DESIGN AND CONTROL OF AN ACTIVE ANKLE-FOOT ORTHOSIS

by

Oğuzhan Kırtaş

B.S., Mechatronics Engineering, Sabancı University, 2014

Submitted to the Institute for Graduate Studies in
Science and Engineering in partial fulfillment of
the requirements for the degree of
Master of Science

Graduate Program in Mechanical Engineering
Boğaziçi University
2018

ACKNOWLEDGEMENTS

It would not be possible to complete my thesis without the help of my advisor, Professor Samur. I sincerely thank him for his valuable guidance and support. I would also like to thank Prof. Can A. Yücesoy and Prof. Volkan Patoğlu who accepted to be jury members of my master's thesis despite their overwhelming busy schedule.

I would like to thank my friends in Haptics and Robotics lab, including Gholamreza, Taylan, Onur, Mohammad, Alican, Bora, Şeref, Timur, Cem, Efe and Mehdi. I would also like to thank Yağız and Prof. Halil İ. Baştürk for their great contribution to the control system design.

Finally, my special thanks go to my family, my lovely mom Fadime, my dad Necmettin and my beloved brother Emrecan for their everlasting support and love.

This work was supported by the Boğaziçi University Research Fund under project number 9781, and in part by the People Programme (Marie Curie Actions) of the European Union Seventh Framework Programme FP7/2007-2013/ under REA grant agreement no [631830].

ABSTRACT

DESIGN AND CONTROL OF AN ACTIVE ANKLE-FOOT ORTHOSIS

Active ankle-foot orthoses are used to assist patients suffering from stroke, multiple sclerosis, cerebral palsy etc. through providing an external force supply to track normal gait cycles. In this study, design of an active ankle-foot orthosis prototype, and an adaptive backstepping control algorithm for it to track desired gait trajectories while reducing the effects of unknown disturbances are proposed. A prototype of the orthosis, which is composed of a series elastic actuator, a lever mechanism and an orthotic shoe, is developed. The prototype is mathematically modeled as a two-degree-of-freedom mass-spring system and the unknown disturbances are modeled. The backstepping control law is designed for a force input supplied to the system. Additionally, the designed algorithm is implemented in a real-time operating system to control the developed ankle-foot orthosis prototype. Simulations are performed to compare the proposed control architecture and a PID controller in trajectory tracking of the prototype. Finally, experimental tests including user tests are conducted to evaluate the prototype in use and to compare the performances of the control algorithms. Results show that the overall system satisfies the design requirements, and proposed control architecture is about two times better than the PID controller.

ÖZET

BİR AKTİF AYAK-BİLEK ORTEZİ TASARIMI VE KONTROLÜ

Aktif ayak-bilek ortezleri; felç, multiple skleroz ve serebral palsi gibi rahatsızlıklara sahip hastaların dışarıdan bir kuvvet sağlanması suretiyle sağlıklı bir yürüyüş döngüsü elde etmeleri için kullanılır. Bu çalışmada, bir aktif ayak-bilek ortezi tasarımını ve istenmeyen bozucu etkileri azaltarak prototipin arzu edilen yürüyüş yörüngesini takip etmesini sağlayan bir adaptif geri adımlamalı kontrol sistemini öneriyoruz. Seri elastik eyleyici, kaldıraç mekanizması ve ortotik ayakkabıdan oluşan bir ortez prototipi geliştirildi. Prototipin iki serbestlik dereceli bir kütle-yay sistemi olarak matematiksel modellemesi yapıldı ve istenmeyen bozucu etkiler modellendi. Dışarıdan sağlanan kuvvet için bir adaptif geri adımlamalı kontrol algoritması tasarlandı. Ayrıca, tasarlanan algoritma geliştirilen ayak-bilek ortezinin kontrolü için gerçek zamanlı bir kontrol platformu kullanılarak gerçeklendi. Önerilen kontrol sistemi ile bir PID kontrolcünün yörünge takip performanslarının karşılaştırılması için simülasyonlar yapıldı. Son olarak, kullanılan prototipi değerlendirmek ve kontrol algoritmalarını karşılaştırmak için kullanıcı testlerini de içeren deneysel testler yapıldı. Sonuçlar sistemin tasarım gereksinimlerini karşıladığını ve önerilen kontrol sisteminin PID kontrolcüden yaklaşık iki kat daha iyi olduğunu göstermiştir.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF SYMBOLS

A	System matrix
B	Control vector
c	Constant disturbance
c_t	Motor torque constant
D	Outer diameter of the ball screw
$d(t)$	Unknown disturbance acting on foot
e_1	Error term between current and reference heel positions
$\dot{e_1}$	Dynamics of the heel position error term
e_3	Error term of the virtual controller
F_a	Axial load
$F(t)$	Force supplied to the system
g	Gravitational acceleration
G	Hurwitz matrix with different eigenvalues
g_i	Amplitude of the disturbance
i	Motor current
k	Spring constant
kz_t	Force acts on the mass connected to the heel
$K_{I_{PWM}}$	Integral gain
$K_{P_{PWM}}$	Proportional gain
l	A vector that makes pair (G,l) controllable
L	Moment arm
M	Unique solution of the Slyvester equation
M_a	Moment acting on the ankle joint
m_1	Mass of the transmission system
m_2	Mass connected to the heel
N	Motor speed
p	Pitch of the ball screw
P	Force generated by the artificial muscle

q	Number of distinct frequencies
S	Matrix depends on unknown frequencies
T	Motor driving torque
$U_{PWM}(t)$	Voltage output in duty cycle
V_1	Lyapunov function
x	Error system
$\dot{x}$	Dynamics of the error system
x_s	Spring deflection
z_h	Position of the heel mass
$z_{h_{ref}}$	Reference heel position
$\dot{z_h}$	Velocity of the heel mass
$\ddot{z_h}$	Acceleration of the heel mass
z_t	Position of the transmission mass
$\dot{z_t}$	Velocity of the transmission mass
$\ddot{z_t}$	Acceleration of the transmission mass

$\delta(t)$	Estimation error
$\dot{\delta(t)}$	Dynamics of the estimation error
$\epsilon(\tau)$	Position error
η	Ball screw efficiency
θ^T	Unknown constant vector
$\hat{\theta}^T$	Estimation of unknown vector
$\tilde{\theta}^T$	Estimation error of unknown vector
$\dot{\hat{\theta}}$	Update law
$\tau(t)$	Unknown time dependent vector
ϕ_i	Phase of the disturbance
ω_i	Frequency of the disturbance

LIST OF ACRONYMS/ABBREVIATIONS

aAFO	Active Ankle-Foot Orthosis
AFO	Ankle-Foot Orthosis
AISI	American Iron And Steel Institute
CAD	Computer-Aided Design
CNC	Computer Numerical Control
CPET	Cardiopulmonary Exercise Testing
DC	Direct Current
FBD	Free Body Diagram
FF	Forefoot
FPGA	Field Programmable Gate Array
FSR	Force Resistive Sensor
GRF	Ground Reaction Force
HF	Hindfoot
HX	Hallux
NI	National Instruments
OS	Operating System
PD	Proportional Derivative
PE	Persistent Excitation
PI	Proportional Integral
PID	Proportional Integral Derivative
PLA	Polylactic Acid
PWM	Pulse Width Modulation
SEA	Series Elastic Actuator
TB	Tibia

1. INTRODUCTION

1.1. Aims of the Study

Exoskeletons act in series or in parallel to a human limb to assist in motion economy by augmenting joint torque and work done [1]. An ankle-foot orthosis (AFO) is a type of exoskeleton that surrounds the ankle and foot to assist ankle motion. AFOs could be divided into two groups: passive and active.

Passive AFOs limit undesired ankle movements or augment endurance by storing energy using passive elements. Active AFOs (aAFO) include actuators, power sources, and control systems. They provide net power to the ankle unlike passive AFOs. Current active AFOs do not provide ankle degree-of-freedom in three anatomical planes. Additionally, most of the active AFO designs have a sensor complexity to adapt gait, and that causes high costs.

Control system design of an aAFO has been approached in several ways including variable impedance, torque, and trajectory tracking control systems. In trajectory tracking control systems, pre-determined gait patterns are generated in order to keep the ankle angle as desired [2]. Undesired responses to unknown interaction forces could be considered as the most significant drawback of trajectory tracking control systems [3].

The aims of this study are to develop an active ankle-foot orthosis that can provide ankle degree-of-freedom in three anatomical planes, has a simple sensor system to adapt gait, and can track desired gait trajectories while reducing the effects of unknown disturbances. Potential users of the ankle foot orthosis could be drop foot, cerebral palsy, or multiple sclerosis patients. Design requirements for the overall system are low response time, low position tracking error, and high torque output.

1.2. Outline

This study is structured as follows. Chapter 2 describes the literature review for medical background, ankle-foot orthosis designs, and control strategies. Chapter 3 explains the design of the active-ankle foot orthosis and development of the prototype. In Chapter 4 problem definition, disturbance representation, and design of the adaptive backstepping and PID controllers are provided. Chapter 5 explains the details of the implementation of the control algorithm. In Chapter 6, evaluation methods for simulations and experiments are explained. Simulation and experimental results, and their discussion are presented in Chapters 7 and 8, respectively. Conclusion is given in Chapter 9.

2. LITERATURE REVIEW

2.1. Medical Background

The foot and ankle form the terminal portion of the lower limb. They are responsible for the propulsion, balance, and support of the body weight during many activities such as walking, running, jumping, and standing. There are two motions of the ankle in the sagittal plane: dorsiflexion and plantar flexion. Dorsiflexion is the flexion movement of the ankle that decreases the angle between foot and leg; while plantar flexion increases that angle. Some muscles are connected to the bones via corresponding tendons and provide ankle movements. The muscles, which carry the most of the force, are tibialis anterior (provides dorsiflexion), gastrocnemius and soleus (calf muscles which provide plantar flexion). Since the movements of dorsiflexion and plantar flexion are antagonist movements which are respect to each other; the muscles which provide these movements are antagonist pairs as well. The muscle contraction is controlled by central nervous system.

The gait cycle is divided into two phases: stance phase and swing phase. Stance phase (support or contact phase) starts with the initial contact (heel strike to the ground) and lasts until the toe off. Swing phase starts with the toe off and ends with the next heel strike. At heel strike, the ankle plantar flexes to 8 degrees in order to lower the foot to the ground. After foot flat, ankle starts to dorsiflex to 13 degrees until the heel is off the ground. During push off, ankle plantar flexes to 20 degrees until the beginning of the swing phase where the toe is off the ground as well. Finally, ankle plantar flexes to the neutral position to prepare the foot for the next heel strike. Thus the gait cycle is completed (Figure 2.1) [4]. Typical ankle angular velocity, moment and power have been published in [5]. At heel strike the ground reaction force passes close to the ankle joint and produces a small plantar flexion moment. After heel strike the ground reaction force moves in front of the ankle and produces a larger dorsiflexion moment (Figure 2.2).

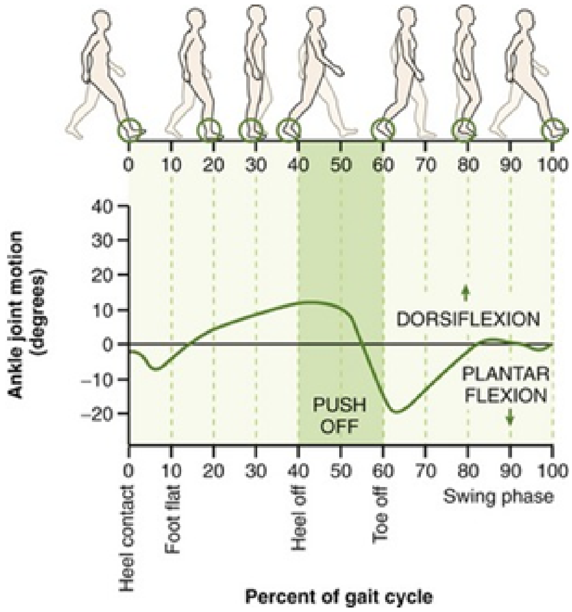


Figure 2.1. Range of motion of ankle joint during normal gait cycle. Reprinted from [4].

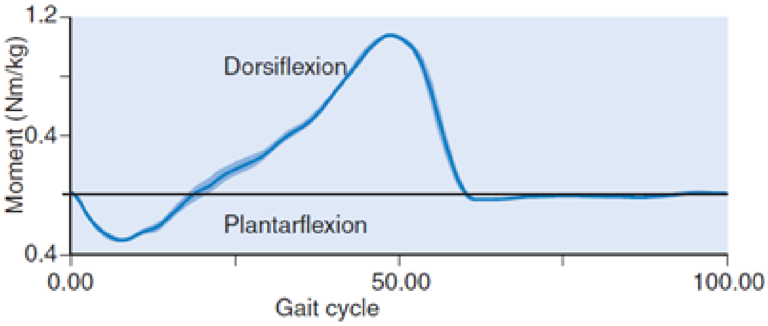


Figure 2.2. Typical ankle moments during normal gait cycle. Reprinted from [5].

Not everyone has a normal gait. The ability to walk can be impaired by neurological and muscular pathologies such as trauma, incomplete signal cord injuries, stroke, multiple sclerosis, muscular dystrophies, and cerebral palsy. Those pathologies affect joint mobility and muscle activity [6]. Damage to the nervous system often after a stroke or old age induced biological failure can cause muscle atrophy and dysfunction leading to trips and falls. An example of such pathology is “drop foot”, where the anterior muscles, which provide dorsiflexion, become weaker while posterior muscles become stiffer [7]. Thus the foot drops in swing phases and toe strikes instead of the heel. For rehabilitation of the gait pathologies, exoskeletons are used. They act in series or in parallel to a human limb, and assist motion by augmenting joint torque and work done [1].

2.2. Ankle-Foot Orthosis Designs

An ankle-foot orthosis (AFO) is a type of exoskeleton that surrounds the ankle and foot. AFOs are externally applied and intended to control position and motion of the ankle, compensate for weakness, or correct deformities. AFOs could be divided into two groups, which are passive and active.

2.2.1. Passive Ankle-Foot Orthosis

Passive AFOs have no powered actuator or control systems. They limit undesired ankle movements or augment endurance by storing energy using passive elements [1]. They can have articulated or non-articulated joints. Passive AFO designs comprise of materials such as metal and leather, composite, and thermoplastics. Traditional metal and leather systems have articulated hinge joints to limit motion [6]. Those AFOs assist the ankle joint by preventing undesired foot movements with direct physical resistance. Additionally, non-articulated thermoplastic passive AFOs integrate energy storage by means of geometry and properties of the material. The design in [8] includes long leaf springs parallel to the legs and augments the running abilities of human. In [9], a tendon like passive assistive technology based on geometrically arranged long elastic cords to provide mechanical advantage to the user was presented. Similarly in [10], developed passive leg orthosis reduces the effects of gravity during walking and improves the locomotion. However, adaptability of these passive devices to different subjects is problematic [11]. In order to increase the adaptability, active AFOs (aAFO) were proposed.

2.2.2. Active Ankle-Foot Orthosis

Active AFOs include actuators, power sources, and control systems. They provide net power to the ankle unlike passive AFOs. To allow ankle motion in anatomical planes, several designs were developed in literature. Hinge joints that allow ankle motion in sagittal plane (dorsiflexion and plantar flexion) were widely used [12–14]. Distinctly in [15], inversion/eversion degree-of-freedom in coronal plane was also provided with the design. Different types of actuators including DC motors [15], series elastic actuators (SEA) [13, 14], and pneumatic systems [12, 16, 17] were used in aAFO designs.

Series elastic actuators include elastic materials to adjust the natural dynamics of the system in order to store energy. They also include powered actuators to provide power to the system. Series elasticity provides safe human-robot interaction by filtering shock loads and reduction of the energy consumption by storing energy [18]. In [13] an active AFO that includes SEA has been developed to assist drop foot which includes SEA. It assists walking with dorsiflexion during the swing phase. Similarly in [14], a series elastic actuator has been designed for the control of an active AFO (Figure 2.3). Additionally, a variety of foot switches and force sensors were used to detect heel strike and determine walking phases [13, 14]. In the literature, there is no aAFO design that provides ankle joint degrees-of-freedom in three anatomical planes. Moreover, most of the designs have a variety of sensors to adapt gait, and thus have high costs due to these sensors.

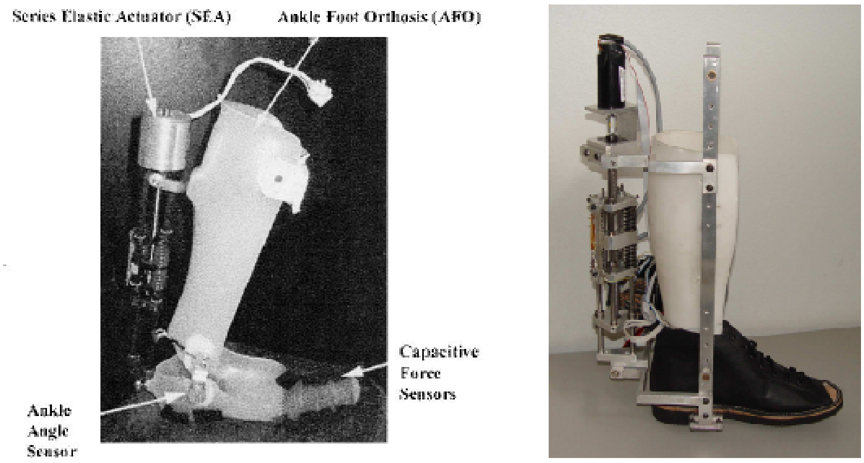


Figure 2.3. Active AFO designs. Reprinted from [13] (left), and [14] (right).

2.3. Control Methods

The control design problem of an aAFO has been approached in various ways including variable impedance, torque, and trajectory tracking control systems. In this section, control strategies for the ankle-foot orthoses are reviewed.

2.3.1. Variable Impedance Control

Variable impedance control systems detect subphases of a gait cycle and implement different low-level controllers for each phase. Finite state machines are used to adjust mechanical impedance of the AFO for each subphases. By controlling the impedance of the system, undesired effects of disturbances could be eliminated. Studies show that use of variable impedance control achieves a remarkable success in tracking the normal gait cycles [2, 13, 19]. However, this method requires triggers to transite between the finite states during a gait cycle. This means that, additional force or position sensors are required depending on transition formulations.

2.3.2. Torque Control

As another popular method, torque control is used to apply the required torque to the ankle joint. In [20], several techniques were proposed to determine ankle joint torque for controlling the AFO to compansate forces acting on the foot. Moreover, in [21], a torque tracking method for a powered ankle-foot prosthesis based on a neuromuscular model was explained. However, torque control methods require joint angle information for state transitions, and it may cause serious problems such as exceeding motion limits of the ankle joint [3].

2.3.3. Trajectory Tracking Control

Trajectory tracking designs rely on the periodicity of a gait. Pre-determined gait patterns are generated in order to keep the ankle angle at desired degrees [2]. In [22], an active AFO with a proportional-derivative (PD) controller, that use a reference gait pattern as a function of stride time, was proposed. Moreover, several trajectory tracking algorithms which relate the ankle degree, stride length, and gait cycle were proposed in [23]. Additionally, a Lyapunov design was proposed to track set-point trajectories [15]. However, trajectory tracking control strategies are practical only when the reference gait cycle and interaction forces are well-defined. Therefore, the most significant drawback of this method can be considered as its undesired response to unknown interaction forces. Disturbances caused by ground reaction forces (GRF) could lead to instability and forces could reach high values [3]. Moreover, ankle motion sticks to a desired trajectory, and active interaction of the patient is not possible with trajectory tracking control methods [24].

3. ACTIVE ANKLE-FOOT ORTHOSIS PROTOTYPE

3.1. Design of the Active Ankle-Foot Orthosis

To determine design requirements for the active ankle-foot orthosis prototype, a free body diagram (FBD) was developed (Figure 3.1). The multi-segment foot model in [25] was used. The model comprises of four segments which are hindfoot (HF), forefoot (FF), hallux (HX), and tibia (TB). These segments were labeled on the FBD. The forces generated by human muscles were assumed to be zero, so all the forces are generated by the artificial muscles. Weights of the segments and other materials such as motor and spring were neglected as well. Moments acting on the ankle joint is denoted by M_a. P is the force generated by the artificial muscle (M) to be designed. The spring that provide series elasticity is represented by S. L is the moment arm for ankle movements and assumed as 0.08 m.

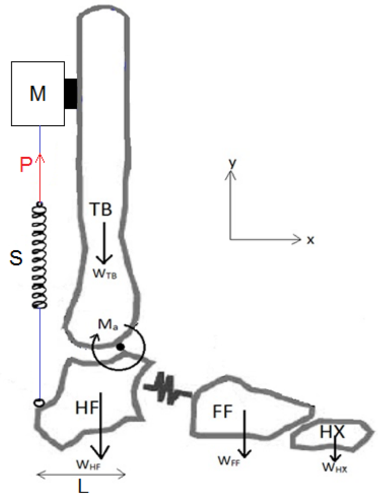


Figure 3.1. Free body diagram of the active ankle-foot orthosis.

