The 3-D spring-mass model reveals a time-based deadbeat control for highly robust running and steering in uncertain environments. *IEEE Transactions on Robotics*, 29(5):1114–1124, May 2013.

[19] H. Geyer, A. Seyfarth, and R. Blickhan. Positive force feedback in bouncing gaits? *Proceedings of the Royal Society of London B: Biological Sciences*, 270(1529):2173–2183, October 2003.

[20] F. Dzeladini, A. R. Wu, D. Renjewski, A. Arami, E. Burdet, E. van Asseldonk, H. van der Kooij, and A. J. Ijspeert. Effects of a neuromuscular controller on a powered ankle exoskeleton during human walking. In *2016 6th IEEE International Conference on Biomedical Robotics and Biomechatronics (BioRob)*, pages 617–622, June 2016.

[21] A. Ostermeier, A. Gawelczyk, and N. Hansen. A derandomized approach to self-adaptation of evolution strategies. *Evolutionary Computation*, 2(4):369–380, 1994.

[22] N. Hansen and A. Ostermeier. Completely derandomized self-adaptation in evolution strategies. *Evolutionary Computation*, 9(2):159–195, June 2001.

[23] J. K. Moore, S. K. Hnat, and A. J. van den Bogert. An elaborate data set on human gait and the effect of mechanical perturbations. *PeerJ*, 3:e918, April 2015.

[24] D. A. Winter. *Biomechanics and Motor Control of Human Movement*. John Wiley and Sons, Inc, 2nd edition, 1990.

[25] S. K. Hnat and A. J. van den Bogert. Inertial compensation for belt acceleration in an instrumented treadmill. *Journal of Biomechanics*, 47(15):3758–3761, November 2014.

[26] J. K. Moore, S. K. Hnat, and A. J. van den Bogert. An elaborate data set on human gait and the effect of mechanical perturbations. *PeerJ*, 3:e1248, 4 2015.

[27] K. G. M. Gerritsen, W. Nachbauer, and A. J. van den Bogert. Computer simulation of landing movement in downhill skiing: Anterior cruciate ligament injuries. *Journal of Biomechanics*, 29(7):845–854, July 1996.

[28] E. R. Kandel, J. H. Schwartz, T. M Jessel, and S. A. Siegelbaum. *Principles of Neural Science*. McGraw-Hill, New York, 2000.

[29] E. Todorov. Direct cortical control of muscle activation in voluntary arm movements: A model. *Nature Neuroscience*, 3:391–398, April 2000.

[30] S. Hirokawa, M. Solomonow, Z. Luo, Y. Lu, and R. D'Ambrosia. Muscular co-contraction and control of knee stability. *Journal of Electromyography and Kinesiology*, 1(3):199–208, September 1991.

[31] S. Heitmann, N. Ferns, and M. Breakspear. Muscle co-contraction modulates damping and joint stability in a three-link biomechanical limb. *Front Neurorobot*, 5:5, January 2011.

[32] N. Hogan. The mechanics of multi-joint posture and movement control. *Biological Cybernetics*, 52(5):315–331, September 1985.

[33] M. Millard, T. Uchida, A. Seth, and S. L. Delp. Flexing computational muscle: Modeling and simulation of musculotendon dynamics. *ASME Journal of Biomechanical Engineering*, 135(2):0210051–02100511, February 2013.

PART III.

Implementing a Real-Time Virtual-Muscle

Reflex Controller on a Powered Orthosis

CHAPTER VIII

A REAL-TIME VIRTUAL MUSCLE REFLEX MODEL

FOR A POWERED EXOSKELETON

With the VMR controller working in simulations of the hip, knee, and ankle joints, we then program the model on the Parker Hannifin Indego® exoskeleton using C-code compiled by MATLAB and Simulink. In this chapter, we detail how we built the Simulink model for use on the Indego and describe the results of preliminary proof-of-concept tests.

ABSTRACT

We hypothesize that a neurologically-inspired controller can achieve more natural gait mechanics in a powered orthosis when compared to the phase-based PD controllers typically used in these devices. Here, we design a controller based on Virtual Muscles and Reflexes (VMR), that produces a joint torque signal to operate the electric motors of a powered exoskeleton. The controller was modeled and built in MATLAB® Simulink® for use on the Parker Hannifin Indego®. The virtual muscles of the controller were tested by setting joint angles at a constant of 0° while prescribing a step-input neural control signal ranging from 0 to 1. Additionally, we used human experimental gait data as sensor input and manually tuned our controller in order to observe the corresponding joint torques. Not only can the controller operate in real-time at a sampling rate of 200 Hz, but the results from our preliminary tests also suggest the VMR controller produces intuitive joint torques in response to the prescribed neural control signals. Therefore, using the VMR controller onboard the Indego is an acceptable platform for future studies.

8.1 Introduction

Powered orthotic devices aim to improve the quality of life for persons with a spinal cord injury (SCI). *Exoskeletons* are anthropomorphic devices worn by the user which enhance or assist the operator's performance, while *powered orthoses* are classified as a type of exoskeleton that restores function to individuals with a SCI or other type of pathological condition which affects ambulation [1]. Recently, several powered orthoses have been developed for rehabilitative gait training. These devices can lessen strain on the therapist while measuring the patient's progress throughout the treatment, such as improvements in force production and joint angle patterns [2]. Many of these devices, including the ALEX I and II [3, 4] and the LOPES robot [5], have been designed as overhead treadmill gait trainers, operating in parallel to the body to guide paralyzed or weakened limbs through

walking motions [6].

Outside of rehabilitation environments, a few powered orthoses have been developed and marketed for home use. These devices can be used to supplement the patient's capabilities or to completely replace walking for individuals with paraplegia. Some commercialized models include the ReWalk[TM] by ReWalk Robotics[TM] [7, 8], the Hybrid-Assistive-Leg[®] (HAL[®]) by Cyberdyne Inc. [9–13], Ekso[TM] by Ekso Bionics[®], [14–16], and the Parker Hannifin Indego[®] exoskeleton [17]. Typically, these devices are equipped with torque-driven electric motors at the joints. The motion from the electric motors reproduce a predefined set of joint trajectories in order to replicate standard walking patterns observed in able-bodied human gait [18] and to perform sit-to-stand maneuvers.

Various control strategies have been designed and implemented for powered orthoses, though phase-based impedance controllers are the current standard [19, 20], in which the joint torque is calculated as a function of the joint angle and joint velocity. Instead of an impedance controller, the spring-damper characteristics can also be achieved through the nonlinear viscoleastic and dynamics properties of muscles. In simulation, virtual muscles have been shown to greatly increase stability during walking when compared to direct torque-driven control [21]. Recently, a simulation of muscle dynamics using a Hill-type muscle model was achieved in real-time [22], which is a requirement for orthotic devices (for details concerning this model, refer to Part II, Chapter V).

To generate neural control signals for the virtual muscles, an autonomous, positive-feedback reflex model has been previously modeled by Geyer and Herr [23] and tested in a simulated walking model [24], and an actuated knee and ankle prosthesis [25]. This reflex controller is also used in Herr's BiOM single-joint ankle prosthesis (BionX, [26, 27]). Recently, a reflex controller was evaluated on an exoskeleton with an actuated ankle, resulting in a reduction of net metabolic cost during rehabilitation exercises [28]. We also used a similar approach to compare a neuromuscular reflex controller to an impedance controller in the ankle joint (see Part II, Chapter VI) and the hip, knee, and ankle (Part II,

Chapter VII).

In this study, we will expand the controller for use in the Parker Hannifin Indego exoskeleton (hereafter, Indego). The Indego is a battery-powered, relatively lightweight (12-kg) device designed with brushless motors at the hip and knee joints and a rigid ankle-foot orthosis (AFO). The electric motors of the Indego are capable of operating within the range of motion and torque magnitudes observed during able-bodied walking [18].

Here, we implement a controller based on Geyer's model [23], including real-time virtual muscles, reflex control, and vestibular control (Virtual Muscle Reflex, VMR). Subsequently, we will build the controller in MATLAB® Simulink® (Mathworks, Santa Rosa, California, USA) and compile the model to run on the Indego. To verify the model is properly implemented, we first test the controller under two test conditions: 1) step-input neural control signals to each muscle independently and 2) observing the prescribed joint torque when human experimental gait data is used in place of sensors.

8.2 Methods

In this chapter, we provide a brief overview of the VMR controller architecture (8.2.1) and detail how it can be modeled in Simulink to operate the Indego (8.2.2 - 8.2.5). We also discuss the test conditions for verifying that the controller was properly implemented (8.2.6).

8.2.1 Overview of Virtual Muscle Reflex (VMR) Controller

First, we summarize the virtual muscle model and the reflex control equations for stimulating each muscle. For a more in-depth explanation of this controller, refer to Part II, Chapter VII.

Simulation of Muscle Dynamics

For this study, we used a planar model of the lower leg with 5 muscles in the sagittal plane to represent a powered orthoses with an actuated knee and hip joint. Since each joint must be capable of moving in both directions (flexion/extension), our model includes the: *Gastrocnemius*, *Vasti* group, *Hamstrings* group, *Gluteal* group, and *Iliopsoas*. Note our sign convention defines negative hip and knee torque as flexion.

Each muscle was represented by a standard 3-element Hill-type model, including a contractile element (CE), series/parallel nonlinear elastic elements (SEE/PEE), and a small amount of viscous damping for numerical stability (see Figure 19 in Part II, Chapter V, Section 5.2.1). The nominal properties of these muscles were obtained from previous literature [29]. Inputs to the muscle model include the muscle length (derived from the joint angles measured by the encoders) and neural excitation signals from the reflex controller.

Muscle dynamics were represented by a set of first-order implicit differential equations (IDE) for muscle contraction and muscle activation (see Part II, Chapter V, Section 5.2.1 for details). These relationships can be combined into a single, implicit state equation:

$$f(x, \dot{x}, u) = 0 \tag{8.1}$$

with state variables $x_n = \begin{bmatrix} L_{CE,1}...L_{CE,n} & a_1...a_n \end{bmatrix}^T$ and state derivatives $\dot{x}_n = \begin{bmatrix} \dot{L}_{CE,1}, ...\dot{L}_{CE,n} & \dot{a}1, ...\dot{a}_n \end{bmatrix}^T$, where n is the number of muscles, L_{CE} is the length of the contractile element, and a is the muscle activation signal. The control inputs, u, are the neural excitation control signals generated by the reflex controller.

Real-time muscle dynamics were simulated in MATLAB using a first-order, implicit Rosenbrock solver [22], outlined in Part II, Chapter V, Section 5.2.2. The muscle model produces a muscle force which is multiplied by the moment arm to obtain the joint torque used to control the electric motors of the Indego.

Reflex Controller

We used the reflex controller previously described by Geyer and Herr [23], with additional modifications including vestibular feedback and hip position control. For each muscle (m), the neural excitation (u) is obtained through spinal reflex control equations of the general form:

$$u_m = S_{0,m} + G_m Feedback_m(t - \Delta t_{m,R}) + k_m \ddot{\theta}(t - \Delta t_{m,V}) \tag{8.2}$$

where $S_{0,m}$ is a pre-stimulation term, and G_m is a gain multiplied by a feedback term $Feedback_m$ subject to the time delay between the particular muscle and spinal cord ($t - \Delta t_{m,R}$). This feedback term can be either muscle force or muscle fiber length. A vestibular gain k_m is multiplied by the trunk forward acceleration $\ddot{\theta}$, subject to the time delay between the muscle and the cerebellum ($t - \Delta t_{m,V}$).

However, reflexes alone cannot describe the aspects of balance regulated by the cerebellum, brain stem, and visual and vestibular cortices of the cerebrum [30]. To address the limitations of reflexes, a proportional-derivative (PD) control is added to the *Hamstrings*, *Gluteals*, and *Iliopsoas* to maintain postural stability of the trunk. This term of the controller can be described as

$$u_m = k_p[\theta(t - \Delta t_{m,R}) - \theta_{ref}] + k_d \dot{\theta}(t - \Delta t_{m,R}) \tag{8.3}$$

where proportional gain k_p and derivative gain k_d control the trunk forward angular position (θ) by a reference position (θ_{ref}) and forward velocity of the trunk ($\dot{\theta}$). Note that any sensor signal used as feedback must be delayed by the reflex time delay.

Additional features of the reflex controller include:

- Separate control strategies for each muscle during swing and stance gait phases

- Hyperextension prevention of the knee

162

- Inhibition or excitation of muscles during double-support phase

- Control of hip extension/flexion during early stance and late swing

A detailed description of the reflex controller can be found Part II, Chapter VII where we perform optimizations to determine the parameters of the VMR system, in which we minimized the root-mean-square error (RMSE) between the torque generated by our model and the torque measured through human perturbed walking data (see Part I, Chapter II and [31] for details concerning our dataset.) Results suggest that a controller of this structure can reproduce joint torque patterns observed in able-bodied human walking and can be evaluated for use on a powered orthoses.

