



Faculty of Engineering

**Modelling and Control of Ankle Foot Orthosis (AFO) for Children
Utilising Soft Computing Towards Intelligent Approach**

Aida Suriana binti Abdul Razak

**Master of Engineering
2024**

Modelling and Control of Ankle Foot Orthosis (AFO) for Children
Utilising Soft Computing Towards Intelligent Approach

Aida Suriana binti Abdul Razak

A thesis submitted

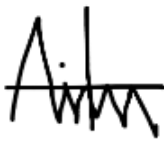
In fulfillment of the requirements for the degree of Master of Engineering

(Master of Engineering)

Faculty of Engineering
UNIVERSITI MALAYSIA SARAWAK
2024

DECLARATION

I declare that the work in this thesis was carried out in accordance with the regulations of Universiti Malaysia Sarawak. Except where due acknowledgements have been made, the work is that of the author alone. The thesis has not been accepted for any degree and is not concurrently submitted in candidature of any other degree.



.....

Signature

Name: Aida Suriana binti Abdul Razak

Matric No.: 19020156

Faculty of Engineering

Universiti Malaysia Sarawak

Date : 5 January 2024

ACKNOWLEDGEMENT

First of all, I am grateful to Allah S.W.T, I have successfully completed this research after facing various challenges.

Secondly, I would like to thank Dr Annisa binti Jamali as my supervisor for the support, patience, advice and guidance that contributed to this research. All the knowledge and help that she provided has taught me a lot to be a student who is always working to solve problems. It is also not forgotten to the lecturers and staffs at the Faculty of Engineering and the Centre of Graduate Studies, University Malaysia Sarawak who has provided exposure and teaching throughout this study.

Thanks to my best friend, Nur Jannah Al Firdaus binti Abd Al Fatah for the help and enthusiasm so that I do not give up on my studies and can help with chess training to represent the university and country. Thanks also to my beloved family for giving me the love, passion, advice, prayers and tips I really need.

Finally, thank you to all those who have directly or indirectly assisted in providing suggestions, assistance and enthusiasm in preparing me to complete this research.

ABSTRACT

Ankle foot orthosis (AFO) is frequently utilised to offer practical help for patients with lower limb injuries or defects. However, ground response force modelling has historically proven difficult, especially when the robot is modelled with a high degree of freedom and has numerous points of contact with the ground. Therefore, this research presents the development of a dynamic system and controller of AFO. In order to accurately reflect the system dynamics, a prototype AFO model for kids was created and constructed. Using data immediately obtained from the experimental setup, the system's dynamic behaviour was simulated. The research uses both non-parametric and parametric techniques, namely Multilayer Perceptron Neural Network (MLP NN) and Particle Swarm Optimisation (PSO), for system modelling. The modelling stage comprised breaking down the dynamic AFO models into single-input single-output models. Validation methods such as the mean-squared error (MSE) and correlation tests were utilized to verify the performance of the models. The results suggested that MLP NN excelled over PSO in recognizing the AFO dynamic system by utilizing the model structure of delay 8, with unbiased results whereby the confidence level fell within the 95% range and showed the lowest mean squared prediction error is 1.1034×10^{-5} . In comparison, it can be concluded that MLP NN provided a more accurate estimation of the AFO dynamic model. In controllers' development, the research employed both conventional and intelligent Proportional–Integral–Derivative (PID) based controllers. For the conventional controller, Proportional–Integral–Derivative Ziegler-Nichols (PID-ZN) was tuned by using the mathematical calculation based on the identified models obtained earlier. Meanwhile, for the intelligent controller, Proportional–Integral–Derivative Particle Swarm Optimisation (PID-PSO) was utilized to find the optimum value of PID controller parameters to track the desired position during the ankle

training. The PID controllers were tuned automatically by the PSO algorithm based on the same identified models. The performance of both developed controllers was compared and analyzed. The system's conduct in tracking the position of the AFO along its trajectory was observed. Both control approaches' comparative evaluation was provided and discussed. Based on the results, it is evident that PID-PSO performs superiorly due to its lower steady-state error and shorter settling time when compared to PID-ZN. This comparison indicates that the intelligent controller, PID-PSO showed a significant improvement as compared to the conventional controller, PID-ZN. It was found that the percentage of improvement achieved by the intelligent controller over the conventional controller for settling time is 62.6 % and the steady state error is improved by 75.56 %.

Keywords: Ankle foot orthosis, system identification, Neural Network Multilayer Perceptron, Particle Swarm Optimisation, Proportional–Integral–Derivative Ziegler-Nichols, Proportional–Integral–Derivative Particle Swarm Optimisation

Pemodelan dan Kawalan Ortosis Pergelangan Kaki (AFO) untuk Kanak-kanak dengan Menggunakan Penyelakuan Komputer ke Arah Pendekatan Pintar

ABSTRAK

Orthosis kaki pergelangan kaki (AFO) sering digunakan untuk menawarkan bantuan praktikal bagi pesakit dengan kecederaan atau kecacatan anggota badan yang lebih rendah. Walau bagaimanapun, pemodelan kekuatan tindak balas tanah secara historis terbukti sukar, terutama ketika robot dimodelkan dengan tahap kebebasan yang tinggi dan mempunyai banyak titik kontak dengan tanah. Oleh itu, penyelidikan ini menyajikan pengembangan sistem dinamik dan pengawal AFO. Untuk mencerminkan dinamika sistem dengan tepat, model prototaip AFO untuk anak-anak dibuat dan dibina. Dengan menggunakan data yang segera diperoleh dari penyediaan eksperimen, tingkah laku dinamik sistem disimulasikan. Penyelidikan ini menggunakan teknik bukan parametrik dan parametrik, iaitu MLP NN (perceptron berbilang lapisan rangkaian neural) dan PSO pengoptimuman kerumuman zarah), untuk pemodelan sistem. Tahap pemodelan merangkumi pemecahan model AFO dinamik menjadi model output tunggal input tunggal. Kaedah pengesahan seperti ralat kuadrat rata-rata (MSE) dan ujian korelasi digunakan untuk mengesahkan prestasi model. Hasilnya menunjukkan bahawa MLP unggul daripada PSO dalam mengenali sistem dinamik AFO dengan menggunakan struktur model penundaan 8, dengan hasil yang tidak berat sebelah di mana tahap keyakinan jatuh dalam 95 % julat dan menunjukkan ralat ramalan kuasa dua min terendah ialah 1.1034×10^{-5} . Sebagai perbandingan, dapat disimpulkan bahawa MLP memberikan anggaran yang lebih tepat mengenai model dinamik AFO. Dalam pengembangan pengawal, penyelidikan menggunakan pengawal berasaskan pengawal berkadaran-kamiran-terbitan (PID). Untuk pengawal konvensional, pengawal berkadaran-kamiran-terbitan Ziegler-Nichols (PID-ZN)

diselaraskan dengan menggunakan pengiraan matematik berdasarkan model yang dikenal pasti yang diperoleh sebelumnya. Sementara itu, untuk pengawal pintar, pengawal berkadaran-kamiran-terbitan pengoptimuman kerumuman zarah (PID-PSO) digunakan untuk mencari nilai optimum parameter pengawal PID untuk mengesan kedudukan yang diinginkan semasa latihan pergelangan kaki. Pengawal PID diselaraskan secara automatik oleh algoritma PSO berdasarkan model yang dikenal pasti sama. Prestasi kedua-dua pengawal yang dikembangkan dibandingkan dan dianalisis. Tingkah laku sistem dalam mengesan kedudukan AFO di sepanjang lintasannya diperhatikan. Penilaian perbandingan kedua-dua pendekatan kawalan diberikan dan dibincangkan. Berdasarkan hasilnya, terbukti bahawa PID-PSO berprestasi lebih baik kerana kesalahan keadaan tetap yang lebih rendah dan masa penyelesaian yang lebih pendek jika dibandingkan dengan PID-ZN. Perbandingan ini menunjukkan bahawa pengawal pintar, PID-PSO menunjukkan peningkatan yang ketara berbanding dengan pengawal konvensional, PID-ZN. Didapati bahawa peratusan peningkatan yang dicapai oleh pengawal pintar berbanding pengawal konvensional untuk menyelesaikan masa adalah 62.6% dan kesalahan keadaan tetap ditingkatkan sebanyak 75.56 %.

Kata kunci: Ortosis pergelangan kaki, sistem pengenalanpastian, pengoptimuman kerumuman zarah, rangkaian neural perceptron berbilang lapisan, pengawal berkadaran-kamiran-terbitan Ziegler-Nichols, pengawal berkadaran-kamiran-terbitan pengoptimuman kerumuman zarah

TABLE OF CONTENTS

	Page
DECLARATION	i
ACKNOWLEDGEMENT	ii
ABSTRACT	iii
ABSTRAK	v
TABLE OF CONTENTS	vii
LIST OF TABLES	xi
LIST OF FIGURES	xii
LIST OF ABBREVIATIONS	xiv
LIST OF SYMBOLS	xviii
CHAPTER 1: INTRODUCTION	1
1.1 Background	1
1.2 Problem Statement	2
1.3 Objectives	4
1.4 Research Scope	5
1.5 Organization of The Thesis	5
CHAPTER 2: LITERATURE REVIEW	7
2.1 Introduction	7
2.2 Biomechanics of Ankle Foot	7
2.3 Types of Ankle Foot Orthosis	8
2.4 Parameter settings for Children	10

2.5	System Modelling	12
2.5.1	Mathematical Modelling	12
2.5.2	System Identification	15
2.5.2.1	Parametric Modelling	16
2.5.2.2	Non-parametric Modelling	18
2.6	Control System	20
2.7	Research Gap	22
CHAPTER 3: METHODOLOGY		28
3.1	Introduction	28
3.2	Research Framework	29
3.3	Experimental Setup	31
3.3.1	AFO Experimental Rig	31
3.3.2	Data Acquisition	34
3.3.3	Linear Actuator	36
3.3.4	Motor Driver	37
3.3.5	IMU Sensor	38
3.3.6	Circuit Design	39
3.4	Software Development	40
3.5	System Integration	46
3.6	Verification of Experimental Setup	51

3.7	Development of System Modelling	52
3.7.1	Particle Swarm Optimisation	52
3.7.2	Artificial Neural Network	55
3.8	Model Validation	59
3.8.1	One Step Ahead Prediction	59
3.8.2	Mean Squared Error	61
3.8.3	Correlation Test	61
3.9	Design Controller	62
3.9.1	Conventional PID Controller (Ziegler-Nichols Rules)	63
3.9.2	Intelligent Controller - PID-PSO Controller	64
3.10	Summary	68
CHAPTER 4: RESULTS AND DISCUSSION		69
4.1	Introduction	69
4.2	System Modelling Results	69
4.2.1	Particle Swarm Optimization	71
4.2.2	Artificial Neural Network	76
4.2.3	Comparison and Analysis	79
4.3	Control Scheme	80
4.3.1	Proportional Integral Derivative	80
4.3.2	Particle Swarm Optimization	83

4.3.3	Comparison and Analysis	86
4.4	Summary	86
CHAPTER 5: CONCLUSION AND RECOMMENDATIONS		88
5.1	Conclusion	88
5.2	Recommendations	89
REFERENCES		90
APPENDICES		99

LIST OF TABLES

	Page
Table 2.1 The comparison of the development of AFO by experimentation	26
Table 2.2 The comparison of the development of AFO by simulation	27
Table 3.1 AFO movements in relation to axis and angle	33
Table 3.2 The features of the microcontroller	35
Table 3.3 Specification of the LIN-ACT1-02 Windynation linear actuator	37
Table 3.4 the alignment of the IMU sensor axes (The MathWorks, 2020)	45
Table 3.5 The average time of 1 complete walking cycle of normal children under the age of 12	50
Table 3.6 Method 1 PID-ZN parameters	64
Table 4.1 Optimal result parameters in PSO modelling optimization	72
Table 4.2 Comparison of PSO optimization performance in different numbers of model order	76
Table 4.3 Comparison of MLP NN performance	78
Table 4.4 Performance of parametric and non-parametric modelling approaches	79
Table 4.5 The results of the P, PI, and PID controller	82
Table 4.6 Optimal result parameters in PSO controller optimization	84
Table 4.7 The results of the PID-PSO control	85
Table 4.8 The comparison performance between the PID (Ziegler–Nichols) controller and PID-PSO controller	86

LIST OF FIGURES

	Page
Figure 2.1 Gait phases (Caldas et al., 2017)	8
Figure 2.2 Example of PID control system	22
Figure 3.1 Flowchart of the research framework	29
Figure 3.2 AFO's dorsiflexion	32
Figure 3.3 AFO's plantarflexion	32
Figure 3.4 Arduino Mega 2560 R3 board	35
Figure 3.5 LIN-ACT1-02 Windynation linear actuator	36
Figure 3.6 L298N motor driver	38
Figure 3.7 Location of IMU sensor placed on AFO rig	39
Figure 3.8 Circuit design	40
Figure 3.9 The support package	41
Figure 3.10 Data collection (input) using Simulink	41
Figure 3.11 Data collection (output) using Simulink	42
Figure 3.12 Low-pass filter (a) and High-pass filter (b)	43
Figure 3.13 The fundamental design of a complementary filter (Narkhede et al., 2021)	43
Figure 3.14 The experimental setup of AFO	47
Figure 3.15 Schematic diagram of AFO	47
Figure 3.16 Block diagram for data collection of AFO	49
Figure 3.17 (a) Dorsiflexion (b) Plantarflexion	52
Figure 3.18 The flowchart of the PSO algorithm	53
Figure 3.19 The flowchart of the MLP Neural Network	56
Figure 3.20 MLP-NN algorithm diagram	57
Figure 3.21 Feed-forward dynamic neural network architecture	58

Figure 3.22	The basic function of MLP-NN structure (Jamali, 2017)	58
Figure 3.23	Block diagram of the overall closed-loop system	63
Figure 3.24	PID control of a plant (Ogata, 2010)	63
Figure 3.25	S-shaped response curve (Ogata, 2010)	64
Figure 3.26	The flowchart of the PID-PSO algorithm (Sultan, 2016)	67
Figure 4.1	The experimental IMU angle (roll)	70
Figure 4.2	Bang-bang signal responses	71
Figure 4.3	The output and estimated output of roll axis angle (PSO)	73
Figure 4.4	(a) Correlation of error (PSO) (b) Correlation of the input and the error (PSO)	74
Figure 4.5	The stability test for roll axis angle (PSO)	75
Figure 4.6	The output and estimated output of roll axis angle (MLP)	77
Figure 4.7	(a) Correlation of error (MLP NN) (b) Correlation of the input and the error (MLP NN) (c) Correlation of the square input and the error (MLP NN) (d) Correlation of square of the input square of the error (MLP NN) (e) Correlation of multiplication of the input by the error and the error (MLP NN)	78
Figure 4.8	Actual signal (without controller)	80
Figure 4.9	Simulink setup of the P, PI, PID controller	81
Figure 4.10	The comparison between the P, PI, and PID controller and the one without a controller	82
Figure 4.11	Simulink setup of the PID controller	83
Figure 4.12	The comparison between the PID (Ziegler–Nichols) controller, PID-PSO controller, and the one without a controller	84
Figure 4.13	MSE convergence of PSO for hub angle	85

LIST OF ABBREVIATIONS

2D	Two dimensions
3D	Three dimensions
AFO	Ankle foot orthosis
ANN	Artificial neural network
APSO	Adaptive particle swarm optimization
BAS	Beetle antennae search
BBO	Biogeography-based optimization
BSO	Beetle swarm optimisation
BPNN	Back propagation neural network
DC	Direct current
DOF	Degree of freedom
EEPROM	Electrically erasable programmable read-only memory
EMG	Electromyography
FFNN	Feedforward neural network
GND	Ground
GRFs	Ground reaction forces
HMM	Hidden Markov Model
HPF	High pass filter

IAE	Integral of the absolute error
IC	Initial contact
IMU	Inertial measurement unit
IPSO	Improved particle swarm optimisation
ISE	Integral of the squared error
ITAE	Integral of time multiplied of absolute error
IT-AFO	Ideal training ankle foot orthosis
LCF	Linear complementary filter
LPF	Low pass filter
LR	Loading response
MLP NN	Multilayer perceptron neural network
MS	Mid stance
MSE	Mean squared error
NARX	Nonlinear autoregressive network with exogenous inputs
NN	Neural network
NNARX	Neural network autoregressive network with exogenous inputs
OCP	Optimal control problem
OSA	One step-ahead
P	Proportional

PD	Proportional-derivative
PI	Proportional-integral
PID	Proportional-integral-derivative
PS	Pre-swing
PSO	Particle swarm optimisation
RASPLib	Rensselaer Arduino Support Package Library
RBFNN	Radial basis function neural network
RBFNNs	Radial basis function neural networks
RISE	Robust integral of the sign of the error
RMSE	Root mean square error
ROS	Robot operating system
SCL	Serial clock
SDA	Serial data
SE	Subtalar eversion
SIT	System identification toolbox
SMD	Spring-mass-damper
SQP	Sequential quadratic programming
SRAM	Static random access memory
ST	Stance

SVM	Support vector machine
SW	Swing
TDLs	Tapped delay lines
TS	Terminal stance
USB	Universal Serial Bus
UT	Utah
VCC	Voltage common collector
VDC	Volts of direct current
Z-N	Ziegler-Nichols

LIST OF SYMBOLS

α	The weighted factor for gyroscope and accelerometer, alpha
β	Weighing factor
C_1, C_2	Acceleration coefficients/ learning functions
c_r	The dynamic parameters of the ankle joint's second layer
$C(s)$	The PID controller
ϵ	The set membership symbol (is an element of)
e	Error
$e(t)$	The predicted error at time, t
e_{ss}	Steady state error
$\varepsilon(t, \theta)$	The residual as a function of time, t , and a parameter set, θ
$\varepsilon u(t)$	Impulse function
f	Function
$f(\cdot)$	Non-linear function
$fit(P_i)$	Fitness equation for PSO
$G_L(s)$	Examples of transfer functions of the human exoskeleton
$G_H(s)$	Examples of transfer functions of the lower limb
$G(s)$	The transfer function for the low-pass filter
$\bar{G}(s)$	The transfer function for the high-pass filter

g	The function of the neural network
i	The i -th particle in the d -dimensional search space
J	The ankle joint
k	The number of iterations
k_r	The dynamic parameters of the ankle joint's second layer
K_p	Proportional gains
K_i	Integral gains
K_d	Derivative gains
L^i	The approximated error function at iteration i of the LM algorithm
M_p	Maximum overshoot
N	Number of input or output samples
$\circ$	The zero
q	The number of particles
t_s	Settling time
t_r	Rise time
$\forall \tau$	for all values of value of the time lag, τ
T_{max}	The maximum number of time steps the swarm
u	The inputs

$v_{id}(t)$	The current velocity of the i-th particle in the d-dimensional search space
V	Particle velocity
$W(K)$	Performance function
W	Inertia weight
w_{ij}	Hidden layer
w	Inertia
w_{start}	The starting points of linearly decreasing inertia weight
w_{end}	The ending points of linearly decreasing inertia weight
w_{max}	Maximum inertia weight
w_{min}	Minimum inertia weight
X	Particle position
$\times$	The poles
$\dot{x}_g$	The state vector of the gyroscope
x_a	The state vector of the accelerometer and magnetometer sensors
$\hat{x}$	Linear complementary filter structure
$X_i = (x_{i1}, x_{i2}, \dots, x_{id})$	The d-dimensional vector
$x_{id}(t)$	Position vector of the i-th particle in the d-dimensional search space
x_i	Input layer

y_j	Output layer
$\hat{y}$	The output
y	The outputs
$y(t)$	Output system at time t
$y(n)$	The system's actual outputs
$\hat{y}(n)$	The anticipated output
Z^N	The training data set
θ	Angle
$\dot{\theta}$	The pitch axes of the gyroscope
θ_a	The pitch axes of the accelerometer
θ_{act}	Actual angle
$\theta_{des.}$	Desired angle
$\theta_{di}(t)$	Reference hub angle
$\theta_i(t)$	The actual hub angle
$\varphi_{u\varepsilon}(\tau)$	The cross-correlation function between $u(t)$ and $\varepsilon(t)$
$\varphi(t)$	denotes the regression vector at time step, t
$\dot{\phi}$	The roll axes of the gyroscope
$\dot{\psi}$	The yaw axes of the gyroscope
ψ_m	The yaw axes of the magnetometer

CHAPTER 1

INTRODUCTION

1.1 Background

One of the most crucial daily activities for people is walking (Abe et al., 2017). Foot drop, stroke, and other disabilities are the main causes of patients who walk abnormally (Amiri et al., 2020). A patient who has a drop foot is unable to dorsiflex, which causes them to drag their toe while swinging. The lower leg joint contributes most to walking compared to other repetitive joints of a limb because it is vulnerable to damage due to its high torque (Habeeb Mohamed & Zakaria, 2019). According to the World Health Organization (2018), the number of people with disabilities worldwide is over a billion, or about 15% of the total population. Consequently, gait preparation is a top priority in rehabilitation medicine (Abe et al., 2017).

One method to help with walking assistance, such as providing stability and maintaining range of motion, is the use of ankle foot orthosis (AFO) during rehabilitation training. Gait speed appeared to be faster in all studies involving AFO conditions than in those without them (Yamamoto et al., 2019). Clinicians must fully comprehend the mechanical characteristics of AFO and how they affect gait in order for patients to benefit fully from their orthoses (Kobayashi et al., 2017). It is crucial to adjust the ankle foot orthosis' torque during clinical practice in order to accommodate each patient's body function and gait capability (Yamamoto et al., 2019).

It has always been challenging to model ground reaction forces (GRFs), especially when the robot is modelled with a high degree of freedom and has several points of contact

with the ground. The complexity of the robot model (Caldas et al., 2017), the difficulty of acquiring precise data (Bykov et al., 2018), and the difficulties of integrating it with control systems (Al-Shuka et al., 2019) make it a challenge when modelling GRFs. This is why modelling the AFO is important (Eldirdiry et al., 2020). The direct multiple shooting techniques (Sreenivasa et al., 2017), functional electrical stimulations (Eldirdiry et al., 2020), Monte Carlo simulations (Hegarty et al., 2017), 2D and 3D computational simulations (Fatone et al., 2014), transition using a genetic algorithm (Chamnikar et al., 2017) and mathematical modelling of the musculoskeletal system (Rosenberg & Steele, 2017) are the most frequently employed methods for developing the models.

It is essential to create an appropriate control system so that the lower extremity exoskeleton legs of the walker's stride can follow a preset trajectory (Gilbert, Zhang, & Yin, 2016). PID controllers facilitate signal design by comparing desired input signals with actual output signals. The lower limb gait rehabilitation robot trajectory tracking control can then be drawn using a Simulink multi-joint coordination simulation diagram. Previous research experimented with different controllers for lower limb gait rehabilitation such as proportional–integral–derivative (Habeeb Mohamed & Zakaria, 2019), Adaptive Particle Swarm Optimization (Amiri et al., 2020), and Ant Colony Optimization (Y.Hassan & S. Ghintab, 2016).

1.2 Problem Statement

When a virtual robot is employed instead of testing similar scenarios on humans, the discomfort that may ensue from a human rolling over his foot or dropping it is not taken into account (Eldirdiry et al., 2020). Although mathematical modelling is widely used in simulation, it has not yet been put to the test through experiments, so it cannot capture the

true cost function for children walking (Skaaret et al., 2019; Kribus-Shmiel et al., 2018; Rosenberg & Steele, 2017). AFO are crucial for kids for a number of reasons, including bettering gait kinematics, perhaps lowering walking energy expenditure, customising for various daily activities, preventing bone abnormalities, and possibly facilitating neuromuscular control training. According to Rosenberg & Steele (2017), the real cost function for children walking for under 14 years old is the sum of muscular forces in one leg, integrated over single-limb support and double-limb support, known as leg impulse. Although this is not a direct measure of metabolic cost, muscle mechanical work rate plays an important role. An AFO used motion capture has a disadvantage in the slow time response as it is dependent on a computer processor which is very costly (Jin et al., 2021). The reliability is questionable if a potentiometer is used in AFO to adapt in the real application (Habeeb Mohamed & Zakaria, 2019).

For the study that includes AFO that uses an actuator, modelling will be much more effective if system identification is used. System identification assists in capturing the actuator's dynamic features, such as its responsiveness to various inputs and the link between inputs and outputs (Aliman, Ramli, & Haris, 2018). This allows for the building of a model that closely reflects the actuator's real behaviour, resulting in greater exoskeleton control and manoeuvring. System input and output data obtained from experimentation are used in dynamic mathematical modelling to create a system identification. Although system identification is not frequently used for AFO, it is constantly used in other applications, such as the assessment of actuator performance. Studies in the past revealed that only a small number of studies used system identification (Habeeb Mohamed & Zakaria, 2019; Yagi et al., 2018).

To be more precise, it is necessary to research the issue of learning human gait parameters, which leads to a better dead-reckoning algorithm (Kasebzadeh, 2019). Non-parametric algorithms use a flexible number of parameters for modelling, which typically increases over the course of learning, and make fewer assumptions about the data. The computational efficiency of parametric algorithms can be higher than that of non-parametric ones. Performance degradation, however, might not be insignificant if the data does not support the underlying hypotheses. In summary, parametric modelling is based on particular assumptions and parameter estimates, whereas non-parametric modelling is data-driven and makes no significant assumptions about the underlying system (Mat Darus & Tokhi, 2006).

Traditional methods make it difficult and inaccurate to fine-tune a proportional-integral-derivative (PID) controller for a complicated multi-joint construction, such as an exoskeleton (Amiri et al., 2020). PSO is simple to understand and used in a variety of optimization issues. As a result, it has been used to regulate the PID parameters' robustness.

Therefore, this research will be investigating the modelling of the system using system identification and identify the most suitable control system to make it simplified according to the specification of the available rig and controller used. The best algorithms whether it consists of a combination or otherwise will be carefully analysed and studied depending on the specification of the motor that will be used and the ability to acquire data. The control system will be identified thoroughly and chose the simplest with the suitability of the rig.

1.3 Objectives

The main objectives of this project are:

- (a) To model the dynamic of the AFO for children by using parametric and non-parametric dynamic modelling via system identification utilizing PSO and ANN as a model estimation.
- (b) To develop both conventional and intelligent PID-based controller for position tracking of AFO based on the identified model.
- (c) To verify and validate the developed controllers and perform the comparative assessment between those controllers.

1.4 Research Scope

The scope of this project will be limited to:

- a) The modelling and control do not consider the environment effect and vibration.
- b) The input data was taken using 1-axis movement only.
- c) The AFO rig is suitable for rehabilitation exercises for children under 12 years old who can already walk independently.
- d) The number of participants is limited to availability during the research period.

1.5 Organization of The Thesis

The organization of this thesis is as follows:

Chapter 1 introduces the modelling and control of AFO through the background, problem statement, objectives, research scope, and organization of the thesis.

Chapter 2 will preview all the current and previous research done according to the modelling and control of AFO based on the biomechanics, application, system modelling,

and control system. Lastly, the research gap was highlighted in order to conduct this research.

Chapter 3 will provide a description of the research framework and the development of the system by experimentation via rig, instrumentation and data acquisition, linear actuator, motor driver, IMU sensor, circuit design, and system integration. Besides that, the flowchart of the algorithm and control system with the method of model validation will be explained in detail in this chapter. The experimental setup was then verified before proceeding to the modelling stage.