According to Figure 2.2, the maximum moment on the ankle joint is about 1.1 Nm/kg for dorsiflexion, and it is 0.3 Nm/kg for plantar flexion. Since the measurements are based on an 80 kg subject, the maximum moment acts on the ankle is 88 Nm. Considering the moment arm (L), the maximum force generated by the artificial muscle is 1100 N.

A DC motor was selected as the actuator of the system because of the ease of control. Additionally, to convert rotational motion to linear motion, a ball screw was selected due to its high efficiency characteristics. The selections were made with respect to the ball screw driving torque equation,

$$T = \frac{F_a p}{2\pi\eta} \tag{3.1}$$

where T is the driving motor torque (Nm), F_a is the axial load (N), p is the screw lead (m), and η is the ball screw efficiency. A ball screw with 2 mm lead and 8 mm outer diameter was selected (CPC Type SFK 08x02mm). The efficiency of the selected ball screw is 90 %. The axial load is 1100 N provided by the artificial muscle. Using Equation 3.1, required driving torque was calculated as 0.39 Nm. A Maxon DC brushed motor (RE 40 150 Watt, 24 V, 0.48 kg) was selected considering the calculated driving torque. The datasheets of the selected DC motor and ball screw can be seen in Appendix A. The DC motor has a nominal torque of 0.17 Nm, a stall torque of 2.28 Nm, and a nominal speed of 6930 rpm. The selected actuator satisfies following inequality,

$$DN \leq 70000 \tag{3.2}$$

where D is the outer diameter of the ball screw (mm), and N is the speed of the actuator (rpm).

A linear compression spring was selected to provide series elasticity to the actuator with respect to the Hooke's law,

$$F_a = kx_s \tag{3.3}$$

where F_a is the applied axial load (N), k is the spring constant (N/m), and x_s is the spring deflection (m). The free length of the compression spring was decided as 0.084 m. Desired maximum spring deflection is 50 % when maximum force is applied. Using Equation 3.3, k was calculated as 26000 N/m. Considering the calculated spring constant, an AISI Type 302 stainless steel linear compression spring with a 3 mm wire diameter and 16 mm outer diameter was selected.

After selection of the materials, computer-aided design (CAD) models of the series elastic actuator and the active ankle-foot orthosis were developed using SolidWorks software (Figure 3.2). The aAFO model consists of a series elastic actuator (SEA), an orthotic shoe, and a lever mechanism that connects the SEA and the shoe. SEA is comprised of a brushed DC motor, a ball screw, a coupling, a linear compression spring, two buckle preventing pipes, two linear bearings, two shafts, and a ball-and-socket joint. The compliant actuator is powered by the selected Maxon DC brushed motor. To convert rotational motion to linear motion, the selected ball screw was used. To connect the DC motor to the ball screw, an aluminum coupling was used. The selected stainless steel linear compression spring was used to provide series elasticity. In order to prevent buckling of the spring, two stainless steel pipes were located at the outside of the spring. Two 8-mm-diameter induction hard chromed shafts, and two linear bearings were used to provide pure linear motion of the ball screw's nut with no rotation. Four shaft holders were selected in order to fasten the shafts. A ball-and-socket joint was used for the connection of the lever mechanism and the SEA. It allows powered ankle motion in the sagittal plane and passive motion of the ankle in the horizontal and coronal planes. A shoe was selected for the design. It should be noted that, the designed SEA can be mounted to any shoe with different sizes. Two brackets were used to attach the aAFO to the leg. Some parts were designed to connect the lever mechanism and lower bracket. Four radial ball bearings were used to allow plantar flexion and dorsiflexion movements in the sagittal plane. Two extra radial ball bearings were placed into the aluminum alloy parts to prevent restriction of inversion and eversion movements of the ankle.

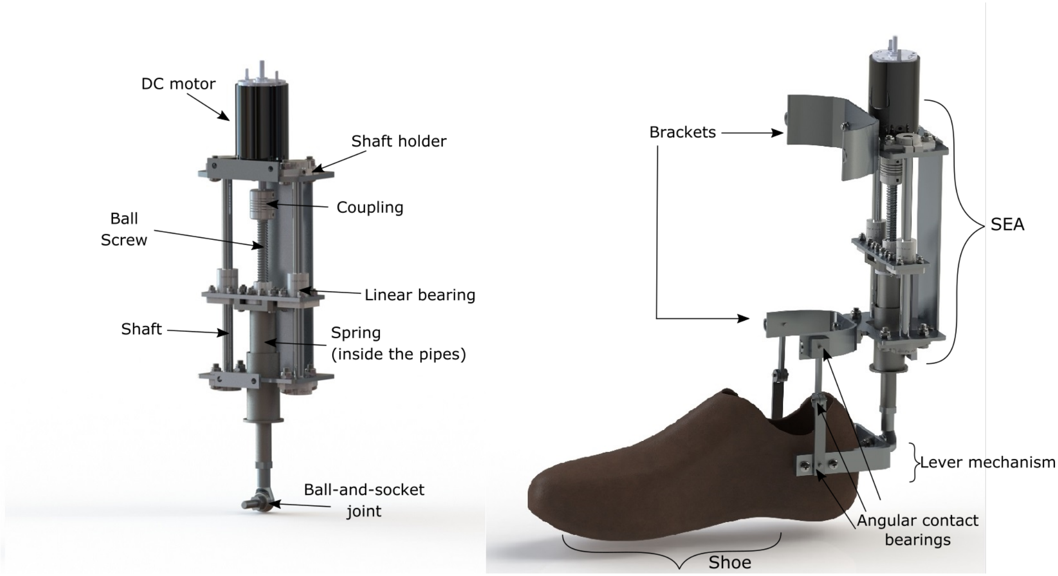


Figure 3.2. CAD models of the series elastic actuator (left) and the active ankle-foot orthosis (right). The system consists of a series elastic actuator, an orthotic shoe, and a lever mechanism.

3.2. Prototype Development

The selected DC motor, ball screw, coupling, shafts, linear bearings, shaft holders, ball-and-socket joint, radial ball bearings, buckle preventing pipes, shoe, nuts and bolts were purchased. The list of the purchased parts is provided in Table 3.1. A linear compression spring was manufactured with respect to the selected properties. The lever mechanism, brackets, and parts which form the structure of the prototype were manufactured using CNC machining. Manufactured parts were indicated by numbers in an exploded view of the CAD model of the ankle-foot orthosis (Figure 3.3). The list of the manufactured parts is given in Table 3.2. The medium carbon steel radial bearing shafts were cut to different lengths of 12 mm, 13 mm, and 17 mm. The material of the parts which were welded to the spring is 304 Type stainless steel. Remainder of the parts were manufactured using 7075 aluminum alloy. Finally, all of the materials were assembled. Hook-and-loop fasteners were attached to the brackets. The total weight of the developed aAFO prototype is 1.8 kg.

Table 3.1. Purchased parts of the ankle-foot orthosis

Part	Model/Material	Quantity
DC motor	Maxon RE 40 Brushed DC motor	1
Ball screw	CPC Type SFK 08x02mm	1
Coupling	SJCB30C-RD	1
Induction hard chromed shaft	SFC 8	2
Linear bearing	LMEK 8 UU	2
Shaft holder	SHF 08	4
Ball-and-socket joint	BL6D	1
Radial ball bearings	623	4
	SMF 63 2Z	2
Shoe	Dockers	1
Buckle preventing pipes	AISI 304 stainless steel	2
Bolts	ISO 7045 M2.5 5	6
	ISO 7045 M3 10	8
	ISO 7045 M3 12	4
	ISO 7045 M3 16	4
	ISO 7045 M4 10	6
	ISO 7045 M4 12	4
	ISO 7045 M4 16	4
	ISO 7045 M5 16	8
Nuts	DIN EN ISO 10511 M3	16
	DIN EN ISO 10511 M4	14
	DIN EN ISO 10511 M5	8
Incremental encoder	Encoder MR, Type L, 1024 CT	1
Linear potentiometer	Vishay LMF2D103W, 50 mm	1
Force sensitive resistor	Interlink Electronics, FSR 402	1

Table 3.2. Manufactured parts of the ankle-foot orthosis

Part #	Part Name	Manufacturing Method	Material	Quantity
1	Calf bracket	CNC milling & Turning	Al 7075	1
2	Ankle bracket	CNC milling & Turning	Al 7075	1
3	Motor connection part	CNC milling & Turning	Al 7075	1
4	Linear bearing connection part	CNC milling & Turning	Al 7075	1
5	Ankle bracket connection part	CNC milling & Turning	Al 7075	1
6	Upper connection part of the spring	CNC milling & Turning	304 SS	1
7	Connection part for 3 and 5	Bending & Turning	Al 7075	1
8	Lower connection part of the spring	CNC milling & Turning	304 SS	1
9	Lever mechanism	CNC milling & Turning	Al 7075	1
10	Upper shoe connection part	CNC milling & Turning	Al 7075	2
11	Lower shoe connection part	CNC milling & Turning	Al 7075	2
12	Connection part for 11 and shoe	Turning	Al 7075	2
13	Compression spring	Coiling & Hardening	302 SS	1
14	Housing for the linear potentiometer	3D Printing	PLA	1

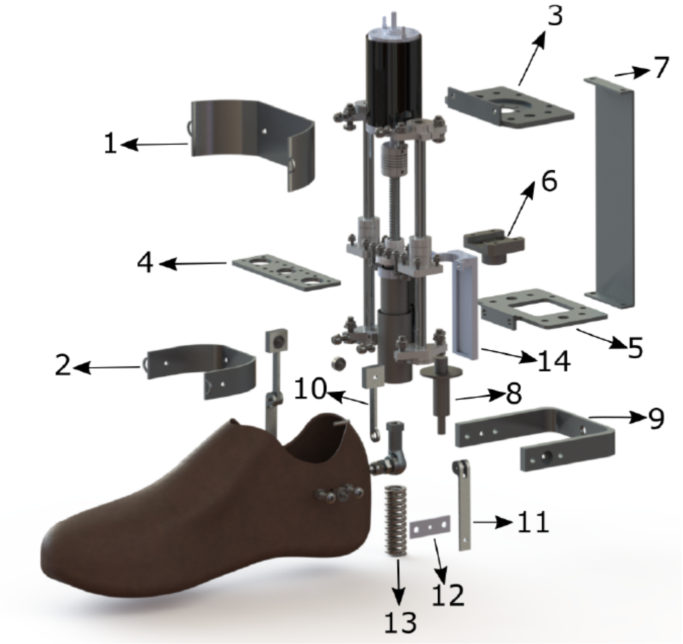


Figure 3.3. Exploded view of the ankle-foot orthosis model with the indicated part numbers.

The developed prototype can be seen in Figure 3.4. The system includes three sensors. An incremental encoder with 1024 count per turn was used to measure the DC motor's position. A linear potentiometer with 50 mm travel length was used for the spring deflection measurements (refer linear potentiometer specification sheet in Appendix A), and it was assembled to the stainless steel pipe with a 3D printed polylactic acid (PLA) housing. By encoder readings and displacement measurements, we can easily calculate the heel position displacement and convert it to the ankle joint degree without needing an extra joint degree sensor. A force sensitive resistor (FSR) was placed under the arch support of the shoe. It was used for detecting the beginning of a gait cycle and measuring the elapsed time between two heel strikes (gait cycle period). So, we can synchronize the system with the gait cycle and provide adaptation to different stride frequencies with the use of a single force sensor.

In the developed prototype, the ankle range of motion is between -25° and 15° for sagittal plane, $\pm 10^\circ$ for transverse plane, and $\pm 2^\circ$ for coronal plane.

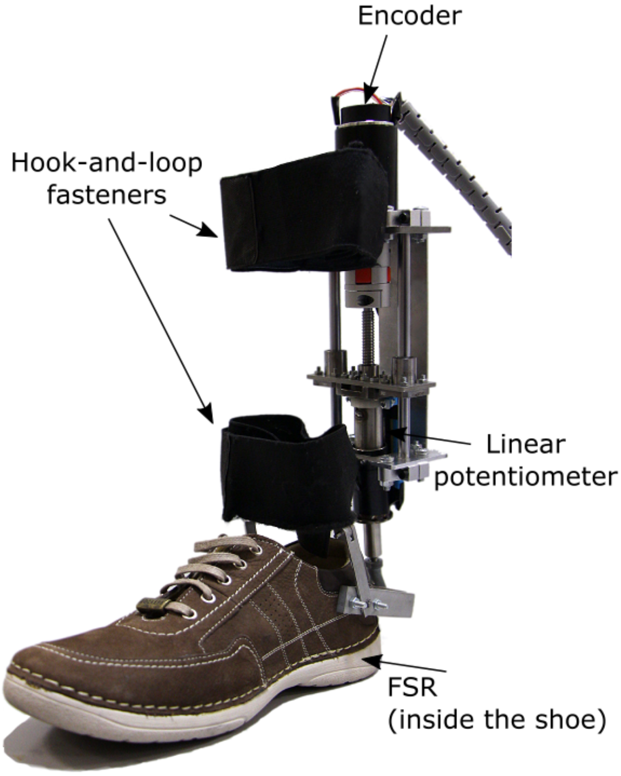


Figure 3.4. A photo of the developed ankle-foot orthosis prototype.

4. CONTROL SYSTEM

The main aim of an active ankle foot orthosis is assisting users to track normal gait behaviour despite the existence of unknown disturbances. This aim can be satisfied by approaching the problem as a trajectory tracking problem with the use of heel position trajectories during normal gait cycles. To this end, we designed a control algorithm for the force input to maintain the stability of the equilibrium and to satisfy the convergence of vertical heel position to a desired reference signal. The details of the control algorithm can be found in [26].

4.1. Problem Definition

The developed ankle foot orthosis driven by a compliant actuator can be mathematically modeled as a two-degree-of-freedom mass-spring system as shown in Figure 4.1. In the model, transmission system consists of a coupling and a coupling spindle through which the rotational motion of the actuator is transformed into a linear motion. Additionally, lower parts of the system were considered as a lumped mass which is connected to the transmission system and the ankle joint through a spring and a lever arm, respectively.

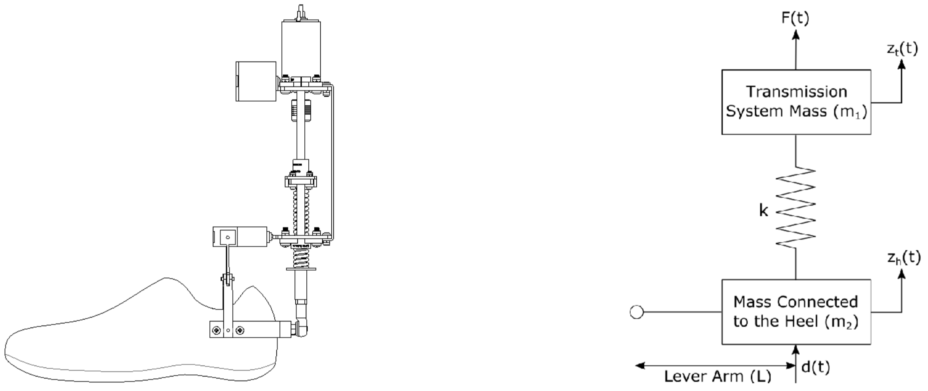


Figure 4.1. Side view of the aAFO CAD model (left) and its mass-spring system representation (right).

The system of equations for this model is given by,

$$m_1\ddot{z}_t(t) = -m_1 g - k(z_t(t) - z_h(t)) + F(t) \tag{4.1}$$

$$m_2\ddot{z}_h(t) = -m_2 g + k(z_t(t) - z_h(t)) + d(t) \tag{4.2}$$

where m_1 and m_2 represent the mass of the transmission system and the lumped mass which is connected to the heel, respectively. Spring constant is represented by k, $F(t)$ stands for the force supplied to the system by an actuator through a rigid transmission and g is the gravitational acceleration. Vertical displacements of m_1 and m_2 are given by $z_t(t)$ and $z_h(t)$, respectively. Finally, unknown disturbance is provided by $d(t)$.

There are two main external forces denoted by disturbances that may diverge an AFO from a desired trajectory of a normal gait cycle: first, ground reaction force (GRF), a variable force applied by the ground to the foot during the stance phase, and second, weight of the foot during the swing phase. The ankle moment value in Figure 2.2 can be given as vertical force values acting on the ankle by multiplying it with the weight of the subject and dividing it with the lever arm L. Moreover, during the swing phase, only the weight of the foot acts as disturbance. As a result, total disturbance acting on the system, in Newtons, during a gait cycle, is given by Figure 4.2. Note that there is a large negative disturbance during stance phase since the heel moves downwards throughout this phase and the disturbance is defined as heading upwards in the mathematical model.

Using the idea that any bounded signal can be composed of different numbers of sinusoidals and considering representation provided in Figure 4.2, the disturbance was assumed to be a finite sum of sinusoidal signals and constants. All assumptions are listed as follows.

Assumption 1: The measurements of vertical displacements, $z_t(t)$, $z_h(t)$, and vertical velocities $\dot{z}_t(t)$, $\dot{z}_h(t)$ are available for the control design.

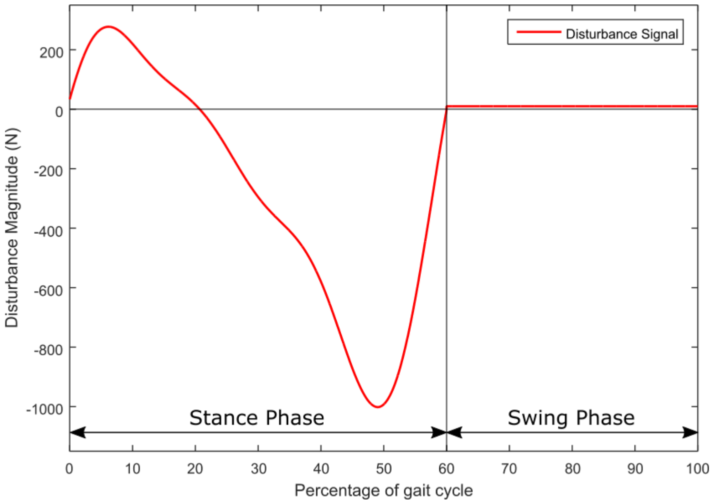


Figure 4.2. Representation of disturbances acting on the system during a gait cycle. Ground reaction force acts on the system during the stance phase and the weight of the foot causes the disturbance during swing phase.

Assumption 2: The disturbance is represented as $d(t) = c + \sum_{i=1}^{q} g_i \sin(\omega_i t + \phi_i)$ where constant disturbance c, the amplitude, g_i, the frequency, ω_i, and the phase ϕ_i of the sinusoidal disturbance are unknown. The number of maximum distinct frequencies q is known.

Assumption 3: The parameters of the AFO, m_1, m_2 and k are known.

There are two important points that should be clarified regarding the assumptions: (i) In Assumption 1, vertical velocities of masses were supposed to be measured which seems a bit unrealistic. However, vertical velocity information of the transmission mass can be measured directly through the encoder of the actuator since they are connected rigidly. Additionally, the vertical velocity information of the heel can be gathered numerically from the heel position measurements. (ii) In Assumption 2, the controller does not require the condition of persistent excitation (PE). Therefore, it is capable of maintaining the stability and convergence even if the number of distinct frequencies, q, is overestimated or $d(t) = 0$.

4.2. Disturbance Representation

The disturbance acting on the system was parametrized to approach the problem as an adaptive control problem. It is important to note that using Assumption 2, the unknown disturbance can be transformed into a form so that its undesirable effect can be removed regardless of its amplitude, frequency and phase and it is only necessary to know the maximum number of distinct frequencies which the disturbance consists of. This technique allows the design to be suitable for subjects with any weight and at any walking speed since the design actually becomes independent of the disturbance properties once parametrized.

By using Assumption 2 and defining a new parameter $a_4{=}1/m_2$, the disturbance term $d(t)/m_2$ can be represented as $a_4 d(t){=}c{+}\sum_{i=1}^{q} g_i sin(\omega_i t + \phi_i)$. Additionally, since the number of maximum distinct frequencies that the disturbance consists of, was assumed to be known, it can be represented as the output of a linear exosystem,

$$\dot{w}(t) = Sw(t) \tag{4.3}$$

$$a_4 d(t) = h^T w(t) \tag{4.4}$$

where $w{\in}\mathbb{R}^{2q+1}$. Amplitude and phase uncertainties of the disturbance, $a_4 d(t)$, is related to the initial conditions of (4.4) and the matrix, S, depends on unknown frequencies.

Parametrization of the disturbance was performed by following [27]. Let $G \in \mathbb{R}^{(2q+1)\times(2q+1)}$ be a Hurwitz matrix with distinct eigenvalues, $l \in \mathbb{R}^{2q+1}$ and (G, l) be a controllable pair. Since (h^T, S) is observable and the spectra of S and G are disjoint, the only solution $M \in \mathbb{R}^{(2q+1)\times(2q+1)}$ of the Sylvester equation

$$MS - GM = lh^T \tag{4.5}$$

is invertible [28]. A state transformation with the use of $\tau(t) = Mw(t)$ yields,

$$\dot{\tau}(t) = G\tau(t) + la_4 d \tag{4.6}$$

$$a_4 d(t) = \theta^T \tau(t) \tag{4.7}$$

where $\theta^T = h^T M^{-1} \in \mathbb{R}^{2q+1}$.