8.2.2 Software and Hardware Dependencies

Operation of the Indego requires the use of the following MATLAB and Simulink toolboxes: Simulink Coder^{TM}, MATLAB Coder^{TM}, and Simulink Desktop Real-Time^{TM} (SL-DRT). After building a Simulink model, these toolboxes compile the model into C-code, which can then be uploaded onto the microcontroller of the Indego. The Indego communicates to the PC through a USB + Vector VN1610 CAN interface (Vector, Stuttgart, Germany) at a sample rate of 200 Hz. Sensor feedback from the Indego's encoders and gyrometer, including angular position, pelvic tilt, and motor temperature, are accessible through Simulink.

The motors (hip and knee) are operated through prescribed currents (mA). To operate the Indego's motors, the joint torques generated by the VMR controller must be converted into current commands. First, the desired torque is divided by the transmission ratio of the knee or hip, and is then divided by a motor torque constant to obtain the current command.

Figure 32 provides an overview for controlling the Indego. Sensor data from the Indego is used as input to the VMR controller, which then outputs a joint torque for the knee or hip motor. The signal is saturated to remain within the hardware specifications of the motor and then converted into a current command. The VMR controller and unit conversions are

compiled into C-code using Simulink and the associated toolboxes. Note that a separate workflow is required for each leg, though the control structure is identical.
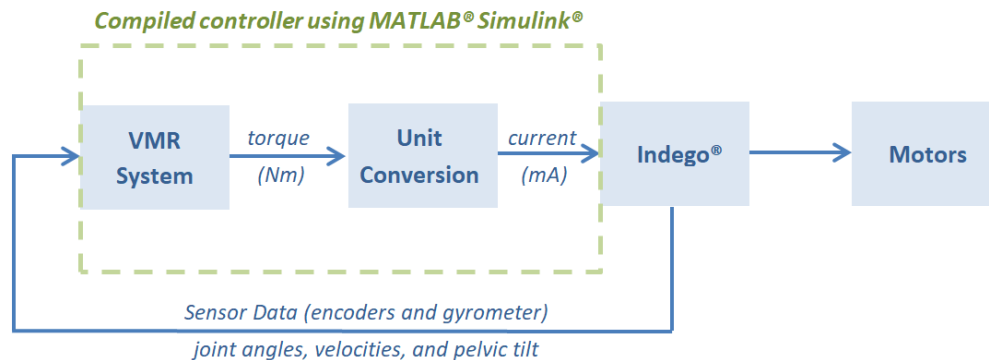

Figure 32: Workflow for operating the Indego® with the VMR system. The VMR receives feedback from the Indego sensors (i.e. angular positions and velocities) which is then used to drive muscle dynamics, controlled by reflexes. The output torque signal is then converted into milliamps before being sent to the Indego's electric motors.

The entire VMR Simulink model, for controlling both legs, can be viewed in Figure 40.

8.2.3   Sensor Inputs and VMR Outputs

Figure 33 summarizes the inputs and outputs of the VMR controller. The outputs of the controller are the joint torques (Nm), muscle forces (N), muscle activations, and length of the contractile element (m). For more information concerning how we obtain these output parameters, refer to muscle model in Part II, Chapter V. Here, we focus on the main inputs of the VMR system: simulation time, muscle co-activations, ground reaction forces (GRFs), and angles, velocities, and accelerations for the joints and trunk.

1. Simulation Time

The VMR controller requires the simulation time for the following reasons:

- the implicit muscle model, particularly the Rosenbrock fixed-step solver (detailed in Part II, Chapter V) requires a time counter to calculate the two muscle states for the next-step of the muscle dynamics simulation
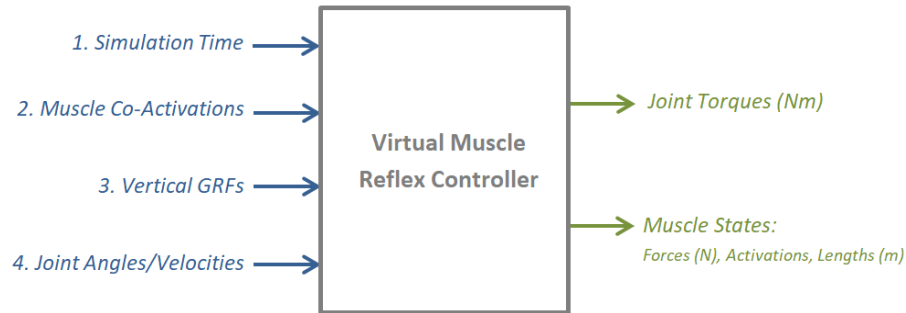
Figure 33: Inputs and outputs of the VMR system in MATLAB®Simulink®. Simulation time, muscle-co-activations, ground reaction forces (GRFs), and joint angles and velocities drive the muscle model, which then outputs the joint torques (hip and knee) and muscle forces, lengths, and activations

- every feedback term for the reflex equations must be delayed with either the reflex time-delay between the muscle and the spine, or vestibular time-delay between the muscle and the cerebellum. In the Simulink model, these real-time feedback terms are delayed using interpolation and a stored time vector

For simulation time, we use Simulink's clock, sampled at 200 Hz.

2. Muscle Co-Activations

To operate the muscles independently from the reflex controller, we can prescribe a manual neural input (u). Depending on the application, these neural signals can be generated through a variety of different forms, such as a constant, a sine wave, or a square wave. Co-Activations must be a scalar value between 0 and 1.

For preliminary model testing, we can use a step-input for each muscle and observe the respective motor action. For instance, activating the *Vasti* should result in a corresponding extension torque in the Indego's knee motor.

3. Ground Reaction Forces (GRFs)

The reflex equations of the VMR controller require vertical ground reaction forces (GRFs). Not only do GRFs trigger the transition between swing and stance phases for each muscle,

but they are also used as feedback terms in the *Vasti*, *Hamstrings*, *Glutes*, and *Iliopsoas* (refer to Part II, Chapter VII, Section 7.2.1 for more details of these controller equations).

The Indego is not equipped with force sensors. Instead of integrating force sensors on the Indego itself, our approach was to use analog GRF signals from the instrumented treadmill (R-Mill, Motekforce Link, Culemborg, Netherlands). Each belt of the treadmill has a separate force plate in which 6 analog force signals can be measured (± 10 V). At Cleveland State University's Parker Hannifin Laboratory for Human Motion and Control, all analog signals (force plate, accelerometer, and EMG) are read through a National Instruments USB-6255 data acquisition unit (DAQ), sampled at 1000 Hz (refer to Part I, Chapter II).

The six raw force plate analog signals, for each force plate, are converted into GRFs (F_{xyz} and M_{xyz}) through a calibration matrix. For the VMR controller, only the vertical GRF is needed (positive Y-axis in our coordinate system). Figure 34 shows the Simulink model for recording vertical GRF data in real-time. Simulink's Data Acquisition Toolbox$^{\text{TM}}$ acquires the 12 raw analog signals from the NI-USB-6255 before they are split into the six raw signals of the respective force plate (left or right belt). The signals are then transformed into GRFs through the calibration matrix.

Next, we extract the vertical GRFs, in which a saturation block limits the minimum F_y signal to 0 N of force. Negative forces from noise in the signal can cause abnormal behavior in the VMR system since these feedback terms are multiplied by large gain. Sign errors will result in the opposite of the intended action (i.e. *Gluteal* excitation during swing phase). We also subtracted a small offset from each F_y signal, calibrated based on an object of a known mass.

However, the Indego microcontroller, which executes the Simulink model, does not have access to these GRF signals. A thorough description of this limitation is discussed in Section 8.4.1.
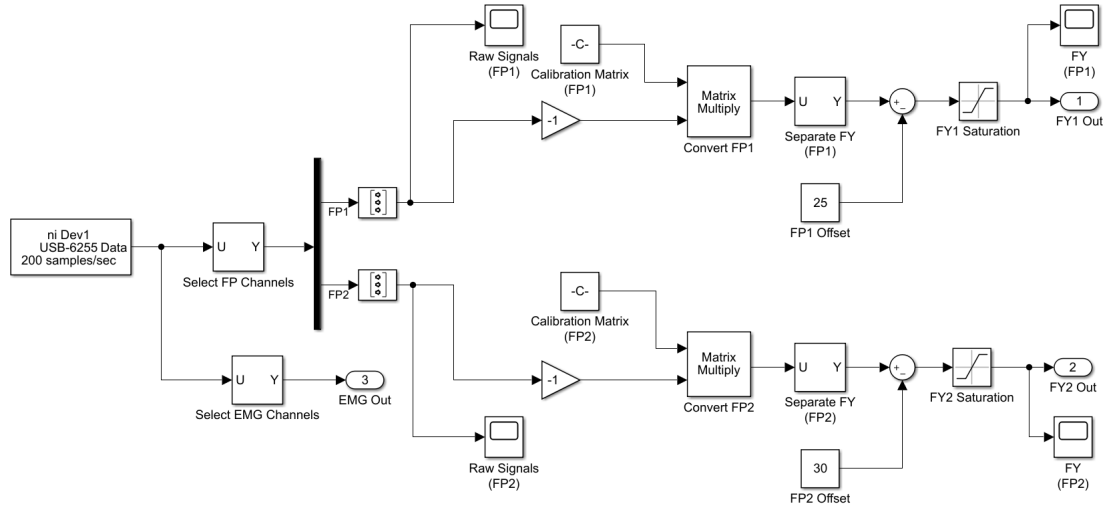
Figure 34: Simulink diagram for reading real-time GRFs, where 12 analog force plate channels (V) are split into left- and right-belt before being multiplied by a calibration matrix to obtain 3-D GRFs. Then, vertical GRF (F_y) is parsed, translated to account for signal offset, and saturated to remove negative force signals due to noise

4. Joint and Trunk Position, Velocity, and Acceleration

Joint angles can be obtained directly through the Indego's encoders, which are then converted to match the sign convention of the VMR controller (negative torque corresponding to flexion). An onboard gyrometer measures the trunk forward lean angle. Velocities and accelerations of the joints and pelvis can then be obtained through Simulink's derivative block.

8.2.4   VMR Control Implementation

Muscle dynamics, governed by the reflex control equations, are estimated during each time-step through a Simulink function block. In accordance to the human biological control system, our controller uses delayed feedback variables. In the real-time Simulink model, each feedback variable must be stored in order to obtain delayed values from a previous time-step. We used this method instead of a built-in Simulink delay block for debugging purposes so the muscle function could be tested independently of the Simulink environment. Additionally, some states must be stored for use in the next time-step, particularly for the Rosenbrock fixed-step solver. Outputs of interest (i.e. joint torque and muscle

167

states) must be stored for off-line analysis. Therefore, each input/output variable for the VMR function requires its own read (input), write (output), and store blocks. Figure 35 provides an example of delaying a signal, in which both time and the knee angle (ϕ) have the current value and a vector of previous values as inputs to the VMR controller.
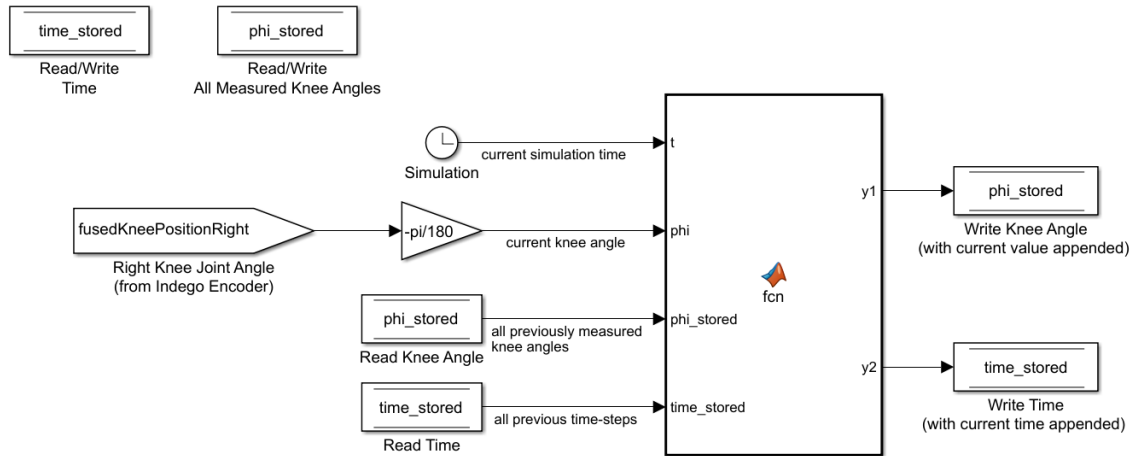

Figure 35: Simulink diagram to delay real-time data for the VMR controller. All reflex control parameters must account for the physiological time delay between the muscles and the spine or cerebellum. In this representative example, knee angle (ϕ) is delayed, in which the real-time value and a vector of all previous values are required as inputs to the controller, along with the time vector for performing interpolation

The Simulink VMR function requires 33 separate inputs and outputs, summarized in Table XVI. Note that all sensor inputs and outputs must have a stored vector for calculating the time-delays.

The full inputs and outputs for the VMR system can be viewed in Figure 41.