Chapter 4 contains all the algorithms used in the parametric and non-parametric modelling of the AFO. Both types of modelling were compared and analyzed before proceeding with the control scheme. The best modelling was controlled by using PID and PSO which were also compared and analyzed.

Chapter 5 concludes the research and provides future recommendations.

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction

In this chapter, the biomechanics of the ankle-foot and the use of ankle-foot orthoses will be briefly explained based on prior research. Additionally, it will describe how system modelling compares to system identification in terms of mathematical modelling. Based on earlier research on ankle-foot orthoses, a number of control systems will also be covered. Last but not least, a research gap will be identified in order to address and improve the issues from the modelling and control of ankle-foot orthoses research conducted in the past.

2.2 Biomechanics of Ankle Foot

Walking is defined as moving on foot, step by step, so that one foot is always on the ground, including walking, sailing, moving forward, backward, and sideways (Owen, 2014). Walking is separated into two stages: complete gait and stepping. The trunk maintains a vertical orientation and is interpreted forward or backward whether standing and swinging, jumping, or walking with full gait cycles. To accomplish this, the segments must be optimally aligned.

One gait cycle may be broken down into two stages, stance (ST) and swing (SW), which can then be further split into five and three stages, respectively, as illustrated in Figure 2.1 (Caldas et al., 2017). The instant the heel of one foot contacts the ground during a physiological gait is known as the initial contact (IC), which marks the beginning of the ST process. The loading response (LR), which is started by the IC, is finished with the toe-off (EC). The mid stance (MS) continues until the reference leg's heel comes off the ground.

The terminal stance (TS), which comes after the MS, is followed by the contralateral IC. The pre-swing (PS), followed by the reference leg toe-off, marks the end of the ST phase. The first swing, as per the SW phase, follows the PS and comes to a stop when the supporting leg's ankle joint crosses that of the reference leg's ankle. The mid swing comes next, ending with the reference leg's tibia being vertical to the ground. The terminal swing, which starts with the reference leg's first contact, completes the gait cycle.

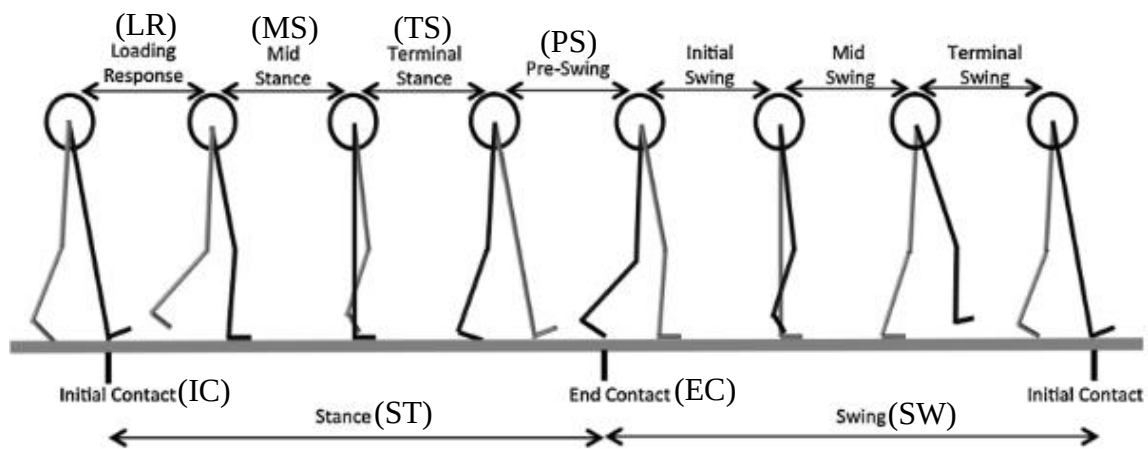


Figure 2.1: Gait phases (Caldas et al., 2017)

2.3 Types of Ankle Foot Orthosis

Orthoses are among the most commonly used childhood interventions (Owen, 2014). Both treatments should be recommended optimally if they are to produce the desired outcomes in the necessary realms of the International Classification of Disability and Health, Children and Youth Version (2007). Firstly, by type of orthosis as defined by the International Standards Organisation. Secondly, the dosage which includes the design, alignments, proportions, stiffness, and prescription profiles for orthoses-the algorithms and the clinical rationale for the algorithms. Finally, by the level of administration, which is a summary of the activities for which the orthosis is to be used and for how long it is to be

worn every day or week, depending on the expected results in all International Classification of Disability and Health, Children and Youth Version domains.

Current AFO are made with or without articulated joints and a variety of motion control components, including springs, dampers, four-bar mechanisms, and series elastic actuators (Alam et al., 2014). Devices classified as passive, semi-active, and active AFO are the three main categories. No electrical, electronic, or power sources may be found in passive AFO devices. This can consist of mechanical elements such as dampers or springs for regulating ankle-foot complex motion. Those devices are articulated and non-articulated in two types. Non-articulated AFO typically consist of a single piece made of lightweight thermoforming or thermoplastic material that covers the dorsal portion of the leg and the bottom of the foot. Passive articulated AFO have lightweight thermoplastic or carbon-composite shells and articulated joints. Passive components, which have limited effectiveness for persons with impairments and are not adaptive to changing settings, limit the passive movement regulation of AFO.

Through the use of computer power, semi-active AFO devices are able to modify the flexibility of the ankle joint. Active AFO can interact with people while walking and will act in a manner consistent with that interaction. These AFO include an actuator, tethered or untethered power supply, and stiffness control features like the magnetorheological brake for enhanced ankle motion regulation. The control device typically consists of elements such as a power sensor, angle measuring sensor, accelerometer, and microcontroller.

When using a powered AFO to help disabled people with daily tasks or improve healthy people's walking skills, portability is a crucial requirement (Molledo et al., 2018). Research on the experimental comparison of passive and powered (active) AFO showed that

the primary criticism of powered AFO was the additional mass, and all participants noted that their preference for powered AFO would increase if the system were lighter.

As a result, the type of AFO to use varies on the patient's disability. A passive AFO can be helpful for people with less severe disabilities whereas an active AFO can be used by patients with severe disabilities (Adiputra et al., 2019).

2.4 Parameter settings for Children

According to Aliman et al. (2018), the relevance of addressing parameter settings for children when utilising AFO stems from the need to guarantee that the AFO are suited to the unique needs of paediatric patients. Children have distinct physiological and biomechanical characteristics that differ from those of adults, which influence the design and functionality of AFO. As a result, modifying the specifications of AFO to accommodate children's growth and development is critical for maximising their effectiveness and providing correct orthotic support. Furthermore, evaluating parameter settings for children aids in the resolution of difficulties such as gait development, muscle tone, and overall comfort, all of which are critical for encouraging mobility and independence in paediatric users.

The parameter settings for the children in the Rosenberg & Steele (2017) study included an AFO stiffness range of 0.0 Nm/deg to 5.0 Nm/deg, which was determined based on prior studies for children with cerebral palsy and unimpaired adults. Each participant's AFO equilibrium angles varied from the smallest to the highest ankle angle during walking. Recent powered AFO for children can weigh as little as 1.1 kg per foot, whereas commercial passive AFO for children can weigh as little as 0.3 kg per foot, which, when combined with shoes, would be less than 5.0% of the mass of the majority of the children in the study.

Hegarty et al. (2017) included two children who presented with spastic diplegia and gross motor function classification scores of II. Child 1 was 14 years old, 1.6 metres tall, and weighed 50.5 kilogrammes. Child 2 was 16 years old, 1.6 metres tall, and weighed 48.7 kilogrammes. Both children wore their prescription AFOs, with Child 1 wearing a leaf spring AFO and Child 2 wearing an articulated ankle AFO.

Contini et al. (2019) study comprised ten children diagnosed with diplegic cerebral palsy, aged four to eleven years. The participants comprised of three females and seven males, with an average body mass of 19.8 kg and a height of 1.10 metres. The children's leg length was an average of 0.49 m. The study's qualifying criteria included being between the ages of 3 and 12 years old and having a Gross Motor Function Classification System I or II. AFO sizing is commonly calculated using a variety of measurements, such as foot length, ankle circumference, calf circumference, and leg length. These measurements are then utilised to determine the most appropriate AFO size and design for each child.

In conclusion, while analysing the effects of AFO mechanical qualities on gait simulation muscle force findings for children, parameter parameters such as age range, gender, weight, and height are critical. This is critical because these characteristics have a substantial impact on the biomechanics of walking. Children's musculoskeletal development varies with age, which can influence how their musculoskeletal system responds to AFO mechanical qualities (Hegarty et al., 2017). Similarly, gender differences in muscle mass and distribution can influence how AFOs interact with the body while walking. Furthermore, weight and height have an important effect in determining the load and lever arms acting on the lower limbs, which influences the forces exerted by muscles during walking.

2.5 System Modelling

The foot's inherent curvature and its effect on how the rest of the body moves are ignored when simulating the foot as a flat surface, which simplifies the resolution of the interface dynamics (Sreenivasa et al., 2017). The orthosis increases ankle torque, especially during push-off. As a result, the model walking with the orthosis may move more quickly and with more extended knees than the model going barefoot. Both mathematical modelling and system identification can be used to model an ankle foot orthosis.

2.5.1 Mathematical Modelling

The goal of modelling that is only done through simulation by using mathematical calculation is to achieve precise alignment. The exact alignment of the mechanical joint axis with the anatomical joint axis is a key idea for the development of articulated AFO (Fatone et al., 2014). The leg model developed by Zygo Media Group (American Fork, UT) served as the basis for the reference frame and a set of leg segment coordinates utilised in earlier investigations. Additionally, users can choose different viewing angles, change the animation speed, zoom in to show more detail, and select which part of the gait process to animate. The goal of this project was to create a three-dimensional model depicting the effects of AFO on ankle joint misalignment. Since the stiffest springs in the model are those on the anterior surface of the leg, deviations from the 2D model are likely the result of the 3D model's efforts to reduce the energy stored in the stiffer springs.

Therefore, additional model assumptions may introduce errors or restrict the model's applicability. The model's assumption that the ankle joint is a fixed hinge joint may not be accurate, which could cause errors in the calculated displacements. Due to the assumption that the AFO joint rotates freely about a fixed axis, the applicability of this model is limited

to AFO with fixed joints and no mechanical stops. To make the model tractable, it was necessary to assume that the AFO does not move in relation to the foot. This assumption, however, restricts model predictions to the maximum and minimum possible values of shank displacement. Despite these drawbacks, the model is still able to convey broad ideas that aid in understanding the connections between relative motion and ankle joint misalignment.

According to Eldirdiry et al. (2020), using a simulated model allows us to analyse hypothetical cases, which can aid in the resolution of potential issues. Due to its straightforward implementation in MATLAB/ Simscape for the GRF, the model spring-mass-damper (SMD) model is chosen in this study. It is therefore crucial to represent the full dynamic structure of a biped robot (as a simplified model of a patient) as well as the walking environment. The computed transfer function in Equation 2.1 is 70.4% more similar to the real transfer function that reflects the afflicted ankle joint, according to the system identification toolbox's (SIT) fitting percentage.

$$\frac{1}{Js^2 + c_r s + k_r} = \frac{231.8}{s^2 + 9.999s + 926.65} \quad \text{Equation 2.1}$$

The notations k_r and c_r in this equation represent the dynamic parameters of the ankle joint's second layer. The values of k_r and c_r were calculated using Equation 2.1 and found to be 8.71 N.m/rad and 23.18 N.m.s/rad, respectively. The use of an estimated transfer function rather than the actual one is since this study employs a robotic foot while human data are used to represent the foot drop behaviour of a human case. The exact mass distribution of the human foot would be difficult to model in the model that will determine the value of J. As a result, while similar mass is employed to mimic human foot, the mass

distribution on each is different. To correct the foot drop, an estimated transfer function is employed to adjust and identify optimal settings for the foot drop controller. The research's restriction is the use of a simplified calculation for the moment of inertia, as well as the requirement to reduce the total mass value to half of the maximum value throughout the simulation time to avoid excessive PID benefits.

Another study demonstrates that a robot operating system (ROS) with average human anthropometric height was used to create and control the human exoskeleton simulation model (Amiri et al., 2020). Each link between the exoskeleton and human is modelled by a one DOF pendulum model, which replicates a single joint. In the dynamic equation of the pendulum model, the Lagrangian formulation is employed. The Gazebo simulation environment contained a 3D human model to simulate the patient's interaction with the exoskeleton. The mass and inertia of the human model correspond to an individual whose height was 175.5 cm and weight was 74.2 kg. Parallel bars were added to the simulation to provide stability and prevent the human model from falling, ensuring that it faithfully replicated the gait training task.

The use of simulated models, on the other hand, presents several complex challenges to the simulation system design, such as ground interaction with the modelled foot and biped stability (Eldirdiry et al., 2020). Modelling GRFs have always been a difficult task, especially when the robot is portrayed with many degrees of freedom and multiple points of contact with the ground. Because ground reaction forces play such an important role in locomotion stability, the model must accurately depict them. This is because of their immense oscillation effect on the upper body. For ethical reasons, the independence provided by this computer simulation model cannot be achieved with human subjects.

Based on Sreenivasa et al. (2017), a simulation-based study focuses on the latter by using optimised control-based methods to predict motions, GRFs, and neuromuscular dynamics when walking with and without orthoses. The study develops an objective function that fits the patient's recorded joint kinematics as closely as possible to the Least-Squares Fit Barefoot Optimal Control Problem (OCP). The objective here is to apply computational tools and patient-specific models to aid with a critical clinical operation – fitting an orthosis to a patient. The patient was recorded walking in bare feet on flat terrain while wearing a bilateral ankle foot orthosis. As the patient used a passive type of AFO, the child walked without the help of an actuator or sensors for this research.

While this is a positive result because it leads to a less tiring gait, it is unclear at this time whether such improvements occurred in the patient's actual muscular efforts. Because there are gaps in the experimental EMG recordings for this gait, their predicted decrease in muscle activations must be taken as reasonable but unproven. In this early research, they do not simulate the feedback dynamics of muscle reflexes. Solving the OCP becomes more challenging when these closed-loop dynamics are included. Reflexes are normally subdued during normal movement, but they are crucial in ensuring that gait is perturbation rejection resistant (Geyer & Herr, 2010).

2.5.2 System Identification

The system identification approach, which has four stages—experiment setup, model structure selection, parameter estimation, and model validation—is used for actuator modelling (Aliman et al., 2018). Identification of the system's true model is its main objective. However, it's always a good idea to remember that every model is unrealistic, which implies that the model should always be anticipated to be imperfect or unresolved

phenomena (Reggeti, 2017). The ability of the models to be either linear or nonlinear is their most significant characteristic from the perspective of system identification. The first step in solving this optimization problem is to identify a function that can approximate the system's output in terms of its input. This characteristic is crucial because most system identification techniques rely on a linear system model, which in the case of contact can omit some crucial details about the phenomenon.

In the first case, the data can be modelled using a fixed set of parameters because it is assumed that they conform to a particular set of properties (Kasebzadeh, 2019). On the other hand, non-parametric algorithms make fewer assumptions about the data and use a flexible number of modelling parameters that typically increase over time. In terms of computation, parametric algorithms may be superior to non-parametric ones. However, if the data do not support the underlying hypotheses, performance degradation might not be insignificant. As a result, there are accuracy and computational cost trade-offs between the two types of algorithms.

2.5.2.1 Parametric Modelling

Parametric refers to the use of optimisation techniques to determine system parameters and optimise controllers for lower limb rehabilitation with actuators (Abdelhad et al., 2017). Previous research from Abdelhady et al. (2017) used algorithms such as biogeography-based optimization (BBO), PSO, and sequential quadratic programming (SQP) for system identification and PID controller optimisation in a prosthetic knee with a DC motor. BBO and PSO are simple to deploy and have demonstrated significant success in past studies. BBO has the best overall average ISE, although PSO has the fastest convergence during PID controller optimisation. The limitation is that SQP converges

slower than PSO and BBO. Increasing shank length increases the optimal proportional gain while decreasing the ideal derivative and integral gains for the PID controller.

A study by P. Zhang & Zhang (2022) improves the control of a leg exoskeleton robot by quickly recognising dynamic parameters. The study uses beetle swarm optimisation (BSO), a method that combines beetle antennae search (BAS) with PSO, to identify parameters faster and more accurately than existing algorithms. Experimental results demonstrate BSO's usefulness in forecasting torque for improved robot control, outperforming standard techniques. BSO's possible drawbacks include hyperparameter sensitivity, applicability varies among problems, interpretability issues, computational complexity considerations, and the requirement to analyse robustness to noisy data.

According to an experiment that examined the quasi-stationary condition and took the mechanism of the suggested procedure into account, the algorithm first calculates the mechanical entry of the human joint, and the impedance is then simply obtained by working the other way around (Yagi et al., 2018). The established algorithm considers the obtained angular response as the impulse response in the discrete-time domain, which is a well-known fundamental component in system identification because the perturbation torque is based on the equivalent area theory within a sufficiently brief time frame. The suggested method does not demand a correlation between the angle response data recorded by the angle sensor and the torque perturbation data recorded by the force sensor when using this algorithm. The correlation between angle response and torque perturbation is required to ensure that the intrinsic joint feature is the primary contributor to mechanical impedance. However, the intra-subject results show a negligible amount of dispersion, making it impossible to accurately estimate the joint impedance using a single collection of measurement data.

As a result of its good robustness, the support vector machine (SVM) algorithm's complexity has little relationship to the dimension of feature samples (Geng et al., 2021). By using the SVM one-to-one classification method, the lower limb movements were recognised and categorised. In the five experimental tests, the lower limb joint rehabilitation system's recognition and classification of forward and backward movements were more accurate, with only two errors occurring. The two errors mean that out of the five experimental tests, the lower limb joint rehabilitation system recognised and classified forward and backward movements accurately, with only two errors, while the recognition and classification of left-turn and right-turn movements were less accurate, with only one correct identification. Forward and backward actions accounted for 34% of the times with 100% accuracy in the five recognition and classification tests for the four actions (forward, backward, left, and right), while the left-turn and right-turn actions accounted for 11% of the total number of times. This demonstrates that the system has a high accuracy in recognising forward and backward movements but a slightly lower accuracy in recognising left-turn and right-turn movements that requires further improvement.

2.5.2.2 Non-parametric Modelling

Most scientific fields have used nonlinear systems identification and modelling to forecast the behaviour of dynamic systems in the future. Control theory has a thriving system identification community, and it is a crucial strategy for studying, exploring, and comprehending the world by using a formal description of events as a model (Arcolezi et al., 2021). Black-box model construction is largely dependent on the calibre of system measurement data. The core idea behind this strategy is to model the direct input-output relationship, or simply identify and model from data, with the primary goal being to determine the weights and other coefficients (also referred to as hyperparameters) of the NN.

Additionally, the foundation of NNs is a network of interconnected units known as neurons. These neurons have three or more layers, including input, hidden layers, and output layers.

Based on Arcolezi et al. (2021), the nonlinear mapping function was approximated in this study using a feedforward NN (multilayer perceptron, or MLP), which was built on a series-parallel structure. ELMAN Neural Networks have drawbacks such as slow convergence, a tendency to fall into local minima, and rigidity (Y. Zhang et al., 2021). The MLP Neural Network in the healthcare sector has the advantage that it can be used to solve challenging non-linear problems, performs well with large input data sets, and makes prompt predictions after training (Colakoglu & Akkaya, 2019).

When the suggested solution was used by the MLP NN and the robust integral of the sign of the error (RISE) controller was not tuned with empirical gains, the lower limb made a valiant effort to track the reference angle for 60 seconds for most healthy individuals (Arcolezi et al., 2021). If they had taken pretrial tests, which might have caused muscle fatigue from earlier stimulations, this would not have been possible. The root means square error (RMSE) was typically higher during the first sessions with healthy subjects, which may have been brought on by anxiety or discomfort from the electrical stimulation or the subjects' voluntary movements. One of several explanations for why some people's tracking scores improved as sessions went on is the use of a more representative model incorporating earlier rehabilitation data. As a result, the RISE controller was tuned better, which in turn improved the tracking outcomes in real-world situations.

Other research also uses neural networks for modelling lower limb rehabilitation (Saeed et al., 2022; Sivakumar et al., 2019). A study from Sivakumar et al., (2019) tests the efficacy of a wearable foot insole system's GRFs in conjunction with a feedforward neural

network (FFNN) to estimate ankle angles. The use of a single wearable sensor instead of multiple ones could potentially eliminate the need for long-term ankle angle estimation. However, this research did not use any actuator to make foot movement. Radial basis function neural networks (RBFNNs) are used to approximate the system function in the model-free control method, but an adaptive proportional-derivative RBFNN controller was created to accomplish the required results with the least amount of tracking error (Saeed et al., 2022). Both approaches result in a robust and stable system.

Research from Jin et al. (2021) suggested improving the Back Propagation Neural Network (BPNN) algorithm to predict the future joint angle of the human lower limb. In terms of final optimisation results and iterative convergence speed, Improved Particle Swarm Optimization (IPSO) algorithm beat PSO algorithm. It also offers a greater optimisation efficiency for BPNN models. By incorporating two cognitive inflection points on top of the original search path to strengthen individual cognition and historical cognition, respectively, IPSO improved the fundamental particle movement process of PSO. These three search directions strengthened the local, global, and historical inheritance capabilities in turn, which broadened the particle swarm's search space laterally. In addition, IPSO received feedback from the BPNN model to increase the complexity of the search path vertically. Finally, a three-search path competitive optimization model was developed, which sped up the updating of each particle's extremum.

2.6 Control System

The distinguishing effects of the various controllers will determine whether a particular controller is better suited for a particular group of subjects or a specific purpose (Molledo et al., 2018). The degree and pace of human adaption to the active system are

directly impacted by the manner of control utilised by enhancing the exoskeleton (Arnez-Paniagua et al., 2019). Different monitoring strategies have been employed to achieve these objectives, including the study of trajectory comparisons in lower limb kinematics through control methods for lower limb exoskeletons.

The research from Habeeb Mohamed and Zakaria (2019) compared four different controllers (P, PI, PD, and PID). The dorsi-plantar ankle joint flexion is studied using a proportional-integral (PI) model. The test results showed that the PI performed well against the device torque control parameter. This study employs only an actuator and an AFO rig, with no sensors.

In the Manchola et al. (2019) study, the reference detection device was based on footswitches. While foot-mounted sensors have many advantages, such as low cost and simple signal processing, interference with wired connections and incorrect placement can reduce their accuracy. The device proposed here, which consists of a single-axis gyroscope on the foot inspection, not only has functional benefits in terms of aesthetics, research expense, ease of positioning, and donning/doffing times but also offers an efficient gait phase detection method for the control system of a robotic AFO using a Hidden Markov Model (HMM) based algorithm that prevents the detection of robotic AFO.

Proportional-integral-derivative (PID) controllers have been widely utilised for a long time due to their consistent performance, straightforward operation, and sturdy design. Previous studies have shown that the lower limb rehabilitation robot, which uses the PID control approach, can swiftly monitor the target joint angular displacement, demonstrating high performance and good outcomes (Gilbert et al., 2016).

The actual deviation from the desired trajectory based on Equation 2.2 is used as the input to the PID controller, and the plant torque is used as the output (Amiri et al., 2020). As stated below, the error is:

$$e = \theta_{act} - \theta_{des}. \quad \text{Equation 2.2}$$

An illustration of a PID-based control system with a feedback loop for the joint angles at each joint is shown in Figure 2.2.

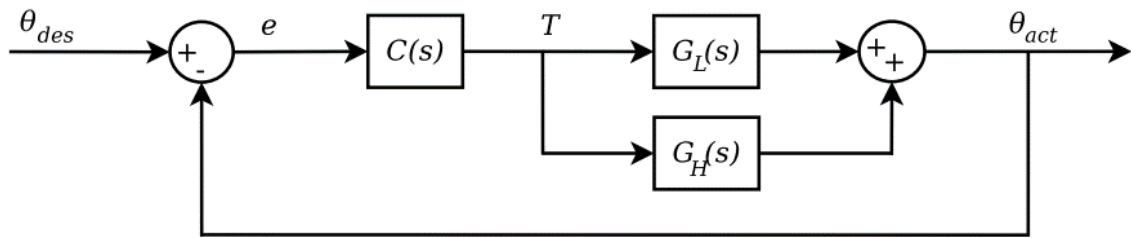


Figure 2.2: Example of PID control system

The PID controller and plant model are depicted as follows in the example equation:

$$C(s) = K_p + \frac{K_i}{s} + K_d s \quad \text{Equation 2.3}$$

$$G(s) = G_L(s) + G_H(s) \quad \text{Equation 2.4}$$

where $C(s)$ is the PID controller and K_p , K_i , and K_d are the proportional, integral, and derivative gains to be tweaked to minimise error, respectively. $G_L(s)$ and $G_H(s)$ are examples of transfer functions of the human exoskeleton and lower limb from the earlier study.

2.7 Research Gap

Ten research papers from the years 2017 to 2022 were used to compare current research with prior research. The categories, input, output, modelling, control system, and performances of the research papers were compared in Table 2.1 and Table 2.2.

Table 2.2 focuses on the development of AFO through simulation, specifically emphasizing mathematical modelling. The researchers conducted by using a large number of freedoms with simulations but the safety and the side effects are questionable. In order to address these problems, the study will model and control AFO through an experimental approach. This strategy ensures that the created technique can be used successfully in rehabilitation training that takes place in real time. Both Table 2.1 and Table 2.2 conducted research on the different types of AFO which is active, passive, and 3D motion capture. By comparing these types of AFO, the study will be conducted using an active type of AFO. When compared to a passive AFO, an active AFO performs more tasks thanks to its intricate structure and algorithm, including movement generation and body balancing (Adiputra et al., 2019).

Various type of modelling has been conducted in the study of AFO such as in Table 2.1 and Table 2.2. The goal of modelling, which has been used in most scientific disciplines, is to forecast the future behaviour of dynamic systems. An important method for exploring, studying, and understanding the world through a formal description of events as a model is system identification, which has been a lively area of study in control theory (Arcolezi et al., 2021). When comparing the system modelling approaches, recent studies have used Neural Networks for ankle foot orthosis while previous studies used mathematical and another dynamic modelling. Extreme parallelism, distributed representation and computation, generalisation capability, adaptability, and integrated contextual information processing are just a few of the desirable properties of neural networks (Mat Darus & Tokhi, 2006). A dynamic system is mathematically modelled through parametric system identification using data from measurements is crucial to create a strong model for most model-based control

approaches. Therefore, this study will compare the parametric and non-parametric modelling of AFO.

Table 2.1 and Table 2.2 show a variety of control systems being used. The desired trajectory of the joints needs to be maintained for the AFO wearer to walk correctly, which calls for a strong control system that minimises steady-state errors and lowers disturbances (Amiri et al., 2020). The Ziegler-Nichols (Z-N) method and adaptive particle swarm optimization (APSO) are combined in the PID controller tuning algorithm to speed up convergence and increase the likelihood of finding the global optimum. The reported work on controlling the mechanical properties of the AFO demonstrated that accurate output reference tracking was obtained utilising well-established approaches, such as a PID controller (Adiputra et al., 2019). Another earlier study used the PID regulation method to compare the input data with the reference value and feedback on the comparison results in real-time, which can enhance the system's accuracy and stability (Geng et al., 2021). Hence, this study will use PID and PID tuned by PSO for controlling the AFO system. Although there are other tuning techniques, PSO is the method of choice for PID tuning because of its track record of effectiveness and success in fine-tuning PID controllers for complex systems. While there are other approaches, it is thought that a targeted strategy utilising PSO is in line with the objectives of the research to regulate the AFO system.