In (4.7), the unknown disturbance was represented as a product of two unknown vectors, namely unknown constant vector θ^T and unknown time dependent vector $\tau(t)$. Through designing a conceptual observer, the value of $\tau(t)$ can be estimated and the following lemma establishes the properties of this observer.

Theorem 4.1. *The inaccessible disturbance $a_4 d(t)$ can be represented in the form,*

$$a_4 d(t) = \theta^T \delta(t) + \theta^T \zeta(t) \tag{4.8}$$

where

$$\zeta = \eta + l\dot{z}_h(t) \tag{4.9}$$

$$\dot{\eta} = G(\eta + l\dot{z}_h(t)) - l(-g + a_3(z_t(t) - z_h(t))) \tag{4.10}$$

and $a_3 = k/m_2$. The estimation error $\delta(t) \in \mathbb{R}^{2q+1}$ is defined as follows,

$$\delta(t) = \tau(t) - \zeta(t) \tag{4.11}$$

and its dynamics is represented as,

$$\dot{\delta}(t) = G\delta(t) \tag{4.12}$$

4.3. Adaptive Backstepping Controller Design

Throughout the design, backstepping methods, explained in [29], were used. In backstepping approach, some signals were considered as virtual inputs whose dynamics allow the designer to reach actual inputs. Then, by designing control laws for those virtual inputs and stepping back by taking derivatives, final controllers were designed for real inputs. Since the main aim of the control system is to make patients be able to hold their heels at desired positions during gait cycles, the virtual controller is seen as $z_t(t)$ in Figure 4.3.

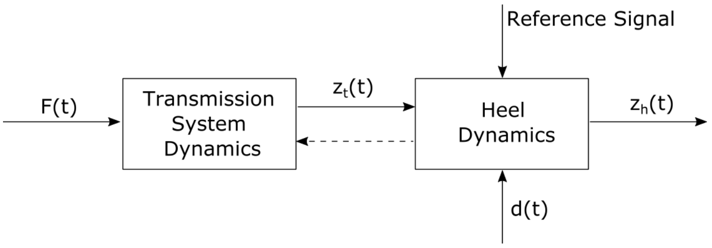


Figure 4.3. Block diagram representation of the model which is used during the application of backstepping technique.

Considering that our main aim is forcing the heel to track a certain trajectory, i.e. gait cycle of healthy subjects, we start with defining an error term between current and desired (reference) positions of the heel and its dynamics as follows,

$$e_1 = z_h - z_{h_{ref}} \tag{4.13}$$

$$\dot{e}_1 = \dot{z}_h - \dot{z}_{h_{ref}} \tag{4.14}$$

By taking time derivative of (4.14), replacing $\ddot{z}_h$ term with (4.2), and using (4.8), we can represent the error system in a state-space form as follows,

$$\dot{x} = Ax + B(a_3 z_t - a_3 z_{h_{ref}} - g + \theta^T \zeta + \theta^T \delta - \ddot{z}_{h_{ref}}) \tag{4.15}$$

where $x=\begin{bmatrix} e_1 \\ \dot{e}_1 \end{bmatrix}$, $A=\begin{bmatrix} 0 & 1 \\ -a_3 & 0 \end{bmatrix}$ and $B=\begin{bmatrix} 0 \\ 1 \end{bmatrix}$.

In order to achieve convergence of z_h to $z_{h_{ref}}$, state space system (4.15) can be forced by using $a_3 z_t$ as a virtual input and its desired value is given by,

$$(a_3 z_t)_{desired} = g + a_3 z_{h_{ref}} + \ddot{z}_{h_{ref}} - \hat{\theta}^T \zeta + Kx \tag{4.16}$$

where $K \in \mathbb{R}^2$ is a matrix which makes $A{+}BK$ Hurwitz and $\hat{\theta}^T$ is the estimation of unknown vector θ^T. Throughout the design, the estimate of θ^T and the estimation error were denoted by " ˆ " and " ˜ ", respectively. In example, $\tilde{\theta}^T{=}\theta^T{-}\hat{\theta}^T$.

Since $a_3 z_t$ is not actuated directly, it is not possible to assign its value. Therefore, another error term appears in the design which is represented by,

$$e_2 = a_3 z_t - (a_3 z_t)_{desired} \tag{4.17}$$

Reaching the real input $F(t)$ is only possible by reaching $\ddot{z}_t$ and it can be achieved by taking time derivative of (4.17) twice. Therefore, we continue with defining the dynamics of e_2 as following,

$$\dot{e}_2 = a_3 \dot{z}_t - a_3 \dot{z}_{h_{ref}} - \dddot{z}_{h_{ref}} + \hat{\theta}^T \zeta + \hat{\theta}^T \Xi - K\Gamma + (\hat{\theta}^T l - KB)(\tilde{\theta}^T \zeta + \theta^T \delta) \tag{4.18}$$

where $\Xi = \dot{\eta} + l(-g + a_3(z_t - z_h) + \hat{\theta}^T \zeta)$ and $\Gamma = (A + KB)x + Be_2$.

As expected, a term which includes $\dot{z}_t$ appears in (4.18). It was used as a virtual controller at this step and its value was determined with the use of the following Lyapunov function,

$$V_1 = \frac{1}{2}(x^T P_x x + e_2^2 + \frac{\tilde{\theta}^T \tilde{\theta}}{\gamma_\theta} + \epsilon \delta^T P_\delta \delta) \tag{4.19}$$

where P_x and P_δ are positive definite matrices whose values were determined at the final step.

By calculating derivative of (4.19) with respect to time and considering cross terms, the virtual controller was designed and its error is given by,

$$e_3 = a_3\dot{z}_t - (a_3\dot{z}_{h_{ref}} + \dddot{z}_{h_{ref}} - \Lambda^T\zeta - \hat{\theta}^T\Xi + K\Gamma - e_2(\hat{\theta}^T\hat{\theta} + K_{e_2}) - x^T P_x B) \tag{4.20}$$

where $\Lambda = x^T P_x B\zeta + e_2(\hat{\theta}^T l\zeta - KB\zeta)$ and $K_{e_2} > 0$.

The actual input to the system, $F(t)$, can be reached by taking time derivative of (4.20) and replacing $a_3\ddot{z}_t$ with (4.1). By the backstepping method, we transform the system (4.1), (4.2) into an error system whose states are x, e_2, e_3. Therefore, we can design the controller for $F(t)$ at this step.

Final control algorithm for the force input $F(t)$ is given by,

$$\begin{aligned} F = \Big(\frac{1}{a_2a_3}\Big)\Big(& a_3(g + a_1(z_t - z_h)) + a_3\ddot{z}_{h_{ref}} + \ddddot{z}_{h_{ref}} - (x^T P_x B\zeta + G^T\hat{\theta})^T\Xi + \Gamma^T P_x B \\ & - \dot{\hat{\theta}}^T G(\eta + l\dot{z}_h + l\hat{\theta}^T\zeta) - (x^T P_x B\Xi + \Gamma^T P_x B\zeta + \dot{\hat{\theta}} l^T\hat{\theta})^T\zeta \\ & - e_2((\hat{\theta}^T l\zeta + \hat{\theta} l^T\hat{\theta} - KB\zeta)^T\Xi + ((KB + \hat{\theta}^T l)\Xi + \dot{\hat{\theta}}^T l\zeta)^T\zeta - \dot{\hat{\theta}}^T\hat{\theta}) \\ & - (\zeta^T(\hat{\theta}^T l - KB)^T\zeta + \hat{\theta}^T\hat{\theta} + K_{e_2} - KB)\Psi - K_{e_3}e_3 - \Phi - e_2\nu\Big) \end{aligned} \tag{4.21}$$

and the update law is given by,

$$\dot{\hat{\theta}} = \gamma_d(\Lambda + e_3\nu) \tag{4.22}$$

where

$$\Psi =(\dot{\hat{\theta}} - (x^T P_x B\zeta + e_2(\hat{\theta}^T l\zeta - KB\zeta)))^T\zeta - x^T P_x B - e_2(\hat{\theta}^T\hat{\theta} + K_{e_2}) \tag{4.23}$$

$$\Phi =(x^T P_x B\zeta)^T(x^T P_x B\zeta) + ((\hat{\theta}^T l\zeta)^T(\hat{\theta}^T l\zeta))^2 + ((\hat{\theta}^T l\zeta)^T\zeta)^2 + (\zeta^T\zeta)^2 + (\hat{\theta}^T\hat{\theta})^3 + (\hat{\theta}^T l)^4 \tag{4.24}$$

$$\begin{aligned}\nu =&(((x^T P_x B\zeta)^T + \hat{\theta}^T G + (\hat{\theta}^T l\hat{\theta}^T)) + (x^T P_x Bl)^T\zeta - K(A+BK)B + B^T P_x B)\zeta\\ &+ e_2((\hat{\theta}^T l\zeta)^T l + l^T\hat{\theta} l^T\zeta(KB\zeta)^T l - (KBl)^T\zeta)\zeta\\ &+ (((\hat{\theta}^T l\zeta)^T - (KB\zeta)^T)\zeta + (\hat{\theta}^T\hat{\theta} - KB + K_{e_2}))\Lambda \end{aligned} \tag{4.25}$$

and $a_1 = 1/m_1$, $\gamma_\theta, K_{e_3} > 0$.

Following theorem summarizes the properties of the closed loop system.

Theorem 4.2. *Consider the closed loop system that is represented by* (4.1), (4.2), *forced by unknown disturbance* (4.4), *disturbance observers* (4.9), (4.10), *control law* (4.21) *and update law* (4.22). *Under Assumptions 1–3, the following holds;*

The equilibrium of the closed loop system is stable and the signals $\|x(t)\|, \|e_2(t)\|, \|e_3(t)\|, \|\delta(t)\|$, *specifically* $\|z_h(t) - z_{h_{ref}}(t)\|$ *converge to zero as* $t \to \infty$.

4.4. PID Controller Design

The PID controller was designed by using two feedforward loops. The closed loop system is depicted in Figure 4.4. The main difference between the backstepping and PID designs is the disturbance representation. Since PID controller is only able to reduce the effects of constant disturbances, the disturbance term, $d(t)$, appearing in (4.2) is not replaced with its equivalent representation which is provided in Section 4.2. Hence, the design starts with a modified version of (4.15), given as,

$$\dot{x} = Ax + B(a_3 z_t - a_3 z_{h_{ref}} - g + d - \ddot{z}_{h_{ref}}) \tag{4.26}$$

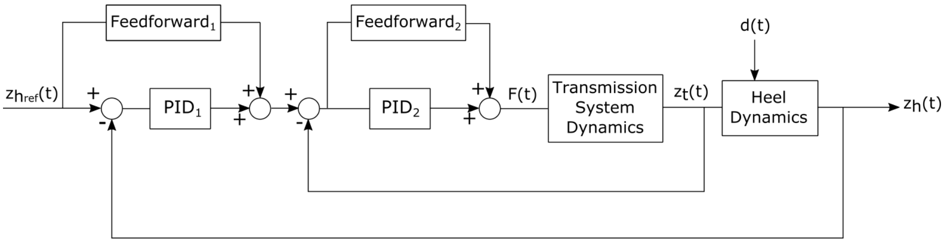


Figure 4.4. Block diagram representation of the closed loop system for the PID design.

As explained in backstepping design, $a_3 z_t$ was chosen as the virtual controller and a PID controller was designed so that it satisfies stability conditions and diminishes the effect of the disturbance thanks to the Integral term. Then, by differentiating the virtual controller twice, the actual controller $F(t)$ was reached and another PID controller which satisfies stability conditions was designed. It should be noted that two feedforward loops appear in the block diagram as a result of this two step design method.

5. IMPLEMENTATION

The designed adaptive controller and the PID controller were implemented using a National Instruments (NI) CompactRIO platform which is based on a field programmable gate array (FPGA) and the LabVIEW software. Block diagram representation of the control system can be seen in Figure 5.1. FPGA was used to read sensor values and generate pulse width modulation (PWM) signals in order to run the DC motor. In addition to the FPGA, the controller codes were deployed to the real-time OS of the CompactRIO to read data from the FPGA, calculate the required motor current, and write data to the FPGA. LabVIEW codes in the FPGA and real-time sides can be seen in Appendix B.

Two modules were inserted into the slots of the chassis of the NI CompactRIO. The first one is NI 9505, a DC Brushed Servo Drive. The NI 9505 returns the motor current data to the LabVIEW FPGA Module. The LabVIEW FPGA Module generates a PWM signal and sends the signal to the NI 9505. Quadrature encoder signals pass through the NI 9505 and processed in the FPGA module. The second module is NI 9381, a multifunction I/O module for CompactRIO systems. It was used to read linear potentiometer and force sensitive resistor (FSR) values which are required for the controller implementation. Two power supplies were used to provide power to the system. The first one is NI PS-15 (24 V, 5 A) and it is used to supply power to the controller of the CompactRIO. The second power supply that can provide voltage of 24 V and current of 5 A is used for the NI 9505.

The wiring between the encoder and NI 9505 can be seen in Figure 5.2. Quadrature encoder uses three output channels (A, B and index) to sense position. By monitoring the number of pulses and the relative phases of A and B signals, the position and direction of the rotation can be tracked. Index signal which provides a single pulse per revolution is also used for determination of a reference position.

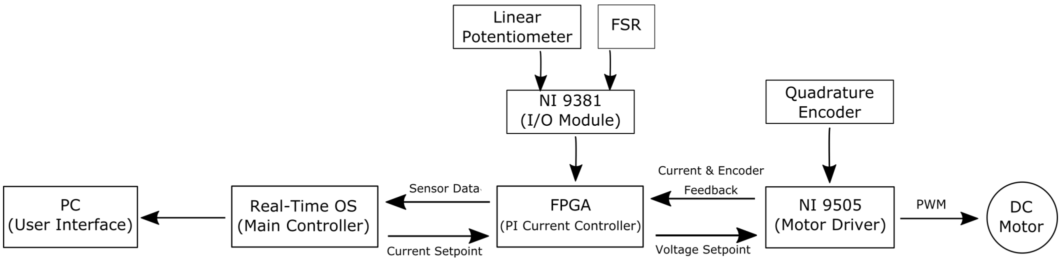
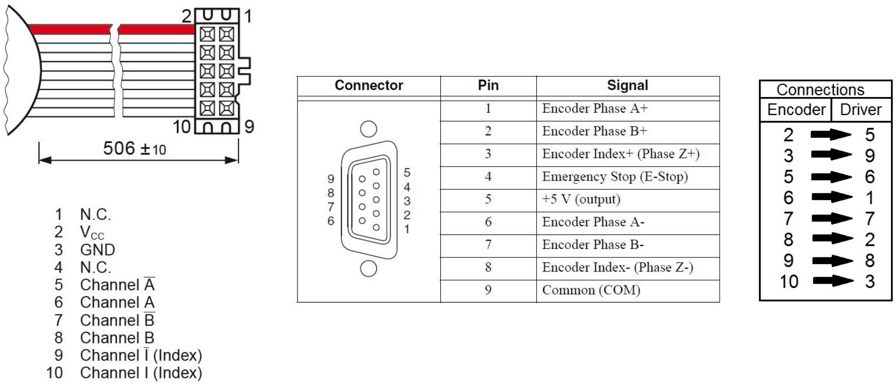


Figure 5.1. Control system block diagram. Main controller was implemented in real-time OS. FPGA was used to process sensor signals and generate PWM signals to run the DC motor.

The connections between the linear potentiometer and NI 9381 multifunction I/O module is shown in Figure 5.3. Three wires were soldered to the soldering pads of the linear potentiometer. The black wire goes to the ground pin, the red wire goes to the analog output pin, and the blue wire was connected to the analog input pin of the NI 9381 module. Electric potential of 5 V was supplied to the analog output pin using LabVIEW.

The wiring of the FSR can be seen in Figure 5.4. A fixed resistor (10 KΩ resistance) as a voltage divider was used for the voltage comparison. The green wire goes to the analog input pin, the red wire goes to the analog output pin, and the black wire at the end of the resistor was connected to the analog input pin of the NI 9381 module. Electric potential of 5 V was supplied to the analog output pin using LabVIEW.

Position measurement of the DC motor was obtained with the LabVIEW functions. With the information of the number of clock ticks between loop increments and mean filtering, accurate motor velocity report was ensured as well. Measured noisy spring displacement was also mean filtered. The velocity of the spring compression was obtained by differentiating the spring displacement. The data was converted to the linear position and velocity values of the m_1 and m_2 (z_t,$\dot{z}_t$,z_h,$\dot{z}_h$). The weights of masses m_1 and m_2 were measured as 0.5 kg. The stride time was also found by measuring the elapsed time between two heel strikes.



1 N.C.
2 V_{CC}
3 GND
4 N.C.
5 Channel $\overline{A}$
6 Channel A
7 Channel $\overline{B}$
8 Channel B
9 Channel $\overline{I}$ (Index)
10 Channel I (Index)

Connector	Pin	Signal
	1	Encoder Phase A+
	2	Encoder Phase B+
	3	Encoder Index+ (Phase Z+)
	4	Emergency Stop (E-Stop)
	5	+5 V (output)
	6	Encoder Phase A-
	7	Encoder Phase B-
	8	Encoder Index- (Phase Z-)
	9	Common (COM)

Connections	
Encoder	Driver
2 ➡	5
3 ➡	9
5 ➡	6
6 ➡	1
7 ➡	7
8 ➡	2
9 ➡	8
10 ➡	3

Figure 5.2. Connections between the quadrature encoder and the NI 9505.

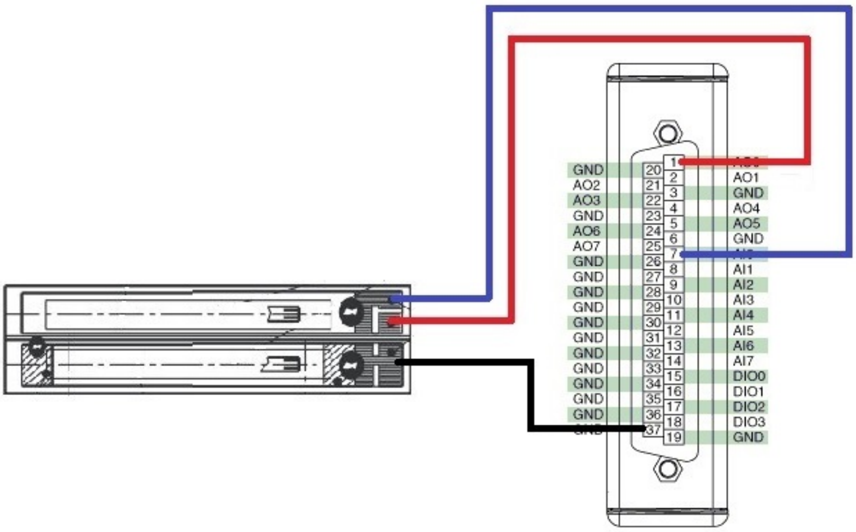


Figure 5.3. Connections between the linear potentiometer and the NI 9381.

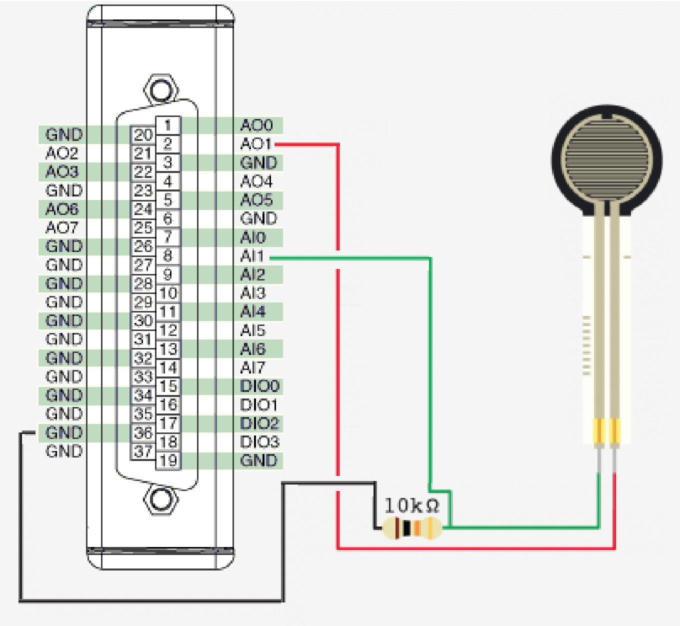


Figure 5.4. Connections between the force resistive sensor and the NI 9381.

The main control algorithm was implemented in the real-time side. The force input calculated in the real-time side was converted to the required motor torque by using the ball screw torque equation (Equation 3.1). Then, obtained motor torque was converted to the motor current using,

$$i = \frac{T}{c_t}, \tag{5.1}$$

where c_t is the torque constant of the motor. Calculated motor current setpoint was transferred to the FPGA side. A proportional-integral (PI) current controller was designed to obtain the required voltage setpoint in percentage of PWM duty cycle. The current controller is based on the continuous PI equation,

$$U_{PWM}(t) = K_{p_{PWM}}\varepsilon(t) + K_{I_{PWM}} \int_0^t \varepsilon(\tau)d(\tau), \tag{5.2}$$

where $U_{PWM}(t)$ is the voltage output in duty cycle, $K_{p_{PWM}}$ is the proportional gain, $K_{I_{PWM}}$ is the integral gain, and $\varepsilon(\tau)$ is the position error (difference between the current setpoint and the current feedback). To run the DC motor, PWM signals with 20 kHz frequency were generated using the calculated voltage output.