8.2.5 Safety Features

To safely operate the Indego with our controller, we recognized the need to introduce three precautionary capabilities:

- An option to switch the commanded current to 0 mA (passive-mode) at any time

- An option to deactivate the VMR controller at any time

Table XVI: List of variables required by the Simulink VMR function. The fixed-step Rosenbrock solver uses state variables and derivatives from the previous time-step. Any sensor variable must be stored in a vector of all previous measurements, in order to delay the signal. Outputs of the VMR controller are recorded for off-line analysis

Outputs of VMR Controller:
1) joint torques, 2) muscle forces, 3) muscle lengths, 4) control signals 5) joint angles (encoders)
Rosenbrock fixed-step solver variables:
6) x: state variables (L_{ce} and a) for each muscle
7) $\dot{x}$: state derivatives ($\dot{L}_{ce}$ and $\dot{a}$) for each muscle,
8) u_{dyn}, stored u for next time-step
Real-Time Sensor inputs:
9) simulation time, 10) ipsilateral GRF, 11) contralateral GRF, 12) knee angle, 13) knee velocity, 14) hip angle, 15) trunk lean angle, 16) trunk lean velocity, 17) trunk lean acceleration, 18) trunk lean at previous take-off
*Note that each real-time variable (other than the Rosenbrock) requires a stored vector of previous values, therefore totaling 33 inputs

- A method to ensure the commanded torques can operate within the Indego's motor capabilities

In this section, we describe our approach for designing each of these safety features.

Motor Current On/Off Switch

We implemented a simple switch to modulate between the torque commands generated from the VMR controller and a "passive-mode" of 0 mA. At the beginning of each Indego testing session, we set these switches to passive-mode for both the right and left legs. At any point during the Indego's operation, the current can be set to zero. Note that while in passive-mode, the hip and knee joint are not locked and can be manually moved into different positions without damaging the motors. An overview of this safety feature is illustrated in Figure 36.

Reflex Control On/Off Switch

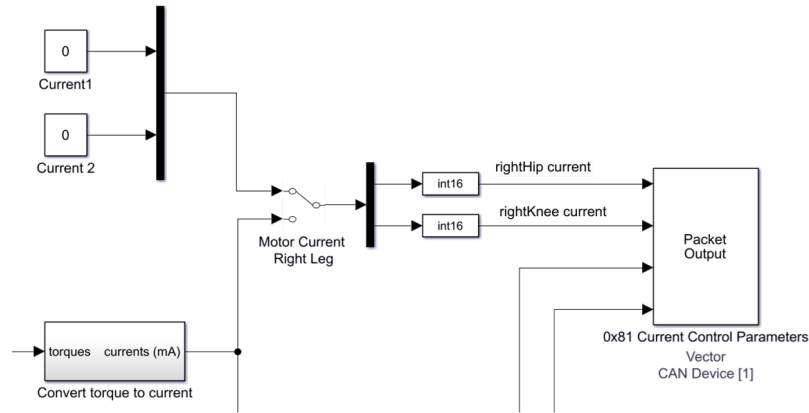The reflex controller is manually turned on or off using a constant block, where 0 cor-

Figure 36: Simulink diagram of the passive-mode safety feature for the Indego, in which the current signal sent to the motors can be set to 0 mA at any time during the operation

responds to "off" and 1 corresponds to "on." This allows the user to operate the virtual muscles independently of the reflexes, without changing the structure of the VMR function. In the controller, the "off" flag will set all neural signals to zero while still simulating muscle dynamics. Each muscle can be controlled using the co-activations, even if the reflexes are off.

Torque Scaling and Maximum Torque Cap

The nominal parameters used for our virtual muscles are capable of producing large muscle forces, and therefore, powerful joint torques. For instance, the maximum isometric force of the *Vasti* muscle group is 7000 N, which will result in a joint torque exceeding the motor's maximum limits. First, we scale the joint torques by a tenth of the generated value from the VMR controller. Next, we use a saturation block, which clips the signal to the maximum torque. The torque scaling and capping precaution is shown in Figure 37.

8.2.6 Model Validation

In order to test if the VMR system was correctly implemented on the Indego, we conducted two preliminary verifications. For both tests, we switched the Indego to passive-mode
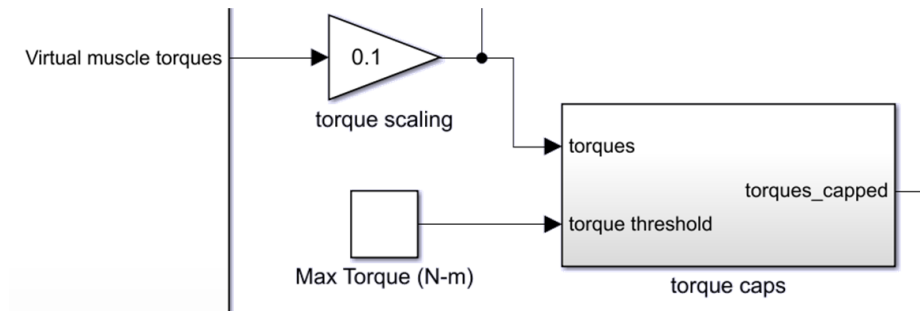
Figure 37: Simulink diagram of the torque-scaling safety feature for the Indego, where the output joint torques of the muscle model are first scaled to a value within the operation limits of the Indego's motors, and saturated by the maximum torque limit

where the output joint torques were not prescribed the electric motors.

1. Verification of Muscle Behavior

First, we will test the virtual muscles independently of the reflex controller, where we activate each muscle separately and observe the output torques performed by the Indego. Instead of using sensor data, we use constant joint angles of $0°$ for each joint. Intuitively, activating the *Hamstrings* should produce an extension torque in the hip and a flexion torque in the knee. For this test, we activate each muscle using a step-input command of 1 for maximum joint torque, and then set the control for the muscle back to 0. We activate the muscles in the following order: *Gastrocnemius*, *Vasti*, *Hamstrings*, *Gluteals*, *Iliopsoas*. Note that the maximum joint torque is scaled by a factor of $0.1\ \mathrm{N\,m}$.

2. Human Experimental Data as Sensor Input

Next, in place of using Indego sensor data as input to the VMR system, we prescribe joint positions and GRFs from recorded data of able-bodied human walking (see Part I, Chapter II for details concerning this dataset). Parameters for the reflex controller were adjusted to Geyer and Herr's nominal parameters [23] with some additional tuning in order to generate cyclical torque patterns that resembled walking. For demonstration purposes, we tuned the *Vasti* control parameters to produce larger muscle forces. This will demonstrate how an

adjustment in the control signal of a particular muscle should alter the gait pattern; in this case, larger knee extension torque.

8.3   Results

The muscle forces and joint torques of the Indego's VMR controller respond appropriately to the prescribed muscle excitations. Figure 38 demonstrates the first test case where joint angles were held constant at 0° while a step-control fully actuated each muscle at random times throughout the simulation. Note that these results are the recorded torque commands and not a measurement of the torque produced by the motor. Negative torque corresponds to both knee flexion and hip flexion. Predicted torques in the hip (blue) and knee (red) correspond as expected to the muscle activations. The torque directions in the hip and knee are influenced by muscle activations in the anterior and posterior of the leg. Additionally, the biarticular properties of muscles are represented when the *Hamstrings* (green) are activated, in which we observe both hip extension and knee flexion.

Figure 39 demonstrates the resulting torque commands when human experimental data is used as sensor input. Negative torque corresponds to both hip and knee flexion. The torque signals are scaled by a factor of 0.1 to remain within the torque limits of the Indego's electric motors.

Muscle force (bottom) responds appropriately to its neural activation (middle) and produces a torque in the proper direction (top). For example, activating the *Hamstrings* and *Gluteal* muscles results in a hip extension torque (blue). Additionally, when the *Hamstrings* are active, we also observe a knee flexion torque (red) at the same time. For a demonstration, we adjusted the control parameters of the *Vasti* to produce high muscle forces. This adjustment is noted in the larger knee extension torque. The magnitude of the torque varies in response to changes in muscle force, shown clearly by the *Vasti* behavior.
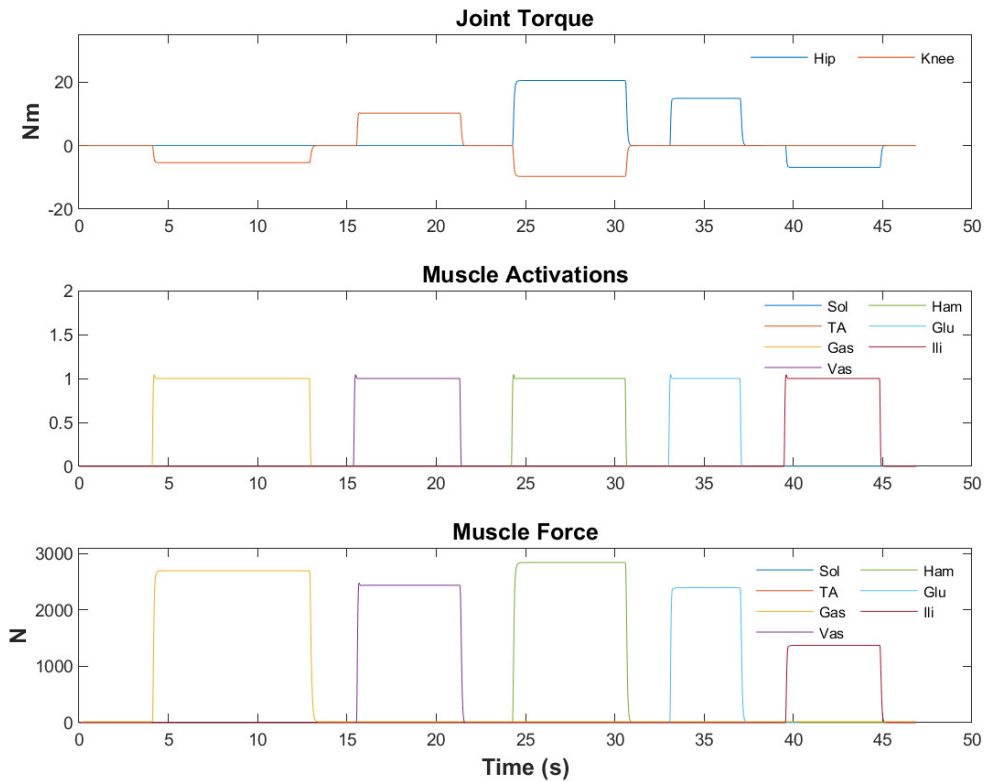
172

Figure 38: Predicted joint torques (top), muscle activations (middle) and muscle forces (bottom) with constant joint angles held at $0°$ and step-control signal u randomly applied throughout the simulation. Negative torque corresponds to knee flexion and hip flexion

8.4 Discussion

In this study, we designed a controller based on the human biological actuation system, including muscles, reflexes, and vestibular feedback, with the goal to provide more natural gait mechanics to users of a powered orthotic device. Our controller produces a joint torque, which can then control the electric motors of an exoskeleton. Using Simulink, we compiled our VMR controller into C-code and uploaded it into the Indego's microcontroller. Sensor information from the Indego was used as input to the VMR controller and processed through Simulink and the SLDRT toolbox. We incorporated several safety features, such as a passive/active mode switch, a controller on/off switch, and a torque-scaling subsystem.
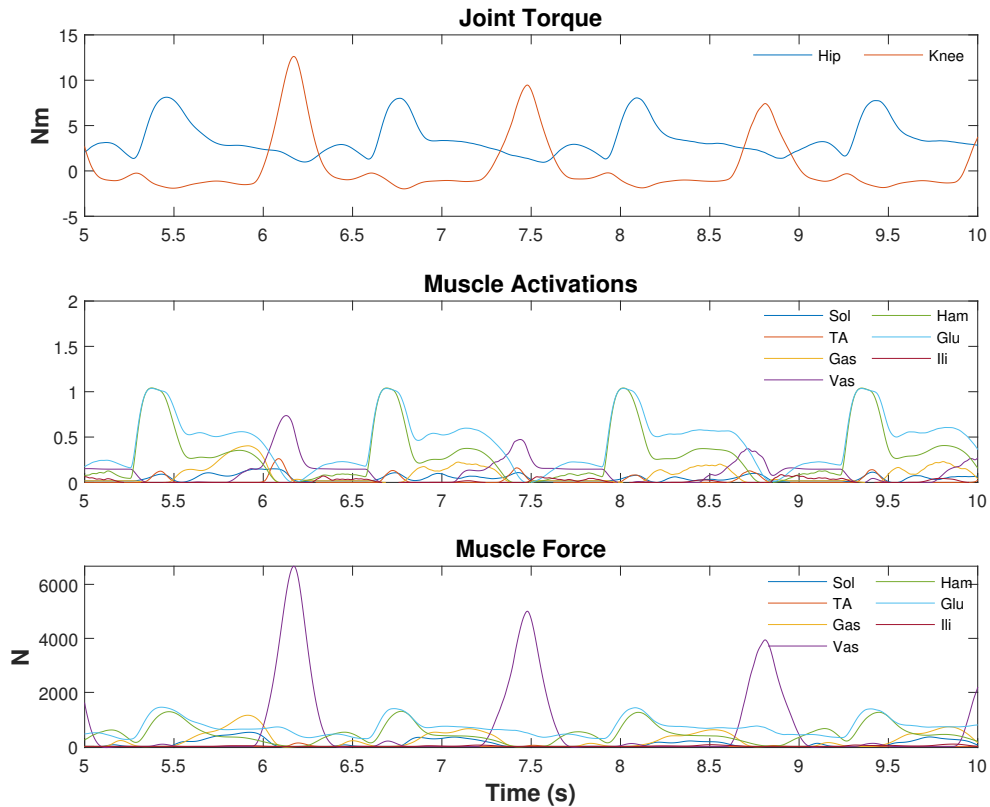
Figure 39: Predicted joint torques (top), muscle activations (middle), and muscle forces (bottom), generated using able-bodied human walking data as sensor inputs to the VMR system

Before physically testing the controller on the Indego, we set the electric motors to passive-mode while performing two preliminary verifications to ensure the controller had been correctly implemented. First, we set the joint angles to a constant of $0°$ and activated each muscle independently using a step-input as a neural control signal. Then, we used able-bodied human walking data as sensor inputs and manually tuned our VMR controller in order to observe the resulting joint torque. Based on both of these tests, the joint torques of the knee and hip respond appropriately given the prescribed muscle activation. The biarticular nature of the *Hamstring* produces a knee flexion and hip extension when activated. Additionally, by tuning the muscle parameters of the *Vasti*, we were able to demonstrate that modifying the controller will result in an intuitive change in the respective joint torques.