In this research, the gap that will be tackled from this previous research is overall gait detection on walking speed (stance and swing), investigating the modelling of the system using system identification of parametric and non-parametric and identify the most suitable control system to make it simplified according to the specification of available rig and controller used. The problems that the previous research needed to be improvised and

figure out the most simplified solution. The purpose of doing the research using the IMU sensor for gait detection is by using the movement of the axis when the actuator moves. Besides that, the actuator will move in plantarflexion and dorsiflexion movement.

Table 2.1: The comparison of the development of AFO by experimentation

Authors	Category	Input	Output	Modelling	Control system	Performance
Saeed et al., 2022	Active	Voltage	Velocity	Mathematical model using bond graph	Adaptive proportional-derivative RBFNN	Torque tracking accuracy is good. The rise time is 0.8 s. Limitation: no disturbance included.
Jin et al., 2021	3D motion capture	Angle and time span	Length of the data	Back Propagation Neural Network	Improved Particle Swarm Optimization Algorithm	Position tracking is based on a motion prediction algorithm. Limitation: the more complicated motion environment might prevent the current model from adjusting.
Arcolezi et al., 2021	Active	Pulse width	Angular position	MLP NN	The robust integral of the sign of the error (RISE)	Torque tracking accuracy is good. Limitation: the RMSEs were typically higher in the first sessions of healthy subjects, which may have been a result of their discomfort or fear of the electrical stimulation or their voluntary movements.
Habeeb Mohamed & Zakaria, 2019	Active	Current sensing	Angular displacement	System identification (linear)	P, PI, PD, and PID	Torque tracking using the continuous transfer function. The rise time is 3.94 s. Limitation: only use an encoded DC motor and potentiometer.
Thalman et al., 2019	Active	Pressure	Tension force	Finite Element Analysis	Controller logic based	Torque tracking accuracy is good. Limitation: inflation time of the actuator.

Table 2.2: The comparison of the development of AFO by simulation

Authors	Category	Input	Output	Modelling	Control system	Performance
Eldirdiry et al., 2020	Passive and active	Torque	Ankle-joint	Piecewise-linear method, ground reaction force model.	PID	Torque tracking accuracy is good. DOF is 14. Limitation: instead of testing such cases on humans, a simulated robot is used; the pain that could result from a human rolling over his foot or dropping has not been considered.
Amiri et al., 2020	Active	Controller gains	Torque	Lagrangian approach and transfer function	PID controller tuned by Z-N and APSO	Torque tracking accuracy is good. DOF is 6. Limitation: the optimization procedure needs to be repeated for a new trajectory and distinct training circumstances.
Hegarty et al., 2017	Passive	Stiffness, damping, and equilibrium angle	Muscle force and torque profile	Monte Carlo simulations	-	Torque tracking accuracy is good. DOF is 21. Limitation: the restricted ability to quantify the simulations' response to SE rigidity.
Sreenivasa et al., 2017	Passive	Gait recordings	Torque	Modelling the muscles as lumped torque generators	-	Torque tracking accuracy is good. DOF is 10. Limitation: no muscle studies.
Rosenberg & Steele, 2017	Passive and active	Mass and walking speed	Leg and muscle impulses	Musculoskeletal modelling (linear)	-	Torque tracking accuracy is good. DOF is 19. Limitation: to determine potential muscle demand reductions regardless of changes in gait pattern, inverse simulations were limited by one set of experimental cinematics and kinetics for everyone.

CHAPTER 3

METHODOLOGY

3.1 Introduction

The research framework and experimental setup for ankle foot orthosis are described in this chapter. Besides that, the chapter will explain the flowchart of the algorithm, control system, and how the model validation is conducted in this research. Additionally, it will outline the steps taken to complete the project, which will be built around the goals outlined in Chapter 1 of the thesis.

3.2 Research Framework

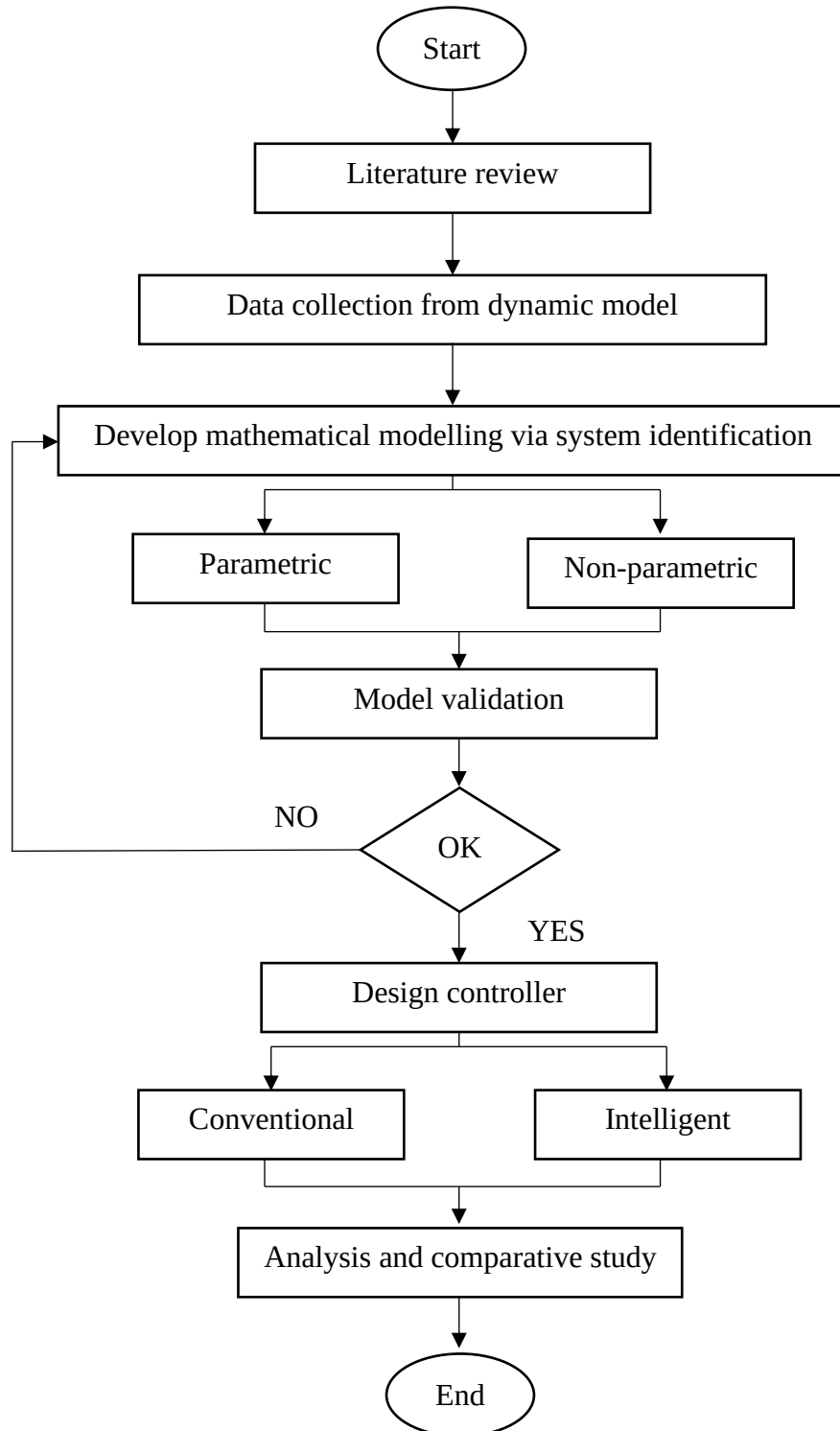


Figure 3.1: Flowchart of the research framework

Figure 3.1 shows the flow chart of the research framework that was used in the modelling and control of AFO. The research began by doing a literature review to compare the characteristics and to find the improvements from the previous research for modelling and control that will be conducted for this project. As depicted in Figure 3.1, the research is divided into three key phases: system identification, controller design, and experimental validation. Prior to that, the experimental rig was designed and built. The rig's construction aims to mimic the dynamics of the real systems. The instrumentation and data collecting equipment was set up and integrated with the AFO rig.

The model of AFO was identified through system identification. The input-output data will be collected from the AFO rig by using an Arduino Mega with MATLAB software. The simulink program was developed as the tool for collecting the data. One output was collected from the motor inertial measurement unit (IMU) sensor which represents the hub angles of the angle movement. The model was estimated by using the parametric modelling to compare with the non-parametric modelling. The parametric modelling used was the PSO algorithm and the non-parametric modelling used was Multilayer Perceptron Neural Network (MLP NN). One-step-ahead prediction (OSA), mean squared error (MSE), and correlation tests were used to verify the model. As the base or plant for the controller design in the simulation environment, the best model was chosen.

Once the model has been selected, a controller for the system was designed by comparing it with conventional and intelligent for torque control parameters. The conventional controller used was the Ziegler-Nichols PID controller and the intelligent controller used was the PSO algorithm. It was conducted and reported on a comparison study

between several controller designs. Comparative research's goal is to simultaneously track the performance differences between the two controllers.

3.3 Experimental Setup

The experimental setup is used for data collection to develop dynamic modelling of the system which addresses objective 1. Details explanation in the next section.

3.3.1 AFO Experimental Rig

This research uses an active AFO that is measured for children. This AFO experimental rig was developed by Tan (2016). The technical drawings and 3D drawings can be observed in Appendix E and F. The functionality of each mechanical design is thoroughly explained in Appendix G. In terms of its intricate structure and algorithm, an active AFO performs more tasks than a passive one, including movement generation and body balancing (Adiputra et al., 2019). The plantarflexion and dorsiflexion movements of the foot are allowed one degree of freedom of motion by the designed AFO. The mechanism supports dorsiflexion movement while limiting uncontrolled plantarflexion. The foot brace's rear side is connected to an actuator, which transfers kinetic energy to the foot brace and eventually enables the mobility function. Since the patient's affected foot is fastened to the AFO braces, when the AFO performs the mobility function, the patient's other foot would be moved as well. Figure 3.2 shows the AFO's dorsiflexion while Figure 3.3 shows the AFO's plantarflexion.

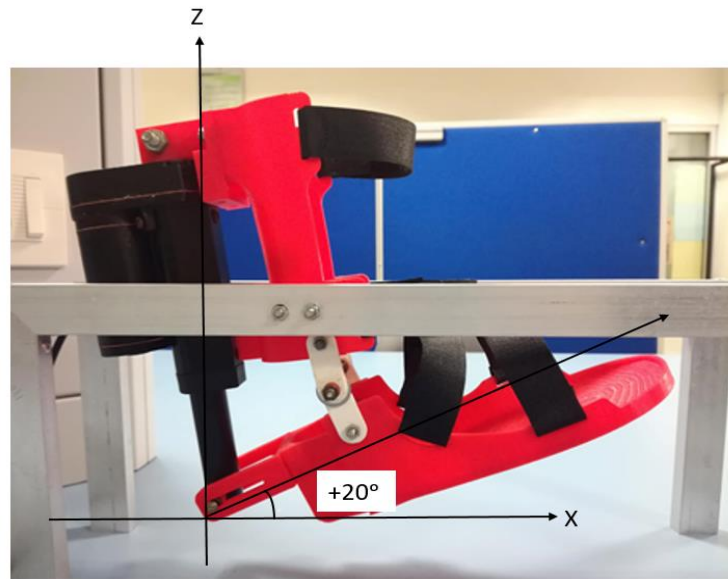


Figure 3.2: AFO's dorsiflexion

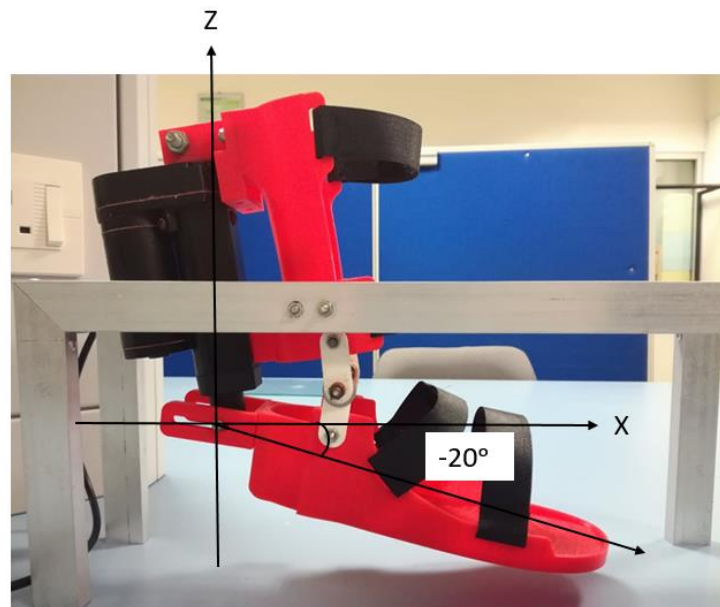


Figure 3.3: AFO's plantarflexion

The proper angle of plantarflexion and dorsiflexion is determined using the biomechanics of the human foot and ankle as well as the angle of the ankle when walking. According to Thalman et al. (2019), the ankle is assumed to be in its most natural, relaxed position at an angle of 115° , which allows for 20° of dorsiflexion and 30° of plantarflexion.

A new, effective training AFO (IT-AFO) with dynamic control improves the gait's overall kinematics, which benefits the hemiparetic lower limb by improving the loading response and moving weight there (Hsu et al., 2020). The minimum and maximum angles that the IT-AFO can affect are 21.810° and 21.342° , respectively. These outcomes might enhance gait training in patients with hemiplegia brought on by a stroke.

For proximal-distal misalignments between 20° dorsiflexion and 20° plantar flexion in both 2D and 3D models of AFO, previous study used relative motion between the tracking sites (Fatone et al., 2014). Anterior-posterior and proximal-distal misalignments of the AFO's mechanical axis and the associated relative motion with respect to the lower leg can be up to 20 mm. This is accomplished by taking into account the ankle angles that develop during typical gait. Since the stiffness distribution within the model might affect estimating the AFO joint angle and predicting the forces imposed on the leg, it is essential to include actual leg tissue stiffness in the model to improve model accuracy.

As a result, in this study's foot brace, plantarflexion would occur at -20° when the actuator is fully extended to a length of 50 mm, dorsiflexion would occur at $+20^\circ$ when the actuator is fully retracted, and the foot brace would remain in its natural position at 0° . Table 3.1 shows the AFO movements in relation to the axis and angle.

Table 3.1: AFO movements in relation to axis and angle

Movement	Axis	Angle
Dorsiflexion	X-Z	$+20^\circ$
Plantarflexion	X-Z	-20°

3.3.2 Data Acquisition

In this research, data acquisition used a microcontroller to manage the data collected from actuators and sensors. Although the clock speeds of the Arduino Mega and Arduino Uno are the same at 16 MHz, they differ in terms of memory and storage. While Arduino Uno's flash memory is only 32 kB, Arduino Mega's is 256 kB. Given the size of the code of the research, an Arduino Mega with more memory is preferable.

The Arduino Mega 2560 R3 board is the microcontroller made in Italy by the affiliates of Arduino S.R.L.. The microcontroller serves as a command centre, processing input, sending signals to the actuator and sensors, and ultimately carrying out the task. First, a signal was sent to the Arduino Mega 2560 R3 board from Simulink. After processing the signal, the microcontroller sends the electrical signal to the actuator and sensors. As a result, the actuator and sensors carry out the task in accordance with the Simulink programming. The features of the microcontroller are shown in Table 3.2 and the Arduino Mega 2560 R3 board is shown in Figure 3.4.

Table 3.2: The features of the microcontroller

Microcontroller	ATmega2560
Operating Voltage	5V
Input Voltage	7-12V
Digital I/O Pins	54
Analog Input Pins	16
DC Current per I/O pins	20 mA
Flash Memory	256 kB of which 8 kB used by the bootloader
SRAM	8 kB
EEPROM	4 kB
Clock Speed	16 MHz



Figure 3.4: Arduino Mega 2560 R3 board

3.3.3 Linear Actuator

The intended driven ankle foot orthosis employs a linear actuator of the AFO. A DC motor that only moves in one linear direction from fully extended to fully retracted is a linear actuator. The main benefits of a linear actuator include high torque, precise stroke duration, and long-term dependability. The chosen linear actuator has a 51 mm stroke length and a 900 N load rating. The LIN-ACT1-02 Windynation linear actuator, 12 VDC, weighs 102 kg. Figure 3.5 shows the LIN-ACT1-02 Windynation linear actuator and Table 3.3 shows the specification of the LIN-ACT1-02 Windynation linear actuator.



Figure 3.5: LIN-ACT1-02 Windynation linear actuator

Table 3.3: Specification of the LIN-ACT1-02 Windynation linear actuator

Specification of the LIN-ACT1-02 Windynation linear actuator	
Stroke length	51 mm
Rated load	900 N
Maximum travel speed	10 mms ⁻¹
Rated voltage	12 VDC
Maximum current draw	≤ 2.5 A
Minimum install dimension	161 mm
Mounting holes	8 mm
Limit switches	Not adjustable
Operating temperature	-26°C to +65°C
Duty cycle	25%
Noise level	≤ 50 dB

3.3.4 Motor Driver

A driver board that contains all the electrical parts required to control a motor is known as a motor driver. In order for the actuator to receive signals from the microcontroller, they must first pass through the motor driver. This research uses L298N motor driver as shown in Figure 3.6. Two DC motors may be simultaneously controlled for speed and direction with the dual H-Bridge motor driver L298N. The module can power DC motors with peak currents up to 2 A and operating voltages between 5 and 35 V.

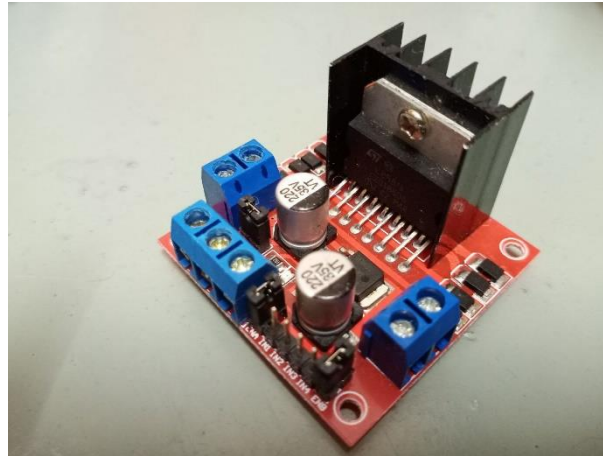


Figure 3.6: L298N motor driver

3.3.5 IMU Sensor

Figure 3.7 illustrates the installation of an inertial measurement unit (IMU) sensor on the AFO rig's footplate. For the purpose of detecting gait cycles, the IMU sensor (MPU 6050) was used to measure acceleration. For input-output purposes, this sensor was connected to an Arduino Mega R3. The Arduino Mega R3's serial data (SDA) and serial clock (SCL) pins were connected to the SDA and SCL pins of the IMU. The Arduino Mega R3 GND was connected to the IMU's ground (GND) pin. The Arduino Mega R3's 5V pin was connected to the IMU's VCC pin. Pitch and roll readings were taken as the measurements.

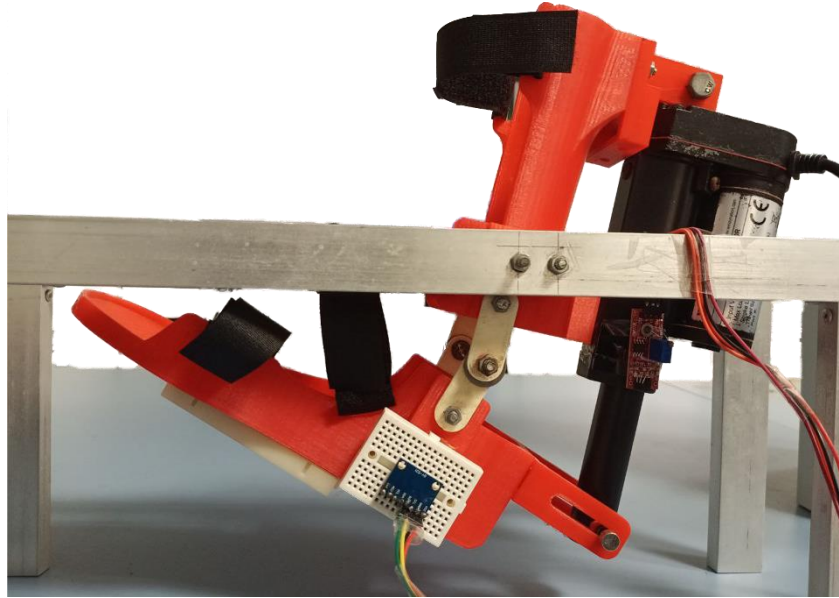


Figure 3.7: Location of IMU sensor placed on AFO rig

3.3.6 Circuit Design

For circuit design, Fritzing software version 0.9.3 is utilised. The circuit design is displayed in Figure 3.8. The components used for this research are Arduino Mega 2560, IMU sensor, magnetic hall effect sensor, linear actuator, voltage sensor, motor driver L298N, 12V battery, breadboard, and jumper wires. The linear actuator is attached to the AFO rig. The Arduino Mega is connected to the computer directly via a USB type B cable.

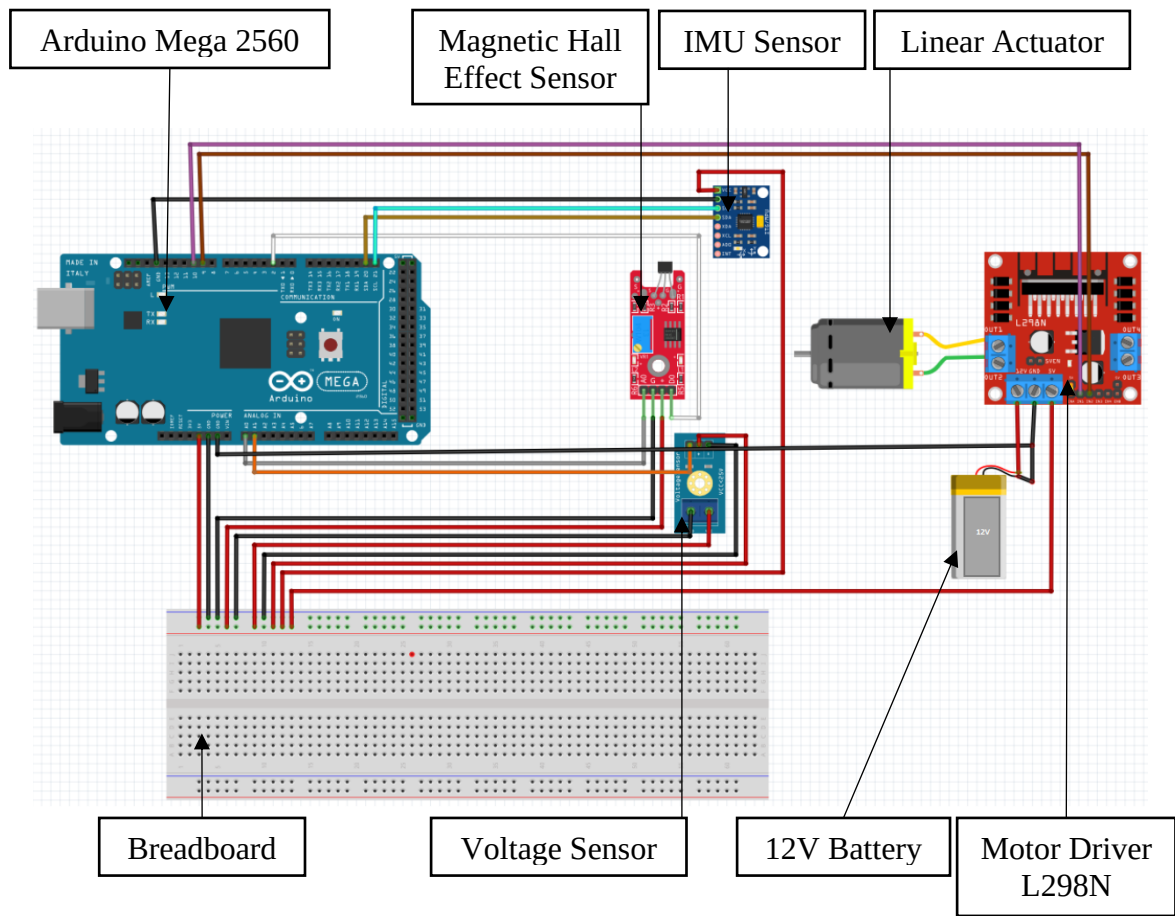


Figure 3.8: Circuit design

3.4 Software Development

This research used MATLAB R2020b edition. In order to enable Arduino programs to sync with MATLAB, there was some hardware support package needed to be added on. Figure 3.9 shows the support package that is required to add on. Three of the add on were Rensselaer Arduino Support Package Library (RASPLib), MATLAB Support Package for Arduino Hardware, and Simulink Support Package for Arduino Hardware.

Name	Type	Author
 Rensselaer Arduino Support Package Library (RASPLib) version 1.52	Collection	Joshua Hurst
 MATLAB Support Package for Arduino Hardware version 20.2.0	Hardware Support Package	
 Simulink Support Package for Arduino Hardware version 20.2.2	Hardware Support Package	

Figure 3.9: The support package

After all the installation setup is completed, this research proceeds with the data collection. Figure 3.10 shows how the data collection was done to input voltage to the linear actuator from Simulink MATLABR2020b. The full Simulink setup for data collection can be observed in Appendix A.

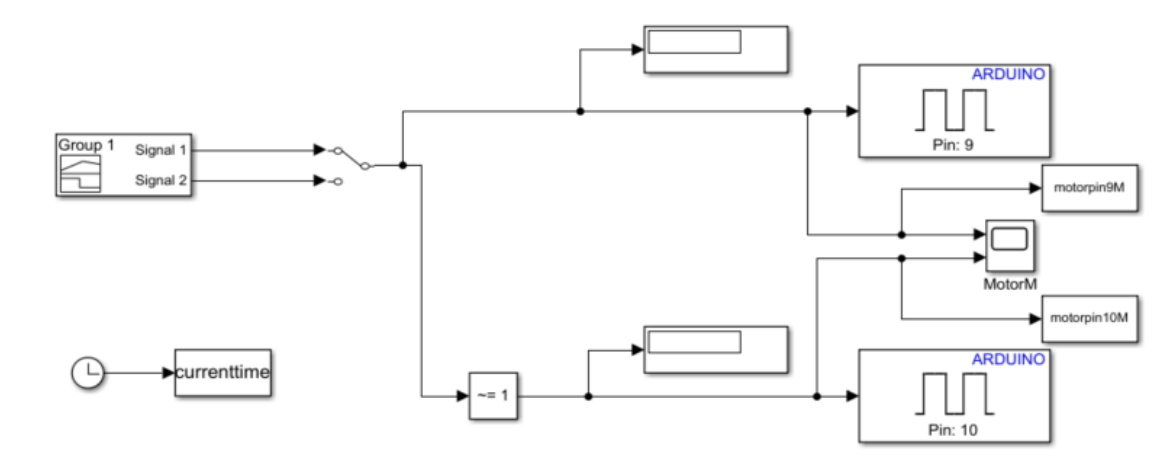


Figure 3.10: Data collection (input) using Simulink

The data collection begins after clicking run the Simulink file. First and foremost, the USB cable connected to the Arduino must be connected before clicking the run button. All the data from various sensors and actuator. Due to the reasons that the linear actuator did not have an encoder, the data collection requires switching pins 9 and 10 manually for plantarflexion and dorsiflexion movement. All the data was sent to the MATLAB workspace.