6. EVALUATION METHODS

In this chapter, evaluation methods for the simulations and experiments are given.

6.1. Simulations

Numerical simulations were performed on MATLAB to observe and compare the performance of proposed adaptive backstepping method with one of the designed PID controllers. MATLAB code for the performance comparison of the controllers can be seen in Appendix C. Parameters of the PID controller were chosen in a way that the controller has the least possible tracking error while tracking the trajectory in a desired shape. It is noted that higher PID terms than the chosen ones resulted in oscillations around the trajectory.

Disturbance representation was prepared by considering the effects of the GRF and the weight of the foot. A disturbance diagram indicating the disturbance magnitudes acting on an 80 kg subject during a gait cycle, is provided in Figure 4.2.

Additionally, reference signal that the heel should track was generated by using the ankle joint motion of healthy individuals during a gait cycle (Figure 2.1). By transforming joint degree information to vertical heel position data, reference signal is provided in Figure 6.1.

Simulations for eight cases, indicated in Table 6.1, were performed to compare the performances of adaptive controller and PID controller. In the simulations, subjects with two different weights each walking at two different speeds on two different roads that consist of different maximum number of distinct frequencies were considered. The duration of the simulations were determined as the time required for completion of three gait cycles in each case. Specifications of simulations can be found in Table 6.1 in which speed is given in terms of gait cycle period. During all simulations, system parameters were taken as $m_1 = 0.5kg$, $m_2 = 0.5kg$, $k = 26kN/m$ and moment arm $L = 0.08m$

by considering the actual system. Moreover, road profiles were assumed to have either one or two distinct frequencies. The controllable pair (G, l) was taken as $l = [0_2 \; 1]^T$, with $0_n = [0 \; ... \; 0]^T \in \mathbb{R}^n$ and $G = \begin{bmatrix} 0_2 & I_2 \\ 0_2^T & 0 \end{bmatrix} - l[7.99e{+}6 \; 1.20e{+}5 \; 601.3]$ for $q = 1$, whereas $l = [0_4 \; 1]^T$, and $G = \begin{bmatrix} 0_4 & I_4 \\ 0_4^T & 0 \end{bmatrix} - l[5.27e{+}11 \; 1.20e{+}10 \; 1.10e{+}10 \; 49982 \; 1121]$ for $q = 2$.

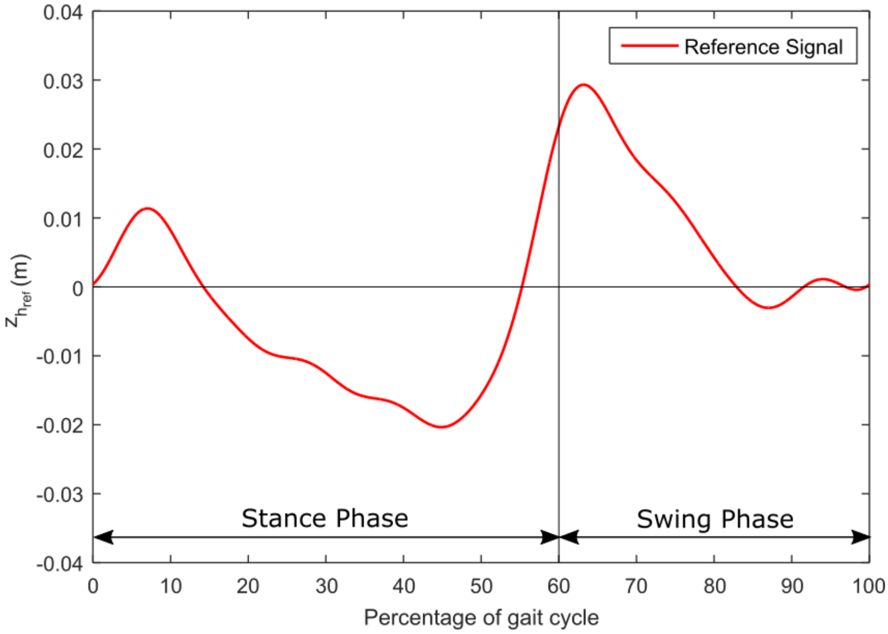


Figure 6.1. Reference vertical heel position signal for simulations performed using subjects with 80 kg. Stance and swing phases start when the heel first strikes and leaves the ground, respectively.

Table 6.1. Simulation specifications for the evaluation of the control algorithms

Case Number:	1	2	3	4	5	6	7	8
Subject Weight (kg):	80	60	80	60	80	60	80	60
Gait Cycle Period (s):	1	1	0.7	0.7	1	1	0.7	0.7
Number of freq. (q):	1	1	1	1	2	2	2	2

6.2. Experiments

In order to characterize the designed actuator, to show the performance of the proposed aAFO system as a whole, and to compare performances of the proposed adaptive controller and the PID controller, five tests were conducted: step response, torque output, trajectory tracking, stride frequency, and cardiopulmonary exercise tests. Three of these tests involve participation of users. The procedure of the user tests has been approved by the Institutional Review Board of Bogazici University (INAREK).

In order to compare the time behaviour of the controllers, first, step response test was conducted. The prototype was fixed to a rigid stationary table from the brackets and was not worn by a user. Three step inputs of 0.01 m, 0.02 m, and 0.03 m were given to the system as the reference position. Responses of the adaptive and PID controllers to the position step inputs were analyzed.

To understand the torque characteristics of the developed compliant actuator, a torque test was conducted. A test setup was prepared as seen in Figure 6.2. The active ankle-foot orthosis prototype was fixed between two stationary tables using a clamp. The compression end of the dynamometer was fixed to the ground, while the tension end was hanged to the ball-and-socket joint. 0.02 m step input was given to the system as the reference position and the force values on the dynamometer were observed.

As a third test, tracking performances of the adaptive and PID controllers were compared when the prototype was worn by a user. One woman and two men with different weights (68 kg, 58 kg, and 56 kg) volunteered to participate the study. They wore the prototype on their right foot and walked with a constant speed on a treadmill (Figure 6.3). The speed of the treadmill was determined as 1.7 km/h (0.67 Hz stride frequency). The system starts to run with the first heel strike. After two heel strikes, the system adapts itself to the treadmill speed. Each of the designed controllers were tested for three minutes with a 10 minutes of rest between two walking tests. Root-mean-square (RMS) errors of the controllers that indicate the deviations from the desired trajectory were measured for 30 consecutive gait cycles.

Figure 6.2. Torque test setup. A step input was given to the system and the force values on the dynamometer were observed.

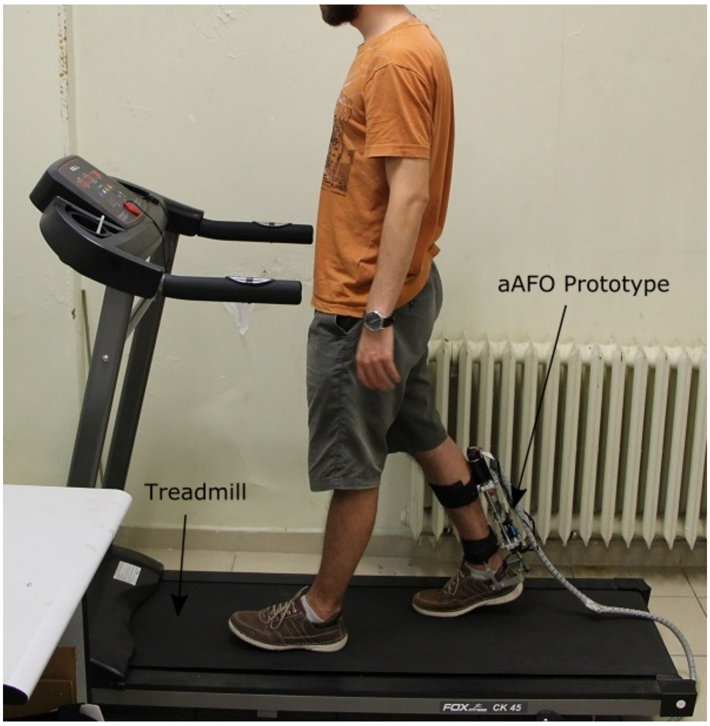


Figure 6.3. A seen from the trajectory tracking test for performance comparison of the controllers. The user walked on the treadmill with a constant speed of 1.7 km/h.

Stride frequency test was conducted to characterize the tracking performance of the adaptive controller for varying walking speeds. 58-kg subject wore the prototype on his right foot and walked on a treadmill with different stride frequencies of 0.6 Hz, 0.8 Hz, and 1 Hz. The subject was given 10 minutes of rests between three tests.

Finally, to evaluate the efficiency of the developed aAFO prototype, a cardiopulmonary exercise test (CPET) was conducted using COSMED Fitmate Pro, a metabolic assessment system (Figure 6.4). The device measures oxygen consumption during CPET, and measured maximum oxygen consumption rate (VO2 max) gives an idea for the spent effort. The first test was conducted with a non-hinged thermoplastic passive AFO. The weight of the passive AFO is 0.5 kg, so an extra weight of 1.3 kg was attached to it to provide equal conditions (weight) with the active AFO prototype. 58-kg user wore a VO2 mask (a reusable mask for oxygen consumption measurements) and walked on a treadmill for 3 minutes with a constant speed of 1.7 km/h (0.67 Hz stride frequency) wearing the passive AFO. After a 1-hour rest, the user wore the active AFO prototype and walked another 3 minutes on the treadmill with the same speed). VO2 values were measured with 30 seconds of intervals. Average oxygen consumptions were calculated for both tests.

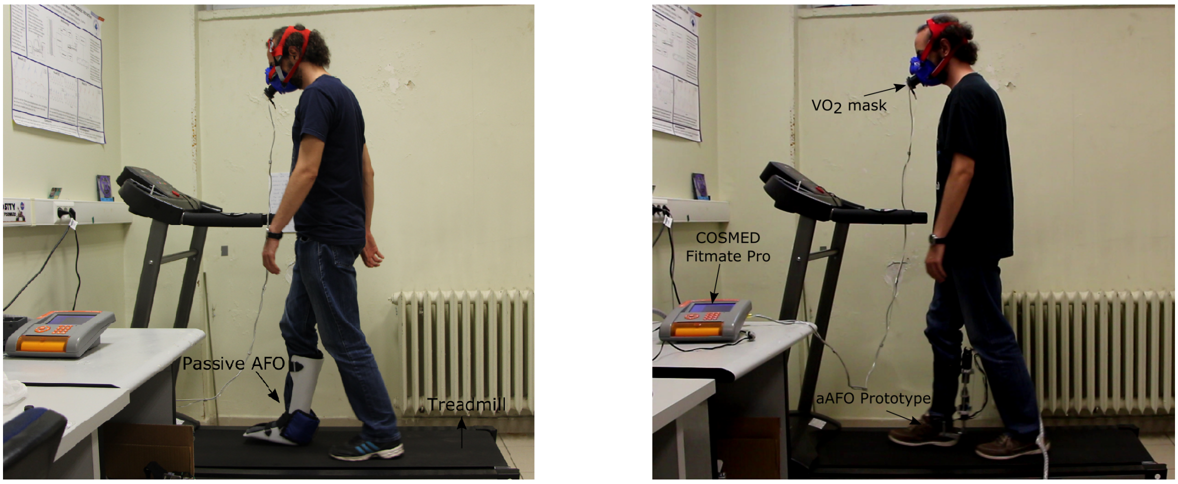


Figure 6.4. Cardiopulmonary exercise test. User wears a passive ankle-foot orthosis (left). User wears the active ankle-foot orthosis prototype (right).

7. RESULTS

In this chapter, simulation and experimental results are presented.

7.1. Simulation Results

Performance comparison of the adaptive controller and the PID controller was achieved by calculating the error between the heel position and the reference signal with the use of second, $\|.\|_2$, and infinity, $\|.\|_\infty$, norms. The second norm error values indicate the root-mean-square (RMS) error of the trajectory tracking during one gait cycle, whereas the maximum norm values indicate maximum deviations from the trajectory during three gait cycles. Results are provided in Table 7.1. It should be noted that, since the working range of AFOs are limited, around ±0.04 m, maximum deviations from the reference signal should be considered as an important performance criteria. Moreover, it was observed that the use of two distinct frequencies, $q = 2$, for the road profile provides almost the same results for all cases. Therefore, they are not explicitly included in the Table 7.1.

Illustrative results for Case 1 are given to make a qualitative performance comparison of controllers. In Figure 7.1, transmission mass displacements are provided. The tracking results are given by Figure 7.2. Additionally, results regarding the total net moments acting on the ankle are provided by Figure 7.3. The total net moment was calculated as $Moment = \frac{kz_t L}{weight}$ where L is the level arm in meters, $weight$ is the weight of the subject in kilograms and kz_t is the force that acts on the mass connected to the heel.

Table 7.1. Quantitative simulation results for the first 4 cases. Results for the cases 5-8 are exactly the same with cases 1-4, respectively.

Case number:	Gait Cycle Period:	Subject Weight:	$\|.\|_2$ Error (m)		$\|.\|_\infty$ Error (m)	
			Adaptive	PID	Adaptive	PID
1	1 s	80 kg	0.48	1.23	0.007	0.018
2	1 s	60 kg	0.36	0.92	0.005	0.013
3	0.7 s	80 kg	0.42	1.04	0.007	0.018
4	0.7 s	60 kg	0.32	0.78	0.005	0.014

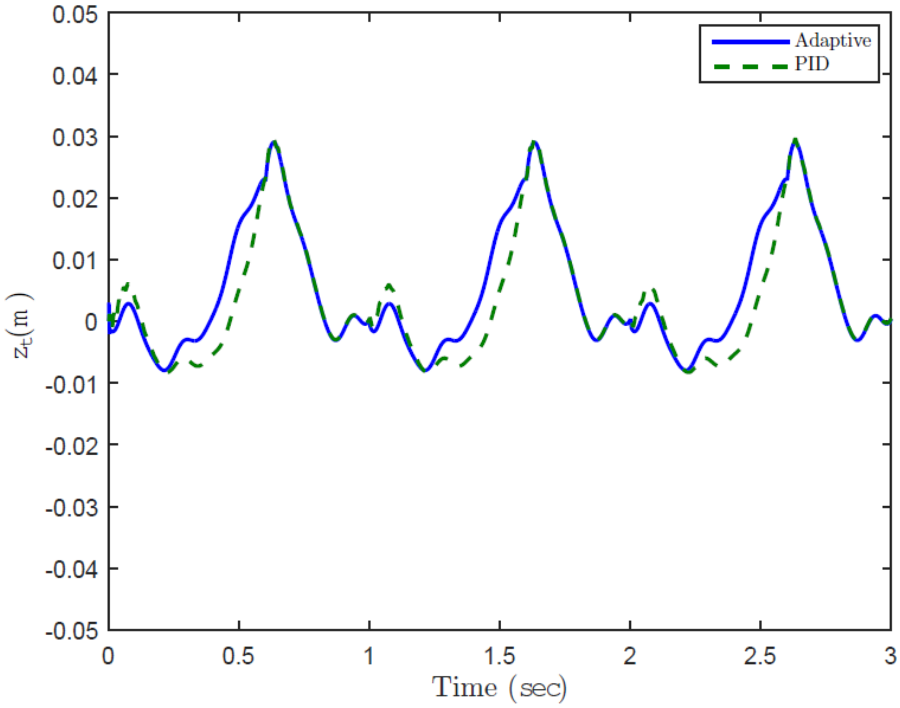


Figure 7.1. Transmission mass displacement

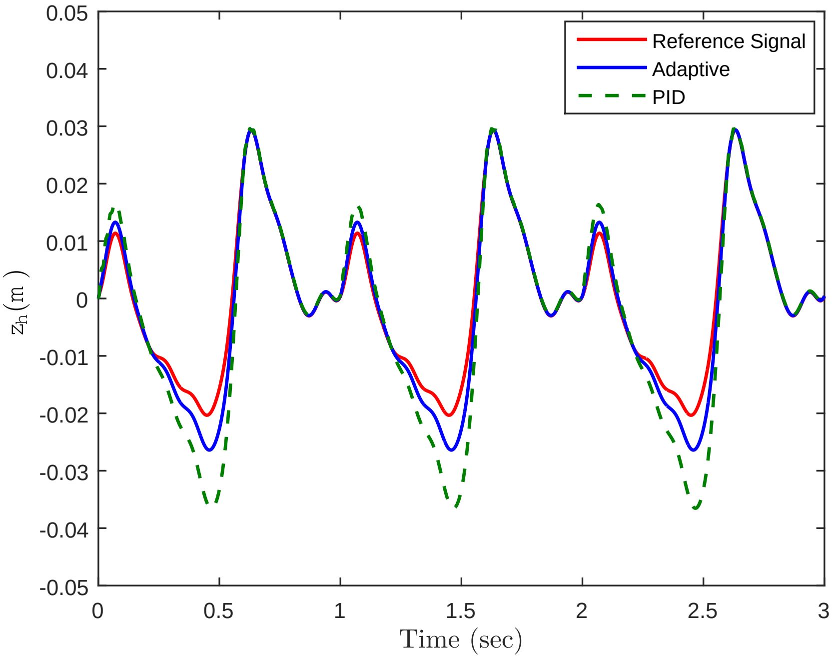


Figure 7.2. Reference vertical heel position and tracking results.

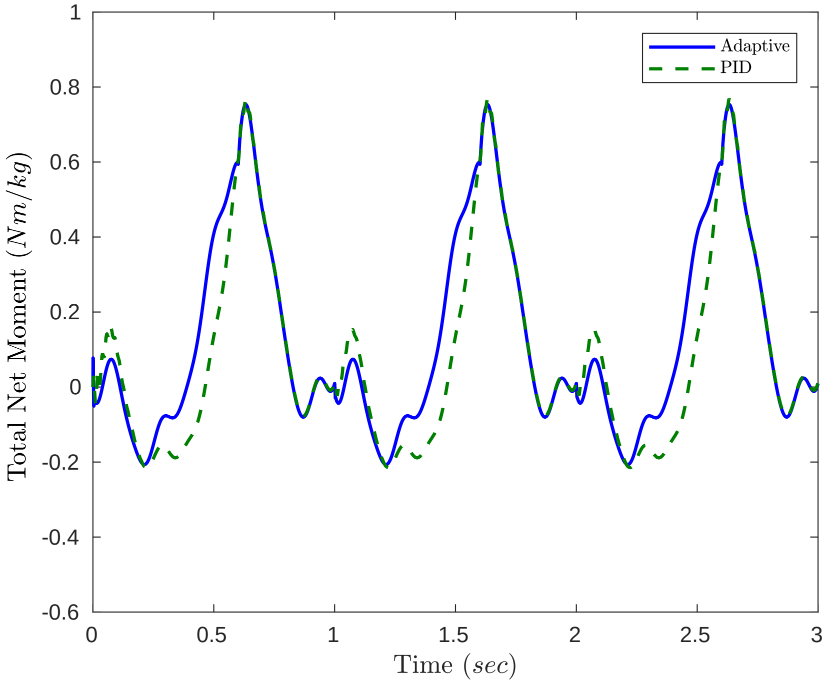


Figure 7.3. Total net moment acting on the ankle joint. Note that the moment is calculated as $Moment = \frac{kz_tL}{subject\ weight}$.

7.2. Experimental Results

Step input response experiment was conducted to compare the time behaviour of the controllers. The measured responses of the PID and adaptive controllers to three position step inputs are presented in Figure 7.4. Red lines represent the position step inputs of 0.01 m, 0.02 m and 0.03 m. The responses of the adaptive controller are represented by blue lines, and green lines stand for the PID controller step input response. Performance indicators of the adaptive controller and PID controller are given in Table 7.2. Percentage overshoot values and settling times (required time to reach and stay within a range of 5% of the desired value) are presented.

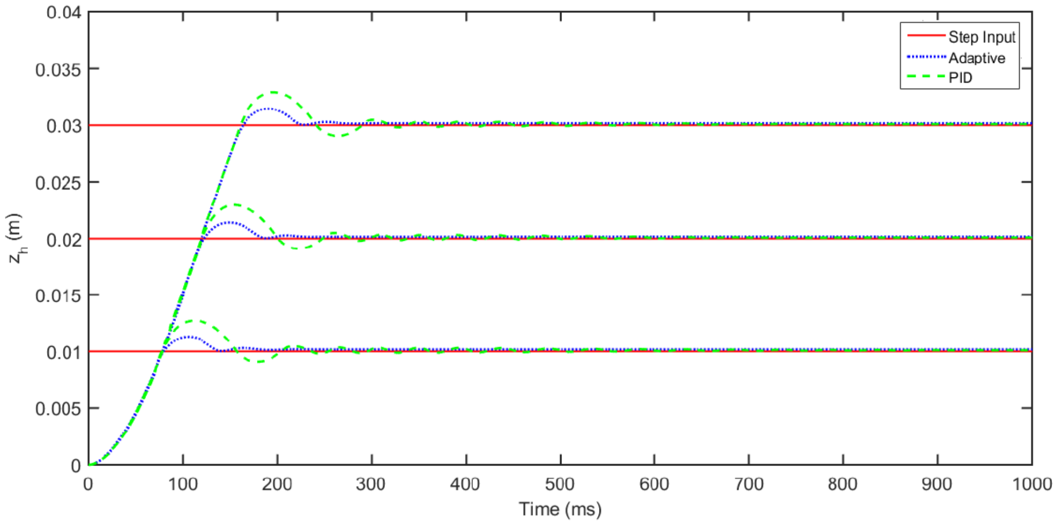


Figure 7.4. Position step input responses of the adaptive and PID controller.