8.4.1 External Sensor Data Limitations

Since the Indego does not have a sensor for measuring GRFs, we intended to read raw analog force plate signals in Simulink measured from a DAQ. However, hardware and software incompatibility did not allow for these signals to be read while simultaneously operating the Indego. With the SLDRT toolbox, any function or Simulink block must be able to be compiled in C-code. Available Simulink toolboxes for reading external signals (the Data Acquisition Toolbox) are not compatible with SLDRT code generation. Similarly, the SL-DRT blockset, which can be compiled into C-code, does not support the model number of our DAQ (NI-USB 6255). Communicating to two separate targets is not possible with the SLDRT toolbox; for instance, Simulink cannot communicate with both the Indego and a more sophisticated DAQ with its own measurement software, such as the MicroLabBox by dSPACE (dSPACE Inc., Wixom, Michigan, USA).

Without these real-time signals, the VMR system cannot be fully tested since vertical GRF is a required feedback for the reflex controller. We recommend the following options for reading GRF signals while operating the Indego:

- **Read analog sensors through an SLDRT-Compatible DAQ:** since code generation is possible for any block within the SLDRT blockset, GRFs can be read in Simulink by a supported DAQ or National Instruments card. However, appropriate PC hardware is required to communicate in real-time to the installed `sldrtkernel`. Check PC hardware for compatibility.

- **Read all sensor signals into the same DAQ:** Simulink models can be compiled for use within other systems, particularly the dSPACE MicroLabBox and ControlDesk software. However, to communicate to the Indego, a method other than a CAN bus should be used, thereby requiring modifications to the Indego hardware itself.

- **Create an extrinsic function to bypass code generation:** a custom MATLAB function can be written to mimic the internals of the Data Acquisition toolbox using

`coder.extrinsic` commands. With an extrinsic function, SLDRT can compile the Simulink model while skipping the internal code of this function.

- **Install force sensors on the Indego:** instead of measuring GRFs from an external source, force sensors can be installed on the Indego directly and read from its micro-controller.

8.4.2 Suggestions for Future Studies

Our aim in designing the VMR system is to provide more natural gait mechanics to the user of a powered orthotic device. For initial controller testing, we recommend performing experiments with an able-bodied test subject. A simple first test compares electromyography (EMG) signals between the Indego's passive-mode and the VMR controller. EMG sensors measure the electrical signal produced by a muscle.

Surface EMGs can be placed on the following muscles: *Lateral gastrocnemius*, *Medial gastrocnemius*, *Vastus lateralis*, *Vastus medialis*, *Gluteus maximus*, and *Semitendinosus*. Though EMG sensors could also be placed on the *Soleus* and *Tibialis anterior*, these muscles are not expected to actively contribute to the motion since the foot and ankle are supported by a rigid AFO while wearing the Indego.

The test subject could walk in three separate walking conditions while EMG measures their muscle activity:

1. *Test 1, Passive-mode:* electric motors receive no current, though the user's gait mechanics are still altered when wearing the Indego. This test can be used as a benchmark when comparing other controllers

2. *Test 2, Impedance or PD controller*: a standard, state-of-the-art controller typically used in other powered orthotics

3. *Test 3, VMR controller*: an optimized muscle-reflex controller

EMG signals can then be compared between the three trials. Ideally, the user would require less muscle force when our VMR controller is active on the Indego over the passive-mode and PD controller.

8.5 Conclusion

We were able to design and implement a VMR controller for a powered orthosis. With the software built, we provide an opportunity for future studies regarding a biologically-inspired controller and their ability to facilitate or augment walking for users of such a device.

Figure 40: Full VMR Simulink model for controlling the hip and knee of the right and left legs
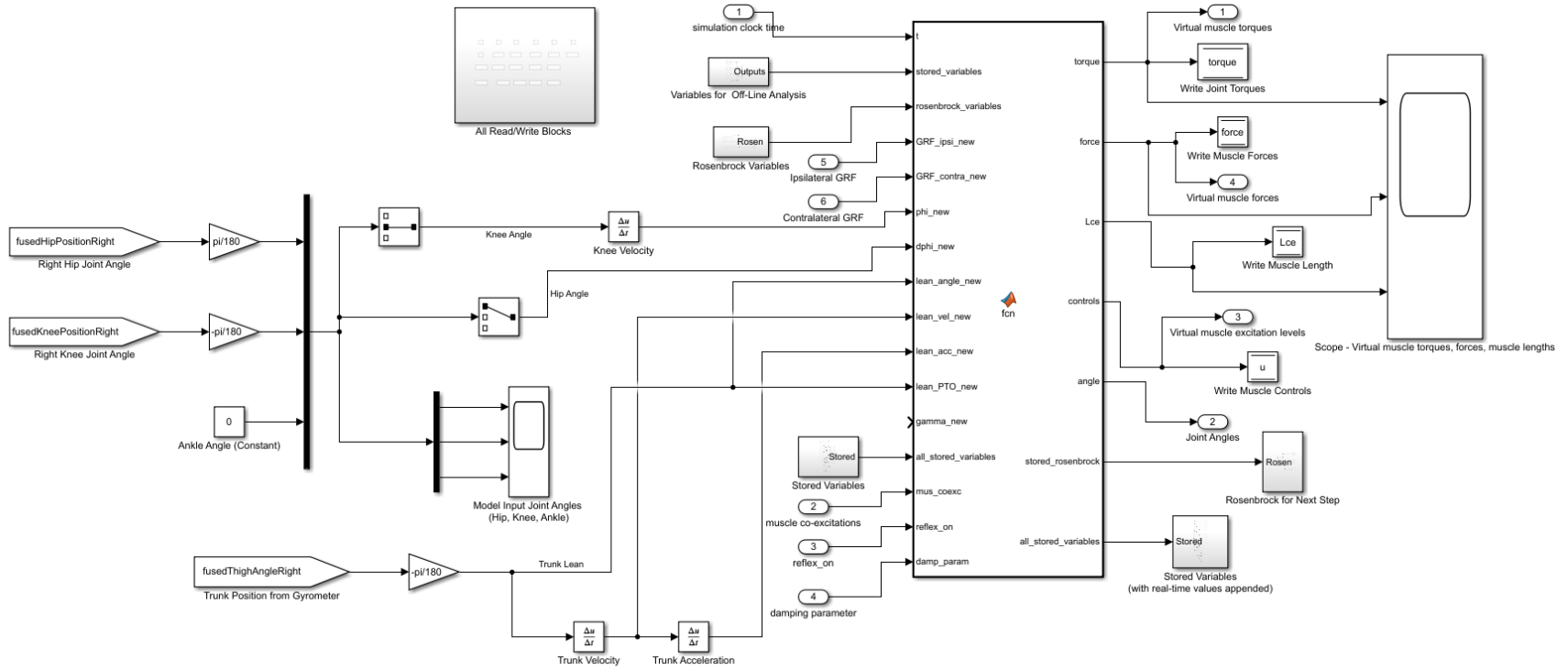
Figure 41: VMR Function Inputs and Outputs for the Indego. For details, see text.

8.6 REFERENCES

[1] A. M. Dollar and H. Herr. Lower extremity exoskeletons and active orthoses: Challenges and state-of-the-art. *IEEE Transactions on Robotics*, 24(1):144–158, February 2008.

[2] I. Diaz, J. J. Gil, and E. Sanchez. Lower-limb robotic rehabilitation: Literature review and challenges. *Journal of Robotics*, 2011(Article ID 759764):1–11, September 2011.

[3] K. N. Winfree, P. Stegall, and S. K. Agrawal. Design of a minimally constraining, passively supported gait training exoskeleton: ALEX II. In *2011 IEEE International Conference on Rehabilitation Robotics*, pages 1–6, July 2011.

[4] S. K. Banala, S. K. Agrawal, and J. P. Scholz. Active leg exoskeleton (ALEX) for gait rehabilitation of motor-impaired patients. In *2007 IEEE 10th International Conference on Rehabilitation Robotics*, pages 401–407, June 2007.

[5] J. F. Veneman, R. Kruidhof, E. E. G. Hekman, R. Ekkelenkamp, E. H. F. Van Asseldonk, and H. van der Kooij. Design and evaluation of the LOPES exoskeleton robot for interactive gait rehabilitation. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 15(3):379–386, September 2007.

[6] K. Anam and A. A. Al-Jumaily. Active exoskeleton control systems: State of the art. *Procedia Engineering*, 41:988–994, December 2012. ISSN 1877-7058.

[7] A. Esquenazi, M. Talaty, A. Packel, and M. Saulino. The ReWalk powered exoskeleton to restore ambulatory function to individuals with thoracic-level motor-complete spinal cord injury. *American Journal of Physical Medicine & Rehabilitation*, 91(11): 911–921, November 2012.

[8] A. Esquenazi. New bipedal locomotion options for individuals with thoracic level

motor complete spinal cord injury. *Journal of the Spinal Research Foundation*, 8(1): 26–28, 2013.

[9] H. Kawamoto, S. Lee, S. Kanbe, and Y. Sankai. Power assist method for HAL-3 using EMG-based feedback controller. In *IEEE International Conference on Systems, Man and Cybernetics*, volume 2, pages 1648–1653, October 2003.

[10] What's HAL? - The World's First Cyborg Type Robot. `https://www.cyberdyne.jp/english/products/HAL/`. Accessed: 2017-04-09.

[11] H. Sankai and Y. Kawamoto. EMG-based hybrid assistive leg for walking aid using feedforward controller. In *Proceedings of the International Conference on Control, Automation and Systems*, pages 190–193, 2001.

[12] K. Suzuki, G. Mito, H. Kawamoto, Y. Hasegawa, and Y. Sankai. Intention-based walking support for paraplegia patients with Robot Suit HAL. *Advanced Robotics*, 21 (12):1441–1469, April 2007.

[13] Y. Sankai. HAL: Hybrid assistive limb based on cybernics. In *Robotics Research*, pages 25–34, 2010.

[14] S. A Kolakowsky-Hayner, J. Crew, S. Moran, and A. Shah. Safety and feasibility of using the EksoTM bionic exoskeleton to aid ambulation after spinal cord injury. *Journal of Spine*, 4:003, September 2013.

[15] E. Strickland. Good-bye, Wheelchair, Hello Exoskeleton. *IEEE Spectrum*, 1:30–32, December 2011.

[16] G. J. Androwis and K. J. Nolan. Evaluation of a robotic exoskeleton for gait training in acute stroke. *Archives of Physical Medicine and Rehabilitation*, 97(10):e115–e116, October 2016.

[17] R. J. Farris, H. A Quintero, S. A. Murray, , K. H. Ha, C. Hartigan, and M. Goldfarb. A preliminary assessment of legged mobility provided by a lower limb exoskeleton for persons with paraplegia. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 22(3):482–490, May 2014.

[18] D. A. Winter. *Biomechanics and Motor Control of Human Movement*. John Wiley and Sons, New York, 1990.

[19] F. Sup, A. Bohara, and M. Goldfarb. Design and control of a powered transfemoral prosthesis. *International Journal of Robotics Research*, 27(2):263–273, February 2008.

[20] B. E. Lawson, B. Ruhe, A. Shultz, and M. Goldfarb. A powered prosthetic intervention for bilateral transfemoral amputees. *IEEE Transactions on Biomedical Engineering*, 62(4):1042–1050, April 2015.

[21] K. G. Gerritsen, A. J. van den Bogert, M. Hulliger, and R. F. Nernicke. Intrinsic muscle properties facilitate locomotor control - A computer simulation study. *Motor Control*, 2(3):206–220, July 1998.

[22] A. J. van den Bogert, D. Blana, and D. Heinrich. Implicit methods for efficient musculoskeletal simulation and optimal control. *Procedia IUTAM*, 2:297–316, January 2011.

[23] H. Geyer and H. Herr. A muscle-reflex model that encodes principles of legged mechanics produces human walking dynamics and muscle activities. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 18(3):263–273, June 2010.