Simulink MATLABR2020b had a built-in IMU sensor, and it is better for data collection of pitch and roll accurately. Figure 3.11 shows how the data collection was done from the IMU sensor. The hall effect sensor Simulink setup for data collection can be observed in Appendix B. The data from the hall effect sensor was used as the control experiment to ensure the data collected from IMU does not deviate.

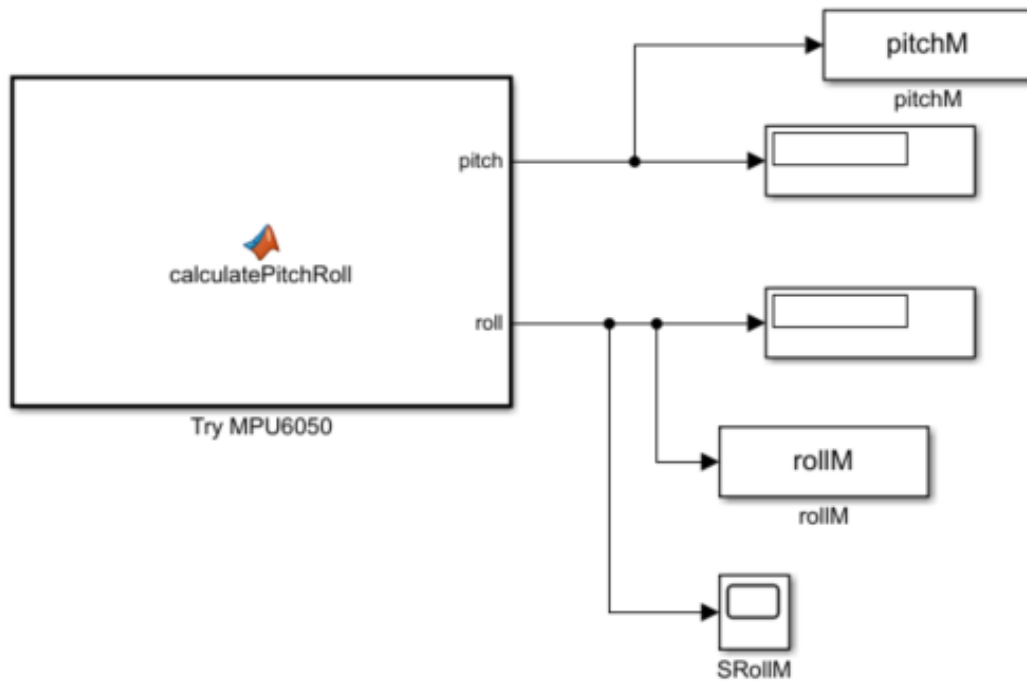


Figure 3.11: Data collection (output) using Simulink

To get the gait movement readings of the IMU sensor accurately, a complementary filter was used to fuse the accelerometer and gyroscope data. The filter coefficient was tuned to 0.98 to improve the result as shown in Appendix C. The complementary filter, a low-pass and high-pass filter combination, is a computationally cheap sensor fusion technique. The dynamic motion characteristics of the gyroscope complement those of the accelerometer. (Narkhede et al., 2021). Figure 3.12 shows the low-pass filter (LPF) and high-pass filter (HPF). Two inputs, x_1 and x_2 , which are low- and high-frequency noise-corrupted versions

of the signal $\hat{x}$, make up the fundamental structure of the complementary filter as shown in Figure 3.13. Equation 3.1 gives the complementary filter output x .

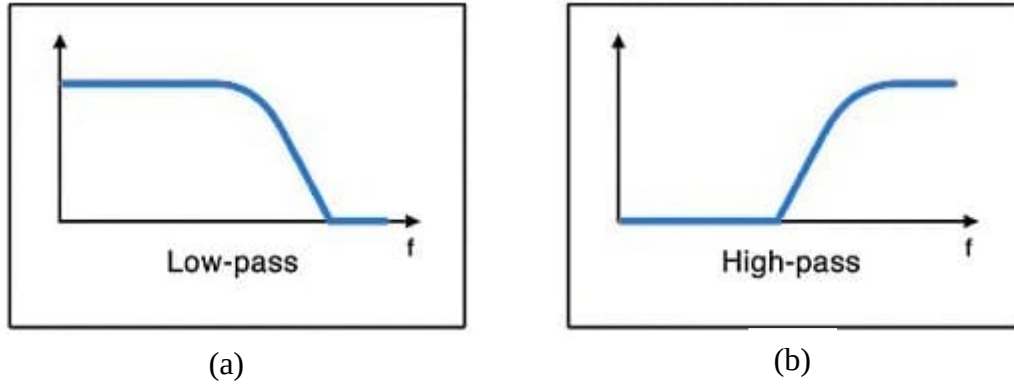


Figure 3.12: Low-pass filter (a) and High-pass filter (b)

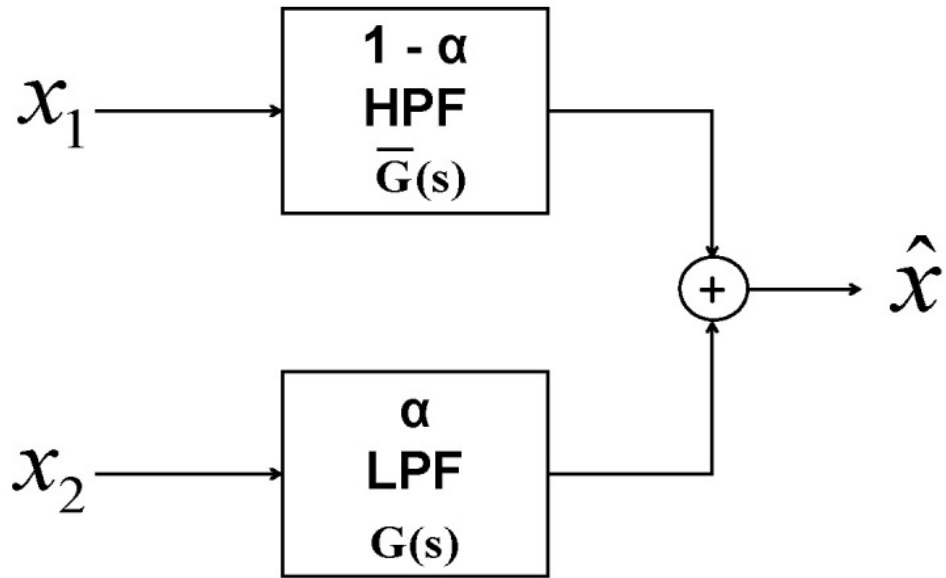


Figure 3.13: The fundamental design of a complementary filter (Narkhede et al., 2021)

$$\hat{x} = x_1 G(s) + x_2 \bar{G}(s) \quad \text{Equation 3.1}$$

The transfer function for the low-pass filter in this case is $G(s)$, while the transfer function for the high-pass filter is $\bar{G}(s)$, resulting in $G(s) + \bar{G}(s) = 1$. Gyroscope estimates $\dot{x}_g (\dot{\phi} \ \dot{\theta} \ \dot{\psi})$ and accelerometer estimates $x_a (\phi_a \ \theta_a \ \psi_m)$ are applied at x_1 and x_2 , respectively, using this structure of a complementary filter. The state vector of the gyroscope, $\dot{x}_g$ consists of $\dot{\phi}$ represents the roll axes of the gyroscope, $\dot{\theta}$ represents the pitch axes of the gyroscope and $\dot{\psi}$ represents the yaw axes of the gyroscope. The state vector of the accelerometer and magnetometer sensors, x_a consists of ϕ_a represents the roll axes of the accelerometer, θ_a represents the pitch axes of the accelerometer and ψ_m represents the yaw axes of the magnetometer. Equation 3.2 provides an analytical estimation using a linear complementary filter structure, $\hat{x}$.

$$\hat{x} = \alpha \left(\int \dot{x}_g dt \right) + (1 - \alpha)x_a \quad \text{Equation 3.2}$$

The weighted factor for gyroscope and accelerometer estimates is determined by parameter $\alpha \in [0 \ 1]$. Equation 3.3 is a transfer function representation of the linear complementary filter (LCF) estimate.

$$\hat{x} = \frac{\tau s}{1 + \tau s} \left(\frac{\dot{x}_g}{s} \right) + \frac{1}{1 + \tau s} (x_a) \quad \text{Equation 3.3}$$

In this case, the separate transfer functions of LPF and HPF can be shown as:

$$LCF_HPF = \frac{\tau s}{1 + \tau s} \quad \text{Equation 3.4}$$

$$LCF_LPF = \frac{1}{1 + \tau s} \quad \text{Equation 3.5}$$

The complementary filter System objectTM in MATLAB combines sensor input from the accelerometer, gyroscope, and magnetometer to determine the orientation and angular velocity of the device (The MathWorks, 2019). By constructing the complementary filter object and setting its parameters, it uses this object to estimate orientation. The object is then called with arguments as though it were a function.

Using data from the accelerometer and gyroscope, this model calculates the pitch and roll angles (The MathWorks, 2020). Table 3.4 shows the alignment of the IMU sensor axes. The pitch and roll angles can be determined depending on the orientation of the sensor after the model has been implemented on the hardware. Pitch and roll angles are between $[-90^\circ, 90^\circ]$. Both values should be zero if the sensor is kept constant and parallel to the ground, for example with the z-axis perpendicular to the ground. With relation to the pitch axis, the pitch angle is positive for rotation going anticlockwise and negative for rotation going clockwise. The same applies to the roll angle.

Table 3.4: the alignment of the IMU sensor axes (The MathWorks, 2020)

Axis	Alignment
X	Longitudinal axis (roll axis)
Y	Transverse axis (pitch axis)
Z	Vertical axis (yaw axis)

While the sensor is kept firm on the ground with the Z-axis perpendicular to the ground, the sensor is read several times and offset values for pitch and roll angle are calculated, to filter out noise at the beginning of the program's execution. Later, when pitch and roll angle are calculated at each step, these values are subtracted. This somewhat reduces

continuous noise in sensor readings. The roll angle readings were the ones that will be used in modelling as the movement of the ankle foot orthosis was more stable based on the arrangement of the IMU on the rig.

After the data collection was completed, this project was ready for modelling. The data that was chosen for modelling is 6 seconds per movement which is approximately 12 seconds per cycle. The modelling begins with delay 1 and model structure 2 2 1 with an increment of 2 until 8 8 1 for NN-MLP and ends at delay 9 with a model structure 8 8 1. Another modelling was conducted using PSO modelling with model order 1 until 10. Then, the best modelling result was chosen for the control stage. Modelling and control of AFO will be discussed in Chapter 4.

3.5 System Integration

The experimental setup and the study's schematic diagram are shown in Figure 3.14 and Figure 3.15, respectively. An ankle foot orthosis (AFO) rig with a single degree of freedom of movement is used in this study. Foot motion will support both dorsiflexion and plantarflexion. An actuator that transfers kinetic energy to the foot brace serves as the mobility function and is fastened to the underside of the foot brace. The rig is set up so that plantarflexion is at -20° and dorsiflexion is at $+20^{\circ}$.

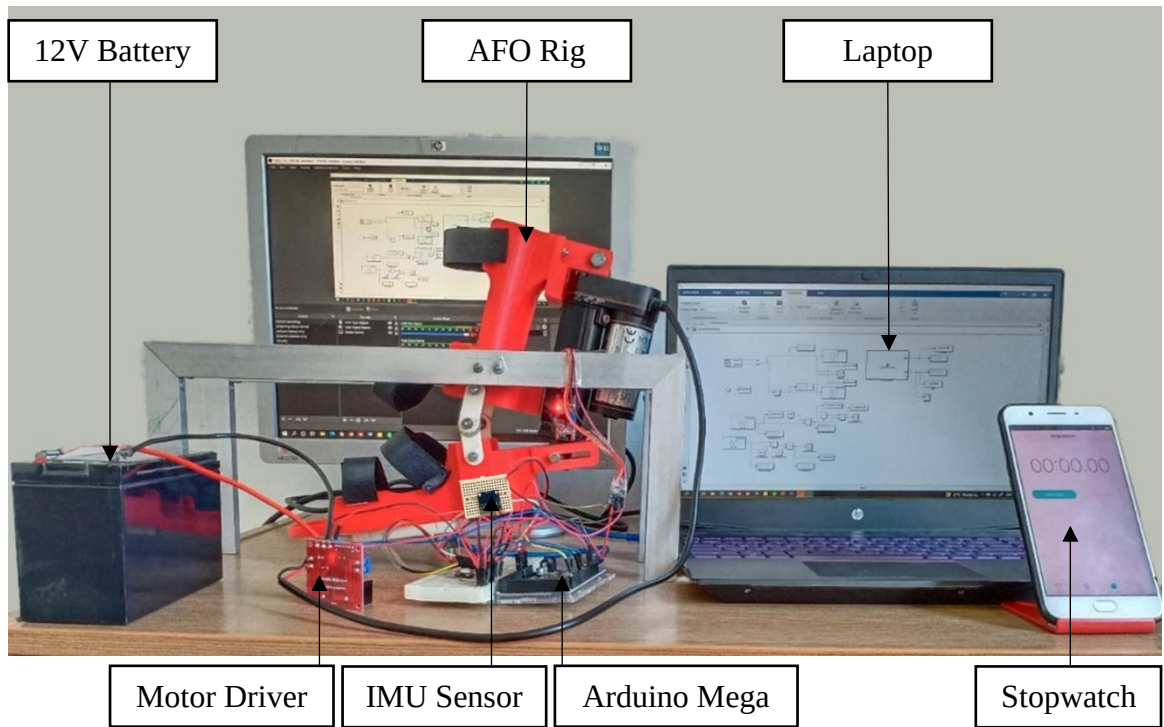


Figure 3.14: The experimental setup of AFO

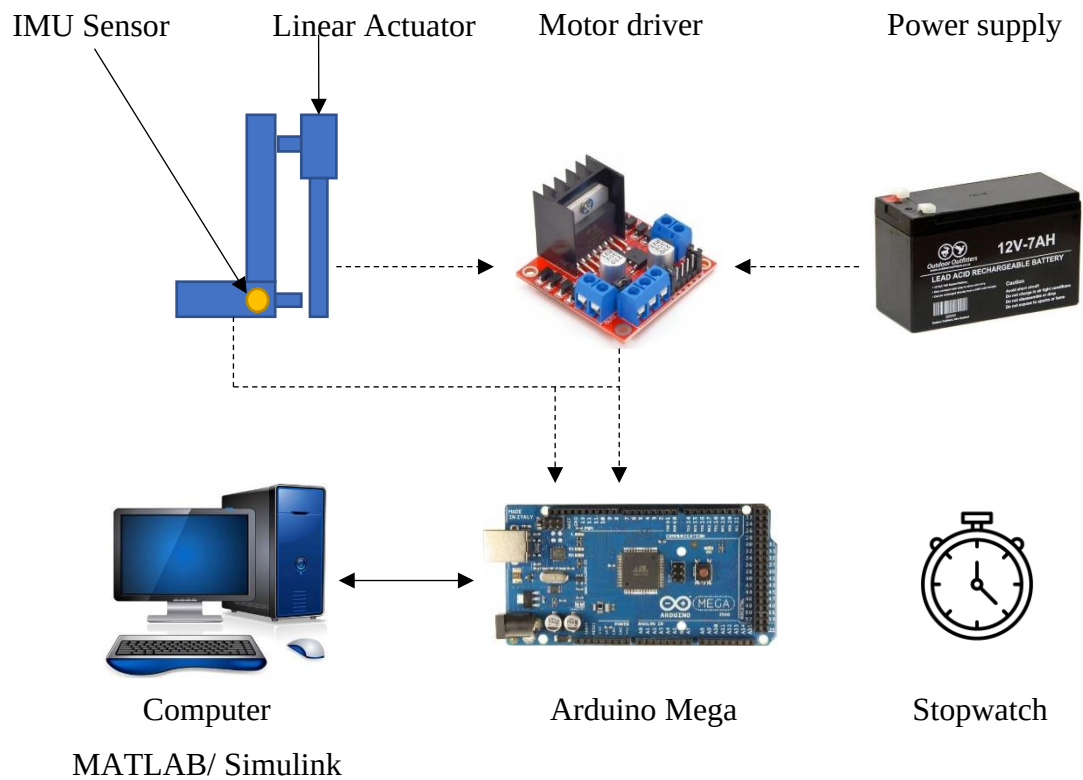


Figure 3.15: Schematic diagram of AFO

A linear actuator is used for the intended driven AFO. The main benefits of a linear actuator include high torque, precise stroke duration, and long-term dependability. The selected linear actuator has a 900 N load rating and a 51 mm stroke length. The stroke was chosen based on the selected angle movement which is 20° . A free-body diagram was used to calculate the necessary stroke length and set the position of the actuator, which serves as the artificial muscle of the lower limb, in order to give the AFO the desired range of motion (see Appendix D for a free body diagram). The 12 VDC, 225-pound LIN-ACT1-02 Windynation linear actuator is made of 6061 Aluminium Exterior. Because it has a supply range of 5 V to 35 V and can handle 2 A continuous current per channel, the L298N motor driver complements the linear actuator very well. The motor driver receives power from a 12 V power supply. A magnetic hall effect sensor is attached to the side of the actuator to track the distance travelled when the actuator moves.

The IMU sensor (MPU 6050) located at the side of the footplate measures the position of the ankle as the actuator moves. To collect input and output data, the Arduino Mega is connected to the motor driver and the sensors. Connecting the Arduino Mega to the computer requires a USB cable. Modelling development is carried out in a MATLAB / Simulink environment.

Experimental data on input-output was obtained using the AFO test rig, which is covered in this chapter. Figure 3.16 depicted the block diagram used to gather the data. The requisite energy for the linear actuator was supplied by means of bang-bang signals. The output representing the hub angle was accumulated using an IMU sensor. The experiment was conducted during a 60-second period with a sample time of 0.03 that consists of 2189 datapoints. MATLAB is utilised for the online real-time application. Interface between

input-output signals and computer is achieved through the use of the Arduino Mega. A motor driver receives the actuation signal from the Arduino Mega after receiving the digital input from MATLAB.

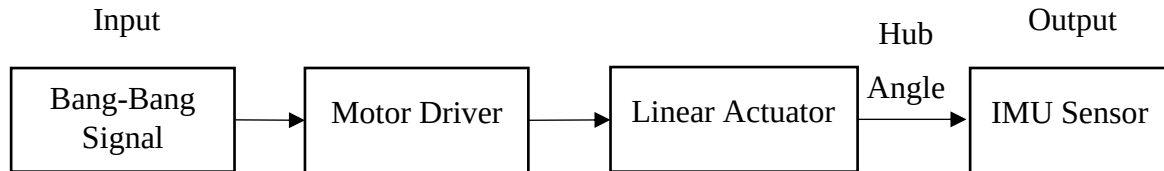


Figure 3.16: Block diagram for data collection of AFO

The data were collected for five cycles of 3 seconds, 4 seconds, and 6 seconds for half-cycle movement either dorsiflexion or plantarflexion. The experiment was repeated for 3 times for all 3 seconds, 4 seconds, and 6 seconds. One cycle consists of both dorsiflexion and plantarflexion which double the time. Table 3.4 shows the average time of one complete walking cycle of normal children under the age of 12 is 1 second. However, one complete of gait movement which includes the stance and swing phase for disabled children is subjected to the condition of the children. It is anticipated that the children that use AFO for rehabilitation will need to take a longer time to complete one cycle. Thus, the readings 3 seconds, 4 seconds, and 6 seconds per half cycle which is 6, 8, and 12 times slower time to complete one cycle are considered in this research.

According to data from Jabatan Kebajikan Masyarakat (2023), there were 6044 children in Malaysia between the age of 6 till 12 has physical disability up to 31 January 2023. However, this broad classification does not represent with the specific focus of our research on ankle and foot disabilities. The calculated sample size using Raosoft, Inc (2004), is 67, encompassing various aspects of physical disability such as ankle, foot, knee, hip, hand, shoulder, arm, and wrist. Therefore, this study included only 14 participants,

constituting 20% of the total sample size. This quantity is adequate, considering the practical limitations associated with participant availability during the data collection phase.

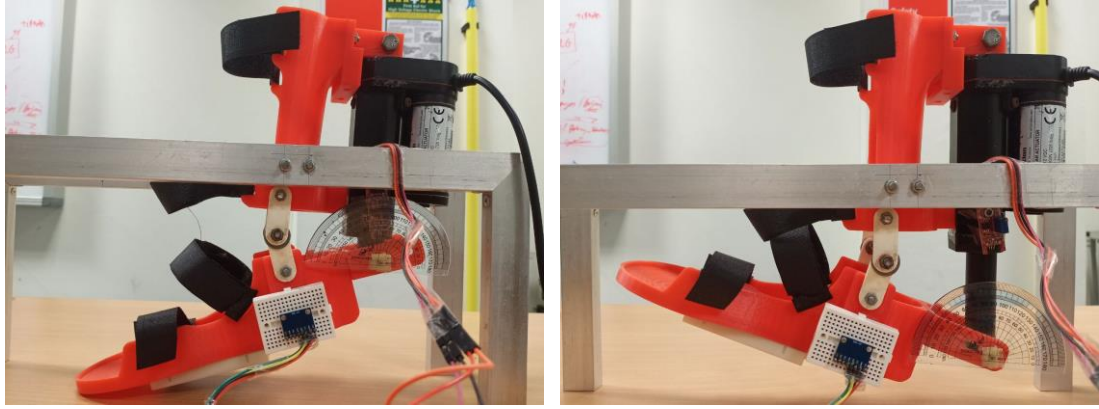
Table 3.5: The average time of 1 complete walking cycle of normal children under the age of 12

No	Gender	Age	Height (cm)	Weight (kg)	Time to complete 1 cycle (s)
1	Female	10	140	30	1.05
2	Male	9	136	28	1.02
3	Female	10	153	65	1.09
4	Female	11	160	72	1.26
5	Female	11	143	21	1.02
6	Male	9	120	30	1.07
7	Male	10	120	35	1.08
8	Female	6	112	18	1.17
9	Male	8	123	20	1.10
10	Female	11	140	30	1.04
11	Female	8	112	21	2.00
12	Male	11	132	33	1.06
13	Male	10	121	19	0.27
14	Male	11	132	33	1.17
Average			131.71	32.5	1.10
Mean, μ			131.71	32.5	1.10
Standard deviation, σ			10.71	16.35	0.24

The smartphone that acts as a stopwatch was important to switch the movement of both dorsiflexion and plantarflexion in Simulink. Although this setup used the magnetic hall effect sensor module and voltage sensor module, both readings did not count for the modelling and control stages. The readings from the voltage sensor module were only for monitoring the voltage of the system during the experiment. The readings from the magnetic hall effect sensor module were not as smooth as the IMU sensor reading as it was affected by the vibration and magnetic device. Thus, IMU sensor readings and the actuator movement with time were the only data used for modelling and control stages.

3.6 Verification of Experimental Setup

The verification of the experimental setup was done by measuring the AFO angle using a protractor to make sure the angle was correct before the experiment begin as shown in Figure 3.17. The accuracy of the protractor is 0.5 cm or about 0.5°. If the angle is not accurate, the linear actuator was adjusted until the angle is on point. The experiment was done by monitoring closely the time on the stopwatch via the smartphone when switching pin 9 and pin 10 of the actuator. If there is any error occurs during data collection, the experiment was repeated until the data collection for 5 cycles of gait is completed.



(a)

(b)

Figure 3.17: (a) Dorsiflexion (b) Plantarflexion

3.7 Development of System Modelling

In this research mathematical modelling is derived from System Identification. Two different algorithms were used that is Particle Swarm Optimization and Neural Network. The details of the algorithm were further details in the following section.

3.7.1 Particle Swarm Optimisation

PSO a population-based stochastic optimisation method, was created by Kennedy and Eberhart (Yatim et al., 2013). Figure 3.18 depicts the stages of a PSO's operating procedure.

PSO begins with a set of random objects that 'fly' through the search region of an optimisation issue. To evaluate a particle's location in the search space, an objective function is used to calculate the fitness value of the particle. Mathematically, the fitness equation for PSO, $fit(P_i)$ appears like this:

$$fit(P_i) = f(X_i) \quad \text{Equation 3.6}$$

where X_i is the d-dimensional factor. Every particle in PSO has a position and velocity in the search space, and the fitness equation is used to update each particle's position and

velocity. By iteratively altering the position and velocity of the particles in the swarm while utilising the fitness equation to assess their performance, PSO seeks to discover the best possible answer.

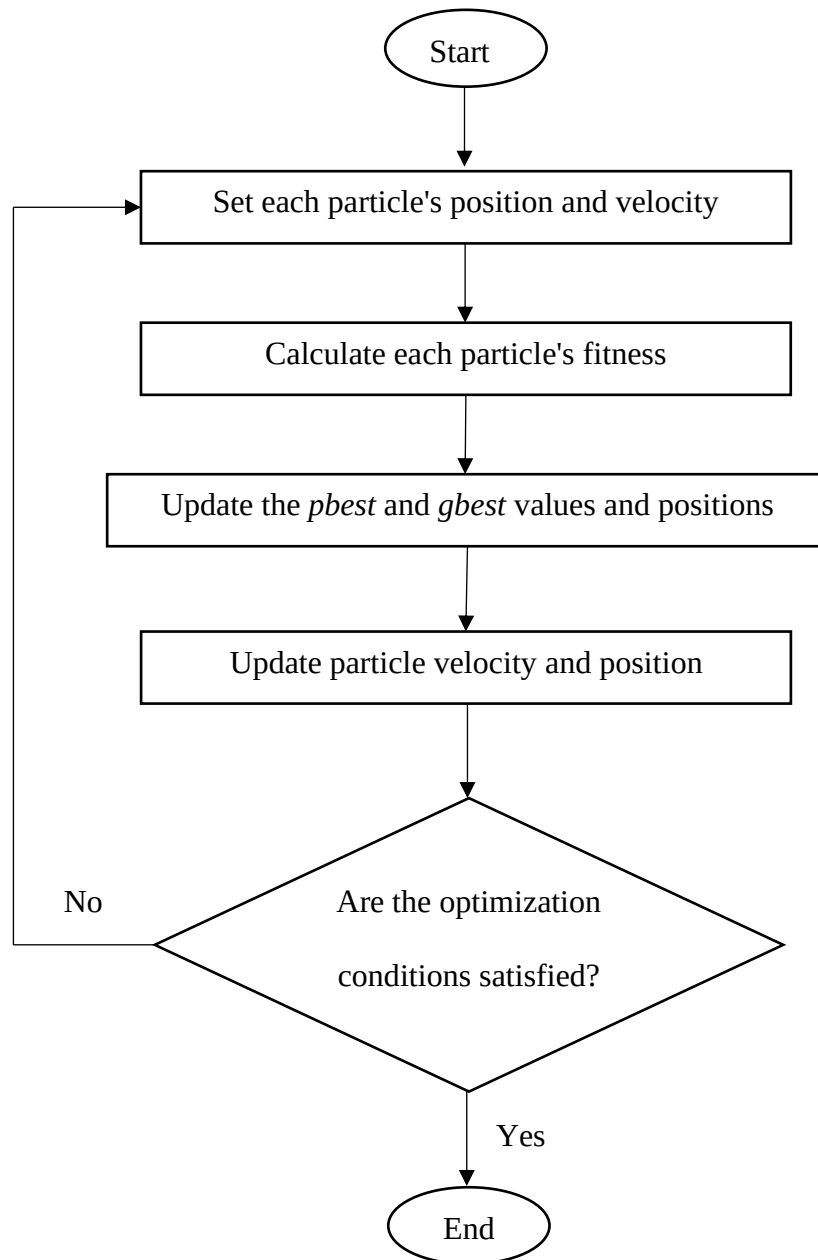


Figure 3.18: The flowchart of the PSO algorithm

The fitness equation is an essential part of the PSO algorithm since it chooses which particles are more likely to be chosen as leaders, which affects how the rest of the swarm

behaves. PSO can more effectively locate the global optimum in a particular problem space by optimising the fitness equation.