Table 7.2. Performance indicators of the step response experiment.

	0.01 m		0.02 m		0.03 m	
	Adaptive	PID	Adaptive	PID	Adaptive	PID
Percentage overshoot (%)	13.2	27.4	7	14.8	4.9	9.8
Settling time (ms)	128	195	166	192	155	227

The mean of the force values observed on the dynamometer is 238.7 N with a standard deviation of 29.3 N in conducted three torque output tests. Since the moment arm of the active AFO prototype is 0.082 m, an average of 19.6 Nm torque is generated on the ankle joint.

Trajectory tracking results of the controllers for three consecutive gait cycles of Subject 3 are given in Figure 7.5 as an example. RMS error measurements in mm for 30 consecutive gait cycles of three subjects are presented in Table 7.3.

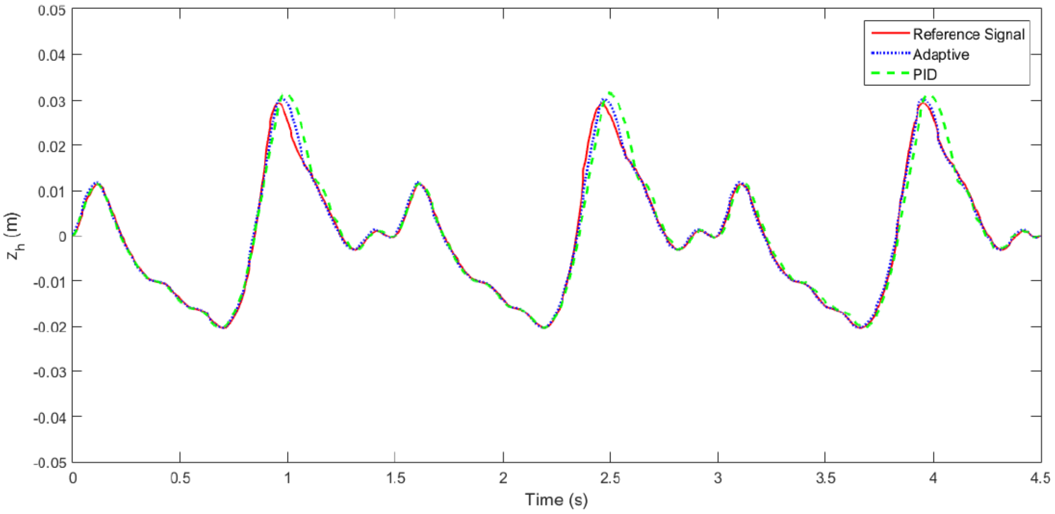


Figure 7.5. Trajectory tracking performances of the adaptive and PID controllers for a single subject.

Table 7.3. RMS error measurements of the adaptive and PID controllers.

Subject number:	Weight (kg):	RMS Error (mm)	
		Adaptive	PID
1	68	0.87	2.3
2	58	0.89	1.92
3	56	0.84	1.95

Trajectory tracking results of the adaptive controller for varying stride frequencies (0.6 Hz, 0.8 Hz and 1 Hz) of Subject 2 are shown in Figure 7.6. Red lines represent reference signals, while blue lines stand for trajectory tracking results.

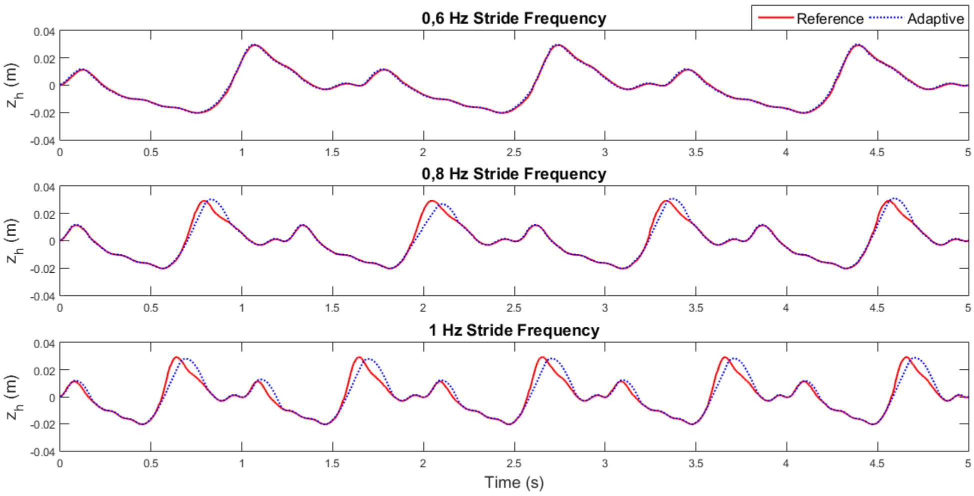


Figure 7.6. Trajectory tracking performances of the adaptive controller for subject 2 for different stride frequencies.

Finally, average oxygen consumptions for the cardiopulmonary exercise tests were measured as 12 ml/(kg min) when wearing the passive AFO and 12.3 ml/(kg min) when wearing the active AFO prototype.

8. DISCUSSION

Simulation results show that the adaptive controller achieves about 2.5 times better performance than the PID controller when the second norm error and the maximum norm values are considered (Table 7.1). Since the magnitude of the disturbance increases with the subject weight, the error values are larger for the 80 kg subject. In Figure 7.1, the difference in the transmission mass displacements reflects the performance difference of two control designs. The superiority of the adaptive controller can easily be seen by looking at the tracking results provided in Figure 7.2. The PID controller cannot reduce the effect of the disturbance as good as the adaptive controller. A disturbance observer for the PID controller may decrease the tracking error. The total net moments acting on the ankle is also an influential indicator for the applicability of the design. As seen in Figure 7.3, the maximum net moment acting on the ankle is about 0.75 Nm/kg. Since the weight of the case 1 is 80 kg in simulations, maximum net moment on the ankle is 60 Nm which a portion of it can be provided by designed actuator.

Experimental results show that the adaptive controller has a better response to the position step inputs than the PID controller as shown in Figure 7.4. Considering the percentage overshoot values given in Table 7.2, the adaptive controller shows approximately two times better performance than the PID controller. If settling times are considered, it can be realized that the adaptive controller outperforms PID the controller. However, the output reached desired position in a short time in both controllers. In the literature, settling times of the active AFO actuators are between 0.5 s and 1 s [7, 30]. Therefore it can be stated that, the designed actuator satisfies the design requirement of low response time. In the torque output test, an average of 19.6 Nm torque is generated on the ankle joint. That value is lower than the maximum net ankle moment in the simulations. Disassembly of the coupling under axial load prevented to reach higher torque values. However, considering muscles of a subject can provide some of the required ankle moments, generated net ankle moment by the actuator is sufficient for the plantar flexion and dorsiflexion motions. So, the high torque requirement is satisfied with the designed actuator. Adaptation to gait was achieved with the use of a force sensor in the trajectory tracking tests. The sensor system of the developed prototype is simpler than [13] and [14]. Considering the trajectory tracking results provided in Figure 7.5 and RMS error measurements of three subjects given in Table 7.3, the adaptive controller achieves almost two times better performance than the PID controller as similar to the simulation results. The developed aAFO outperforms the previous works in [22] and [31] in trajectory tracking, thus it can be concluded that the low position tracking error requirement is satisfied as well. If the tracking results for different stride frequencies given in Figure 7.6 are considered, it can be seen that when the gait cycle frequency is 0.6 Hz, the performance of the implemented adaptive controller is better than higher frequencies. Since the walking speed increases, tracking error increases as well because of the speed limitations of the DC motor. Finally, CPET results indicate that, there is no slight difference between oxygen consumptions when wearing passive AFO and active AFO. This means, the subject spends similar amounts of effort when performing CPET. For a reliable effort comparison of the AFOs and evaluation of the efficiency of the aAFO prototype, an incremental CPET protocol should be performed. However, speed limitation of the DC motor prevents the application of an incremental test protocol. It should be noted that, CPET results might show significant difference in a patient study.

There is a slight difference between the experimental results and the simulation results in terms of the trajectory tracking performance. The magnitude of the tracking error is lower in the experimental results. The reason for that is the larger disturbance magnitudes employed in the simulations, which are conducted to determine whether the proposed control algorithm works well in extreme conditions as well. However, the magnitude of the disturbance acting on the foot is lower in the actual implementation because the heel is in contact with the ground for a little portion of the gait cycle.

9. CONCLUSION

In this study, comparison and selection of the necessary materials were made for an active ankle-foot orthosis prototype. A computer-aided design of the aAFO was developed in SolidWorks environment. Parts were purchased or manufactured using CNC machining, and a prototype was built. The aAFO prototype consists of a series elastic actuator, an orthotic shoe, and a lever mechanism for the SEA and shoe connection. SEA comprises of a brushed DC motor, a linear compression spring, ball screw and coupling. Developed active AFO prototype was modeled as two DOF mass-spring system driven by a compliant actuator system. It was assumed that there is an unknown disturbance caused by ground reaction forces and weight of the foot. The unknown disturbances were represented as a sum of sinusoidal signals and a constant. Moreover, an adaptive control algorithm was designed with the use of backstepping methods. The designed adaptive backstepping controller and a PID controller were implemented in LabVIEW environment. Numerical simulations run under eight cases were performed to compare performances of the controllers. Several experiments including step response, torque output, trajectory tracking, stride frequency, and cardiopulmonary exercise test were conducted using the active AFO prototype. Simulation and experimental results show that the overall system satisfies the design requirements and the proposed adaptive controller is better than a PID controller in trajectory tracking under different conditions.

9.1. Contributions and Originality

In the developed ankle-foot orthosis prototype, ankle joint degrees-of-freedom in three anatomical planes are allowed with the use of a ball-and-socket joint as an advantage to previous designs. Additionally, synchronization with the gait cycle and adaptation to different stride frequencies are provided through a single force sensor. The adaptability to different user weights is also possible with the disturbance parametrization.

Overall, the evaluation results prove the superiority of the proposed adaptive controller over the PID controller in terms of speed and trajectory tracking. The novelty of this study is the application of the designed backstepping algorithm for an active ankle-foot orthosis and performance comparison of the controller with a PID controller.

A poster presentation about the design and control of the active ankle-foot orthosis was delivered on October 6, 2016 at the Cybathlon Symposium in Zurich, Switzerland. Additionally, the paper about the backstepping control design was presented on December 12, 2017 at the 56th IEEE Conference on Decision and Control in Melbourne, Australia. An extended version of the conference paper including the implementation and evaluation of the control algorithm was submitted to an international journal. Finally, a paper on the design and evaluation of the active ankle-foot orthosis was submitted to a national journal.

9.2. Future Work

In this study, there is no intention to claim that designed controller performs better than impedance control approach. Performance comparison of the designed controller with an impedance controller algorithm could be considered in future. Currently, the system needs to be tethered to a power source. An untethered control system relying on lightweight and powerful battery packs and also, the design of an embedded control system will be goals for the future developments.

REFERENCES

1. Herr, H., “Exoskeletons and orthoses: classification, design challenges and future directions”, *Journal of NeuroEngineering and Rehabilitation*, Vol. 6, No. 1, p. 21, 2009.

2. Jimenez-Fabian, R. and O. Verlinden, “Review of control algorithms for robotic ankle systems in lower-limb orthoses, prostheses, and exoskeletons”, *Medical engineering & physics*, Vol. 34, No. 4, pp. 397–408, 2012.

3. Tucker, M. R., 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*, Vol. 12, No. 1, p. 1, 2015.

4. Neumann, D. A., *Kinesiology of the Musculoskeletal System-E-Book: Foundations for Rehabilitation*, Elsevier Health Sciences, 2013.

5. Richards, J., *Biomechanics in clinic and research*, Churchill Livingstone, 2008.

6. Shorter, K. A., J. Xia, E. T. Hsiao-Wecksler, W. K. Durfee and G. F. Kogler, “Technologies for powered ankle-foot orthotic systems: Possibilities and challenges”, *IEEE/ASME Transactions on Mechatronics*, Vol. 18, No. 1, pp. 337–347, 2013.

7. Park, Y.-L., B.-r. Chen, D. Young, L. Stirling, R. J. Wood, E. Goldfield and R. Nagpal, “Bio-inspired active soft orthotic device for ankle foot pathologies”, *Intelligent Robots and Systems (IROS), 2011 IEEE/RSJ International Conference on*, pp. 4488–4495, IEEE, 2011.

8. Yagn, N., “Apparatus for facilitating walking”, Jan. 28 1890, US Patent 420,179.

9. Van den Bogert, A. J., “Exotendons for assistance of human locomotion”, *Biomed-*

ical engineering online, Vol. 2, No. 1, p. 17, 2003.

10. Banala, S. K., S. K. Agrawal, A. Fattah, V. Krishnamoorthy, W.-L. Hsu, J. Scholz and K. Rudolph, “Gravity-balancing leg orthosis and its performance evaluation”, *IEEE Transactions on robotics*, Vol. 22, No. 6, pp. 1228–1239, 2006.

11. Smits, J., “Ankle-foot orthosis”, May 3 2005, US Patent 6,887,213.

12. Ferris, D. P., J. M. Czerniecki and B. Hannaford, “An ankle-foot orthosis powered by artificial pneumatic muscles”, *Journal of applied biomechanics*, Vol. 21, No. 2, pp. 189–197, 2005.

13. Blaya, J. A. and H. Herr, “Adaptive Control of a Variable-Impedance Ankle-Foot Orthosis to Assist Drop-Foot Gait”, *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, Vol. 12, No. 1, pp. 24–31, 2004.

14. Jardim, B. and A. A. Siqueira, “Development of series elastic actuators for impedance control of an active ankle foot orthosis”, *20th International Congress of Mechanical Engineering*, pp. 15–20, 2009.

15. Agrawal, A., S. K. Banala, S. K. Agrawal and S. A. Binder-Macleod, “Design of a two degree-of-freedom ankle-foot orthosis for robotic rehabilitation”, *Proceedings of the 2005 IEEE 9th International Conference on Rehabilitation Robotics*, Vol. 2005, pp. 41–44, 2005.

16. Sawicki, G. S. and D. P. Ferris, “A pneumatically powered knee-ankle-foot orthosis (KAFO) with myoelectric activation and inhibition”, *Journal of neuroengineering and rehabilitation*, Vol. 6, No. 1, pp. 1–16, 2009.

17. Shorter, K. A., Y. Li, T. Bretl and E. T. Hsiao-Wecksler, “Modeling, control, and analysis of a robotic assist device”, *Mechatronics*, Vol. 22, No. 8, pp. 1067–1077, 2012.

18. Vanderborght, B., R. Van Ham, D. Lefeber, T. G. Sugar and K. W. Hollander, "Comparison of mechanical design and energy consumption of adaptable, passive-compliant actuators", *The International Journal of Robotics Research*, Vol. 28, No. 1, pp. 90–103, 2009.

19. Au, S. K., P. Dilworth and H. Herr, "An ankle-foot emulation system for the study of human walking biomechanics", *Robotics and Automation, 2006. ICRA 2006. Proceedings 2006 IEEE International Conference on*, pp. 2939–2945, IEEE, 2006.

20. Dariush, B., "Active control of an ankle-foot orthosis", Jan. 19 2010, US Patent 7,650,204.

21. Eilenberg, M. F., 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*, Vol. 18, No. 2, pp. 164–173, 2010.

22. Hitt, J., A. M. Oymagil, T. Sugar, K. Hollander, A. Boehler and J. Fleeger, "Dynamically controlled ankle-foot orthosis (DCO) with regenerative kinetics: Incrementally attaining user portability", *Robotics and Automation, 2007 IEEE International Conference on*, pp. 1541–1546, IEEE, 2007.

23. Holgate, M. A., A. W. Bohler and T. G. Suga, "Control algorithms for ankle robots: A reflection on the state-of-the-art and presentation of two novel algorithms", *Biomedical Robotics and Biomechatronics, 2008. BioRob 2008. 2nd IEEE RAS & EMBS International Conference on*, pp. 97–102, IEEE, 2008.

24. Meng, W., Q. Liu, Z. Zhou, Q. Ai, B. Sheng and S. S. Xie, "Recent development of mechanisms and control strategies for robot-assisted lower limb rehabilitation", *Mechatronics*, Vol. 31, pp. 132–145, 2015.

25. Carson, M., M. Harrington, N. Thompson, J. O'connor and T. Theologis, "Kinematic analysis of a multi-segment foot model for research and clinical applications: a repeatability analysis", *Journal of biomechanics*, Vol. 34, No. 10, pp. 1299–1307,

2001.

26. Savas, Y., O. Kirtas, H. I. Basturk and E. Samur, “A backstepping control design for an active ankle-foot orthosis”, *Decision and Control, 2017. Proceedings. 56th IEEE Conference on*, pp. 262–267, IEEE, 2017.

27. Nikiforov, V. O., “Observers of external deterministic disturbances. I. objects with known parameters”, *Automation and Remote Control*, Vol. 65, No. 10, pp. 1531–1541, 2004.

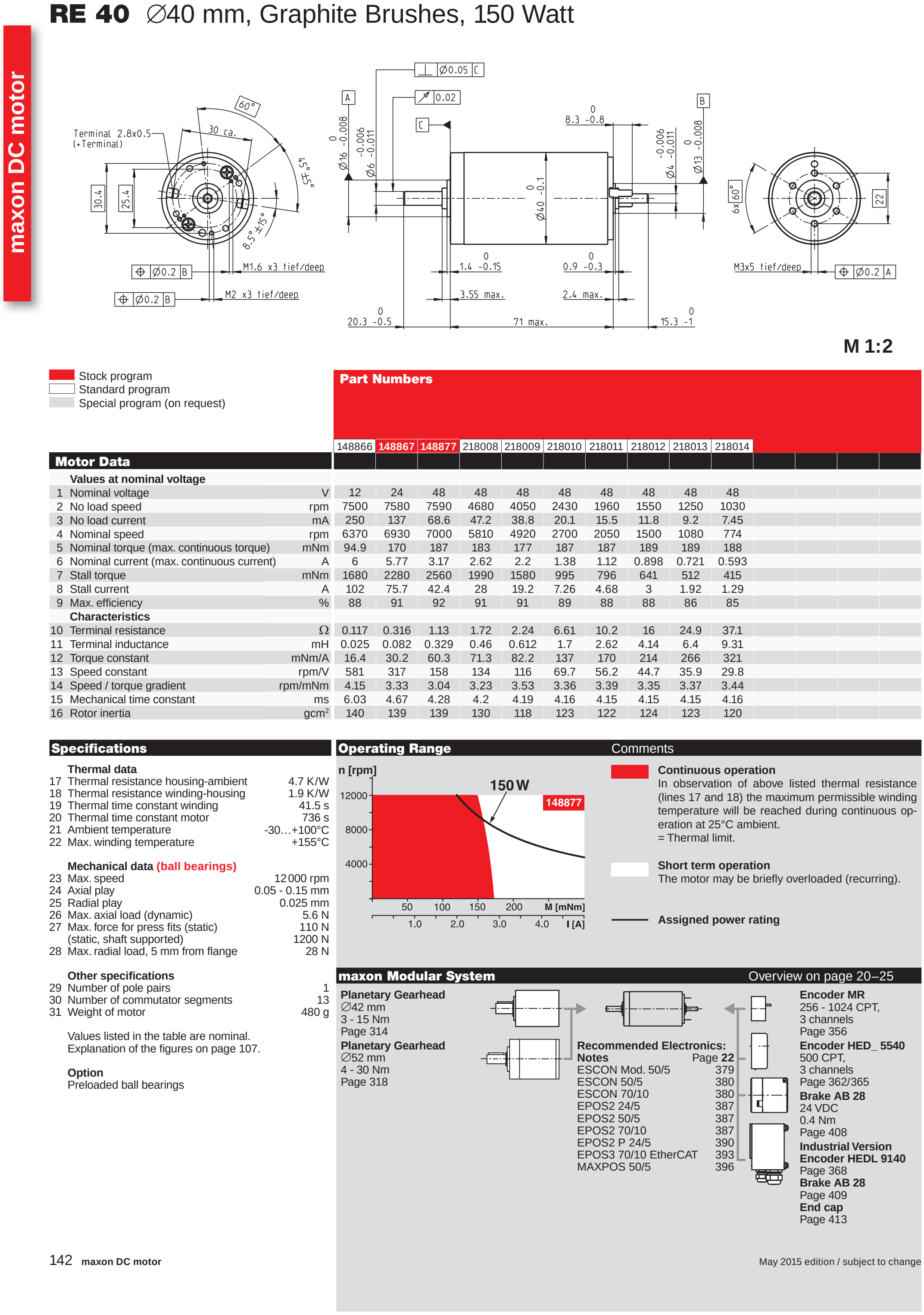
28. Chen, C., “Linear System Theory and Design”, *Holt, Rinehart and Winston, New York*, p. 572, 1984.