[24] S. Song and H. Geyer. A neural circuitry that emphasizes spinal feedback generates diverse behaviours of human locomotion. *The Journal of Physiology*, 593(16):3493–3511, August 2015.

[25] N. Thatte and H. Geyer. Toward balance recovery with leg prostheses using neuromuscular model control. *IEEE Transactions on Biomedical Engineering*, 63(4): 904–913, May 2016.

[26] H. Herr and A. M. Grabowski. Bionic ankle–foot prosthesis normalizes walking gait for persons with leg amputation. *Proceedings of the Royal Society of London B: Biological Sciences*, 279:457–464, February 2012.

[27] M. F. Eilenberg, H. Geyer, and H. Herr. Control of a powered ankle-foot prosthesis based on a neuromuscular model. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 18(2):164–173, April 2010. doi: 10.1109/TNSRE.2009. 2039620.

[28] F. Dzeladini, A. R. Wu, D. Renjewski, A. Arami, E. Burdet, E. van Asseldonk, H. van der Kooij, and A. J. Ijspeert. Effects of a neuromuscular controller on a powered ankle exoskeleton during human walking. In *2016 6th IEEE International Conference on Biomedical Robotics and Biomechatronics (BioRob)*, pages 617–622, June 2016.

[29] K. G. M. Gerritsen, W. Nachbauer, and A. J. van den Bogert. Computer simulation of landing movement in downhill skiing: Anterior cruciate ligament injuries. *Journal of Biomechanics*, 29(7):845–854, July 1996.

[30] V. Dietz. Proprioception and locomotor disorders. *Nature Reviews Neuroscience*, 3 (10):781–790, October 2002.

[31] J. K. Moore, S. K. Hnat, and A. J. van den Bogert. An elaborate data set on human gait and the effect of mechanical perturbations. *PeerJ*, 3:e1248, 4 2015.

CHAPTER IX

EVALUATION OF A VIRTUAL MUSCLE MODEL

USING A POWERED EXOSKELETON

With the VMR controller built for use on the Indego, we then performed a test using the virtual muscles. Here, we describe our experiment where we compared torque command sent to the Indego motors and the measured torque recorded by a Biodex-2 isokinetic dynamometer typically used in rehabilitation exercises. A Biodex measures angular position, velocity, and joint torques. We programmed the Indego to perform the same isometric and isokinetic knee exercises used in human rehabilitation to determine if the virtual muscles follow similar force-length and force-velocity patterns observed in these movements.

Intended Publication:

1. S. K. Hnat, Huawei Wang, Raviraj Nataraj, and A. J. van den Bogert. Evaluation of a Virtual Muscle Model using a Powered Exoskeleton *In preparation.*

ABSTRACT

Phase-based impedance controllers are the current standard for controlling a powered orthosis. We hypothesize that the spring-damper characteristics of an impedance controller can be achieved with virtual muscles. Here, we design a muscle controller that produces a joint torque signal prescribed to the electric motors of a powered exoskeleton. The controller was modeled and built in MATLAB® Simulink® for use on the Parker Hannifin Indego® exoskeleton. To test the force-length and force-velocity relationships of our muscles, we program the Indego to perform standard isokinetic and isometric knee rehabilitation exercises, in which the joint angle, velocity, and torque is measured by a Biodex-2 isokinetic dynamometer. In the isometric results, root-mean-square (RMS) error between the measured and commanded extension and flexion torques were 3.28 Nm and 1.25 Nm, respectively. In the isokinetic trials, we receive RMS error between the measured and commanded extension and flexion torques of 0.73 Nm and 0.24 Nm, respectively. The muscle model demonstrates typical force and length dependencies as shown in human muscles undergoing the same exercises. Therefore, we conclude a virtual muscle model is capable of the same stabilizing properties as observed in an impedance controller.

9.1 Introduction

Powered orthotic devices aim to improve the quality of life for persons with a spinal cord injury (SCI). *Exoskeletons* are anthropomorphic devices worn by the user which enhance or assist the operator's performance, while *powered orthoses* are classified as a type of exoskeleton that restores function to individuals with a SCI or other type of pathological condition which affects ambulation [1]. Recently, several powered orthoses have been developed for rehabilitative gait training. These devices can lessen strain on the therapist while measuring the patient's progress throughout the treatment, such as improvements in force production and joint angle patterns [2]. Many of these devices, including the ALEX

I and II [3, 4] and the LOPES robot [5], have been designed as overhead treadmill gait trainers, operating in parallel to the body to guide paralyzed or weakened limbs through walking motions [6].

Outside of rehabilitation environments, a few powered orthoses have been developed and marketed for home use. These devices can be used to supplement the patient's capabilities or to completely replace walking for individuals with paraplegia. Some commercialized models include the ReWalk[TM] by ReWalk Robotics[TM] [7, 8], the Hybrid-Assistive-Leg® (HAL®) by Cyberdyne Inc. [9–13], Ekso[TM] by Ekso Bionics®, [14–16], and the Parker Hannifin Indego® exoskeleton [17]. Typically, these devices are equipped with torque-driven electric motors at the joints. The motion from the electric motors reproduce a predefined set of joint trajectories in order to replicate standard walking patterns observed in able-bodied human gait [18] and to perform sit-to-stand maneuvers.

Various control strategies have been designed and implemented for powered orthoses, though phase-based impedance controllers are the current standard [19, 20], in which the joint torque is calculated as a function of the joint angle and joint velocity. Instead of an impedance controller, the spring-damper characteristics can also be achieved through the nonlinear viscoelastic and dynamics properties of muscles. In simulation, virtual muscles have been shown to greatly increase stability during walking when compared to direct torque-driven control [21]. Recently, a simulation of muscle dynamics using a Hill-type muscle model was achieved in real-time [22], which is a requirement for orthotic devices (for details concerning this model, refer to Part II, Chapter V).

In this study, we evaluated the performance of an implicit, virtual muscle model on the Parker Hannifin Indego exoskeleton (hereafter, Indego). The Indego is a battery-powered, relatively lightweight (12-kg) device designed with brushless motors at the hip and knee joints and a rigid ankle-foot orthosis (AFO). The electric motors of the Indego are capable of operating within the range of motion and torque magnitudes observed during able-bodied walking [18].

186

To test the muscle model in hardware, we programmed the Indego to perform the standardized isokinetic and isometric tests used during many knee rehabilitation exercises [23–25]. The objectives of this study are to: 1) compare the prescribed motor torque at the knee joint and the measured torque recorded by an isokinetic dynamometer (Biodex System 2, Biodex Medical Systems, Shirley, New York, USA), and 2) qualitatively compare the resulting torque-angle and torque-velocity relationships to previous literature studies to verify if the virtual muscle model behaves similarly to a human subject undergoing the same exercises.

9.2 Methods

Here, we describe the virtual muscle model (9.2.1) and its implementation on the Indego (9.2.2), as well as the rehabilitation exercise testing protocol to be measured by the Biodex (9.2.3).

9.2.1 Muscle Model

For this study, we used a planar leg model with three muscle groups (*Gastrocnemius*, *Vasti*, and *Hamstrings*) to actuate the knee joint in both flexion and extension (shown in Figure 42). Note that our sign convention defines knee flexion as negative and knee extension as positive.

Each muscle is represented by a standard 3-element Hill-type model, including a contractile element (CE) based on standard force-length and force-velocity relationships, series/parallel nonlinear elastic elements (SEE/PEE), and a small amount of viscous damping for numerical stability. The nominal properties of these muscles were obtained from previous literature [26]. Inputs to the muscle model include the muscle length (derived from the joint angles measured by the encoders) and neural excitation signals.

Using this muscle architecture, activation and contraction dynamics are represented as a set of two differential equations, where the state variables are the length of the contractile
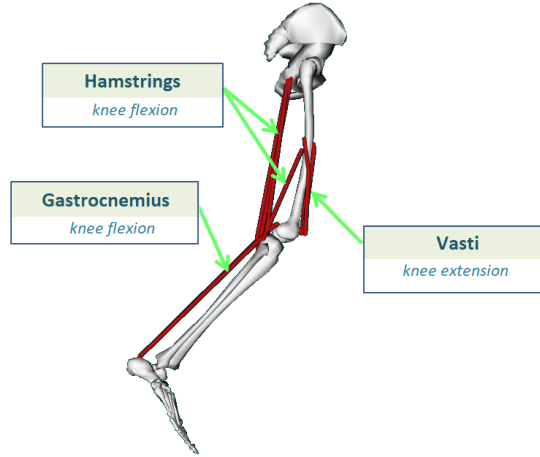
Figure 42: Virtual muscles used in the study, in which muscle pairs are capable of producing flexion and extension at the knee joint. These muscles include: *Gastrocnemius*, the *Vasti*, and the *Hamstrings*

element (L_{CE}) and the muscle activation signal (a). Activation dynamics are driven by neural excitation signals (u) from the brain or spinal cord,

$$\dot{a} = (u - a)\left(\left(\frac{u}{T_{act}} - \frac{1 - u}{T_{deact}}\right)\right)$$
(9.1)

where T_{act} and T_{deact} are the the time constants of muscle activation and deactivation, respectively.

Subsequently, the overall force in the muscle (F_{SEE}) can then be calculated based on the individual force contributions of the elements of the Hill-type muscle model,

$$F_{SEE} = a \cdot F_{max} \cdot f_{FL}(L_{CE}) \cdot f_{FV}(\dot{L}_{CE}) + F_{PEE} + F_D$$
(9.2)

where F_{max} is the maximum isometric force of the muscle, f_{FL} and f_{FV} are the force-length and force-velocity relationships, respectively, and F_{PEE} and F_D are the force contributions due to the PEE and damping element.

The muscle contraction and muscle activation relationships can be combined into a

single, implicit state equation

$$f(x, \dot{x}, u) = 0 \tag{9.3}$$

with state variables $x_n = \begin{bmatrix} L_{CE,1}...L_{CE,n} & a_1...a_n \end{bmatrix}^T$ and state derivatives

$\dot{x}_n = \begin{bmatrix} \dot{L}_{CE,1}...\dot{L}_{CE,n} & \dot{a}_1...\dot{a}_n \end{bmatrix}^T$, where n is the number of muscles, L_{CE} is the length of the contractile element and a is the muscle activation signal. The control inputs, u, are the neural excitation control signals generated by a neural controller. Real-time muscle dynamics can be simulated in MATLAB® (Mathworks, Santa Rosa, California, USA) using a first-order, implicit Rosenbrock solver [22]. For more details concerning the muscle model architecture, refer to Part II, Chapter V.

The muscle model produces a muscle force which is multiplied by the moment arm to obtain the joint torque used to control the electric motors of the Indego. Figure 43 summarizes the input and outputs of the virtual muscle model.
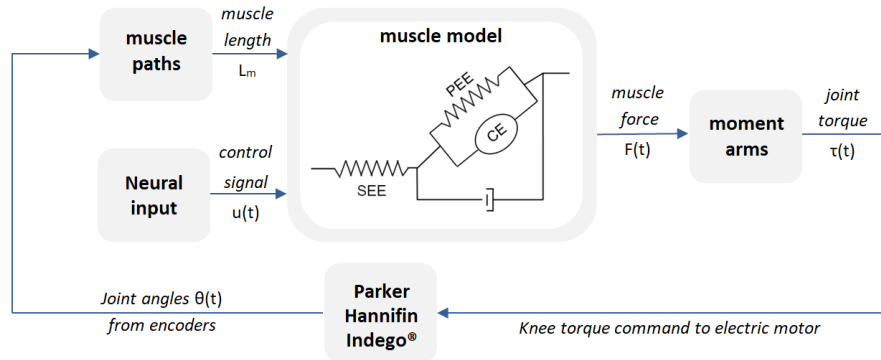
Figure 43: Virtual muscle model, where the inputs to the model are the muscle length L_m (determined from joint angles from encoder measurements), and the neural excitation signals $u(t)$. The output is the muscle force which can be multiplied by the moment arm to obtain the joint torque, $\tau(t)$ prescribed to the electric motors of the Indego®

9.2.2   Virtual Muscle Implementation on the Indego

Building our model on the Indego requires the use of the following MATLAB and Simulink toolboxes: Simulink Coder™, MATLAB Coder™, and Simulink Desktop Real-Time™.

After building a Simulink model, these toolboxes compile the model into C-code, which can then be uploaded onto the microcontroller of the Indego. The Indego communicates to the PC through a USB + Vector VN1610 CAN interface (Vector, Stuttgart, Germany) at a sample rate of 200 Hz. Joint position is obtained through the Indego's encoders and the motors are operated through current commands (mA).

We include the muscle model in Simulink through function block. The function 1) accepts joint positions and neural control signals as input, 2) estimates real-time muscle dynamics, and 3) outputs a muscle force. Neural command inputs must be a scalar value between 0 and 1, where 0 represents no activation and 1 represents maximum activation. For more details concerning the model implementation, refer to Part III, Chapter VIII.