The i -th particle's position is denoted by the d -dimensional vector $X_i = (x_{i1}, x_{i2}, \dots, x_{id})$ in the problem space, where $i = 1, 2, \dots, q$ (q is the number of particles) (Shi & Eberhart, 1998). As a result, the value of i depends on the quantity of particles. Finding the ideal spot will have a better chance of happening with more particles.

In every iteration, particles are updated with two "best" values. The first is named $pbest$, and it remembers the best spot a particle has visited thus far. Another 'best' value is the global best, or $gbest$, attained by any particle in the population thus far. Utilizing their memory of the optimal positions of $pbest$ and $gbest$, the particle is then propelled toward those two best values by updating the particle position and velocity using the following set of equations:

$$v_{id}(t) = wv_{id}(t-1) + C_1 \cdot \text{rand.} (P_{id} - x_{id}(t-1)) + C_2 \cdot \text{rand.} (P_{gd} - x_{id}(t-1)) \quad \text{Equation 3.7}$$

$$x_{id}(t) = x_{id}(t-1) + v_{id}(t) \quad \text{Equation 3.8}$$

$v_{id}(t)$ and $x_{id}(t)$ are the current velocity and position vector of the i -th particle in the d -dimensional search space, respectively. C_1 and C_2 are acceleration coefficients that are normally. $C_1 = C_2 = 2$, while rand is a random value between 0 and 1. The inertia w acts as a recollection of the particle's former orientation, preventing it from abruptly changing direction. A high w number encourages worldwide exploration and exploitation, whereas a low w value encourages a local search. The conventional strategy is to establish a balance between global and local search by decreasing w linearly over the search process. Inertia decreases with time may be stated as:

$$w(t) = w_{start} - \frac{(w_{start} - w_{end})}{T_{max}} \quad \text{Equation 3.9}$$

where w_{start} and w_{end} are the starting and ending points of linearly decreasing inertia weight, and T_{max} is the maximum number of time steps the swarm is permitted to search.

3.7.2 Artificial Neural Network

Developing and building a neural network is a methodical process (Manalo et al., 2016). MLP NN is utilized in this research. The flowchart of the Classification and Testing of the MLP Neural Network is shown in Figure 3.19.

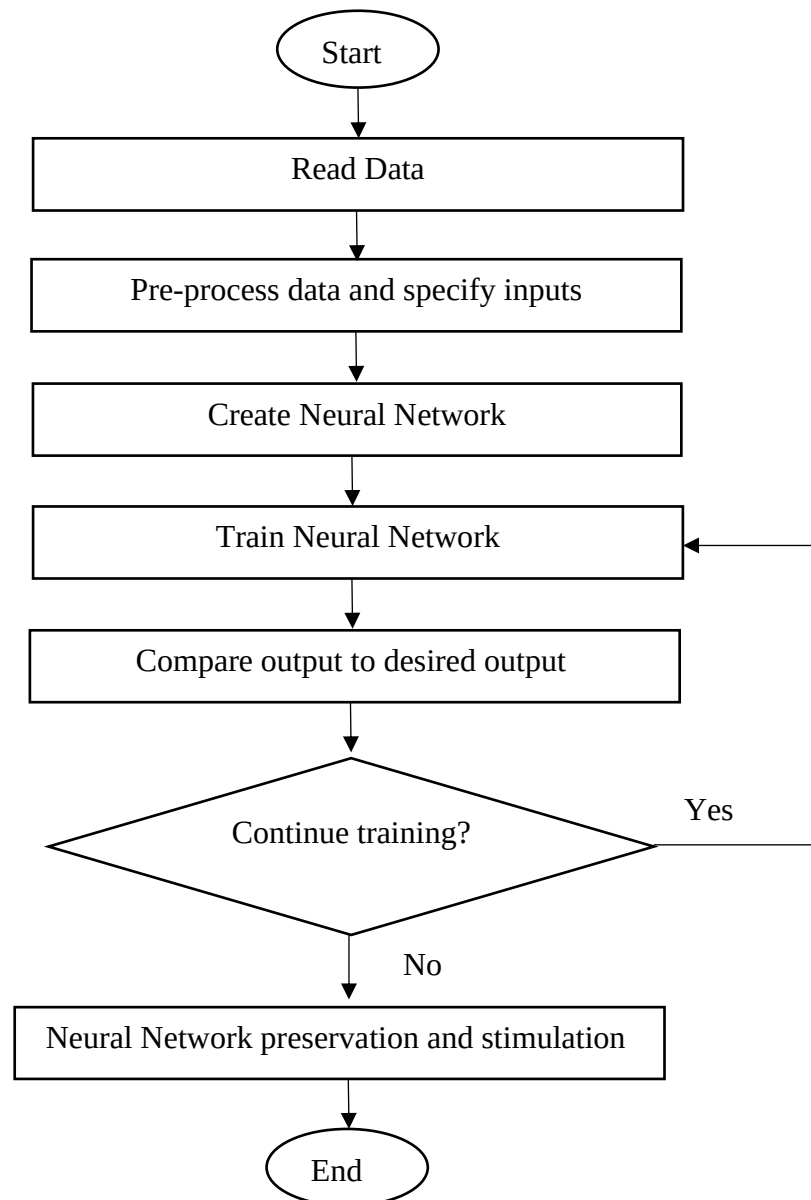


Figure 3.19: The flowchart of the MLP Neural Network

To begin the classification process, the retrieved data will be read and entered into the MATLAB program. The neural network will be built using parameters and inputs that have been supplied. The neural network must then be trained. The neural network's output will be compared to the desired output after a set length of time. The training will continue

if the experimental outcome does not match the expected output. If the required output is obtained, it is preserved and simulated.

Back propagation is used in MLP NN modelling for AFO systems. The design for the MLP neural network algorithm is displayed in Figure 3.20.

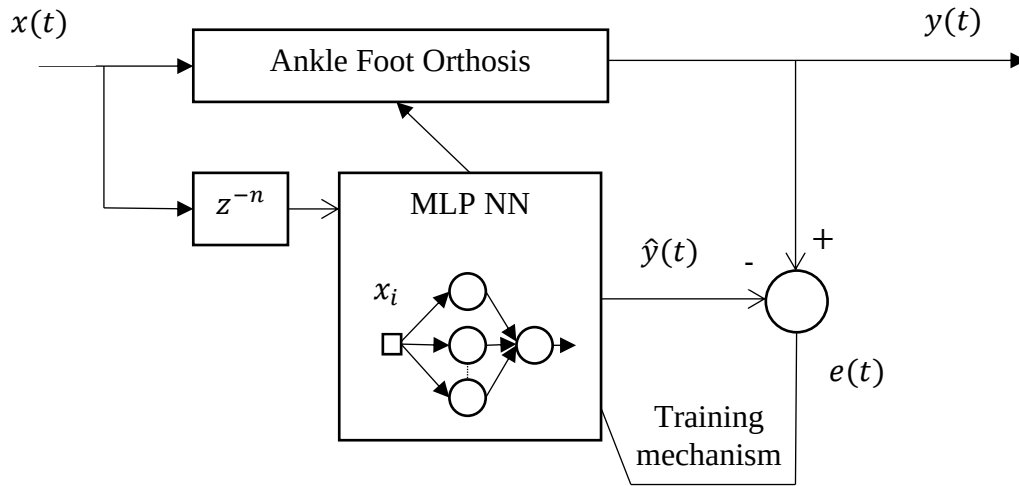


Figure 3.20: MLP-NN algorithm diagram

MLP is the most often used neural network type because it can anticipate a very complicated formula connection with a simple model. In dynamic neural networks, the system's past behaviour and present inputs determine the outputs at a given time. As seen in Figure 3.21, dynamic systems have implanted memory that retains the previous input outcomes. Tapped delay lines (TDLs) can be employed in the network's intermediate nodes, output signal path, or input signal path.

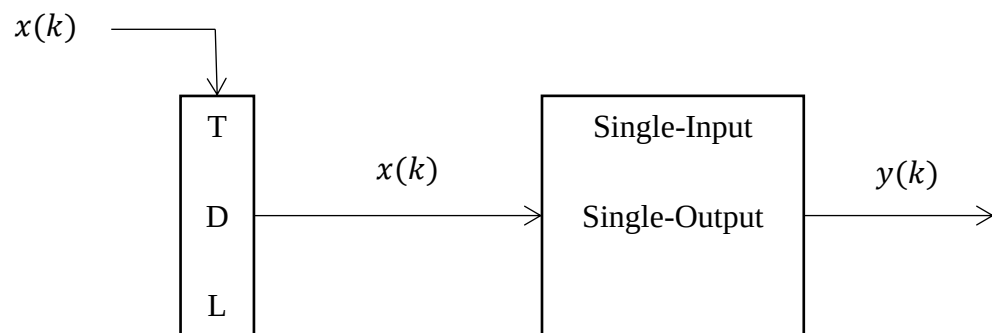


Figure 3.21: Feed-forward dynamic neural network architecture

The MLP is the most well-known member of the neural network family because it can estimate a highly complex formula connection using a straightforward model (Jamali et al., 2018). Figure 3.22 shows the basic function of the MLP-NN structure. The MLP is made up of several intermediate or hidden layers in between the input layer of the NN, which is made up of one layer of nodes, and the output layer of the NN, which is made up of a second layer of nodes.

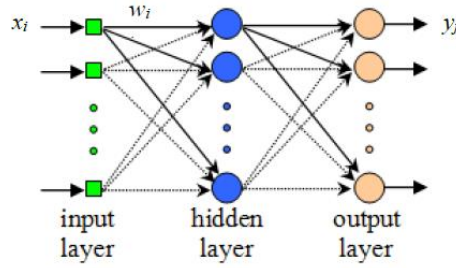


Figure 3.22: The basic function of MLP-NN structure (Jamali, 2017)

Equation 3.10 shows the network layer is made up of an input layer x_i , an output layer y_j , and a hidden layer w_{ij} with changing strength weights. The attributes of the function $f(\cdot)$ are as follows: linear, threshold, sigmoid, hyperbolic tangent, and radial basis. The mapping allows the network to forecast the output, $\hat{y}$, as precisely as feasible. Equation 3.11 depicts the MLP output:

$$y_j = f\left(\sum_{j=1}^q W_{ij} \cdot x_i\right) \quad \text{Equation 3.10}$$

$$\hat{y}(w, W) = F_i\left(\sum_{j=1}^q W_{ij} \cdot f_j\left(\sum_{i=1}^m w_{ij}X_i + w_{j0}\right) + W_{i0}\right) \quad \text{Equation 3.11}$$

Despite requiring more memory than other methods, Levenberg-Marquardt (LM) is preferred for network training due to its fast convergence time. Based on Equation 3.12, the LM optimises the error by minimising the residual, $\varepsilon(t, \theta) = y(t) - \hat{y}(t, \theta)$:

$$L^i(\theta) = \left(\frac{1}{2N}\right) \sum_{t=1}^N \varepsilon^{-2}(t, \theta) \approx P_N(\theta, Z^N), \quad \text{Equation 3.12}$$

where L^i is the approximated error function at iteration i of the LM algorithm and Z^N denotes the training data set.

3.8 Model Validation

The last step in the system identification method is model validation. This stage is critical for ensuring that the obtained model accurately represents the dynamic characteristics of the system. In accordance with earlier literature evaluations, a variety of validation tests were applied to model structure validation. In this study, root locus stability, one-step ahead prediction (OSA), mean square prediction error (MSE), and correlation tests are all taken into account. The following are the model validation principles:

- i. In the time domain, compare the actual output data to the predicted output data.
- ii. In the frequency domain, compare the estimated model frequency response to the actual frequency response.
- iii. Carrying out the correlation test on the input and predicted errors.

3.8.1 One Step Ahead Prediction

In system identification and control, the OSA forecast is a well-known indicator of predictive accuracy. This is stated by:

$$\hat{y}(t) = f \left(\begin{matrix} u(t), u(t-1), \dots, u(t-n_u), \\ y(t), y(t-1), \dots, y(t-n_y) \end{matrix} \right) \quad \text{Equation 3.13}$$

In Equation 3.13, u and y are the inputs and outputs, respectively, while $f(\cdot)$ is a non-linear function. The following equation is the inaccuracy or prediction:

$$e(t) = y(t) - \hat{y}(t) \quad \text{Equation 3.14}$$

where $e(t)$ represents the predicted error at time t , $y(t)$ represents the output system at time t , and $\hat{y}(t)$ represents the output of the one-step ahead prediction model at time t . If the residual value is small, the developed model meets the criteria for validation, which can be determined by calculating the residual between measured and predicted output.

The popular AutoRegressive with eXogenous inputs (ARX) model has a nonlinear variation known as the Nonlinear AutoRegressive with eXogenous inputs (NARX) model. This model is frequently used in linear black-box identification (Jamali et al., 2018). The neural network is used to estimate the nonlinear portion of the ARX structure. The nonlinear component of the ARX arrangement is approximated using a neural network. One way to characterise the ARX model structure is as

$$y(t) = \frac{B(z^{-1})}{A(z^{-1})}u(t) + \frac{\varepsilon(t)}{A(z^{-1})} \quad \text{Equation 3.15}$$

The expressions of $A(z^{-1})$ and $B(z^{-1})$ are;

$$\begin{aligned} A(z^{-1}) &= 1 + a_1 z^{-1} + \dots + a_n z^{-n} \\ B(z^{-1}) &= b_0 + b_1 z^{-1} + \dots + b_n z^{-(n-1)} \end{aligned} \quad \text{Equation 3.16}$$

White noise, $\varepsilon(t) = 0$, n is the model's orders, and $[a_1, \dots, a_n, b_1, \dots, b_n]$ are the model parameters that need to be estimated to determine z^{-1} . The system's output vector is denoted by $y(t)$ and its input vector by $u(t)$.

Equation 3.17 yields the Neural Network Auto Regressive model with eXogenous inputs (NNARX) model structure regression vector. Through the integration of neural networks into the model framework, NNARX overcomes the drawbacks of NARX.

$$\varphi(t) = [y(t-1), \dots, y(t-n_a), u(t-k), \dots, u(t-n_b-n_k+1)]^T. \quad \text{Equation 3.17}$$

where $\varphi(t)$ denotes the regression vector at time step, t . Past values of the system's input and output are used to generate the regression vector. Equation 3.18 gives the OSA forecast of the NNARX model:

$$\hat{y}\left(\frac{t}{\theta}\right) = y\left(\frac{t}{\theta}\right) = g(\varphi(t), \theta), \quad \text{Equation 3.18}$$

where g is the function that the neural network approach has achieved.

3.8.2 Mean Squared Error

In model validations, MSE is most frequently utilised. The MSE is different between the system's actual outputs $y(n)$ and the anticipated output $\hat{y}(n)$, which was acquired from the system's input and the optimised parameters, and it can be represented as:

$$\text{MSE} = \frac{1}{N} \sum_t^N (y(n) - \hat{y}(n))^2 \quad \text{Equation 3.19}$$

where $y(n)$ is the system's actual output, $\hat{y}(n)$ is the anticipated output resulting from the system's input and the optimised parameters as defined in Equation 3.19, and N is the number of input or output samples. If just the MSE value is low, the model will satisfy the requirements.

3.8.3 Correlation Test

The following list represents the five correlation functions that are suitable for the MLP model (Billings & Zhu, 1994) :

$$\varphi_{\varepsilon\varepsilon}(\tau) = E[\varepsilon(t-\tau)\varepsilon(t)] = \delta(\tau), \quad \text{Equation 3.20}$$

$$\varphi_{u\varepsilon}(\tau) = E[u(t-\tau)\varepsilon(t)] = 0, \forall \tau, \quad \text{Equation 3.21}$$

$$\varphi_{\varepsilon^2\varepsilon}(\tau) = E[u^2(t-\tau) - \bar{u}^2(t)\varepsilon(t)] = 0, \forall \tau, \quad \text{Equation 3.22}$$

$$\varphi_{\varepsilon^2 \varepsilon^2}(\tau) = E[u^2(t - \tau) - \bar{u}^2(t) \varepsilon^2(t)] = 0, \forall \tau, \quad \text{Equation 3.23}$$

$$\varphi_{\varepsilon(\varepsilon u)}(\tau) = E[\varepsilon(t) \varepsilon(t - 1 - \tau) u(t - 1 - \tau)] = 0, \tau \geq 0, \quad \text{Equation 3.24}$$

Equation 3.20 calculates the autocorrelation of the error signal at different lags (τ). Equation 3.21 calculates the cross-correlation between the input signal and the error signal at different lags. Equation 3.22 calculates the cross-correlation between the squared error signal and the error signal at different lags. Equation 3.23 calculates the cross-correlation between the squared error signal and the squared error signal at different lags. Equation 3.24 calculates the cross-correlation between the product of the error and the previous error with the input signal at different lags.

$\varepsilon u(t) = \varepsilon(t + 1)u(t + 1)$, $\delta(\tau)$ is an impulse function, while $\varphi_{u\varepsilon}(\tau)$ is the cross-correlation function between $u(t)$ and $\varepsilon(t)$. All five requirements must be satisfied because the NARX structure, a nonlinear system, is the foundation of the MLP model. There are only three requirements to be satisfied in another PSO model that uses a linear system.

3.9 Design Controller

The controller can be designed based on the obtained modelling. In this research, a PID controller is selected to be implemented in the system. The block diagram of the overall closed-loop system is shown in Figure 3.23. For the hub angle motion, $\theta_{di}(t)$ and $\theta_i(t)$ stand for the reference hub angle and the actual hub angle of the system, respectively.

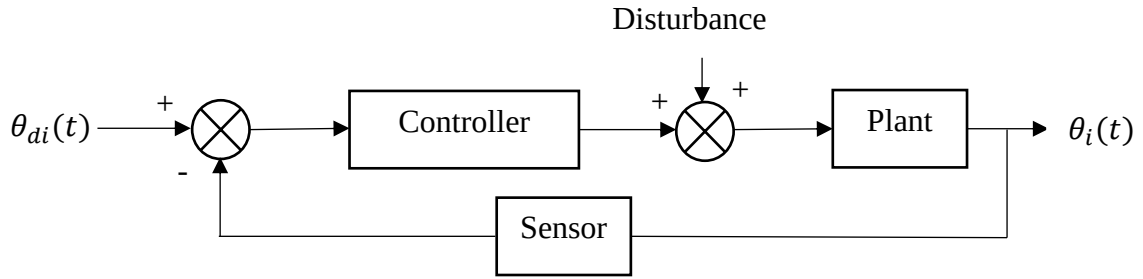


Figure 3.23: Block diagram of the overall closed-loop system

3.9.1 Conventional PID Controller (Ziegler-Nichols Rules)

Among the standard approaches, Ziegler-Nichols was implemented in the system as a comparison to the proposed methods. It is a heuristic PID tuning rule that aims to provide good starting values for the three PID gain parameters.

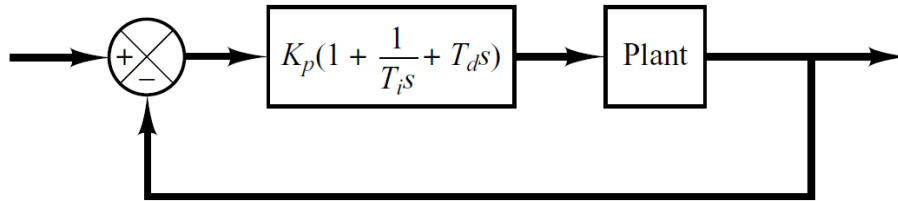


Figure 3.24: PID control of a plant (Ogata, 2010)

A plant's PID control is shown in Figure 3.24. Controller tuning refers to the process of setting controller parameters to match stated performance standards. Ziegler and Nichols proposed rules for tuning PID controllers (that is, setting values K_p , T_i , and T_d based on experimental step responses or the value of K_p that results in marginal stability when only proportional control action is utilised).

Ziegler and Nichols developed two approaches for adjusting the parameters of PID controllers depending on system response behaviour (Ogata, 2010). Ziegler-Nichols gives a starting point for further tuning, but there is no guarantee that the completed system will operate optimally. The first method is used when the system's response to a unit-step is S-

shaped as shown in Figure 3.25. This shows that the plant does not entail pure integration and that the system response is not dominated by a pair of complex-conjugate poles.

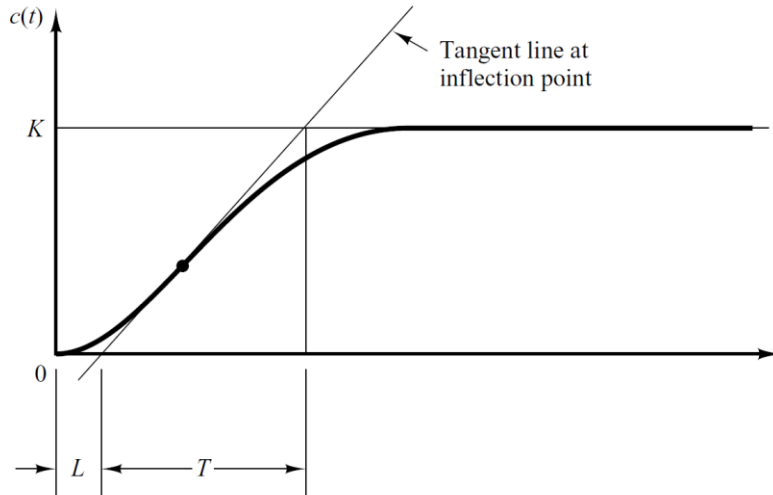


Figure 3.25: S-shaped response curve (Ogata, 2010)

L is the point at which the tangent line intersects the time axis. The time at which the tangent line crosses the steady-state value is given by $L + T$. The controller gains for Method 1 are shown in Table 3.6.

Table 3.6: Method 1 PID-ZN parameters

Controller	K_p	K_i	K_d
P	T/L	0	0
PI	$0.9 T/L$	$0.27 T/L^2$	0
PID	$0.2 T/L$	$0.6 T/L^2$	$0.6 T$

3.9.2 Intelligent Controller - PID-PSO Controller

The utilization of PID-PSO is a technique that employs the PSO algorithm to optimize the parameters of a PID controller. As part of this technique, the performance of

the PID controller is assessed against a desired set point through an objective function or fitness equation. One can define the fitness equation or objective function for the PID PSO controller as either the integral of the squared error (ISE) or the integral of the absolute error (IAE) between the controller's output and the set point throughout a designated period of time. Equation 3.25 shows the performance function by ITAE (Integral of Time Multiplied by Absolute Error) (Gaing, 2004):

$$W(K) = (1 - e^{-\beta})(M_p + e_{ss}) + e^{-\beta}(t_s - t_r) \quad \text{Equation 3.25}$$

$W(K)$ is performance function, β is weighing factor, K is $[K_p, K_i, K_d]$, M_p is maximum overshoot, e_{ss} is steady state error, t_s is settling time and t_r is rise time. The evaluation function, f is defined as follows:

$$f = \frac{1}{W(K)} \quad \text{Equation 3.26}$$

The PSO algorithm relies on the fitness equation to assess the effectiveness of varying sets of PID controller parameters like K_p , K_i and K_d . By searching the parameter space, the PSO algorithm can determine the most suitable combination of these factors that minimizes the objective function or fitness equation. The MATLAB/Simulink closed-loop PID controller uses K_P , K_I , and K_D as the appropriate controller parameters. Every sample's error is calculated, and the MSE is evaluated as the algorithm's fitness score. The target is to reduce the fitness value and optimise the PID controller's parameters.

Then, PSO proceeds with a set of random particles or solutions and then iteratively seeks for the best solution by updating generations. Each particle is updated throughout iteration by alternating between two "best" values. The first is named *pbest*, and it is the best

solution it has thus far. The particle swarm optimizer keeps track of another "best" value and compares it to *pbest*. This best value is known as *gbest* and is a global best. Equation 3.27 and Equation 3.28 shows the velocity and positions updated by the particle.

$$V_{id}^{k+1} = W \times V_{id}^k + C_1 \times R_1 \times (Y_{id}^k - X_{id}^k) + C_2 \times R_2 \times (Y_{id}^k - X_{id}^k) \quad \text{Equation 3.27}$$

$$X_{id}^{k+1} = Y_{id}^k - X_{id}^k \quad \text{Equation 3.28}$$

where V denotes particle velocity, X denotes particle position, W denotes inertia weight, R₁, R₂ denotes a random number, and C₁, and C₂ denote learning factors. The particle number *i* is represented by the *i*-th particle in the d-dimensional search space, and *k* is the number of iterations. Both are chosen based on heuristics.

Figure 3.26 depicts the flowchart of the PID-PSO algorithm. The number of particles and the number of iterations were discovered to have a substantial impact on optimisation performance.

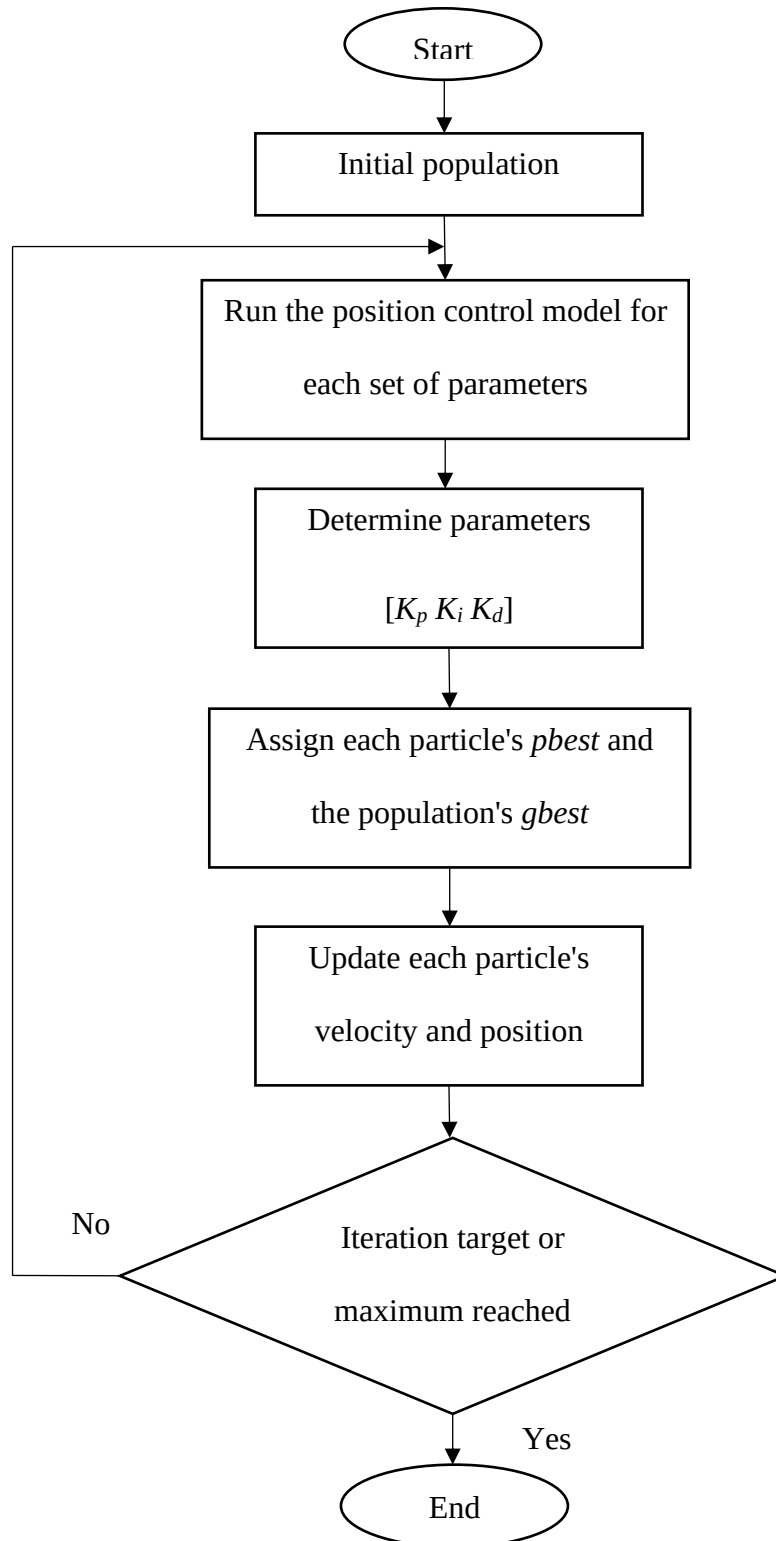


Figure 3.26: The flowchart of the PID-PSO algorithm (Sultan, 2016)

3.10 Summary

In summary, this chapter details the methodology of the research. The setup for this AFO must be done carefully as this will affect the modelling and control of this research. The rig must be able to mimic the movement of children that will use future AFO for rehabilitation in safety usage. Although many other sensor readings were collected, the most consistent and important readings will be chosen for the next stage. There was a brief description of the parametric (PSO) and non-parametric (MLP NN) modelling. The two controllers that will be used to verify ankle foot orthosis's model were lastly presented.