29. Krstic, M., I. Kanellakopoulos and P. V. Kokotovic, *Nonlinear and adaptive control design*, Wiley, 1995.

30. Oymagil, A. M., J. K. Hitt, T. Sugar and J. Fleeger, “Control of a regenerative braking powered ankle foot orthosis”, *Rehabilitation robotics, 2007. ICORR 2007. IEEE 10th international conference on*, pp. 28–34, IEEE, 2007.

31. Arnez-Paniagua, V., H. Rifaï, Y. Amirat and S. Mohammed, “Adaptive control of an actuated-ankle-foot-orthosis”, *Rehabilitation Robotics (ICORR), 2017 International Conference on*, pp. 1584–1589, IEEE, 2017.

APPENDIX A: DATASHEETS



maxon DC motor

RE 40 ∅40 mm, Graphite Brushes, 150 Watt

M 1:2

🟥 Stock program
☐ Standard program
▨ Special program (on request)

Part Numbers

Motor Data		148866	148867	148877	218008	218009	218010	218011	218012	218013	218014
Values at nominal voltage											
1 Nominal voltage	V	12	24	48	48	48	48	48	48	48	48
2 No load speed	rpm	7500	7580	7590	4680	4050	2430	1960	1550	1250	1030
3 No load current	mA	250	137	68.6	47.2	38.8	20.1	15.5	11.8	9.2	7.45
4 Nominal speed	rpm	6370	6930	7000	5810	4920	2700	2050	1500	1080	774
5 Nominal torque (max. continuous torque)	mNm	94.9	170	187	183	177	187	187	189	189	188
6 Nominal current (max. continuous current)	A	6	5.77	3.17	2.62	2.2	1.38	1.12	0.898	0.721	0.593
7 Stall torque	mNm	1680	2280	2560	1990	1580	995	796	641	512	415
8 Stall current	A	102	75.7	42.4	28	19.2	7.26	4.68	3	1.92	1.29
9 Max. efficiency	%	88	91	92	91	91	89	88	88	86	85
Characteristics											
10 Terminal resistance	Ω	0.117	0.316	1.13	1.72	2.24	6.61	10.2	16	24.9	37.1
11 Terminal inductance	mH	0.025	0.082	0.329	0.46	0.612	1.7	2.62	4.14	6.4	9.31
12 Torque constant	mNm/A	16.4	30.2	60.3	71.3	82.2	137	170	214	266	321
13 Speed constant	rpm/V	581	317	158	134	116	69.7	56.2	44.7	35.9	29.8
14 Speed / torque gradient	rpm/mNm	4.15	3.33	3.04	3.23	3.53	3.36	3.39	3.35	3.37	3.44
15 Mechanical time constant	ms	6.03	4.67	4.28	4.2	4.19	4.16	4.15	4.15	4.15	4.16
16 Rotor inertia	gcm^2	140	139	139	130	118	123	122	124	123	120

Specifications

Thermal data	
17 Thermal resistance housing-ambient	4.7 K/W
18 Thermal resistance winding-housing	1.9 K/W
19 Thermal time constant winding	41.5 s
20 Thermal time constant motor	736 s
21 Ambient temperature	-30…+100°C
22 Max. winding temperature	+155°C
Mechanical data (ball bearings)	
23 Max. speed	12000 rpm
24 Axial play	0.05 - 0.15 mm
25 Radial play	0.025 mm
26 Max. axial load (dynamic)	5.6 N
27 Max. force for press fits (static)	110 N
(static, shaft supported)	1200 N
28 Max. radial load, 5 mm from flange	28 N
Other specifications	
29 Number of pole pairs	1
30 Number of commutator segments	13
31 Weight of motor	480 g

Values listed in the table are nominal.
Explanation of the figures on page 107.

Option
Preloaded ball bearings

Operating Range

Comments

🟥 **Continuous operation**
In observation of above listed thermal resistance (lines 17 and 18) the maximum permissible winding temperature will be reached during continuous operation at 25°C ambient.
= Thermal limit.

☐ **Short term operation**
The motor may be briefly overloaded (recurring).

— **Assigned power rating**

maxon Modular System Overview on page 20–25

Planetary Gearhead
∅42 mm
3 - 15 Nm
Page 314

Planetary Gearhead
∅52 mm
4 - 30 Nm
Page 318

Recommended Electronics:	
Notes	Page **22**
ESCON Mod. 50/5	379
ESCON 50/5	380
ESCON 70/10	380
EPOS2 24/5	387
EPOS2 50/5	387
EPOS2 70/10	387
EPOS2 P 24/5	390
EPOS3 70/10 EtherCAT	393
MAXPOS 50/5	396

Encoder MR
256 - 1024 CPT,
3 channels
Page 356

Encoder HED_ 5540
500 CPT,
3 channels
Page 362/365

Brake AB 28
24 VDC
0.4 Nm
Page 408

Industrial Version
Encoder HEDL 9140
Page 368

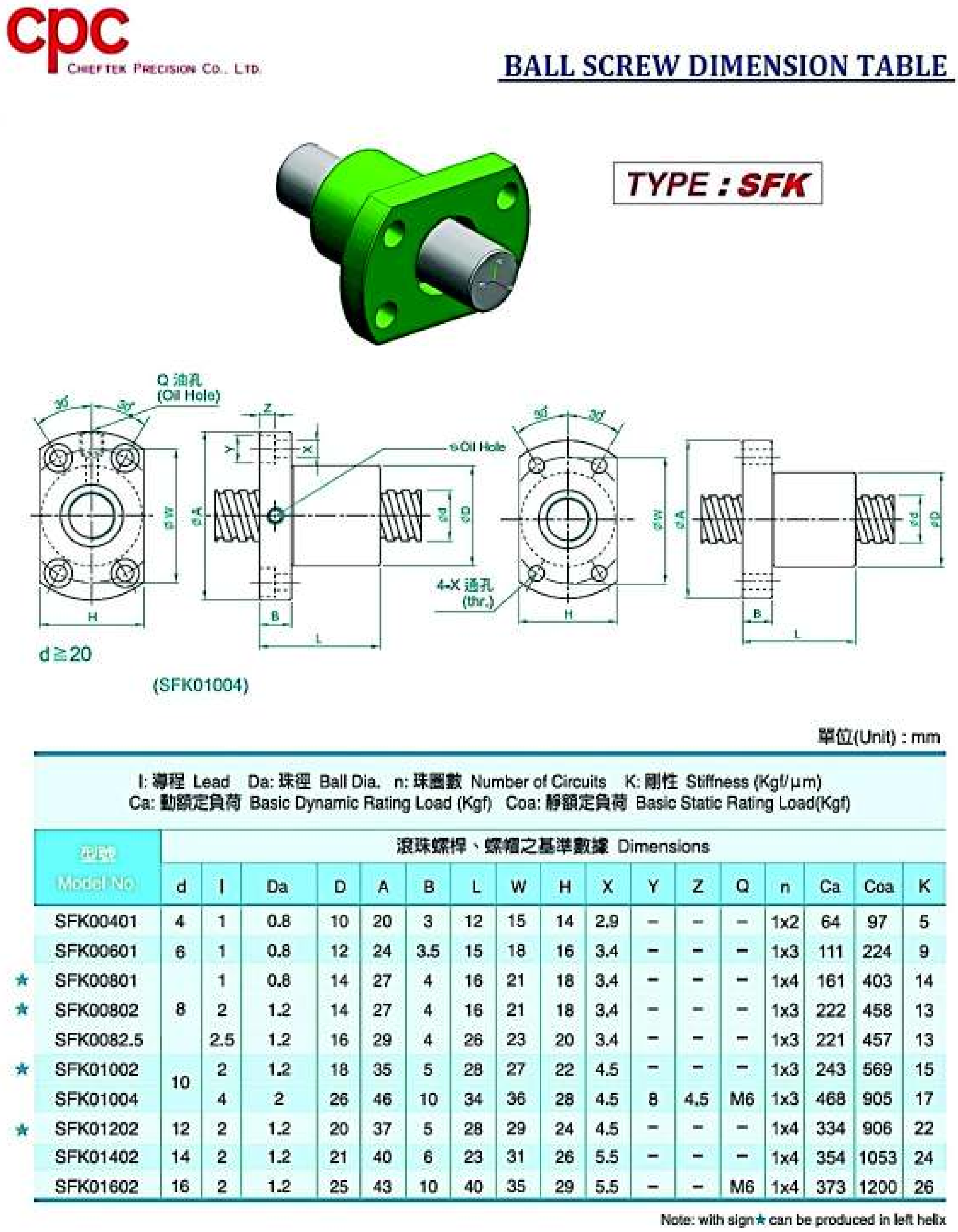
Brake AB 28
Page 409

End cap
Page 413

142 **maxon DC motor**

May 2015 edition / subject to change

Figure A.1. Maxon DC brushed motor (RE 40 150 Watt, 24 V) datasheet.



BALL SCREW DIMENSION TABLE

TYPE : SFK

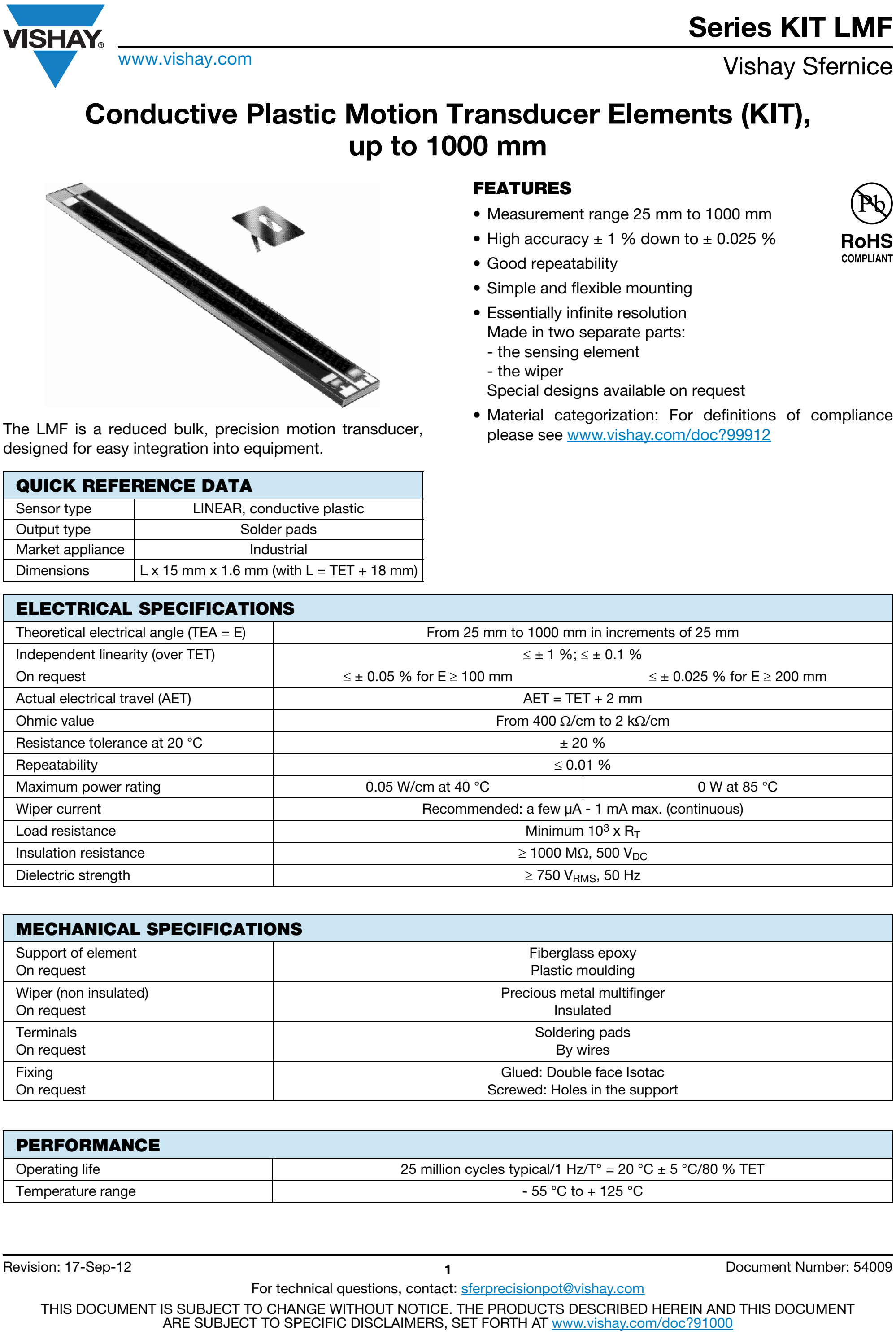
單位(Unit) : mm

l: 導程 Lead　Da: 珠徑 Ball Dia.　n: 珠圈數 Number of Circuits　K: 剛性 Stiffness (Kgf/μm)
Ca: 動額定負荷 Basic Dynamic Rating Load (Kgf)　Coa: 靜額定負荷 Basic Static Rating Load(Kgf)

	型號 Model No	滾珠螺桿、螺帽之基準數據 Dimensions																
		d	l	Da	D	A	B	L	W	H	X	Y	Z	Q	n	Ca	Coa	K
	SFK00401	4	1	0.8	10	20	3	12	15	14	2.9	–	–	–	1x2	64	97	5
	SFK00601	6	1	0.8	12	24	3.5	15	18	16	3.4	–	–	–	1x3	111	224	9
★	SFK00801	8	1	0.8	14	27	4	16	21	18	3.4	–	–	–	1x4	161	403	14
★	SFK00802		2	1.2	14	27	4	16	21	18	3.4	–	–	–	1x3	222	458	13
	SFK0082.5		2.5	1.2	16	29	4	26	23	20	3.4	–	–	–	1x3	221	457	13
★	SFK01002	10	2	1.2	18	35	5	28	27	22	4.5	–	–	–	1x3	243	569	15
	SFK01004		4	2	26	46	10	34	36	28	4.5	8	4.5	M6	1x3	468	905	17
★	SFK01202	12	2	1.2	20	37	5	28	29	24	4.5	–	–	–	1x4	334	906	22
	SFK01402	14	2	1.2	21	40	6	23	31	26	5.5	–	–	–	1x4	354	1053	24
	SFK01602	16	2	1.2	25	43	10	40	35	29	5.5	–	–	M6	1x4	373	1200	26

Note: with sign★ can be produced in left helix

Figure A.2. CPC Type SFK 08x02mm ball screw datasheet.



VISHAY.

www.vishay.com

Series KIT LMF

Vishay Sfernice

Conductive Plastic Motion Transducer Elements (KIT), up to 1000 mm

FEATURES

- Measurement range 25 mm to 1000 mm
- High accuracy ± 1 % down to ± 0.025 %
- Good repeatability
- Simple and flexible mounting
- Essentially infinite resolution
  Made in two separate parts:
  - the sensing element
  - the wiper
  Special designs available on request
- Material categorization: For definitions of compliance please see www.vishay.com/doc?99912

Pb RoHS COMPLIANT

The LMF is a reduced bulk, precision motion transducer, designed for easy integration into equipment.

QUICK REFERENCE DATA	
Sensor type	LINEAR, conductive plastic
Output type	Solder pads
Market appliance	Industrial
Dimensions	L x 15 mm x 1.6 mm (with L = TET + 18 mm)

ELECTRICAL SPECIFICATIONS		
Theoretical electrical angle (TEA = E)	From 25 mm to 1000 mm in increments of 25 mm	
Independent linearity (over TET)	≤ ± 1 %; ≤ ± 0.1 %	
On request	≤ ± 0.05 % for E ≥ 100 mm	≤ ± 0.025 % for E ≥ 200 mm
Actual electrical travel (AET)	AET = TET + 2 mm	
Ohmic value	From 400 Ω/cm to 2 kΩ/cm	
Resistance tolerance at 20 °C	± 20 %	
Repeatability	≤ 0.01 %	
Maximum power rating	0.05 W/cm at 40 °C	0 W at 85 °C
Wiper current	Recommended: a few µA - 1 mA max. (continuous)	
Load resistance	Minimum 10^3 x R_T	
Insulation resistance	≥ 1000 MΩ, 500 V_{DC}	
Dielectric strength	≥ 750 V_{RMS}, 50 Hz	

MECHANICAL SPECIFICATIONS	
Support of element On request	Fiberglass epoxy Plastic moulding
Wiper (non insulated) On request	Precious metal multifinger Insulated
Terminals On request	Soldering pads By wires
Fixing On request	Glued: Double face Isotac Screwed: Holes in the support

PERFORMANCE	
Operating life	25 million cycles typical/1 Hz/T° = 20 °C ± 5 °C/80 % TET
Temperature range	- 55 °C to + 125 °C

Revision: 17-Sep-12 1 Document Number: 54009

For technical questions, contact: sferprecisionpot@vishay.com

THIS DOCUMENT IS SUBJECT TO CHANGE WITHOUT NOTICE. THE PRODUCTS DESCRIBED HEREIN AND THIS DOCUMENT ARE SUBJECT TO SPECIFIC DISCLAIMERS, SET FORTH AT www.vishay.com/doc?91000

Figure A.3. Linear potentiometer datasheet.

APPENDIX B: LABVIEW CODES

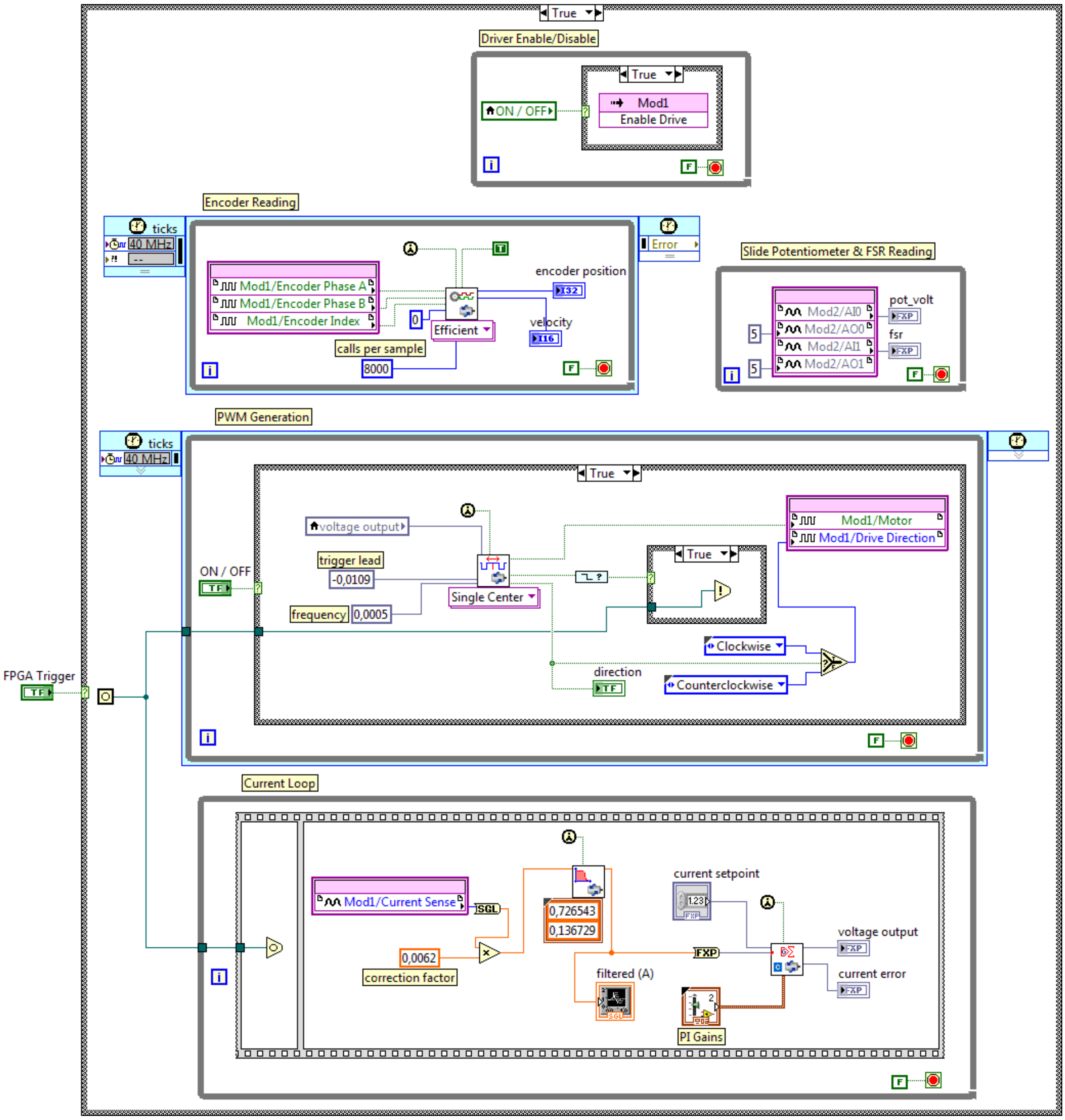


Figure B.1. LabVIEW code at the FPGA side. FPGA is used to read sensor values and generate pulse width modulation (PWM) signals to run the DC motor.