9.2.3 Isokinetic Dynamometers for Measuring Muscle Performance

An isokinetic dynamometer is a popular machine for rehabilitation due to their ability to reliably and accurately measure muscle function and performance [27]. A dynamometer can provide insight regarding muscle properties that are difficult to obtain by resisting motion and controlling the velocity of the movement [28]. Some of these properties include shortening and lengthening rates and net power. In rehabilitation, both isometric and isokinetic exercises are often prescribed to individuals suffering from an injury, or following a surgical operation, particularly the anterior cruciate ligament [29].

Isometric rehabilitation exercises hold the joint in a fixed position, while the patient activates the appropriate muscles for flexion or extension. By repeating the procedure at different angular positions, the resulting torque-angle relationship corresponds to the force-length relationships of the muscles. Each muscle produces a maximum force at its optimal fiber length, with decreasing force as the joint angle deviates from this position. For example, the *Vasti* muscles are capable of larger forces in a flexed knee position, and the least force in full extension since the muscle is short in that position.

Isokinetic rehabilitation exercises guide the movement of the joint through a specified

190

range of motion using a constant angular velocity. In a concentric/concentric exercise, both the anterior and posterior muscles must contract in their respective torque direction in order to overcome the resistance of the dynamometer. Repeating the procedure at different angular velocities will result in torque-velocity relationships, which highlight the force-velocity relationships of a muscle, where the muscle force decreases while shortening velocity increases.

For our study, we use a Biodex-2 isokinetic dynamometer at Cleveland State University's Control, Robotics, and Mechatronics Laboratory, to control the joint position and joint velocity of the Indego. To test the virtual muscles, we programmed the Indego to perform the following standard isometric and isokinetic test conditions:

- *Isometric Test Conditions:* one trial of knee flexion and one trial of knee extension at each of the following angles (°): 15, 30, 45, 60, 75. Each trial contains ten repetitions of maximum joint torque for three seconds, and a rest period of three seconds. A visual representation of knee flexion angles is provided in Figure 44

- *Isokinetic Test Conditions:* concentric/concentric trials at each of the following angular velocities (°/s): 30, 60, 90, and 120. Each trial contained ten repetitions, where one cycle consisted of both knee flexion and extension between joint positions of 15° and 75°
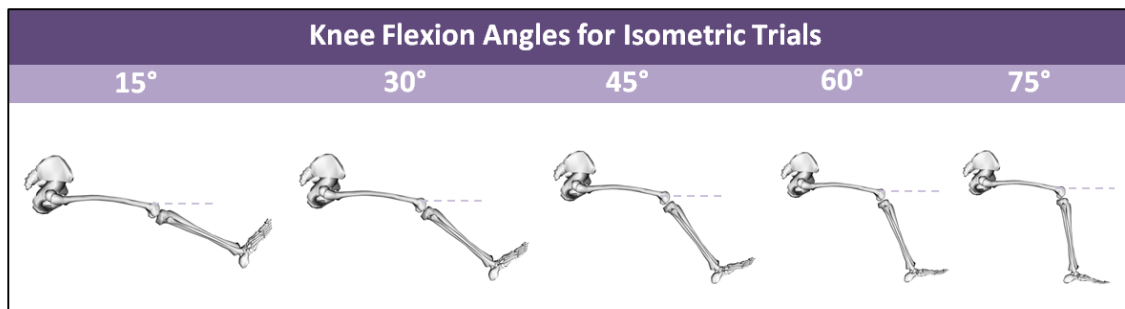
Figure 44: Isometric testing protocol with indicated knee angle positions. Both flexion and extension step-inputs are repeated ten times for each position

For the isometric test conditions, the neural control signals prescribed to the Indego are step-inputs ranging from 0 to 1, repeated ten times. Under the isokinetic conditions, these signals are square waves ranging from 0 to 1, where antagonist muscles are completely out of phase. This is to ensure each muscle can produce the maximum torque without limitations from an opposing torque on the opposite side of the leg. The time period of each pulse was manually adjusted until the desired angular velocity was achieved, while reaching the minimum and maximum angle during the motion.

In each trial, both the Biodex and Indego measure the angular position, velocity, and joint torque. The Biodex samples isometric tests at 60 Hz and isokinetic tests at 125 Hz. Data was manually synchronized between the trials, based on the frame numbers of the first and last peak torque of each trial.

It is important to note that the *Hamstrings* are a biarticular muscle which will produce a hip extension torque when activated. Because the Indego is a lower-limbed exoskeleton, securing it to the Biodex seat is challenging without using the shoulder straps. The flexible frame of the Indego allows the device to bend when a muscle is activated. To limit the twisting motion, we set the current to the Indego's hip motor to 0 mA for passive-mode. We added weights to the Indego's "thigh" in order to more adequately secure it to the seat. Additionally, we inserted foam inside the leg braces to provide some shape to the leg. Figure 45 shows the experimental test setup, with the Indego secured on the Biodex-2.

9.3 Results

Figure 46 provides a representative trial for flexion/extension torques measured in the isometric 30° condition (Figure 46a) and an isokinetic trial at 30 °/s (Figure 46b). In the isometric trials, the measured torque, angles, and velocities from the Biodex (red-dashed) indicate that the Indego is capable of performing the majority of the prescribed torque commands (black), approximately 85% of the commanded value in both flexion and extension.

However, in the isokinetic condition, the Indego consistently reaches the peak flexion

Figure 45: Experimental test setup for the isometric and isokinetic exercises, where the Indego is secured to the Biodex-2 via straps and weights. Foam was added inside the leg braces of the Indego's leg to add shape

torque magnitude, while the extensors still produced approximately 85% of the magnitude. The $10°$/s offset in the Indego's velocity measurement and can be attributed to noise within the encoder joint angle data due to synchronization errors.

Figure 47 illustrates the torque-angle relationship obtained through the isometric trials. Flexion torque of the Biodex (red) and Indego (purple) are negative, whereas the extension torques are positive (blue and yellow, respectively). Root-mean-square (RMS) error between the measured and commanded extension and flexion torques are 3.28 N m and 1.25 N m, respectively. The flexor torque-angle curves are somewhat flat, due to a larger width parameter based on human isometric tests [26].

These results confirm that the virtual muscles demonstrate typical force-length dependencies. The *Vasti* produce 12.73 N m and 23.28 N m of measured extension torque at $15°$ and $75°$, respectively. This is expected since the *Vasti* can produce higher muscle forces
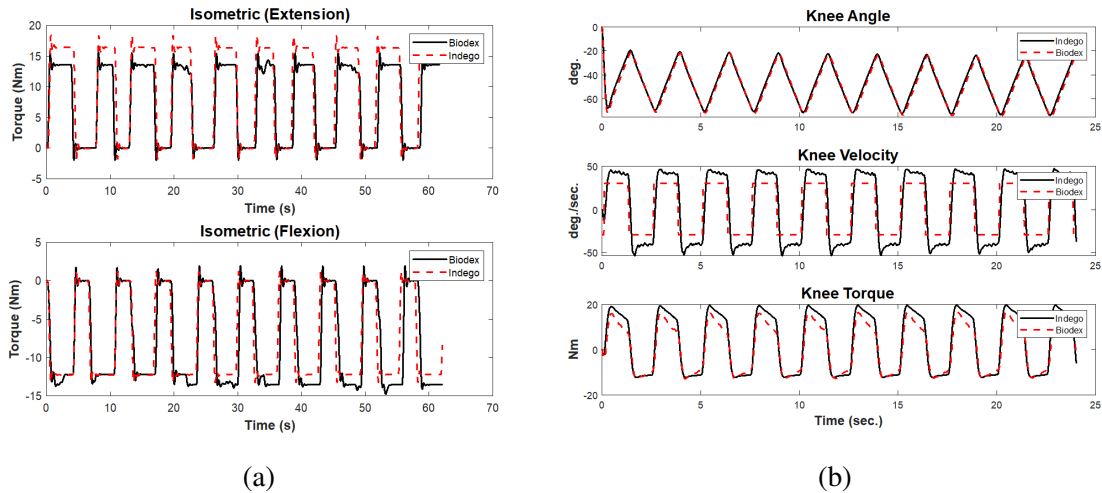
Figure 46: A representative (a) isometric trial with knee angle of 30° and (b) isokinetic trial at 30 °/s, where the data measured on the Biodex (red-dashed) and data prescribed to the Indego (black) are compared. In Figure 46a, knee extension (*Vasti* activation, top) and knee flexion (*Hamstrings* and *Gastrocnemius*, bottom) are shown. In Figure 46b, the angular position (top), angular velocity (middle), and joint torques (bottom) are shown.
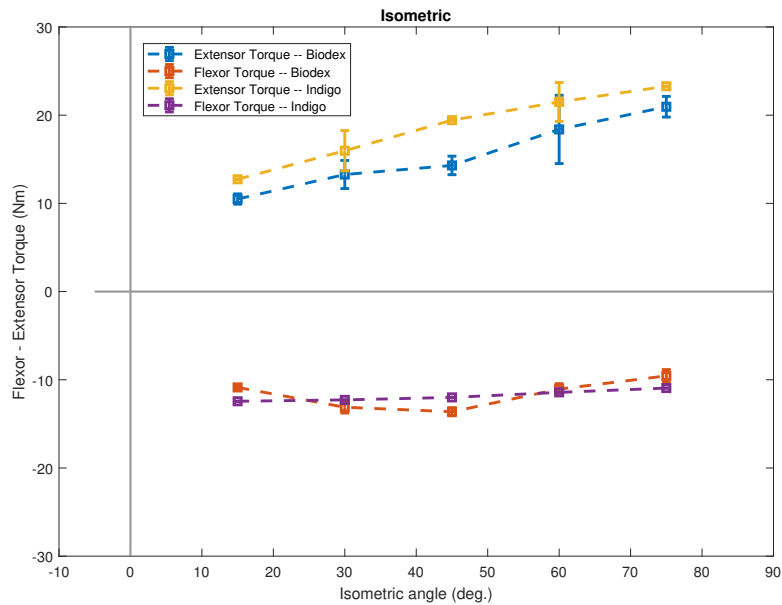
Figure 47: Torque-angle results of all isometric trials. In the extension trials (top), the torque measured by the Biodex (blue) and prescribed to the Indego (yellow) are compared, while the flexion trials (bottom) are also shown (red and purple, respectively)

when the knee is flexed. Likewise, the flexors produce the largest torque of $13.62\,\mathrm{N\,m}$

at 45°. This muscle behavior is similar to torque-angle relationships reported in previous

literature of human subjects undergoing similar isometric test conditions [30].

Figure 48 shows the torque-velocity relationship obtained through the isokinetic trials. Torque values were calculated at a knee angle of $45°$. Flexion torques of the Biodex (red) and Indego (purple) are negative, whereas the extension torques are positive (blue and yellow, respectively). RMS error between the measured and commanded extension and flexion torques are $0.73 \, \mathrm{N\,m}$ and $0.24 \, \mathrm{N\,m}$, respectively.
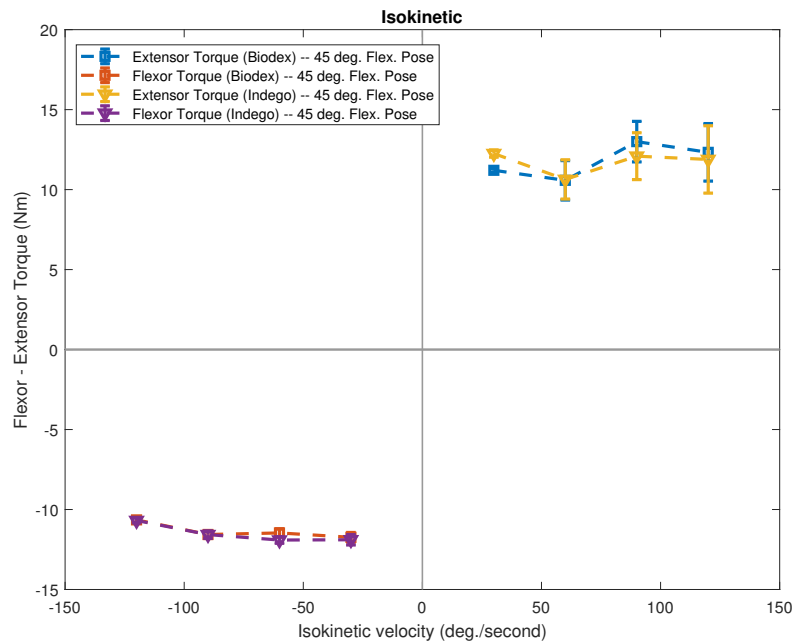
Figure 48: Torque-velocity results of all concentric/concentric isokinetic trials. The extension results (right) compare the torque measured by the Biodex (blue) and prescribed to the Indego (yellow), while the flexion results (left) are also shown (red and purple, respectively)

Our results suggest the virtual muscles demonstrate typical force-velocity properties of these muscles. In flexion, the *Hamstrings* and *Gastrocnemius* show increasing torque with decreasing shortening velocity, with measured torque values of $10.69 \, \mathrm{N\,m}$ and $11.89 \, \mathrm{N\,m}$ at $120 \, °/s$ and $30 \, °/s$, respectively. This behavior is not apparent in the extensors, which can possibly be attributed to the Indego's position shifting in the Biodex-2 seat.