CHAPTER 4

RESULTS AND DISCUSSION

4.1 Introduction

This chapter presents and discusses the final modelling and control of AFO to satisfy the research objectives. Comparison of the parametric and non-parametric dynamic modelling via system identification by using PSO and ANN as model estimation are discussed here. The best modelling result will be controlled by a PID-based controller tuned by PSO for position tracking of AFO. The final developed controller was verified through performance analysis and the results were compared with the conventional controller.

4.2 System Modelling Results

Data from an IMU angle and a bang-bang signal from the linear actuator were gathered in order to perform an experimental test. The results of the tests showed that the information obtained from the experimental setup was suitable for system identification. Using the AFO test rig, data was collected while experimentally acquiring the input-output data required for the modelling process. The Simulink program was used to create the data collection tool. A bang-bang signal with an amplitude of 1 V was used to deliver the required torque to simultaneously activate the actuator.

One output was gathered from the IMU angle, which represents the system's longitudinal axis. For modelling IMU sensor alignment axes, the longitudinal axis (roll axis), which is the IMU's X-axis, was used. Data from the accelerometer and gyroscope are combined to determine the roll angle. The range of the roll is 90 degrees. In relation to the roll axis, the roll angle is positive when rotating anticlockwise and negative when rotating

clockwise. The sensor is repeatedly read and offset values for pitch and roll angle are computed to remove noise at the beginning of the program. These values are subtracted from each step's calculation of the pitch and roll angles. This reduces some of the steady noise in sensor readings.

For system modelling, the 25 seconds long from the 6 seconds data set with a sampling rate of 0.03 seconds was used. In each cycle, there are six seconds of dorsiflexion and six seconds of plantarflexion. Figure 4.1 and Figure 4.2 show the experimental IMU angle (roll) and bang-bang signal responses that were recorded. The gait movement started at dorsiflexion to plantarflexion as shown in Figure 3.2 and Figure 3.3, respectively, which explains why the movement begins from negative values moving up to positive values. The system modelling was conducted by using parametric and non-parametric modelling which will then be compared.

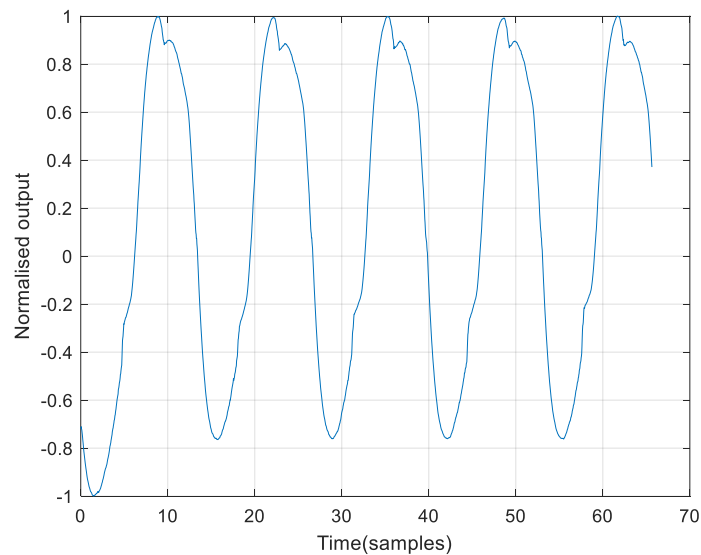


Figure 4.1: The experimental IMU angle (roll)

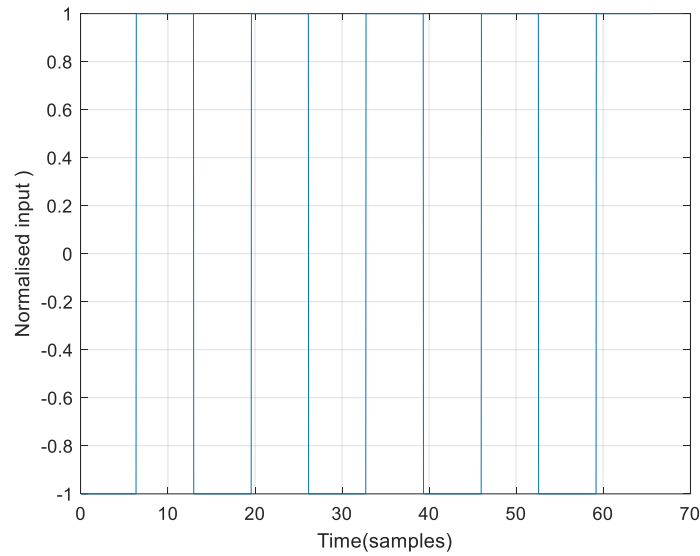


Figure 4.2: Bang-bang signal responses

4.2.1 Particle Swarm Optimization

The 1599-data-point data set for parametric modelling with PSO was divided into two sets of 799.5 data points each. The 1599-data-point data set was chosen as this produce a better result than the overall data which is 2189-data-point. The 0.5 data points were extracted by splitting a data set into two halves, retrieving specific data points including those with indices containing a decimal part, using indexing with non-integer values, and interpolating the data. The first set was used for modelling, while the second (test set) was used for validation. The order of four in the model produced the best results. OSA, MSE, correlation testing, and pole-zero diagram stability were used to validate the created systems by PSO. Table 4.1 shows the optimal result parameters with specifications in PSO modelling optimization.

Table 4.1: Optimal result parameters in PSO modelling optimization

Parameter	Specification
Number of swarms	400
Maximum number of iterations	1000
Maximum generation	50
Inertia weight, w	0.5
Acceleration coefficient, C_1 , C_2	2
Model order	4

The maximum generation, which places a limit on how many times the algorithm can run, is one of the PSO stopping criteria. The PSO algorithm's maximum generation sets a restriction on how often the particle locations and velocities can be changed. PSO employs a number of swarms to strike a balance between exploitation (improving upon already discovered, promising solutions) and exploration (discovering new, prospective solutions). In general, fewer swarms encourage exploitation, whereas more encourage exploration. 400 can be a compromise between these two factors for the particular issue at hand. To properly traverse the search space and prevent becoming stuck, multiple swarms may be required for difficult problems with many dimensions or local optima. The empirically found value that is most effective for this specific issue is 400.

Figure 4.3 depicts the PSO predictions for the roll axis angle. The graph demonstrates that the PSO could track the actual data closely. The actual and predicted PSO outputs differ slightly.

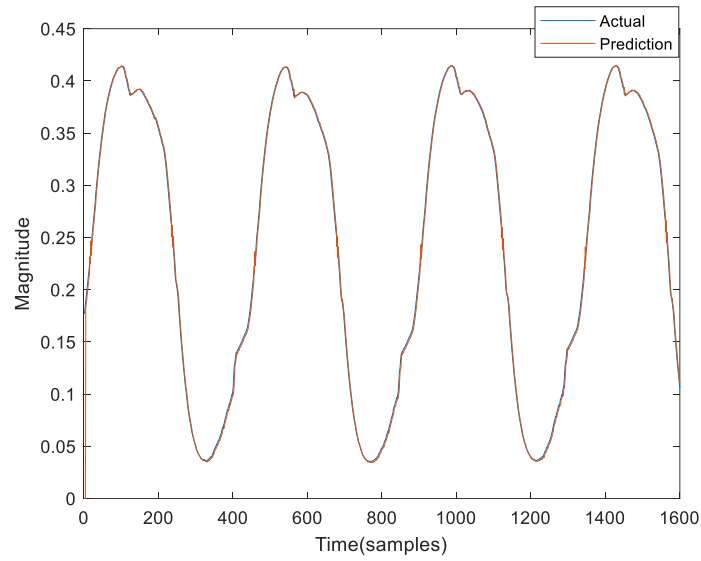


Figure 4.3: The output and estimated output of roll axis angle (PSO)

Figure 4.4 depicts the correlation test for each roll axis angle. The PSO results do not support the model's accuracy and fall outside of its 95% confidence interval which is supported by the result of Equation 3.20 and Equation 3.21. The model's predictions are represented by two horizontal lines, usually dashed, that show the upper and lower bounds of the 95% confidence interval. Considered statistically consistent with the model are data points that lie inside these lines. Points that fall outside of the range shows that PSO results do not support the model's accuracy such as in Figure 4.4.

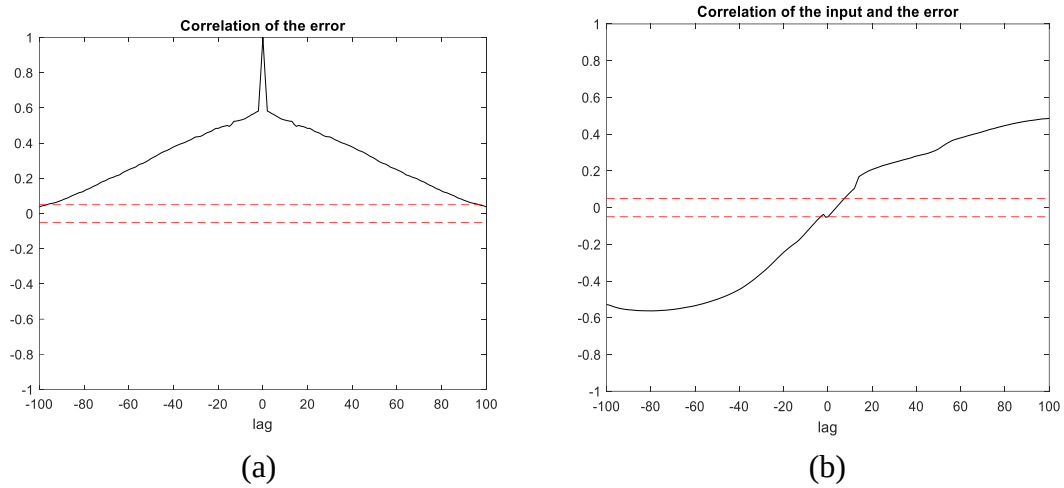


Figure 4.4: (a) Correlation of error (PSO) (b) Correlation of the input and the error (PSO)

The graph shows that the PSO could keep a close eye on the real data in Figure 4.3. Nevertheless, PSO results in Figure 4.4 demonstrate that they do not support the accuracy of the model and are outside of the correlation test's 95% confidence interval. If the underlying link is not straight but instead curved or more complex, correlation will not capture it, even though the data points visually appear to follow a similar trend. The correlation coefficient can be manipulated by a small number of outliers or scale inaccuracies. Although correlation assesses only the linear trend, visual alignment indicates that the model represents the general trend of the data. The greater range of the confidence interval which excludes the claimed accuracy could indicate model performance uncertainty even in cases where the data points appear to be perfectly aligned.

Figure 4.5 depicts the roll axis angle stability test. The PSO demonstrates that the model is controllable. The poles are visible inside the unity circle, and the zeros are at the origin. The zero represents the symbol 'o', while the poles represent the symbol 'x'. The fact that the poles are within the unit circle indicates that the system is stable.

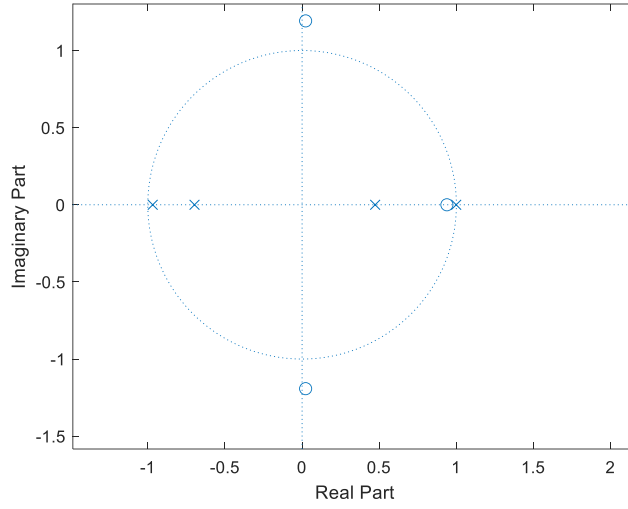


Figure 4.5: The stability test for roll axis angle (PSO)

Equation 4.1 expresses the discrete transfer function of the AFO that was produced by PSO modelling.

$$H(z)_{PSO} = \frac{0.006214z^{-1} - 0.006117z^{-2} + 0.009101z^{-3} - 0.008291z^{-4}}{1 + 0.1928z^{-1} - 1.304z^{-2} - 0.2061z^{-3} + 0.3193z^{-4}} \quad \text{Equation 4.1}$$

Table 4.2 shows the PSO results of the best model order. The primary requirements to consider when selecting the best model are the lowest MSE in testing data, a correlation test with a 95% confidence level, and good stability. Thus, model order 4 was found to be the best model order, with MSE for the training and testing data sets of 1.8075×10^{-4} and 1.7743×10^{-5} , respectively. Model order 2, 3, 7, 8, 9 and 10 are unstable. Between model order 1, 4, 5 and 6, model order 4 has the lowest MSE value thus this shows the best model order for PSO modelling.

Table 4.2: Comparison of PSO optimization performance in different numbers of model order

Model Order	MSE in training data	MSE in testing data	Stability	Correlation Test
1	4.4205×10^{-5}	5.6637×10^{-6}	Stable	Biased
2	1.3612×10^{-4}	6.1793×10^{-5}	Unstable	Biased
3	1.7066×10^{-4}	5.6001×10^{-5}	Unstable	Biased
4	1.8075×10^{-4}	1.4752×10^{-4}	Stable	Biased
5	3.4570×10^{-4}	1.7743×10^{-5}	Stable	Biased
6	2.6758×10^{-4}	1.2621×10^{-5}	Stable	Biased
7	5.5472×10^{-4}	2.6591×10^{-4}	Unstable	Biased
8	4.4309×10^{-4}	8.9236×10^{-5}	Unstable	Biased
9	5.2448×10^{-4}	1.2117×10^{-4}	Unstable	Biased
10	9.7826×10^{-4}	5.2718×10^{-4}	Unstable	Biased

4.2.2 Artificial Neural Network

The 2189-data-point data set was divided into two sets of 1532 and 657 data points for non-parametric modelling with MLP NN. 2189 was the overall data taken from the gait movement and can get a better result when using all the data for the modelling stage using MLP NN. The first set was used for modelling, while the second set (test set) was used for validation. To validate the results, the MLP NN modelling was compared using MSE and 5 correlation tests. Given the lack of prior knowledge about the appropriate delay numbers and the model structure for MLP NN, the structure was realised using a heuristic method.

Three major factors had to be taken into account during the process: the number of delay signals, the size of the NN structure or the number of neurons, and the error. The last factor was considered as part of the process of determining the optimal number of delay signals and structure for each model as the OSA. This was because the procedure for finding

the best model was stochastic. Based on validation MSE, modelling MSE, and correlation tests, the criterion was used to choose the best model.

The number of neurons in this research begins with two neurons in the first hidden layer, two neurons in the second hidden layer, and one neuron in the output layer ([2 2 1] model structure). The delay number denotes the input layer. The model's performance OSA was improved by using eight delay signals, eight neurons in each of the first and second hidden layers, and one neuron in the output layer ([8 8 1] model structure).

Figure 4.6 shows the MLP predictions for the roll axis angle. A yellow vertical line represents the validated data at point 1532. The graph demonstrates that the MLP can closely track the actual data. The difference between actual and predicted MLP output is close to zero.

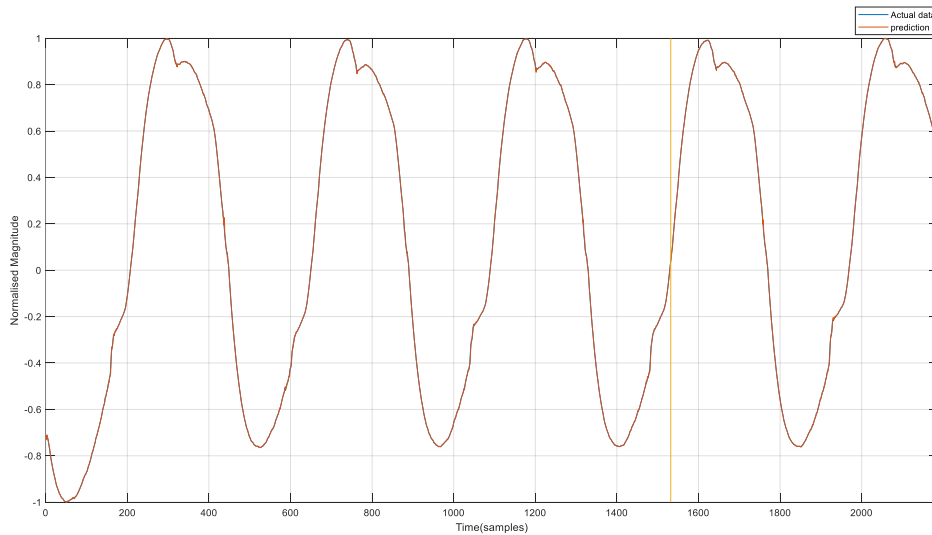


Figure 4.6: The output and estimated output of roll axis angle (MLP)

The correlation test for each roll axis angle is shown in Figure 4.7 which is supported by the result of Equation 3.20, Equation 3.21, Equation 3.22, Equation 3.23 and Equation 3.24. The MLP results are accurate to within 95% of the confidence level.

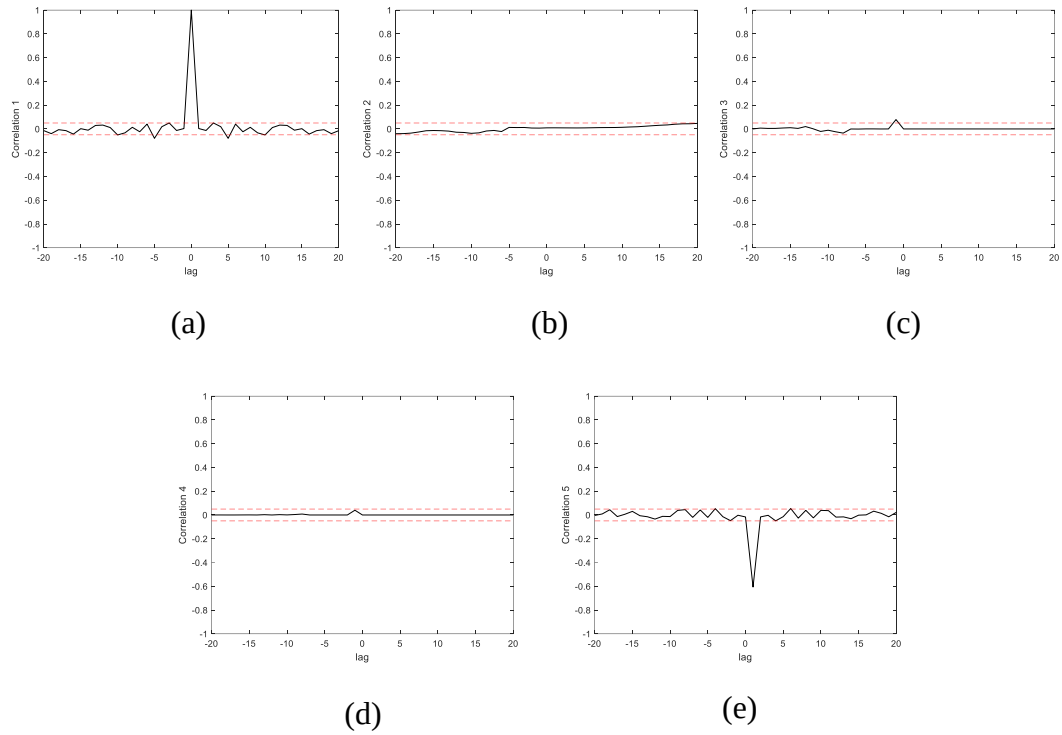


Figure 4.7: (a) Correlation of error (MLP NN) (b) Correlation of the input and the error (MLP NN) (c) Correlation of the square input and the error (MLP NN) (d) Correlation of square of the input square of the error (MLP NN) (e) Correlation of multiplication of the input by the error and the error (MLP NN)

Table 4.3 shows the numerical results of the best model structure and MLP NN delay. Model structure [8 8 1] with 8 delays was discovered to be the best model structure, with the lowest mean squared error of 1.1034×10^{-5} .

Table 4.3: Comparison of MLP NN performance

Model Structure	Delay	MSE	Correlation Test
[2 2 1]	2	2.3829×10^{-4}	Unbiased
[4 4 1]	2	3.5671×10^{-4}	Unbiased
[6 6 1]	6	2.7149×10^{-4}	Unbiased
[8 8 1]	7	1.8625×10^{-4}	Unbiased
[8 8 1]	8	1.1034×10^{-5}	Unbiased
[6 6 1]	9	1.0855×10^{-5}	Biased
[8 8 1]	9	1.7833×10^{-4}	Biased

4.2.3 Comparison and Analysis

Training and testing procedures were used to validate the PSO and MLP NN-based models. Each correlation test set has also been run. All of these tests show that the various modelling techniques investigated in this study performed adequately well. All of these tests show that the various modelling techniques used in this study worked well. Table 4.4 compares the mean-squared error and correlation test performance of parametric and non-parametric modelling approaches.

Table 4.4: Performance of parametric and non-parametric modelling approaches

Algorithm	MSE	Correlation test
MLP NN	0.000011034	Unbiased
PSO	0.00018075	Biased

Table 4.4 compares the performance of MLP NN and PSO modelling, revealing that the MLP NN-based non-parametric modelling technique provides a better approximation to system response than the PSO technique. According to the correlation test results, the MLP NN outperformed the PSO. PSO has a higher mean-squared error than MLP NN.

However, one significant advantage of the PSO is that there are fewer parameters to tune. PSO obtains the best solution through particle interaction, but due to the high-dimensional search space, it advances at a very slow rate toward the global optimum (Gad, 2022). Additionally, when dealing with complicated and huge datasets, it provides poor-quality results.

This research demonstrates that the MLP NN can be used to address difficult nonlinear problems. It manages vast amounts of input data efficiently. It makes prompt forecasts after training. The same accuracy ratio is possible even with smaller samples.

4.3 Control Scheme

To validate the created model, it was evaluated with PID and PID-PSO based controllers. The control system in MATLAB co-simulation is provided to control the actuator with ankle joint movement for tracking the trajectory gait cycle. The time taken used for the controller was 6 seconds as 1 movement either dorsiflexion or plantarflexion in this research took 6 seconds long. Finally, the results of the PID and PID-PSO controllers were compared. Figure 4.8 shows what the actual signal looks like if there is no controller used. The negative voltage occurs in the graph due to the position of the IMU sensor placed on the AFO rig whereby the data is collected from negative to positive angle.

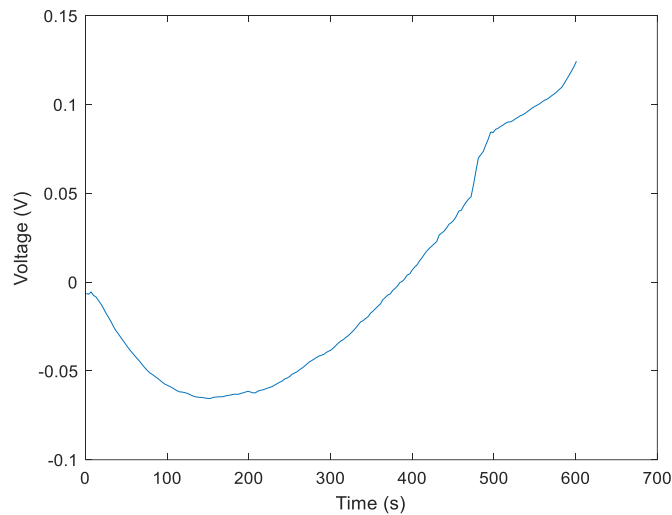


Figure 4.8: Actual signal (without controller)

4.3.1 Proportional Integral Derivative

Figure 4.9 shows the Simulink setup of the P, PI, and PID controller. The P, PI, and PID were adjusted by using Ziegler–Nichols Rules.

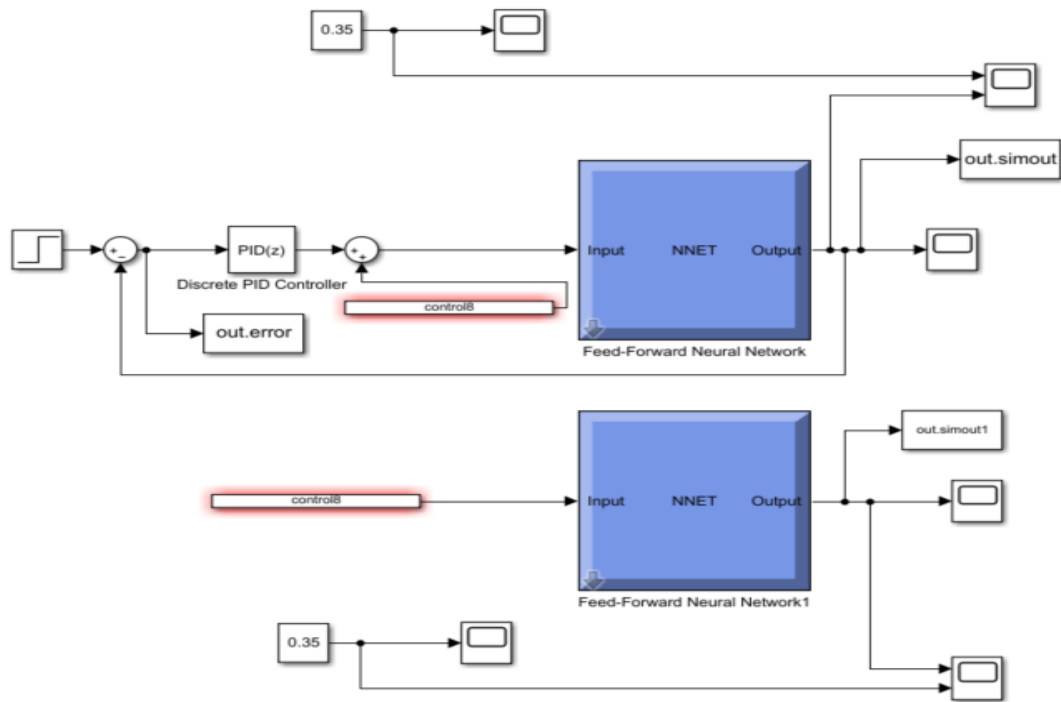


Figure 4.9: Simulink setup of the P, PI, PID controller

Figure 4.10 shows the comparison between the P, PI, and PID controllers and the one without a controller. It can be observed that the PID controller shows the best controller between all 3 controllers. From the graph, it proves that the P controller was not able to control the system as the output is too far from the desired output. This is supported by the evidence of the high steady-state error of the output response by the P controller which is about 0.1716 rad or 9.8320°. Meanwhile, the output response of the PI controller shows obvious oscillation which will cause unstable and jerking in the system. This is supported by the evidence of a high overshoot value which is 28.0946 %. Therefore, the PID result will be compared with the proposed controller which is PID-PSO.