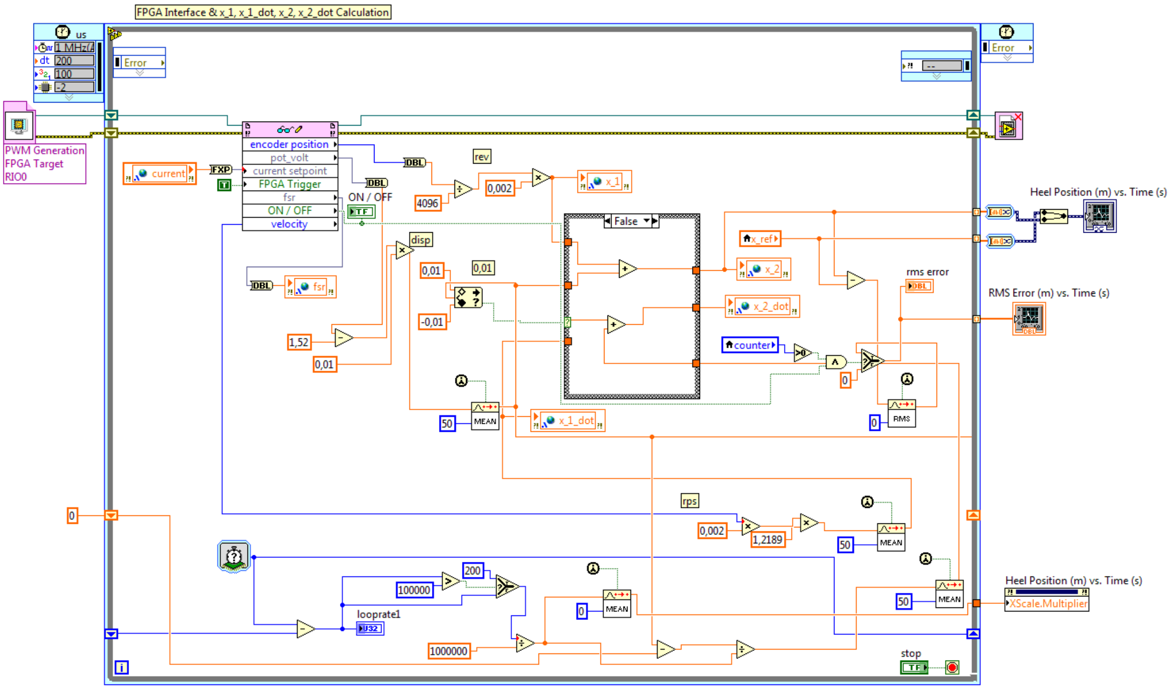
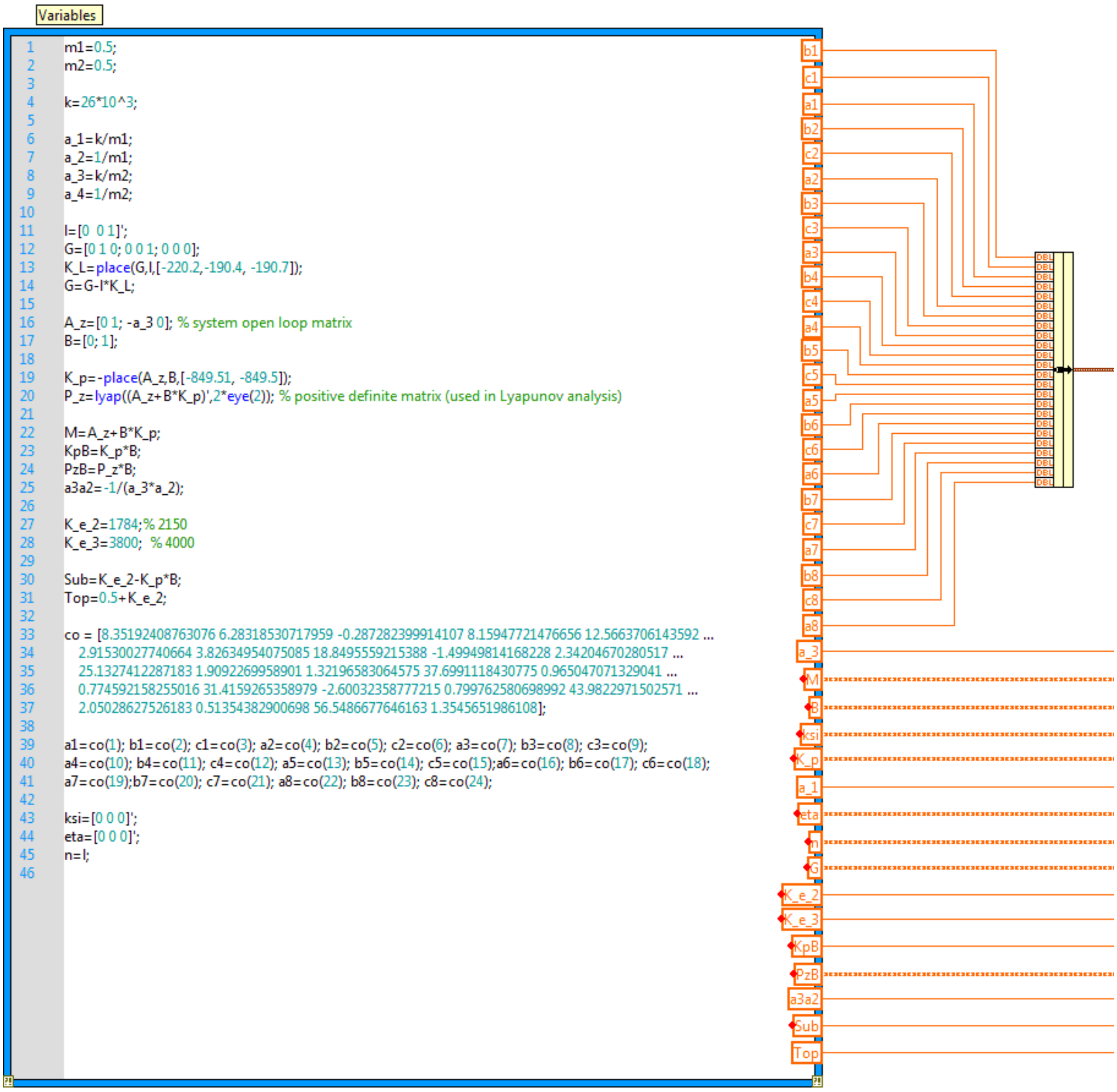


Figure B.2. FPGA interface and derivation of the position and velocity values in the real-time side.



```
m1=0.5;
m2=0.5;

k=26*10^3;

a_1=k/m1;
a_2=1/m1;
a_3=k/m2;
a_4=1/m2;

I=[0  0 1]';
G=[0 1 0; 0 0 1; 0 0 0];
K_L=place(G,I,[-220.2,-190.4, -190.7]);
G=G-I*K_L;

A_z=[0 1; -a_3 0]; % system open loop matrix
B=[0; 1];

K_p=-place(A_z,B,[-849.51, -849.5]);
P_z=lyap((A_z+B*K_p)',2*eye(2)); % positive definite matrix (used in Lyapunov analysis)

M=A_z+B*K_p;
KpB=K_p*B;
PzB=P_z*B;
a3a2=-1/(a_3*a_2);

K_e_2=1784;% 2150
K_e_3=3800;  % 4000

Sub=K_e_2-K_p*B;
Top=0.5+K_e_2;

co = [8.35192408763076 6.28318530717959 -0.287282399914107 8.15947721476656 12.5663706143592 ...
   2.91530027740664 3.82634954075085 18.8495559215388 -1.49949814168228 2.34204670280517 ...
   25.1327412287183 1.9092269958901 1.32196583064575 37.6991118430775 0.965047071329041 ...
   0.774592158255016 31.4159265358979 -2.60032358777215 0.799762580698992 43.9822971502571 ...
   2.05028627526183 0.51354382900698 56.5486677646163 1.3545651986108];

a1=co(1); b1=co(2); c1=co(3); a2=co(4); b2=co(5); c2=co(6); a3=co(7); b3=co(8); c3=co(9);
a4=co(10); b4=co(11); c4=co(12); a5=co(13); b5=co(14); c5=co(15);a6=co(16); b6=co(17); c6=co(18);
a7=co(19);b7=co(20); c7=co(21); a8=co(22); b8=co(23); c8=co(24);

ksi=[0 0 0]';
eta=[0 0 0]';
n=I;

```

Figure B.3. Definition of adaptive controller variables using MathScript node in the real-time side.

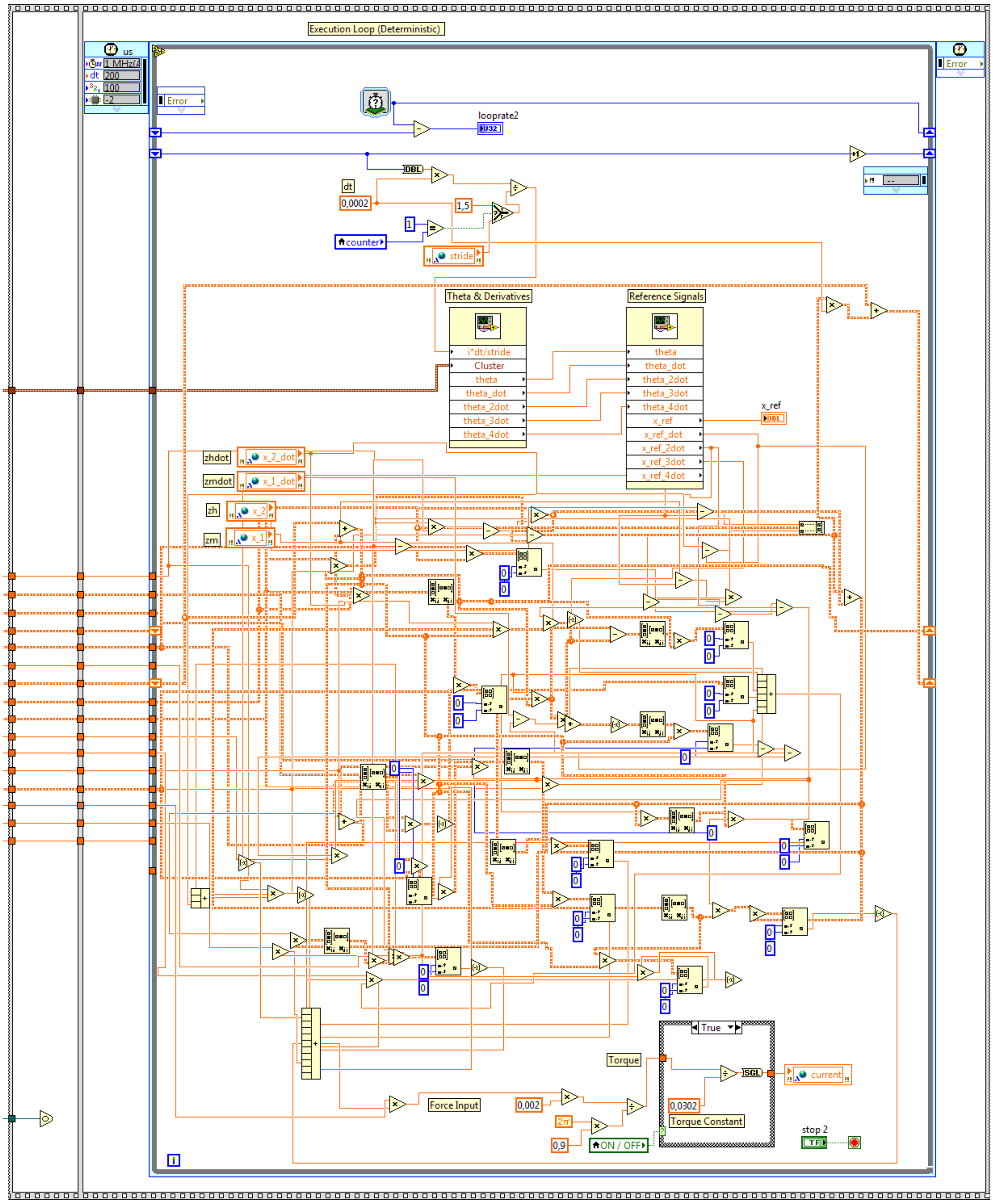
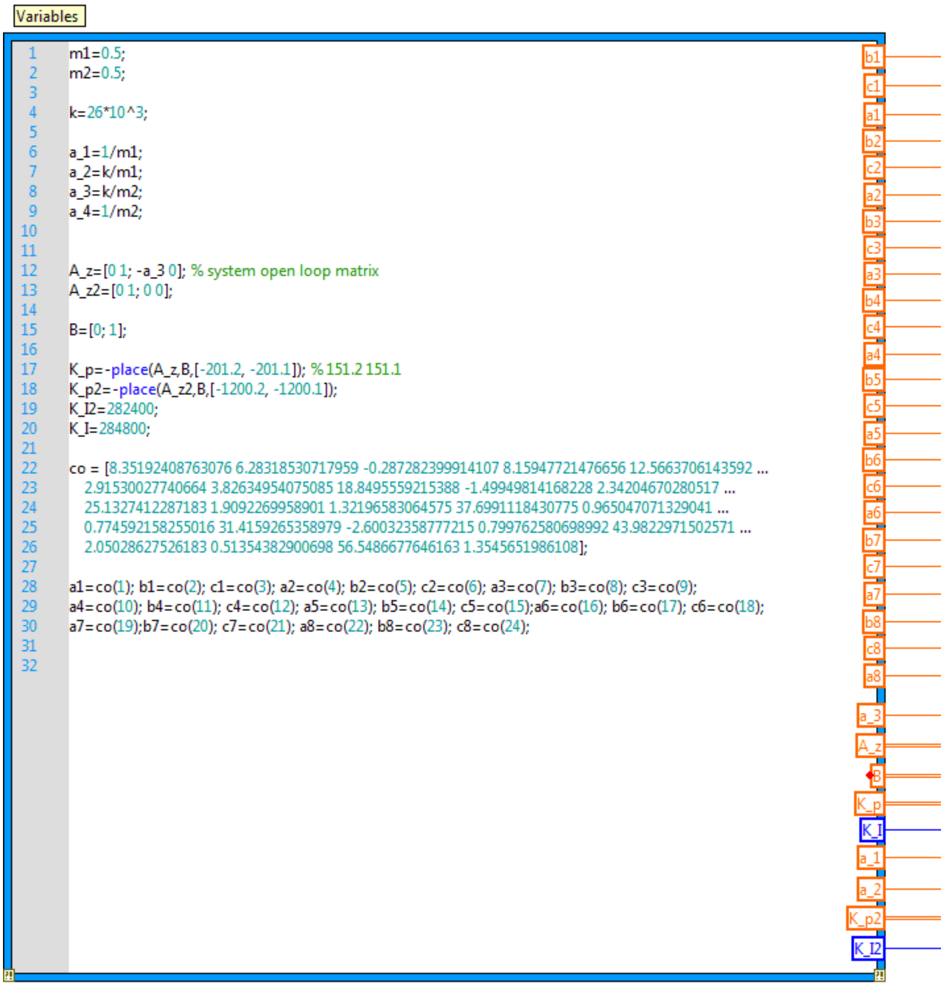
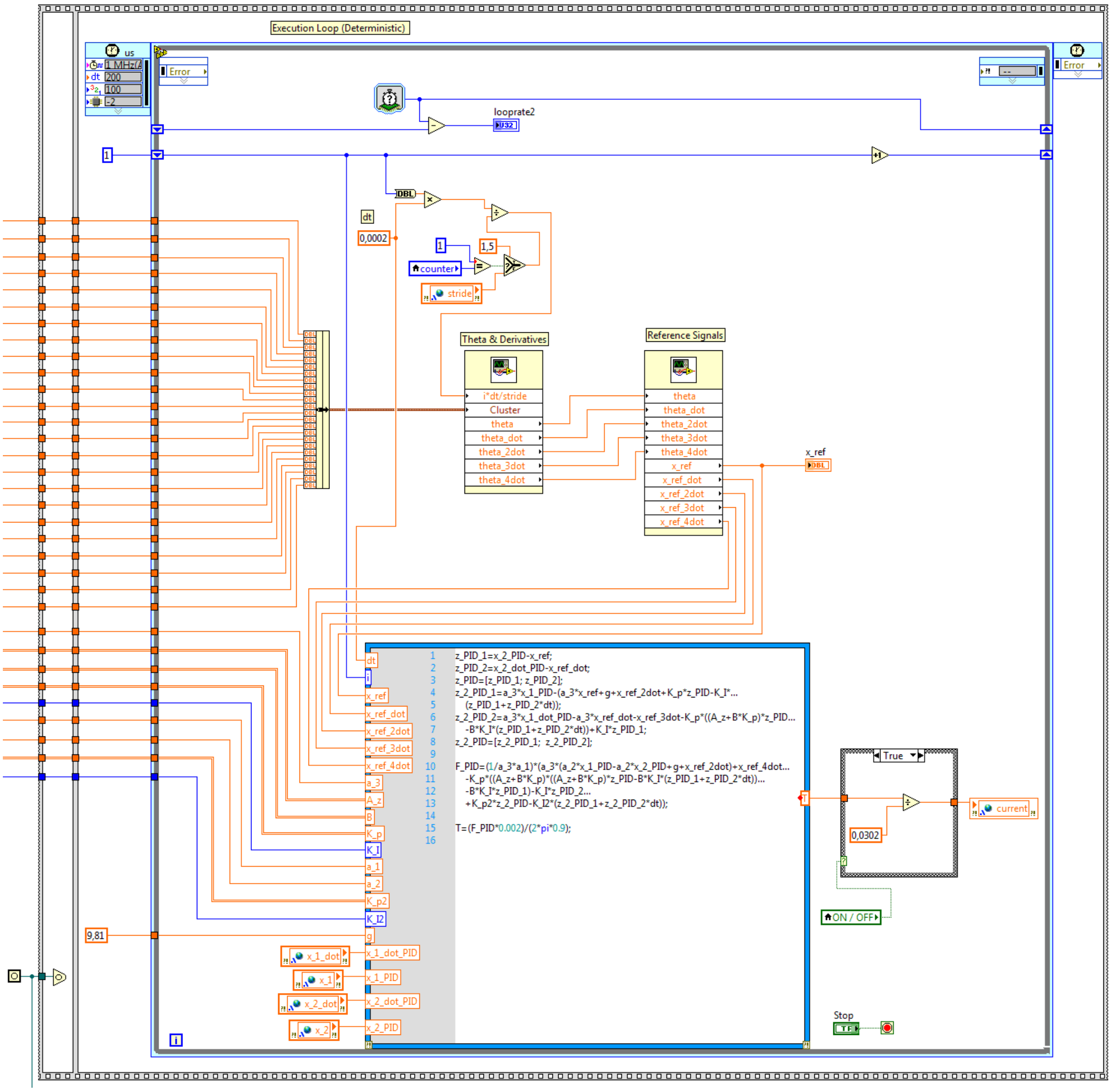


Figure B.4. Adaptive controller code in the real-time side.



```
m1=0.5;
m2=0.5;

k=26*10^3;

a_1=1/m1;
a_2=k/m1;
a_3=k/m2;
a_4=1/m2;


A_z=[0 1; -a_3 0]; % system open loop matrix
A_z2=[0 1; 0 0];

B=[0; 1];

K_p=-place(A_z,B,[-201.2, -201.1]); % 151.2 151.1
K_p2=-place(A_z2,B,[-1200.2, -1200.1]);
K_I2=282400;
K_I=284800;

co = [8.35192408763076 6.28318530717959 -0.287282399914107 8.15947721476656 12.5663706143592 ...
   2.91530027740664 3.82634954075085 18.8495559215388 -1.49949814168228 2.34204670280517 ...
   25.1327412287183 1.9092269958901 1.32196583064575 37.6991118430775 0.965047071329041 ...
   0.774592158255016 31.4159265358979 -2.60032358777215 0.799762580698992 43.9822971502571 ...
   2.05028627526183 0.51354382900698 56.5486677646163 1.3545651986108];

a1=co(1); b1=co(2); c1=co(3); a2=co(4); b2=co(5); c2=co(6); a3=co(7); b3=co(8); c3=co(9);
a4=co(10); b4=co(11); c4=co(12); a5=co(13); b5=co(14); c5=co(15);a6=co(16); b6=co(17); c6=co(18);
a7=co(19);b7=co(20); c7=co(21); a8=co(22); b8=co(23); c8=co(24);
```

Figure B.5. Definition of PID controller variables using MathScript node in the real-time side.



```
z_PID_1=x_2_PID-x_ref;
z_PID_2=x_2_dot_PID-x_ref_dot;
z_PID=[z_PID_1; z_PID_2];
z_2_PID_1=a_3*x_1_PID-(a_3*x_ref+g+x_ref_2dot+K_p*z_PID-K_I*...
   (z_PID_1+z_PID_2*dt));
z_2_PID_2=a_3*x_1_dot_PID-a_3*x_ref_dot-x_ref_3dot-K_p*((A_z+B*K_p)*z_PID...
   -B*K_I*(z_PID_1+z_PID_2*dt))+K_I*z_PID_1;
z_2_PID=[z_2_PID_1;  z_2_PID_2];

F_PID=(1/a_3*a_1)*(a_3*(a_2*x_1_PID-a_2*x_2_PID+g+x_ref_2dot)+x_ref_4dot...
   -K_p*((A_z+B*K_p)*((A_z+B*K_p)*z_PID-B*K_I*(z_PID_1+z_PID_2*dt))...
   -B*K_I*z_PID_1)-K_I*z_PID_2...
   +K_p2*z_2_PID-K_I2*(z_2_PID_1+z_2_PID_2*dt));

T=(F_PID*0.002)/(2*pi*0.9);
```

Figure B.6. PID controller code in the real-time side.