9.4 Discussion

In powered orthotic devices, impedance or PD controllers are typically used to reproduce able-bodied walking joint trajectories. However, spring-damper characteristics can be achieved through the nonlinear viscoelastic and dynamics properties of muscles, which have been reported to improve stability in walking simulations. Similar to an impedance controller, muscles also demonstrate length and velocity dependencies.

In this study, we designed a real-time virtual muscle controller using a Hill-type muscle model, where muscle contraction and activation dynamics were simulated using a fixed-step implicit Rosenbrock solver. The output of the controller is joint torque, which can then control the electric motors of the Indego exoskeleton. Using Simulink, we compiled our muscle controller into C-code and uploaded it into the Indego's microcontroller. Joint angle data from the encoders and a neural command signal (step-input or square wave) were used as inputs to the muscle controller.

The Indego then performed standard isometric and isokinetic knee exercises, measured on a Biodex-2 isokinetic dynamometer. We selected five angular positions for the isometric trials, for both flexion and extension, while the concentric/concentric isokinetic protocol included four angular velocities. The angular position, velocity, and joint torques were recorded by the Biodex and compared to the values prescribed to the Indego.

The torque-angle graphs, obtained through the isometric trials, indicate the Indego's electric motors could produce the majority of the peak extension and flexion torques, with RMS errors of 3.28 N m and 1.25 N m, respectively. In the torque-velocity curves, we receive RMS errors of 0.73 N m and 0.24 N m for extension and flexion torque, respectively, obtained from the isokinetic trials. These results suggest the Indego's motors are capable of producing the majority of the magnitude and frequency of the commanded torques in these isometric and isokinetic conditions. Our virtual *Hamstrings*, *Gastrocnemius* and *Vasti* muscles demonstrate similar force-length and force-velocity relationships as reported in previous literature. For an example, the *Vasti* produce more torque when closer to its

optimum fiber length.

In all trials, the peak extension torque was not reached. Since the Indego could not be fastened tightly enough to the Biodex seat, sudden changes in the extension torque command (i.e. step-input or changing the direction in the isokinetic tests), caused the pelvis to rotate and lift slightly upwards out of the chair. Therefore, the lower values in extension torques could be attributed to this shift in movement. The Biodex records torque at the dynamometer and cannot account for energy losses throughout the other joints. This also describes the larger error in the isometric tests, since this condition is more sensitive to the sudden torque generated by a step-input. A more vigorous method of securing the Indego's pelvis could reduce these errors.

Similarly, synchronization errors between Indego and Biodex could affect the results. The Biodex-2 software does not output a result file containing a time vector or the sampling rate. Instead, these values must be manually estimated through the Indego data by matching the frames of the first and last instances of maximum joint torque. Using this method, we calculated sampling rates of approximately 60 ± 1 Hz and 125 ± 1 Hz for the isometric and isokinetic trials, respectively. Therefore, interpolating the Biodex data to the 200 Hz sampling rate of the Indego is not precise since the Biodex's sampling rate is inconsistent amongst trials.

With the increasing prevalence of powered orthotics, this virtual muscle controller may be capable of achieving similar stabilizing spring-damper properties of a classical impedance controller. The spring-like properties of muscles can prevent abnormal behaviors arising from unstable torque commands and could be used in concert with other control strategies. For neural activation signals sent to the muscles, a biologically-inspired controller, such as a spinal reflex, is an intuitive next step. After developing such a system, the Indego could be tested with the new control architecture and compared to the pre-existing PD or impedance controller typically used in these devices.

Acknowledgments

9.5 REFERENCES

[1] A. M. Dollar and H. Herr. Lower extremity exoskeletons and active orthoses: Challenges and state-of-the-art. *IEEE Transactions on Robotics*, 24(1):144–158, February 2008.

[2] I. Diaz, J. J. Gil, and E. Sanchez. Lower-limb robotic rehabilitation: Literature review and challenges. *Journal of Robotics*, 2011(Article ID 759764):1–11, September 2011.

[3] K. N. Winfree, P. Stegall, and S. K. Agrawal. Design of a minimally constraining, passively supported gait training exoskeleton: ALEX II. In *2011 IEEE International Conference on Rehabilitation Robotics*, pages 1–6, July 2011.

[4] S. K. Banala, S. K. Agrawal, and J. P. Scholz. Active leg exoskeleton (ALEX) for gait rehabilitation of motor-impaired patients. In *2007 IEEE 10th International Conference on Rehabilitation Robotics*, pages 401–407, June 2007.

[5] J. F. Veneman, R. Kruidhof, E. E. G. Hekman, R. Ekkelenkamp, E. H. F. Van Asseldonk, and H. van der Kooij. Design and evaluation of the LOPES exoskeleton robot for interactive gait rehabilitation. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 15(3):379–386, September 2007.

[6] K. Anam and A. A. Al-Jumaily. Active exoskeleton control systems: State of the art. *Procedia Engineering*, 41:988–994, December 2012. ISSN 1877-7058.

[7] A. Esquenazi, M. Talaty, A. Packel, and M. Saulino. The ReWalk powered exoskeleton to restore ambulatory function to individuals with thoracic-level motor-complete spinal cord injury. *American Journal of Physical Medicine & Rehabilitation*, 91(11): 911–921, November 2012.

[8] A. Esquenazi. New bipedal locomotion options for individuals with thoracic level motor complete spinal cord injury. *Journal of the Spinal Research Foundation*, 8(1): 26–28, 2013.

[9] H. Kawamoto, S. Lee, S. Kanbe, and Y. Sankai. Power assist method for HAL-3 using EMG-based feedback controller. In *IEEE International Conference on Systems, Man and Cybernetics*, volume 2, pages 1648–1653, October 2003.

[10] What's HAL? - The World's First Cyborg Type Robot. `https://www.cyberdyne.jp/english/products/HAL/`. Accessed: 2017-04-09.

[11] H. Sankai and Y. Kawamoto. EMG-based hybrid assistive leg for walking aid using feedforward controller. In *Proceedings of the International Conference on Control, Automation and Systems*, pages 190–193, 2001.

[12] K. Suzuki, G. Mito, H. Kawamoto, Y. Hasegawa, and Y. Sankai. Intention-based walking support for paraplegia patients with Robot Suit HAL. *Advanced Robotics*, 21 (12):1441–1469, April 2007.

[13] Y. Sankai. HAL: Hybrid assistive limb based on cybernics. In *Robotics Research*, pages 25–34, 2010.

[14] S. A Kolakowsky-Hayner, J. Crew, S. Moran, and A. Shah. Safety and feasibility

of using the EksoTM bionic exoskeleton to aid ambulation after spinal cord injury. *Journal of Spine*, 4:003, September 2013.

[15] E. Strickland. Good-bye, Wheelchair, Hello Exoskeleton. *IEEE Spectrum*, 1:30–32, December 2011.

[16] G. J. Androwis and K. J. Nolan. Evaluation of a robotic exoskeleton for gait training in acute stroke. *Archives of Physical Medicine and Rehabilitation*, 97(10):e115–e116, October 2016.

[17] R. J. Farris, H. A Quintero, S. A. Murray, , K. H. Ha, C. Hartigan, and M. Goldfarb. A preliminary assessment of legged mobility provided by a lower limb exoskeleton for persons with paraplegia. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 22(3):482–490, May 2014.

[18] D. A. Winter. *Biomechanics and Motor Control of Human Movement*. John Wiley and Sons, New York, 1990.

[19] F. Sup, A. Bohara, and M. Goldfarb. Design and control of a powered transfemoral prosthesis. *International Journal of Robotics Research*, 27(2):263–273, February 2008.

[20] B. E. Lawson, B. Ruhe, A. Shultz, and M. Goldfarb. A powered prosthetic intervention for bilateral transfemoral amputees. *IEEE Transactions on Biomedical Engineering*, 62(4):1042–1050, April 2015.

[21] K. G. Gerritsen, A. J. van den Bogert, M. Hulliger, and R. F. Nernicke. Intrinsic muscle properties facilitate locomotor control - A computer simulation study. *Motor Control*, 2(3):206–220, July 1998.

[22] A. J. van den Bogert, D. Blana, and D. Heinrich. Implicit methods for efficient mus-

culoskeletal simulation and optimal control. *Procedia IUTAM*, 2:297–316, January 2011.

[23] S. C. Webber and M. M. Porter. Reliability of ankle isometric, isotonic, and isokinetic strength and power testing in older women. *Physical Therapy*, 90(8):1165–1175, August 2010.

[24] A. M. Moseley, J. Crosbie, and R. Adams. Normative data for passive ankle plantarflexion–dorsiflexion flexibility. *Clinical Biomechanics*, 16(6):514–521, July 2001.

[25] H. Karnofel, K. Wilkinson, and G. Lentell. Reliability of isokinetic muscle testing at the ankle. *Journal of Orthopaedic & Sports Physical Therapy*, 11(4):150–154, 1989.

[26] K. G. M. Gerritsen, W. Nachbauer, and A. J. van den Bogert. Computer simulation of landing movement in downhill skiing: Anterior cruciate ligament injuries. *Journal of Biomechanics*, 29(7):845–854, July 1996.

[27] C. M. Siqueira, F. R. M. M. Pelegrini, M. F. Fontana, and J. M. D. Greve. Isokinetic dynamometry of knee flexors and extensors: Comparative study among non-athletes, jumper athletes and runner athletes. *Revista do Hospital das Clínicas*, 57:19–24, February 2002.

[28] L. R. Osternig. Isokinetic dynamometry: Implications for muscle testing and rehabilitation. *Exercise and Sport Sciences Reviews*, 4:45–80, January 1986.

[29] D. D. Cvjetkovic, S. Bijeljac, T. Nozica Radulovic S. Palijaand G. Talic, M. Glogovac Kosanovic, and S. Manojlovic. Isokinetic testing in evaluation rehabilitation outcome after ACL reconstruction. *Medical Archives*, 69:21–23, February 2015.

[30] T. S. Yoon, D. S. Park, S. W. Kang, S. I. Chun, and J. S. Shin. Isometric and isokinetic torque curves at the knee joint. *Yonsei Medical Journal*, 32:33–43, March 1991.

CHAPTER X

CONCLUSION

10.1   Contributions

The main goal of this project was to invent a biologically-inspired controller for powered prosthetic/orthotic (P/O) device that uses a combination of virtual muscles, reflexes, and vestibular feedback (VMR controller) to produce more natural able-bodied gait. We identified the controller parameters using optimizations, where we minimized the error between the joint torque produced by our model and those measured from able-bodied walking data. This dissertation described four key objectives in achieving this goal:

Objective 1: *To design human walking experiments under the effect of mechanical perturbations*

- **Chapter II: An Elaborate Data Set on Human Gait and the Effect of Mechanical Perturbations.** Here, we recorded a rich dataset with 15 test subjects walking at three different walking speeds under the effect of mechanical perturbations by varying the treadmill belt speed. The published dataset, complete with seven and a half hours of walking data, benefits the biomechanics community since gait data is not typically published. This allows any researcher in the field to develop and verify other mathematical models using the data we collected.

Publication: J. K. Moore, S. K. Hnat, and A. J. van den Bogert. An elaborate data set on human gait and the effect of mechanical perturbations. PeerJ 3:e918, 2015.

- **Chapter III: Inertial Compensation for Belt Acceleration in an Instrumented Treadmill.** In our perturbation experiments, inertial artifacts are introduced when the treadmill rollers are stopped and started. Previously, no method existed for compensating these inertial errors. Due to the prevalence of perturbation studies, this simple technique, using only the treadmill's belt acceleration, allows for a more accurate calculation of joint torques through inverse dynamics when subjected to similar mechanical perturbations.

   Publication: S. K. Hnat and A. J. van den Bogert. Inertial Compensation for Belt Acceleration in an Instrumented Treadmill. Journal of Biomechanics, 47(15): 3758 – 3761, 2014.

- **Chapter IV: Compensation for Inertial and Gravity Effects in a Moving Force Platform.** Though we did not perturb test subjects by varying the platform position, we invented another compensation method to reduce inertial errors that occur when moving the treadmill platform. This simple, accelerometer-based technique can be easily implemented by any gait lab. The remaining, uncompensated errors are small enough to allow accurate joint torque calculations through inverse dynamic analysis.

   Intended Publication: S. K. Hnat, B. J. H. van Basten, and A. J. van den Bogert. Compensation for Inertial and Gravity Effects in a Moving Force Platform. Journal of Biomechanics. *Accepted.*

Objective 2: *To develop a virtual muscle model that generates real-time joint torque commands*

Chapter V: Development of a Muscle Model for Real-Time Applications. We investigated a real-time, implicit virtual muscle model [1] using joint angles from

203

our experimental data to drive the muscle dynamics. We reduced simulation time by reformulating the Rosenbrock solver into a closed form solution of linear equations. Our muscle model can be used in real-time on a P/O device to produce muscle joint torques that control the electric motors.

Objective 3: *To design a neuromuscular reflex controller to generate muscle activation signals*

- **Chapter VI: Real-Time Virtual Muscle Reflex Control and Phase-Based Impedance Control in Describing Perturbed Walking.** Reflex controllers have been previously used in simulation [2], but were tuned without information from human walking experiments. We identified the controller parameters using optimizations and able-bodied walking data, which had not been attempted. Through a statistical analysis, we showed the VMR controller for the ankle had less root-mean-square (RMS) error and higher correlation coefficients (R^2) than the standard finite-state impedance controller.