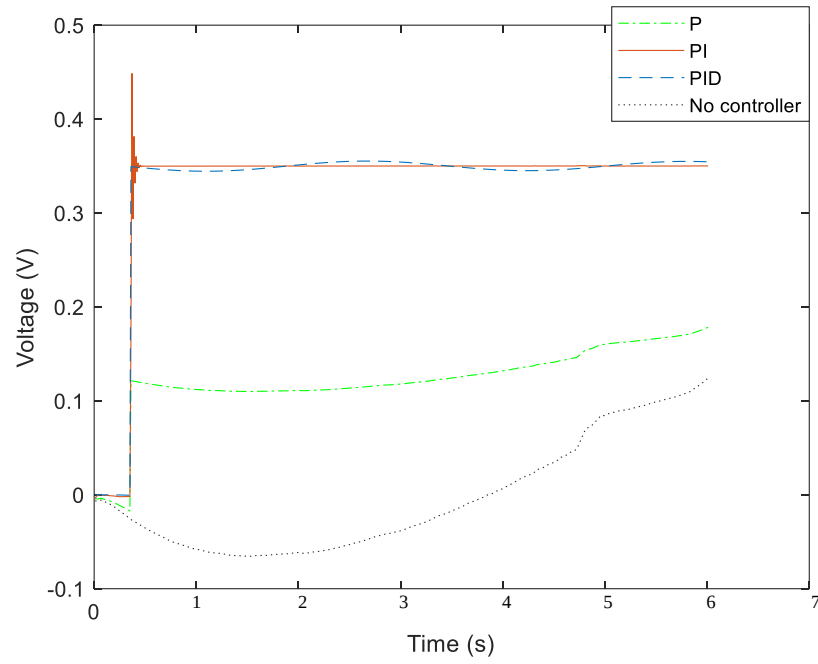


Figure 4.10: The comparison between the P, PI, and PID controller and the one without a controller

The results of the P, PI, and PID controller are tabulated in Table 4.5.

Table 4.5: The results of the P, PI, and PID controller

Results	P	PI	PID
Rise Time (s)	4.6594	0.0131	0.0081
Settling Time (s)	5.9203	0.4018	4.7349
Overshoot (%)	0	28.0946	0.2163
Steady State Error	0.1716	1.554×10^{-4}	0.0045
K_p	2.980132	2.682119	3.576159
K_i	0	503.3333	302
K_d	0	0	75.5
N	100	100	100

4.3.2 Particle Swarm Optimization

The traditional tuning method of observation yields the PID parameters. In order to avoid local optima, the output is set as the initial value, providing the searching space. PSO accelerates convergence and increases the likelihood of locating the global optimum (Amiri et al., 2020). Thus, the PID controller was optimised by combining PID and PSO optimization algorithms. Figure 4.11 shows the Simulink setup of the PID-PSO controller.

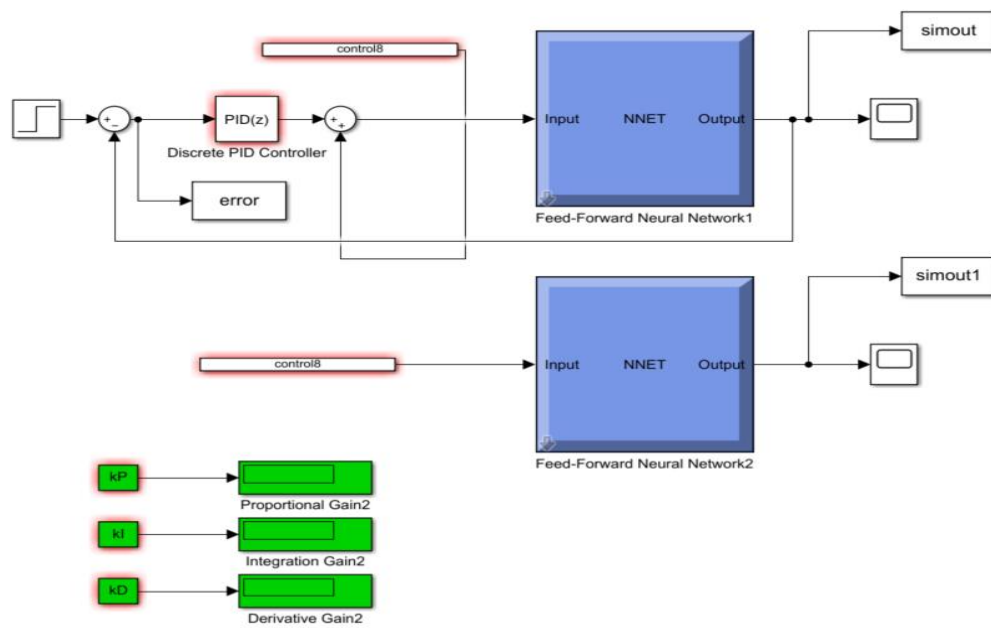


Figure 4.11: Simulink setup of the PID controller

Figure 4.12 shows the comparison between the PID (Ziegler–Nichols) controller, PID-PSO controller, and the one without a controller. Without controller, it can be observed that the trajectory tracking performance is excellent with the acceptable rise time, acceptable settling time, acceptable overshoot value, and a lower steady-state error. Table 4.6 shows the optimal result parameters in PSO controller optimization.

Table 4.6: Optimal result parameters in PSO controller optimization

Parameter	Specification
Number of swarms	20
Number of iterations	20
Maximum inertia weight, w_{max}	0.7
Minimum inertia weight, w_{min}	0.25
Acceleration coefficient, C_1, C_2	2

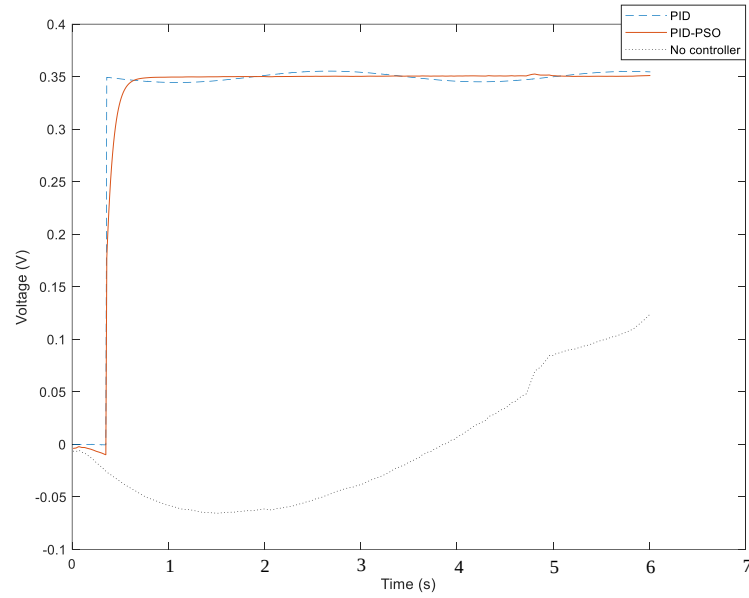


Figure 4.12: The comparison between the PID (Ziegler–Nichols) controller, PID-PSO controller, and the one without a controller

The results of the PID-PSO control are tabulated in Table 4.7.

Table 4.7: The results of the PID-PSO control

Rise Time (s)	0.3665
Settling Time (s)	1.7707
Overshoot (%)	0.4050
Steady State Error	0.0011
K_p	3.6513
K_i	57.8743
K_d	3.4639
MSE	2.0252×10^{-4}

Figure 4.13 depicts the MSE convergence of PSO for angle after 20 iterations. It can be shown that PSO converges quickly and produces a low MSE value for the controller.

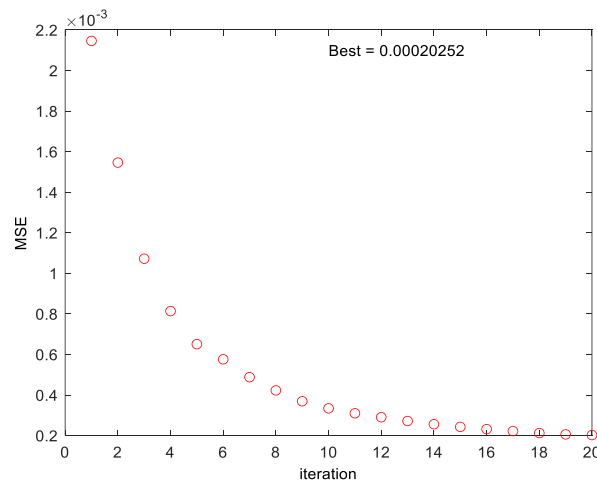


Figure 4.13: MSE convergence of PSO for hub angle

4.3.3 Comparison and Analysis

The model has been successfully tested by using PID (Ziegler–Nichols) and PID-PSO. Table 4.8 shows the comparison performance between both controllers. The rise time for PID (Ziegler–Nichols) controller is lower than the PID-PSO controller makes the system's response faster. PID (Ziegler–Nichols) controller also has a lower overshoot value than the PID-PSO controller which makes the system more stable. Despite these advantages, the PID-PSO controller has a lower settling time than PID (Ziegler–Nichols) which makes the output settles faster to final values. PID-PSO controller also has lower steady-state error than PID (Ziegler–Nichols) which makes the system reach very close to the desired value. Steady-state error and settling time are critical in this application because it involve the rehabilitation of patient users.

Table 4.8: The comparison performance between the PID (Ziegler–Nichols) controller and PID-PSO controller

Results	PID	PID-PSO
Rise Time (s)	0.0081	0.3665
Settling Time (s)	4.7349	1.7707
Overshoot (%)	0.2163	0.4050
Steady State Error	0.0045	0.0011

4.4 Summary

This chapter has presented the modelling and control of ankle foot orthosis. System modelling was identified by using non-parametric and parametric which were MLP NN and PSO. Both algorithms were validated using one step ahead, mean squared error, and correlation test. In this chapter, it can be summarized that MLP NN were able to manage

harder nonlinear problems when compared to the PSO. PSO has difficulty handling complicated and huge datasets. In terms of the controller, PID (ZN) and PID-PSO were used in comparison and analysis. The performance comparison shows that PID-PSO is better as it has lower steady-state error and lower settling time than PID (ZN) which proves that the developed controller is better than the conventional controller.

CHAPTER 5

CONCLUSION AND RECOMMENDATIONS

5.1 Conclusion

In this research, the system identification technique used both parametric and non-parametric modelling of an ankle foot orthosis. The parametric modelling was tested by using Particle Swarm Optimization (PSO) while the non-parametric modelling was tested by using Multilayer Perceptron Neural Network (MLP NN). This supported the first objective of this research. The collection of data inclusive of the input data given to the motor as torque and the output data from the movement of the linear motor captured by the IMU sensor which collects gait movement. The performance result shows that the MLP NN is way much better than PSO in terms of correlation test and mean square error. Although PSO is widely known for solving engineering problems, it does not fulfill the requirement for this research as the results portrayed correlation errors of more than 95%. Thus, non-parametric modelling that is MLP NN is used to solve such complex engineering problems.

Position tracking of the ankle foot orthosis is important after model development. This research compared 2 control systems which are PID (ZN) and PID-PSO supported by the 2 other objectives. The controllers were compared in rise time, settling time, overshoot, and steady-state error. The gait performance results show that there was a huge improvement before using a controller and after using any controller. The conventional controller was tested using P, PI, and PID by Ziegler-Nichols method 1. PID was the best controller among conventional controllers as it has the lowest rise time and lowest overshoot value.

When comparing PID (ZN) and PID-PSO, it can be concluded that PID-PSO is more effective as it has a lower steady state value which makes the system reach very close to the desired value. PID-PSO also has a lower settling time than PID (Ziegler–Nichols) which makes the output settles faster to final values. Due to the fact that this system involves the rehabilitation of patient users, these two aspects are crucial.

5.2 Recommendations

The suggestions listed below can serve as a starting point for extending and broadening this research.

Firstly, using evolutionary algorithms like artificial bee colonies, cuckoo searches, and flower pollination, modelling for AFO can be developed. Additionally, it is advised to modify those methods and utilise them as the learning algorithm in non-parametric models like NN.

Secondly, the application of real-time or adaptive tuning parameters using PSO and ABC algorithms. As long as the plant permits reasonably enough sample times for the algorithm to finish the convergence process, the online tuning parameter can be changed in real-time. If the devices can run the system at a higher capacity, this approach is strongly advised.

Besides that, the environmental effect and disturbance should be considered in future modelling. This ensures that the system able to adapt to various situations and can be adjusted in rehabilitation usage. Additional sensors such as EMG and different axis motor movements will help to monitor the gait cycle closely.

REFERENCES

- Abdelhady, M., Rashvand, A., Moness, M., Richter, H., & Simon, D. (2017). System identification and control optimization of an active prosthetic knee in swing phase. In *Proceedings of the American Control Conference* (pp. 857–862). Institute of Electrical and Electronics Engineers Inc.
- Abe, I., Ishiya, K., Kikuchi, T., Tanida, S., & Yasuda, T. (2017). Preliminary study on ankle-foot orthosis using elastomer- embedded flexible joint. In *2017 39th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)* (pp. 4499–4502). Midview City: Singapore Therapeutic, Assistive & Rehabilitative Technologies (START) Centre.
- Adiputra, D., Nazmi, N., Bahiuddin, I., Ubaidillah, U., Imaduddin, F., Abdul Rahman, M. A., ... Zamzuri, H. (2019). A review on the control of the mechanical properties of ankle foot orthosis for gait assistance. *Actuators*, 8(1), 10.
- Al-Shuka, H. F. N., Rahman, M. H., Leonhardt, S., Ciobanu, I., & Berteau, M. (2019). Biomechanics, actuation, and multi-level control strategies of power-augmentation lower extremity exoskeletons: an overview. *International Journal of Dynamics and Control*, 7(4), 1462–1488.
- Alam, M., Choudhury, I. A., & Mamat, A. Bin. (2014). Mechanism and design analysis of articulated ankle foot orthoses for drop-foot. *The Scientific World Journal*, 2014, 1–14.
- Aliman, N., Ramli, R., & Haris, S. M. (2018). Modeling and co-simulation of actuator control for lower limb exoskeleton. In *2018 3rd International Conference on Control and Robotics Engineering, ICCRE 2018* (pp. 94–98). Institute of Electrical and

Electronics Engineers Inc.

- Amiri, M. S., Ramli, R., Ibrahim, M. F., Wahab, D. A., & Aliman, N. (2020). Adaptive particle swarm optimization of PID gain tuning for lower-limb human exoskeleton in virtual environment. *Mathematics*, 8(11), 1–16.
- Arcolezi, H. H., Nunes, W. R. B. M., de Araujo, R. A., Cerna, S., Sanches, M. A. A., Teixeira, M. C. M., & de Carvalho, A. A. (2021). RISE controller tuning and system identification through machine learning for human lower limb rehabilitation via neuromuscular electrical stimulation. *Engineering Applications of Artificial Intelligence*, 102.
- Arnez-Paniagua, V., Rifaï, H., Amirat, Y., Ghedira, M., Gracies, J. M., & Mohammed, S. (2019). Adaptive control of an actuated ankle foot orthosis for paretic patients. *Control Engineering Practice*, 90, 207–220.
- Billings, S. A., & Zhu, Q. M. (1994). Nonlinear model validation using correlation tests. *International Journal of Control*, 60(6), 1107–1120.
- Bykov, A. V., Burmaka, A. A., Korenevskaya, S. N., & Rodionov, D. S. (2018). An intelligent decision support system for managing patients with lower limb ischemia based on hybrid fuzzy models. *Biomedical Engineering*, 52(4), 271–274.
- Caldas, R., Mundt, M., Potthast, W., Buarque de Lima Neto, F., & Markert, B. (2017). A systematic review of gait analysis methods based on inertial sensors and adaptive algorithms. *Gait and Posture*, 57, 204–210.
- Chamnikar, A. S., Patil, G., Radmanesh, M., & Kumar, M. (2017). Trajectory generation for a lower limb exoskeleton for sit-to-stand transition using a genetic algorithm. In

Proceedings of the ASME 2017 Dynamic Systems and Control Conference (Vol. 1). American Society of Mechanical Engineers.

Colakoglu, N., & Akkaya, B. (2019). Comparison of multi-class classification algorithms on early diagnosis of heart diseases. In *Proceeding Book of the y-BIS Conference*.

Contini, B. G., Bergamini, E., Alvini, M., Di Stanislao, E., Di Rosa, G., Castelli, E., ... Camomilla, V. (2019). A wearable gait analysis protocol to support the choice of the appropriate ankle foot orthosis: a comparative assessment in children with Cerebral Palsy. *Clinical Biomechanics*, 70, 177–185.

Eldirdiry, O., Zaier, R., Al-Yahmedi, A., Bahadur, I., & Alnajjar, F. (2020). Modeling of a biped robot for investigating foot drop using MATLAB/Simulink. *Simulation Modelling Practice and Theory*, 98(10), 1972.

Fatone, S., Johnson, W. B., & Tucker, K. (2014). A three-dimensional model to assess the effect of ankle joint axis misalignments in ankle-foot orthoses. *Prosthetics and Orthotics International*, 40(2), 240–246.

Gad, A. G. (2022). Particle swarm optimization algorithm and its applications: a systematic review. *Archives of Computational Methods in Engineering* 2022 29:5, 29(5), 2531–2561.

Gaing, Z. L. (2004). A particle swarm optimization approach for optimum design of PID controller in AVR system. *IEEE Transactions on Energy Conversion*, 19(2), 384–391.
Retrieved from
https://www.researchgate.net/publication/3270304_A_Particle_Swarm_Optimization_Approach_for_Optimum_Design_of_PID_Controller_in_AVR_System

- Geng, T., Jia, X., & Guo, Y. (2021). Lower limb joint nursing and rehabilitation system based on intelligent medical treatment. *Journal of Healthcare Engineering*, 2021.
- Geyer, H., & Herr, H. (2010). A muscle-reflex model that encodes principles of legged mechanics produces human walking dynamics and muscle activities. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 18(3), 263–273.
- Gilbert, M., Zhang, X., & Yin, G. (2016). Modeling and design on control system of lower limb rehabilitation exoskeleton robot. In *2016 13th International Conference on Ubiquitous Robots and Ambient Intelligence (URAI)* (pp. 348–352). Institute of Electrical and Electronics Engineers Inc.
- Habeeb Mohamed, A. H., & Zakaria, M. A. (2019). Torque analysis of a platform based end effector ankle-foot orthosis for rehabilitation. *Journal of Intelligent Manufacturing & Mechatronics*, 01(01), 35–38.
- Hegarty, A. K., Petrella, A. J., Kurz, M. J., & Silverman, A. K. (2017). Evaluating the effects of ankle-foot orthosis mechanical property assumptions on gait simulation muscle force results. *Journal of Biomedical Engineering*, 139, 1–8.
- Hsu, C. C., Huang, Y. K., Kang, J. H., Ko, Y. F., Liu, C. W., Jaw, F. S., & Chen, S. C. (2020). Novel design for a dynamic ankle foot orthosis with motion feedback used for training in patients with hemiplegic gait: a pilot study. *Journal of NeuroEngineering and Rehabilitation*, 17(1), 1–9.
- Jabatan Kebajikan Masyarakat. (2023). *Statistik Pendaftaran OKU Mengikut Negeri dan Kategori Sehingga 31 Januari 2023*. Available at: <https://www.jkm.gov.my/jkm/index.php?r=portal/full&id=ZUFHVTB1NnJWM0EreGtwNC9Vb1hvdz09> [Assessed on 2 February 2024].

- Jamali, A. (2017). *Modeling and intelligent control of double-link flexible robotic manipulator*. [Doctoral thesis, Universiti Teknologi Malaysia].
- Jamali, A., Mat Darus, I. Z. M., Samin, P. M., & Tokhi, M. O. (2018). Intelligent modeling of double link flexible robotic manipulator using artificial neural network. *Journal of Vibroengineering*, 20(2), 1021–1034.
- Jin, D., Liu, Y., Ma, X., & Song, Q. (2021). Long time prediction of human lower limb movement based on IPSO-BPNN. In *Journal of Physics: Conference Series* (Vol. 1865). IOP Publishing Ltd.
- Kasebzadeh, P. (2019). *Learning human gait*. [Doctoral thesis, Linköping University]. Electronic Press, Linköping.
- Kobayashi, T., Gao, F., LeCursi, N., Foreman, K. B., & Orendurff, M. S. (2017). Effect of shoes on stiffness and energy efficiency of ankle-foot orthosis: bench testing analysis. *Journal of Applied Biomechanics*, 33(6), 460–463.
- Kribus-Shmiel, L., Zeilig, G., Sokolovski, B., & Plotnik, M. (2018). How many strides are required for a reliable estimation of temporal gait parameters? Implementation of a new algorithm on the phase coordination index. *PLoS ONE*, 13(2).
- Manalo, K. D., Linsangan, N. B., & Torres, J. L. (2016). Classification of myoelectric signals using multilayer perceptron neural network with back propagation algorithm in a wireless surface myoelectric prosthesis. *International Journal of Information and Education Technology*, 6(9), 686–690.
- Manchola, M. D. S., Bernal, M. J. P., Munera, M., & Cifuentes, C. A. (2019). Gait phase detection for lower-limb exoskeletons using foot motion data from a single inertial

- measurement unit in hemiparetic individuals. *Sensors (Switzerland)*, 19(13), 2988.
- Mat Darus, I. Z., & Tokhi, M. O. (2006). Parametric and non-parametric identification of a two dimensional flexible structure. *Journal of Low Frequency Noise Vibration and Active Control*, 25(2), 119–143.
- Moltedo, M., Bacek, T., Verstraten, T., Rodriguez-Guerro, Vanderborght, B., & Lefeber, D. (2018). Powered ankle-foot orthoses: the effects of the assistance on healthy and impaired users while walking. *Journal of NeuroEngineering and Rehabilitation*, 15(86).
- Narkhede, P., Poddar, S., Walambe, R., Ghinea, G., & Kotecha, K. (2021). Cascaded complementary filter architecture for sensor fusion in attitude estimation. *Sensors (Basel, Switzerland)*, 21(6), 1–18.
- Ogata, K. (2010). *Modern control engineering* (5th ed.), pp 567-571. New Jersey: Prentice-Hall.
- Owen, E. (2014). From stable standing to rock and roll walking (part 2) designing , aligning and tuning orthoses for standing stepping and gait. *APCP Journal*, 5(2), 4–16.
- Raosoft, Inc. (2004). *Sample size calculator*. Available at: <http://www.raosoft.com/samplesize.html> [Assessed on 2 February 2024].
- Reggeti, J. C. A. (2017). *Parameter identification and modeling of contact properties for robotic applications*. [Doctoral thesis, Universidad Politécnica de Madrid].
- Rosenberg, M., & Steele, K. M. (2017). Simulated impacts of ankle foot orthoses on muscle demand and recruitment in typically developing children and children with cerebral palsy and crouch gait. *PLoS ONE*, 12(7).

- Saeed, M. T., Gul, J. Z., Kausar, Z., Mughal, A. M., Ud Din, Z. M., & Qin, S. (2022). Design of model-based and model-free robust control strategies for lower limb rehabilitation exoskeletons. *Applied Sciences (Switzerland)*, 12(8).
- Shi, Y., & Eberhart, R. (1998). A modified particle swarm optimizer. In *Proceedings of the IEEE Conference on Evolutionary Computation, ICEC* (pp. 69–73). IEEE.
- Sivakumar, S., Gopalai, A. A., Lim, K. H., & Gouwanda, D. (2019). Artificial neural network based ankle joint angle estimation using instrumented foot insoles. *Biomedical Signal Processing and Control*, 54, 101614.
- Skaaret, I., Steen, H., Huse, A. B., & Holm, I. (2019). Comparison of gait with and without ankle-foot orthoses after lower limb surgery in children with unilateral cerebral palsy. *Journal of Children's Orthopaedics*, 13(2), 180–189.
- Sreenivasa, M., Millard, M., Felis, M., Mombaur, K., & Wolf, S. I. (2017). Optimal control based stiffness identification of an ankle-foot orthosis using a predictive walking model. *Frontiers in Computational Neuroscience*, 11, 23.
- Sultan, A. J. (2016). Optimal PID Controller of Position Tracking DC Servomotor Control Based Particle Swarm Optimization. *Journal For Engineering Sciences*, 9(4).
- Tan, F. Y. (2016). *Development of ankle-foot orthosis for hemiplegic children*. Universiti Malaysia Sarawak.
- Thalman, C. M., Hsu, J., Snyder, L., & Polygerinos, P. (2019). Design of a soft ankle-foot orthosis exosuit for foot drop assistance. In *International Conference on Robotics and Automation (ICRA)*.
- The MathWorks. (2019). *Estimate orientation using complementary filter*. Available at:

<https://www.mathworks.com/help/fusion/ref/complementaryfilter-system-object.html>
[Assessed on 3 April 2023].

The MathWorks. (2020). *Pitch and Roll Calculation on Arduino Using IMU Sensor*. Available at: <https://www.mathworks.com/help/supportpkg/arduino/ref/pitch-roll-calculation-arduino-imu-sensor.html> [Assessed on 3 April 2023].

World Health Organization. (2018). *Disability*. Available at: <https://www.who.int/news-room/fact-sheets/detail/disability-and-health> [Assessed on 3 December 2019].

Y.Hassan, M., & S. Ghintab, S. (2016). Ant colony optimization based force-position control for human lower limb rehabilitation robot. *Al-Khwarizmi Engineering Journal*, 12(1), 61–72.

Yagi, K., Suzuki, K., & Mochiyama, H. (2018). Human joint impedance estimation with a new wearable device utilizing snap-through buckling of closed-elastica. *IEEE Robotics and Automation Letters*, 3(3), 1506–1513.

Yamamoto, M., Shimatani, K., Hasegawa, M., & Kurita, Y. (2019). Effect of an ankle–foot orthosis on gait kinematics and kinetics: case study of post-stroke gait using a musculoskeletal model and an orthosis model. *ROBOMECH Journal*, 6(1), 9.