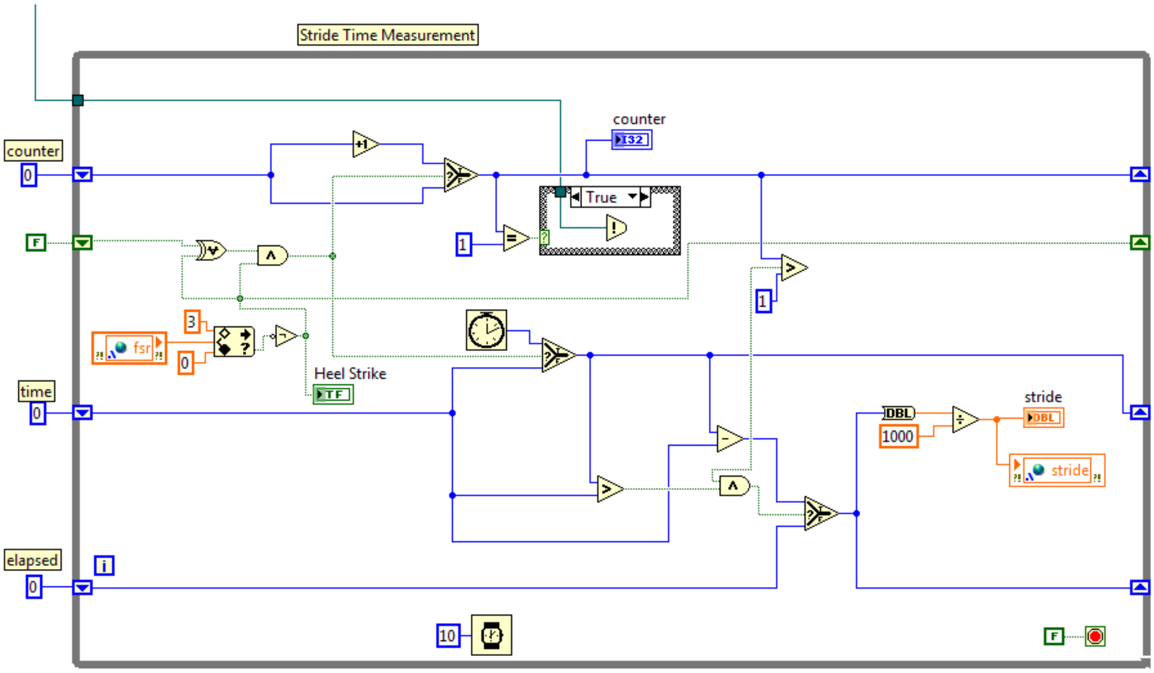


Figure B.7. Stride time measurement code in the real-time side.

APPENDIX C: MATLAB CODE

```
%Final AFO code, including both adaptive and PID controllers
 clear all
 close all
m1=0.5;
m2=0.5;
k=26*10^3;
 a_1=k/m1;
 a_2=1/m1;
 a_3=k/m2;
 a_4=1/m2;
g=9.81;
l=[0  0 1]';
G=[0 1 0; 0 0 1; 0 0 0];
K_L=place(G,l,[-220.2,-190.4, -190.7]);
G=G-l*K_L;
A_z=[0 1; -a_3 0]; % system open loop matrix
B=[0 1]';
K_p=-place(A_z,B,[-850.9, -850.5]); % determine K_p and K_d values using pole placement
P_z=lyap((A_z+B*K_p)',2*eye(2)); % positive definite matrix (used in Lyapunov analysis)
K_e_2=1800;
K_e_3=3800;

A_z2=[0 1; 0 0];
B=[0 1]';
K_p_PID=-place(A_z,B,[-201.2, -201.1]);
K_p2=-place(A_z2,B,[-1200.2, -1200.1]);
K_I2=282400;
```

Figure C.1. MATLAB code for the simulations

```
K_I=284800;
z_PID=[0,0]';
z_2_PID=[0,0]';
x_1_PID=0;
x_1_dot_PID=0;
x_1_ddot_PID=0;
x_2_PID=0;
x_2_dot_PID=0;
x_2_ddot_PID=0;
F_PID=0;
a_1_PID=1/m1;
a_2_PID=k/m1;

grfconst=1;
weight=80;
stride=1;
cycle=0;

cog = [0.466224305239823 5.23598775598299 -0.041046266444951 0.204665281262301 20.943951023932...
    -2.72704287741529 0.0801005769710066 31.4159265358979 -2.02039314975654 0.562893746515834...
    10.471975511966 2.89384616946423];
d1=cog(1); e1=cog(2); f1=cog(3); d2=cog(4); e2=cog(5); f2=cog(6); d3=cog(7); e3=cog(8); f3=cog(9);
d4=cog(10); e4=cog(11); f4=cog(12);

co = [8.35192408763076 6.28318530717959 -0.287282399914107 8.15947721476656 12.5663706143592 ...
    2.91530027740664 3.82634954075085 18.8495559215388 -1.49949814168228 2.34204670280517 ...
```

Figure C.1. MATLAB code for the simulations (cont.)

```
    25.1327412287183 1.9092269958901 1.32196583064575 37.6991118430775 0.965047071329041 ...
    0.774592158255016 31.4159265358979 -2.60032358777215 0.799762580698992 43.9822971502571 ...
    2.05028627526183 0.51354382900698 56.5486677646163 1.3545651986108];
a1=co(1); b1=co(2); c1=co(3); a2=co(4); b2=co(5); c2=co(6); a3=co(7); b3=co(8); c3=co(9);
a4=co(10); b4=co(11); c4=co(12); a5=co(13); b5=co(14); c5=co(15);a6=co(16); b6=co(17); c6=co(18);
a7=co(19);b7=co(20); c7=co(21); a8=co(22); b8=co(23); c8=co(24);

% initialize all states for open and closed loop
z_m=0;
z_m_dot=0;
z_m_ddot=0;
z_m_open=0;
z_m_dot_open=0;
z_m_ddot_open=0;
z_h=0;
z_h_dot=0;
z_h_ddot=0;
z_h_open=0;
z_h_dot_open=0;
z_h_ddot_open=0;
z=[0.005,0.005]';

% initialize disturbance estimation and update laws
theta_hat=[0 0 0]';
theta_hat_dot=[0 0 0]';
```

Figure C.1. MATLAB code for the simulations (cont.)

```
ksi=[0 0 0]';
eta=[0 0 0]';
n=1;
gamma_theta=0.00000000001;
gamma_d=gamma_theta;

t_ini=0; %t initial
t_final=stride*3; %t final
dt=.00001; %time interval

t=t_ini:dt:t_final; %time matrix
y_l=0;
y_l_dot=0;
a=0.55;
KK=1;
% generate zero matrices to keep iteration values
D=zeros(size(t,2),4);
D_real=zeros(size(t));
X_1=zeros(size(t));
X_1_PID=zeros(size(t));
X_2=zeros(size(t));
X_2_PID=zeros(size(t));
X_2_open=zeros(size(t));
Z=zeros(size(t));
F_Vector=zeros(size(t));
F_Vector_PID=zeros(size(t));
F=0; % initialize force
for i=1:t_final/dt
    if (i*dt-cycle*stride)<=0.6*stride
        d = (grfconst*weight/-0.086)*(d1*sin(e1*((i*dt-stride*cycle)/stride)+f1) + ...
```

Figure C.1. MATLAB code for the simulations (cont.)

```
            d2*sin(e2*((i*dt-stride*cycle)/stride)+f2) + d3*sin(e3*((i*dt-stride*cycle)/stride)+f3) + ...
            d4*sin(e4*((i*dt-stride*cycle)/stride)+f4));
    else
        d=10;
    end

    theta=a1*sin(b1*i*dt/stride+c1) + a2*sin(b2*i*dt/stride+c2) + a3*sin(b3*i*dt/stride+c3) + ...
        a4*sin(b4*i*dt/stride+c4) + a5*sin(b5*i*dt/stride+c5) + a6*sin(b6*i*dt/stride+c6) + ...
        a7*sin(b7*i*dt/stride+c7) + a8*sin(b8*i*dt/stride+c8);
    theta_dot=a1*b1*cos(b1*i*dt/stride+c1) + a2*b2*cos(b2*i*dt/stride+c2) + a3*b3*cos(b3*i*dt/stride+c3) + ...
        a4*b4*cos(b4*i*dt/stride+c4) + a5*b5*cos(b5*i*dt/stride+c5) + a6*b6*cos(b6*i*dt/stride+c6) + ...
        a7*b7*cos(b7*i*dt/stride+c7) + a8*b8*cos(b8*i*dt/stride+c8);
    theta_2dot=-1*(a1*b1^2*sin(b1*i*dt/stride+c1) + a2*b2^2*sin(b2*i*dt/stride+c2) + ...
        a3*b3^2*sin(b3*i*dt/stride+c3) + a4*b4^2*sin(b4*i*dt/stride+c4) + a5*b5^2*sin(b5*i*dt/stride+c5)+ ...
        a6*b6^2*sin(b6*i*dt/stride+c6) + a7*b7^2*sin(b7*i*dt/stride+c7) + a8*b8^2*sin(b8*i*dt/stride+c8));
    theta_3dot=-1*(a1*b1^3*cos(b1*i*dt/stride+c1) + a2*b2^3*cos(b2*i*dt/stride+c2) + ...
        a3*b3^3*cos(b3*i*dt/stride+c3) + a4*b4^3*cos(b4*i*dt/stride+c4) ...
       + a5*b5^3*cos(b5*i*dt/stride+c5) + a6*b6^3*cos(b6*i*dt/stride+c6) + ...
```

Figure C.1. MATLAB code for the simulations (cont.)

```
        a7*b7^3*cos(b7*i*dt/stride+c7) + a8*b8^3*cos(b8*i*dt/stride+c8));
    theta_4dot=a1*b1^4*sin(b1*i*dt/stride+c1) + a2*b2^4*sin(b2*i*dt/stride+c2) + ...
        a3*b3^4*sin(b3*i*dt/stride+c3) + a4*b4^4*sin(b4*i*dt/stride+c4) ...
        + a5*b5^4*sin(b5*i*dt/stride+c5) + a6*b6^4*sin(b6*i*dt/stride+c6) + ...
        a7*b7^4*sin(b7*i*dt/stride+c7) + a8*b8^4*sin(b8*i*dt/stride+c8);

    z_h_ref=-0.086*sin(theta*(pi/180));
    z_h_ref_dot=-0.086*(pi/180)*theta_dot*cos(theta*(pi/180));
    z_h_ref_ddot=-0.086*(pi/180)*(theta_2dot*cos(theta*(pi/180))-theta_dot^2*sin(theta*(pi/180))*(pi/180));
    z_h_ref_dddot=-0.086*(pi/180)*(theta_3dot*cos(theta*(pi/180))-theta_2dot*theta_dot*(pi/60)...
        *sin(theta*(pi/180))-(pi/180)^2*theta_dot^3*cos(theta*(pi/180)));
    z_h_ref_ddddot=-(0.00150098*theta_4dot*cos(theta*(pi/180))-0.0000785913*theta_2dot^2*sin(theta*(pi/180))...
        +7.98009e-9*theta_dot^4*sin(theta*(pi/180))-0.000104788*theta_3dot*theta_dot*sin(theta*(pi/180))...
        -2.74335e-6*theta_dot^2*theta_2dot*cos(theta*(pi/180)));

    ksi=eta+n*z_h_dot;
    eta_dot=G*(eta+n*z_h_dot)-n*(-g+a_3*(z_m-z_h));
    z(1)=z_h-z_h_ref; % state tracking error
```

Figure C.1. MATLAB code for the simulations (cont.)

```
    z(2)=z_h_dot-z_h_ref_dot; % derivative of state tracking error
    e_2=a_3*z_m-g+theta_hat'*ksi-a_3*z_h_ref-z_h_ref_ddot-K_p*z;
    e_3=a_3*z_m_dot-a_3*z_h_ref_dot-z_h_ref_dddot+(z'*P_z*B*ksi...
       +e_2*(theta_hat'*n*ksi-K_p*B*ksi))'*ksi+theta_hat'*(G*eta+G*n*z_h_dot+n*theta_hat'*ksi)...
       -K_p*((A_z+B*K_p)*z+B*e_2)+e_2*(0.5+(theta_hat'*theta_hat)/2+K_e_2)+z'*P_z*B;

    ksi_constant=eta_dot+n*(-g+a_3*(z_m-z_h)+theta_hat'*ksi);
    e_2_constant=(theta_hat_dot-(z'*P_z*B*ksi-e_2*(theta_hat'*n*ksi-K_p*B*ksi)))'*ksi...
       -z'*P_z*B-e_2*((theta_hat'*theta_hat)+K_e_2);
    youngs_term=10^-1*((z'*P_z*B*ksi)'*(z'*P_z*B*ksi)+(theta_hat'*n*ksi)'*(theta_hat'*n*ksi)...
       +((theta_hat'*n*ksi)'*(theta_hat'*n*ksi))^2+((theta_hat'*n*ksi)'*ksi)^2+ksi'*ksi...
       +(ksi'*ksi)^2+(theta_hat'*theta_hat)+(theta_hat'*theta_hat)^2+(theta_hat'*theta_hat)^3+(theta_hat'*n)^4);
    correction_term=((z'*P_z*B*ksi)'*n+((z'*P_z*B*n)'*ksi)+e_2*((theta_hat'*n*ksi)'*n)...
       +e_2*(((theta_hat'*n)*n)'*ksi)+(theta_hat'*n*ksi)'*(theta_hat'*n*ksi-K_p*B*ksi)...
       -e_2*((K_p*B*ksi)'*n)-e_2*((K_p*B*n)'*ksi)-(K_p*B*ksi)'*(theta_hat'*n*ksi-K_p*B*ksi))...
       +theta_hat'*G*n+theta_hat'*n*theta_hat'*n-K_p*(A_z+B*K_p)*B...
```

Figure C.1. MATLAB code for the simulations (cont.)

```
        +(theta_hat'*theta_hat+K_e_2-K_p*B)*(theta_hat'*n-K_p*B)+(B'*P_z*B);

    % Adaptive control law for force input F
    F=-1*(1/a_3*a_2)*(a_3*(-g-a_1*(z_m-z_h))-a_3*z_h_ref_ddot-z_h_ref_ddddot...
        +(z'*P_z*B*ksi)'*(ksi_constant)+(z'*P_z*B*ksi_constant)'*ksi...
        +(((A_z+B*K_p)*z+B*e_2)'*P_z*B*ksi)'*ksi+e_2*(theta_hat'*n*ksi)'*ksi_constant...
        +e_2*(theta_hat_dot'*n*ksi)'*ksi+e_2*(theta_hat'*n*ksi_constant)'*ksi...
        +(theta_hat'*n*ksi)'*ksi*e_2_constant-e_2*(K_p*B*ksi)'*ksi_constant...
        -e_2*(K_p*B*ksi_constant)'*ksi-(K_p*B*ksi)'*ksi*e_2_constant...
        +theta_hat_dot'*(G*eta+G*n*z_h_dot+n*theta_hat'*ksi)+theta_hat'*G*eta_dot...
        +theta_hat'*G*n*(-g+a_3*(z_m*z_h)+theta_hat'*ksi)+theta_hat'*n*theta_hat'*ksi_constant...
        +theta_hat'*n*theta_hat_dot'*ksi+(theta_hat'*theta_hat+K_e_2-K_p*B)*e_2_constant...
        +theta_hat_dot'*theta_hat*e_2+((A_z+B*K_p)*z+B*e_2)'*P_z*B+K_e_3*e_3+e_2*correction_term);

    %%%%%%%%%% FILTER %%%%%%%%%%%%
    % Apply filter only when disturbance has discontinuities and sudden
    % amplitude changes. i.e. in 60% and 100% gait cycle.
    %%% a= 0.55, KK=1
     if i*dt==stride-0.405 || i*dt==stride-0.005
```

Figure C.1. MATLAB code for the simulations (cont.)

```
        y_l=F;
    end

    if i*dt>=stride -0.405
        y_l_dot=-y_l*a+F*KK;
        y_l=y_l+y_l_dot*i*dt;
        F=y_l;

        if (i*dt>cycle*0.595 && i*dt<cycle*0.605) || (i*dt>cycle*0.995 && i*dt<cycle*1.005)
            if abs(F)>450
                F=sign(F)*450;
            end
        end
    end
    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%


    %PID

    z_PID(1)=x_2_PID-z_h_ref;
    z_PID(2)=x_2_dot_PID-z_h_ref_dot;
    z_2_PID(1)=a_3*x_1_PID-(a_3*z_h_ref+g+z_h_ref_ddot+K_p_PID*z_PID-K_I*(z_PID(1)+z_PID(2)*dt));
    z_2_PID(2)=a_3*x_1_dot_PID-a_3*z_h_ref_dot-z_h_ref_dddot-K_p_PID*((A_z+B*K_p_PID)*z_PID-B*K_I*(z_PID(1)+z_PID(2)*dt))+K_I*z_PID(1);

    F_PID=(1/a_3*a_1_PID)*(a_3*(a_2_PID*x_1_PID-a_2_PID*x_2_PID+g+z_h_ref_ddot)+z_h_ref_ddddot...
```

Figure C.1. MATLAB code for the simulations (cont.)

```
          -K_p_PID*((A_z+B*K_p_PID)*((A_z+B*K_p_PID)*z_PID-B*K_I*(z_PID(1)+z_PID(2)*dt))-B*K_I*z_PID(1))-K_I*z_PID(2)...
          +K_p2*z_2_PID-K_I2*(z_2_PID(1)+z_2_PID(2)*dt));

     % Update law for disturbance estimation

     theta_hat_dot=gamma_d*(z'*P_z*B*ksi+e_2*(theta_hat'*n*ksi-K_p*B*ksi)...
          +e_3*(correction_term)*ksi);

     % system dynamics in closed loop
     z_m_ddot=-g-a_1*(z_m-z_h)+a_2*F;
     z_m_dot=z_m_dot+z_m_ddot*dt;
     z_m=z_m+z_m_dot*dt;
     z_h_ddot=a_3*z_m-a_3*z_h-g+a_4*d;
     z_h_dot=z_h_dot+z_h_ddot*dt;
     z_h=z_h+z_h_dot*dt;
     theta_hat=theta_hat+theta_hat_dot*dt; % update estimation
     eta=eta+eta_dot*dt;

     %system dynamics in PID
     x_1_ddot_PID=a_1_PID*F_PID-a_2_PID*x_1_PID+a_2_PID*x_2_PID-g;
     x_1_dot_PID=x_1_dot_PID+x_1_ddot_PID*dt;
     x_1_PID=x_1_PID+x_1_dot_PID*dt;
     x_2_ddot_PID=a_3*(x_1_PID-x_2_PID)-g+a_4*d;
     x_2_dot_PID=x_2_dot_PID+x_2_ddot_PID*dt;
     x_2_PID=x_2_PID+x_2_dot_PID*dt;

     % system dynamics in open loop
```

Figure C.1. MATLAB code for the simulations (cont.)

```
    z_m_ddot_open=-a_1*(z_m_open-z_h_open)-g;
    z_m_dot_open=z_m_dot_open+z_m_ddot_open*dt;
    z_m_open=z_m_open+z_m_dot_open*dt;
    z_h_ddot_open=a_3*(z_m_open-z_h_open)-g+a_4*d;
    z_h_dot_open=z_h_dot_open+z_h_ddot_open*dt;
    z_h_open=z_h_open+z_h_dot_open*dt;

    disturbance=theta_hat'*ksi/a_4;
    % write down the calculated values into corresponding matrices
    Z(i)=z(1); % state tracking error
    F_Vector(i)=F; % force
    F_Vector_PID(i)=F_PID;
    X_2(i)=z_h; % x_2 state in closed loop
    X_2_open(i)=z_h_ref; % reference signal
    X_2_PID(i)=x_2_PID;
    D(i)=disturbance; % disturbance estimation
    D_real(i)=d; % real disturbance value
    X_1_PID(i)=x_1_PID;
    X_1(i)=z_m;

    if rem(i*dt,stride)==0
        cycle=cycle+1;
    end
end
```

Figure C.1. MATLAB code for the simulations (cont.)