  Intended Publication: S. K. Hnat and A. J. van den Bogert. Real-Time Virtual Muscle Reflex Control and Phase-Based Impedance Control in Describing Perturbed Walking. In preparation. *In preparation.*

- **Chapter VII: Real-Time Virtual Muscle Reflex Control and Phase-Based Impedance Control for a Powered Exoskeleton.** Here, we expanded the VMR controller to the hip and knee, using optimization to define the nature of our controller. Similar to the VMR controller at the ankle joint, this approach had not yet been attempted. Through our statistical analysis, we conclude that the VMR controller's performance is competitive with a standard finite-state impedance controller.

Objective 4: *To implement the VMR system in the Indego*

- **Chapter VIII: A Real-Time Virtual Muscle Reflex Model for a Powered Exoskeleton.** We designed and implemented a VMR controller on the Indego. Previously, no such controller has been used for a powered hip and knee joint of a powered orthosis. With the software built, we provide an opportunity for future studies regarding a biologically-inspired controller and their ability to facilitate or augment walking for users of such a device.

- **Chapter IX: Evaluation of a Virtual Muscle Model using a Powered Exoskeleton.** Though real-time virtual muscles have been used in simulation studies, they had not been evaluated on hardware. In this study, we programmed the Indego to perform standard isometric and isokinetic knee rehabilitation exercises. Because the muscle model exhibited similar force-length and force-velocity relationships, we can conclude the virtual muscle model on the Indego has the same stabilizing capabilities of an impedance controller.

  Intended Publication: S. K. Hnat, Huawei Wang, Raviraj Nataraj, and A. J. van den Bogert. Evaluation of a Virtual Muscle Model using a Powered Exoskeleton *In preparation.*

10.2 Future Work

Since reflex control is relatively new and an ongoing field of research, many considerations extend beyond the scope of this dissertation. We separate the future work in two categories: 1) controller design and 2) evaluating VMR performance on hardware.

10.2.1 Modifications to the Reflex Controller

The reflex equations for the VMR system are complex and could be simplified. Here, we outline a few suggestions for additional controller modifications and considerations:

- **Consistency within the reflex controller equations.** The reflex equations differ significantly amongst muscles. For instance, some muscles use force feedback while

others are controlled by length feedback. Vestibular feedback is only used in the stance equations, while the cerebellum may influence the swing leg in response to a perturbation that requires a substantial step-length adjustment. To simplify these equations, it may be beneficial to prescribe similar controllers to all muscles.

- **Including a *Rectus femoris* muscle.** The original controller of Geyer and Herr does not include a *Rectus femoris*, a muscle which produces hip flexion and knee extension. Adding this muscle could further improve the fit of the hip and knee joint torques.

- **Reducing controller parameters used in the optimizations.** Since the VMR system is highly coupled, we elected to include all controller variables in the optimizations. Some of these parameters may not have as great of an impact on the overall control for each muscles, such as muscle pre-stimulations. A sensitivity analysis could reduce optimization computation time. With fewer parameters, it may be easier for the optimization to converge to the global optimum.

- **Comparing the VMR controller and impedance controller at other single-joints studies.** Both the VMR controller and the impedance controller demonstrate less RMS error and higher R^2 when designed for the ankle joint only (Chapter 6). When optimized for all three muscles, this performance worsened (Chapter 7). It may provide more insight into the capabilities of both controllers when independently optimized for the hip and knee joints.

- **Reducing the possibility of a passive muscle or joint.** One of our main challenges in designing the VMR controller was the tendency of our optimization to suppress one muscle or find a solution where an entire joint was passive. Though the cost-function promoted load-sharing between muscles, passivity is a possible solution because the muscle's slack-lengths are included as optimization parameters. Either

modifying the cost function, or adding a penalty to muscle/joint shut-off, could improve the VMR controller's performance.

- **Designing an additional perturbation study.** For our study, we perturbed test subjects by varying the belt speed by $\pm 10\%$ of the mean speed. In some cases, this type of perturbation may not be sufficient enough to elicit the stretch reflexes in the muscles. Similarly, the perturbation may not be large enough to trigger the vestibular feedback. A different able-bodied walking experiment, perhaps with larger perturbations or platform perturbations, may be able to capture a broader potential of the VMR controller.

- **Incorporation of other neurological control strategies.** Though simpler controllers are ideal, human responses to perturbations cannot be fully captured by a reflex, nor can reflexes entirely describe human gait. Central pattern generators (CPGs) are known to modulate walking in quadrupeds and can be used as a neural input to a muscle model in addition to the reflexes, but their function is not understood in humans [3]. Also, an investigation of vision feedback could be considered, especially how vision influences the user's interaction with a P/O device. The crossed extensor reflex (contralateral reflex feedback), which was not captured by the VMR system, could be represented as a proportional-derivative (PD) controller on the center of pressure (COP) to handle imbalance and reduce fall risk.

10.2.2 Evaluating the VMR Controller on the Indego

Because we were unable to obtain ground reaction force (GRF) signals, a required feedback parameter of the VMR controller, we could not perform an initial assessment of its performance on the Indego. For controller testing, we recommend performing experiments with an able-bodied test subject. A simple first test compares electromyography (EMG) signals between the Indego's passive-mode and the VMR controller. EMG sensors measure the electrical signal produced by a muscle and could be placed on the following muscles: *Lat-*

eral gastrocnemius, *Medial gastrocnemius*, *Vastus lateralis*, *Vastus medialis*, *Gluteus maximus*, and *Semitendinosus*. The test subject could walk in three separate walking conditions while EMG measures their muscle activity and compares the performance of passive-mode, the VMR controller, and a finite-state impedance controller.

In addition, designing other necessary features, such as fall prevention, was beyond the scope of this project. Another consideration is the reactive nature of a reflexes, in which movement must be occurring in order to trigger the controller. We expect that in a powered orthoses, a PD or impedance controller must first begin walking before the VMR system can be activated. Alternatively, leaning the trunk forward could be enough to trigger the trunk PD controller in the hip muscles and vestibular feedback and start walking. These safety considerations and controller uncertainties must be addressed before proposing the VMR system for a commercialized product.

10.3   REFERENCES

[1] A. J. van den Bogert, D. Blana, and D. Heinrich. Implicit methods for efficient musculoskeletal simulation and optimal control. *Procedia IUTAM*, 2:297–316, January 2011.

[2] H. Geyer and H. Herr. A muscle-reflex model that encodes principles of legged mechanics produces human walking dynamics and muscle activities. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 18(3):263–273, June 2010.

[3] E. R. Kandel, J. H. Schwartz, T. M Jessel, and S. A. Siegelbaum. *Principles of Neural Science*. McGraw-Hill, New York, 2000.
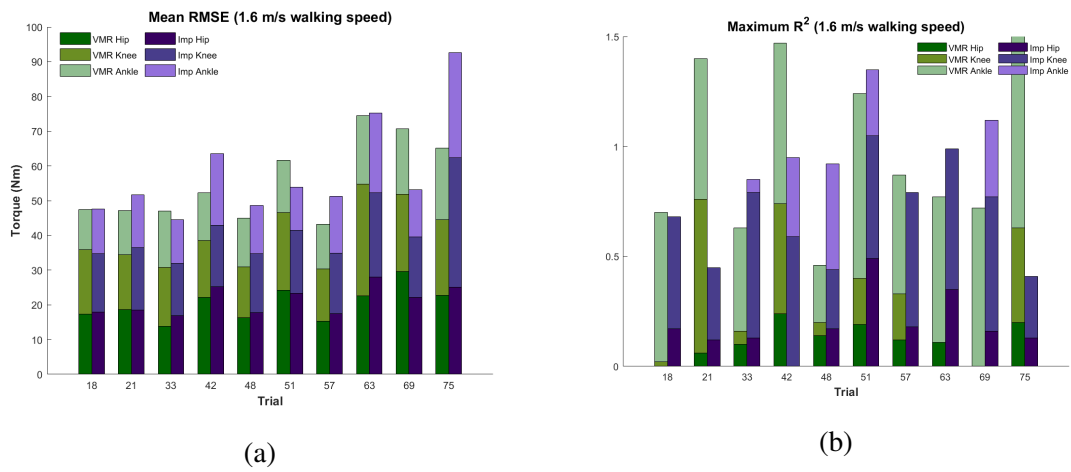
(a)

(b)
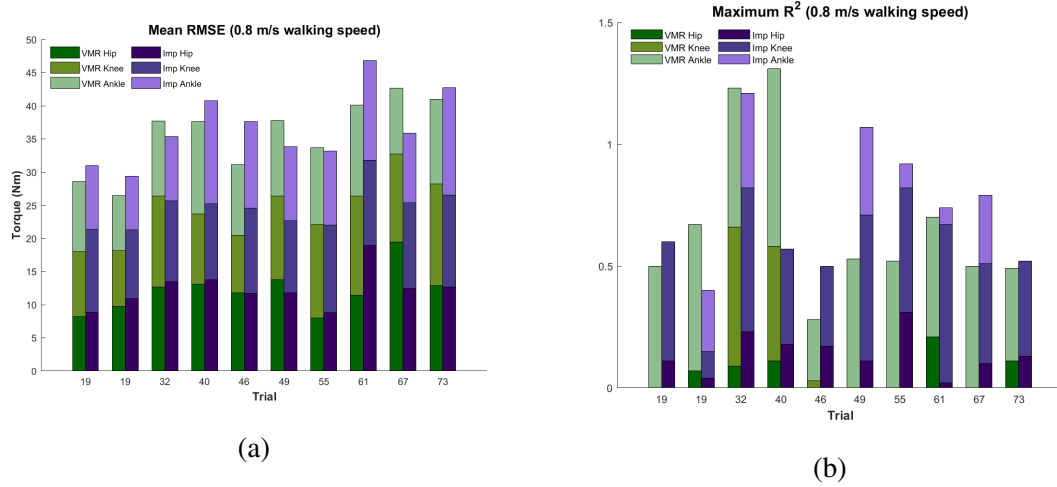
Figure 49: Mean RMS error (left) and maximum correlation coefficient, R^2 (right), as functions of the gait cycle, in which each gait cycle was split into 100 different data points. These results are obtained from 10 trials of the normal walking speed (0.8 m/s). The values for the VMR controller (green) and impedance controller (purple) are compared, with the contribution from each joint indicated. Better controllers have high correlation coefficients R^2 and lower root-mean-square (RMS) errors.

(a)

(b)
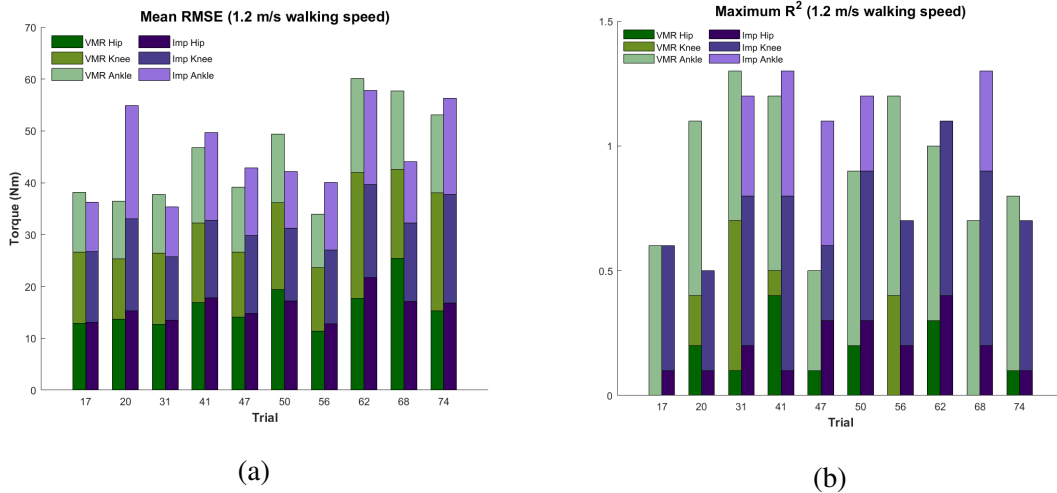
Figure 50: Mean RMS error (left) and maximum correlation coefficient, R^2 (right), as functions of the gait cycle, in which each gait cycle was split into 100 different data points. These results are obtained from 10 trials of the normal walking speed (1.2 m/s). The values for the VMR controller (green) and impedance controller (purple) are compared, with the contribution from each joint indicated. Better controllers have high correlation coefficients R^2 and lower root-mean-square (RMS) errors.

Figure 51: Mean RMS error (left) and maximum correlation coefficient, R^2 (right), as functions of the gait cycle, in which each gait cycle was split into 100 different data points. These results are obtained from 10 trials of the normal walking speed (1.6 m/s). The values for the VMR controller (green) and impedance controller (purple) are compared, with the contribution from each joint indicated. Better controllers have high correlation coefficients R^2 and lower root-mean-square (RMS) errors.

An Elaborate Data Set on Human Gait and the Effect of Mechanical Perturbations

Inertial Compensation for Belt Acceleration in an Instrumented Treadmill