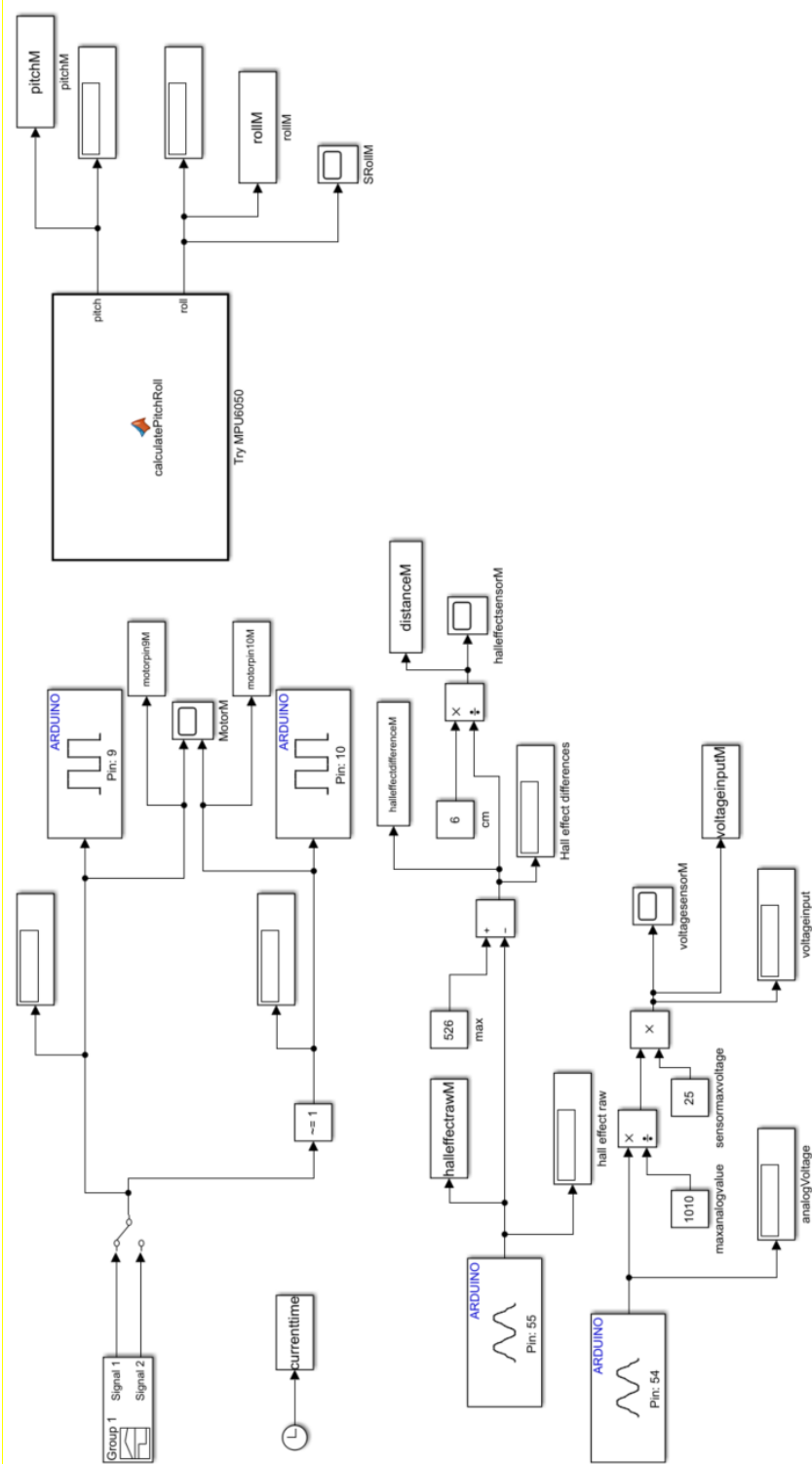
Yatim, H., Mat Dams, I. Z., & Hadi, M. S. (2013). Particle swarm optimization for identification of a flexible manipulator system. In *IEEE Symposium on Computers and Informatics, ISCI 2013* (pp. 112–117). IEEE Computer Society.

Zhang, P., & Zhang, J. (2022). Lower limb exoskeleton robots' dynamics parameters identification based on improved beetle swarm optimization algorithm. *Robotica*, 40(8), 2716–2731.

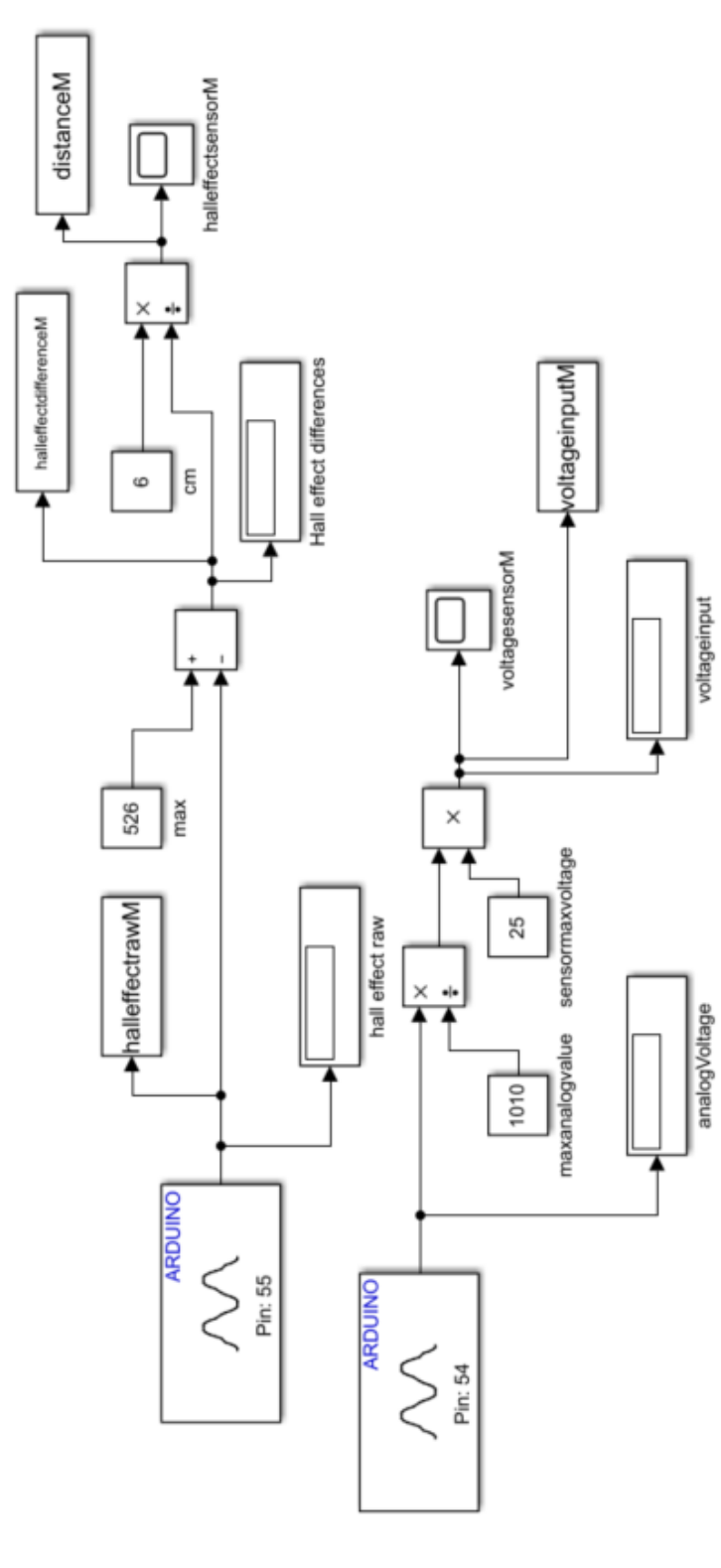
Zhang, Y., Zhao, J., Wang, L., Wu, H., Zhou, R., & Yu, J. (2021). An improved OIF Elman neural network based on CSO algorithm and its applications. *Computer Communications*, 171, 148–156.

APPENDICES

Appendix A



Appendix B



Appendix C

```
% Using complementary filter to fuse accelerometer and gyroscope data.
% Filter coefficient should be tuned to improve result
filterCoeff = 0.98;

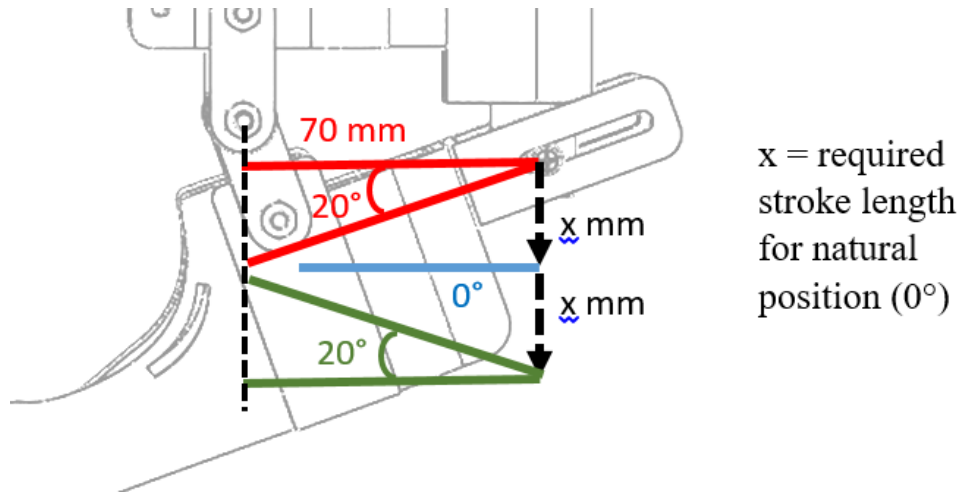
[accel, gyro, ~, timestamp] = read(imuObj);
% Calculate pitch and roll from acceleration values and remove offset
accelPitchAngle = atan2(accel(1), sqrt(accel(2)*accel(2) + accel(3)*accel(3))) - pitchAccelOffset;
accelRollAngle = atan2(-accel(2), sqrt(accel(1)*accel(1) + accel(3)*accel(3))) - rollAccelOffset;
% Subtract the offset values calculated at the beginning
gyro(1) = gyro(1) - rollGyroOffset;
gyro(2) = gyro(2) - pitchGyroOffset;
gyroPitchAngleDiff = gyro(2)*(timestamp - prevTimestamp);
gyroRollAngleDiff = gyro(1)*(timestamp - prevTimestamp);
prevTimestamp = timestamp;

% Use complementary filter to fuse accelerometer and gyroscope data
pitchAngle = filterCoeff*(pitchAngle + gyroPitchAngleDiff) + (1 - filterCoeff)*accelPitchAngle;
rollAngle = filterCoeff*(rollAngle + gyroRollAngleDiff) + (1 - filterCoeff)*accelRollAngle;

% sensor object returns angular velocity in radians/s.

pitch = pitchAngle;
roll = rollAngle;
```

Appendix D



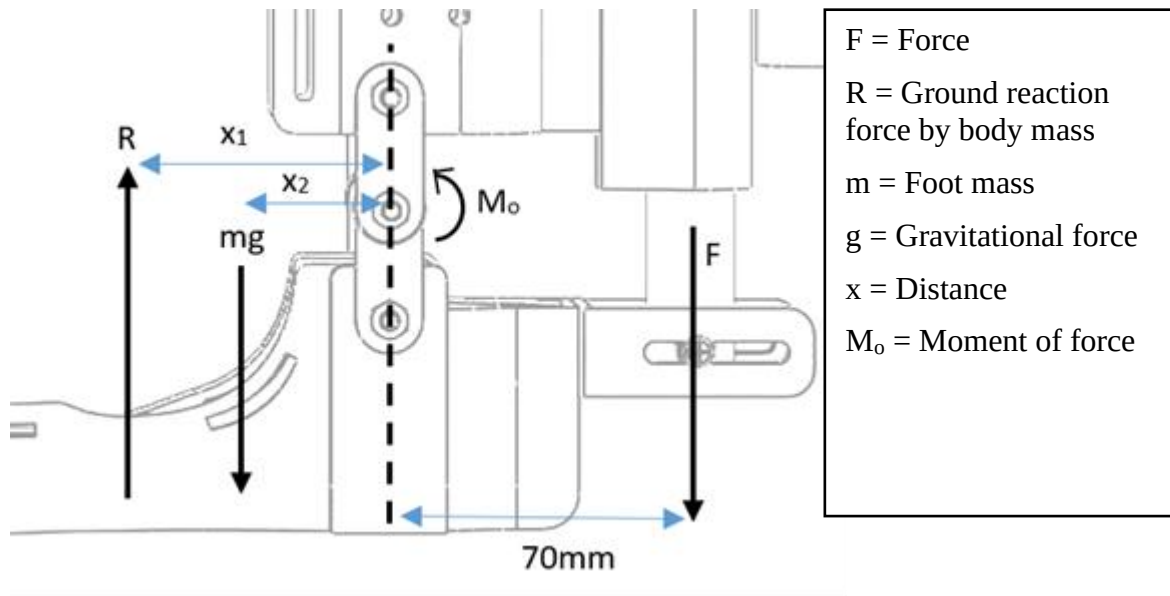
The actuator was placed with a 70 mm horizontal gap between the actuator's end and the hinge pivot. Following the simple trigonometric calculation, the necessary stroke length for the natural position (0°) is as follows:

$$\tan (20) = \frac{x}{70}$$

$$x = 70 \times \tan (20)$$

$$= 25.48 \text{ mm (approximately 25 mm)}$$

Thus, the foot brace would maintain its natural position (0°) when the actuator is fully extended to a length of 50 mm, perform plantarflexion with a -20° angle, and perform dorsiflexion with a +20° angle.



The average total body weight of 14 children is assessed to be 32.5 kg, with a foot weight of 0.49 kg (1.5% of body weight), and the distances x_1 and x_2 are considered to be 50 mm and 20 mm, respectively. The analysis of the necessary is displayed as follows:

$$\sum \tau + M_o = mg(x_2) - R(x_1)$$

$$= (0.49)(9.81)(0.02) - 32.5(9.81)(0.05)$$

$$= -15.85 \text{ Nm}$$

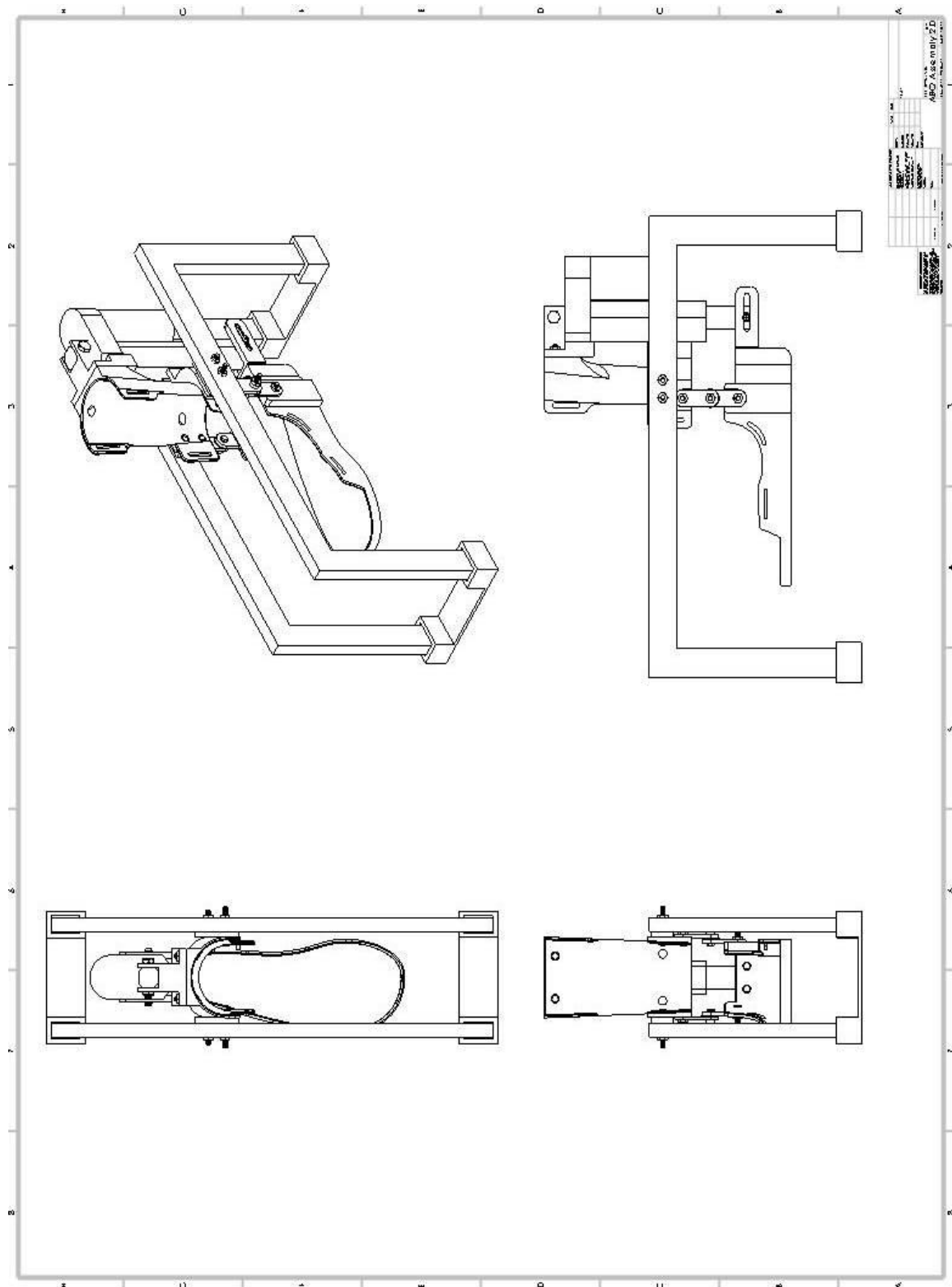
$$\sum \tau + M_o = -F(0.07)$$

$$-15.85 = -F(0.07)$$

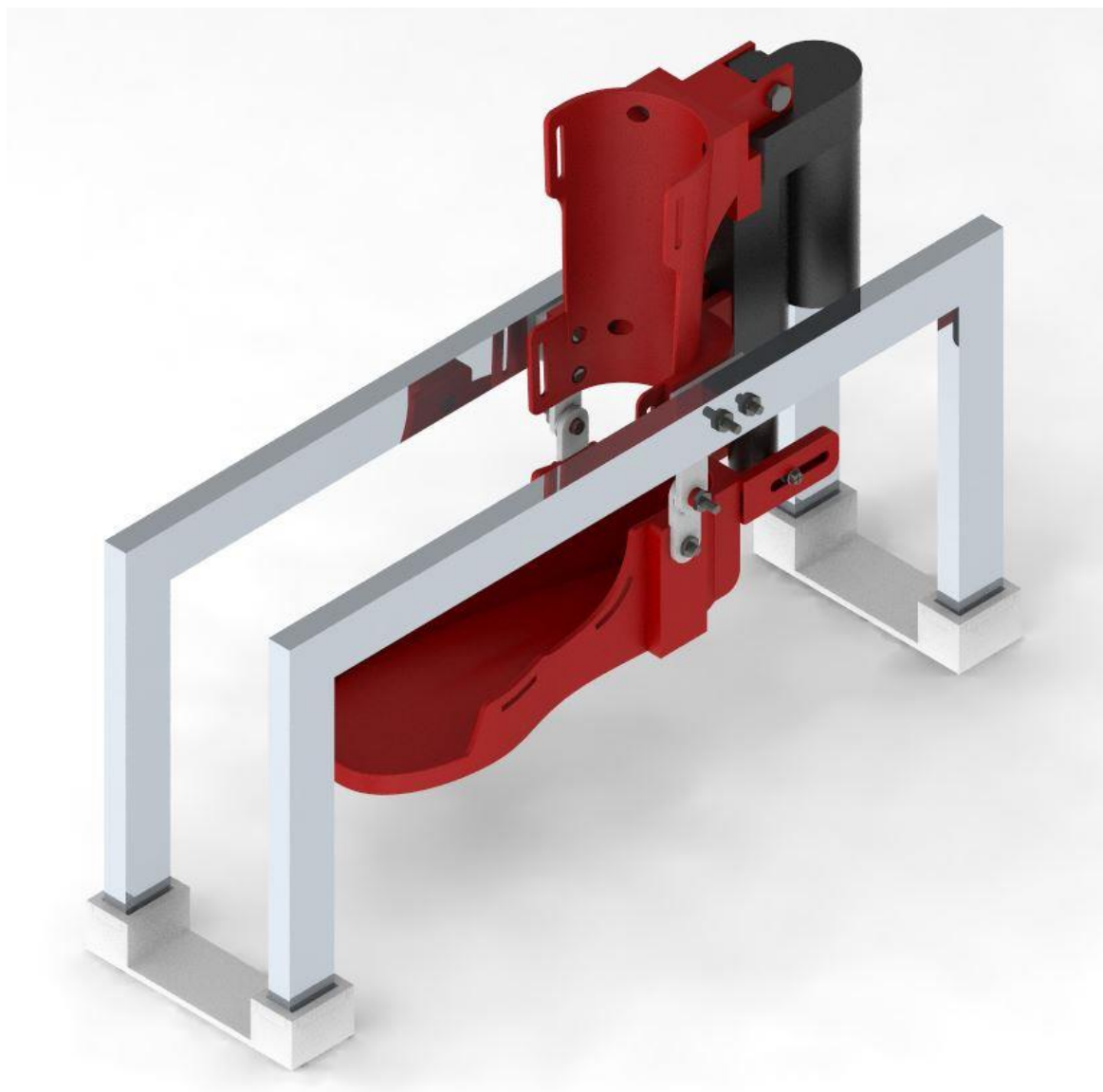
$$F = 226.43 \text{ N}$$

As a result, the actuator needs to exert a force greater than 226.43 N.

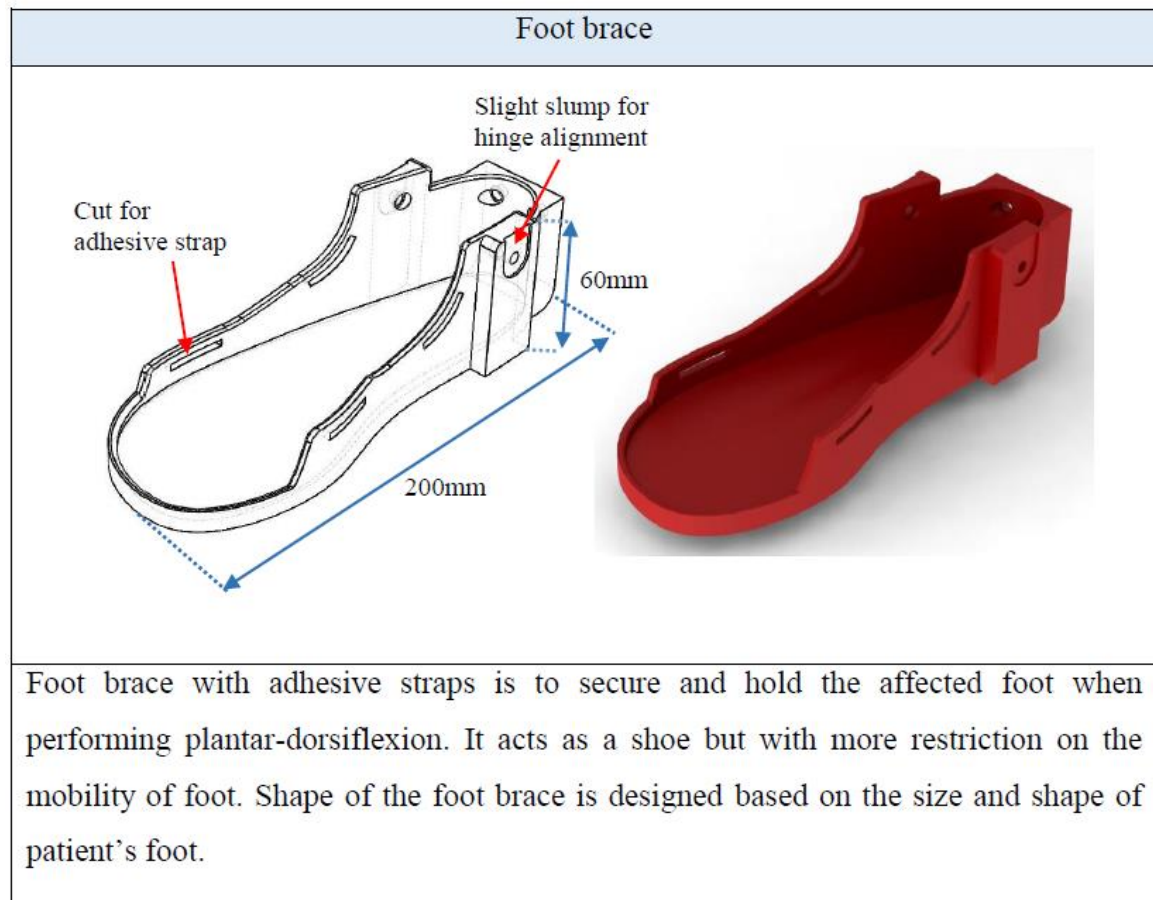
Appendix E



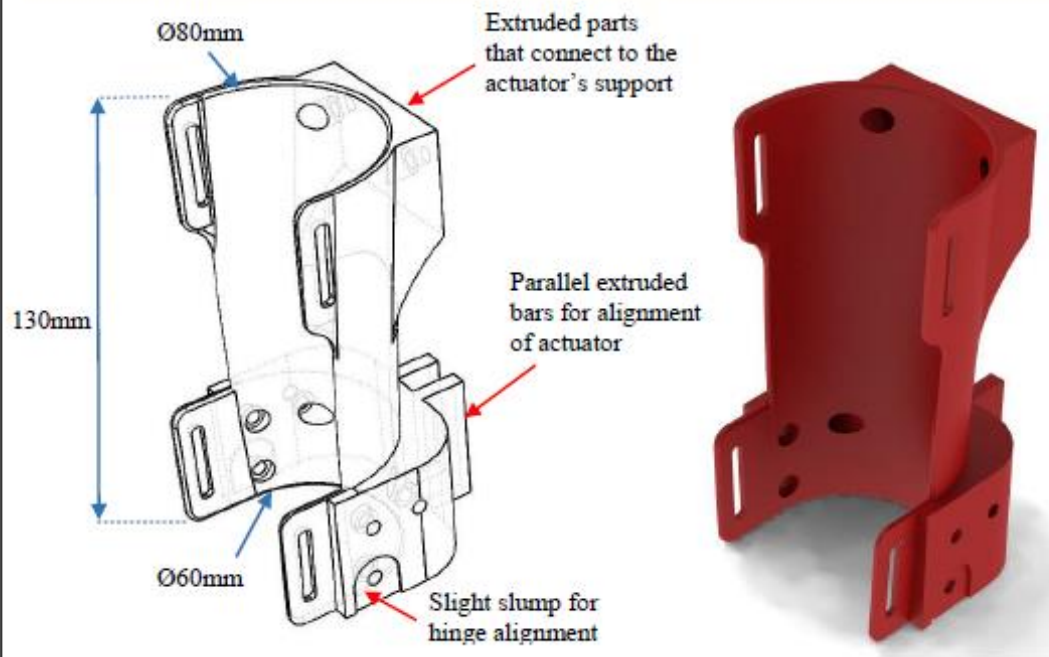
Appendix F



Appendix G

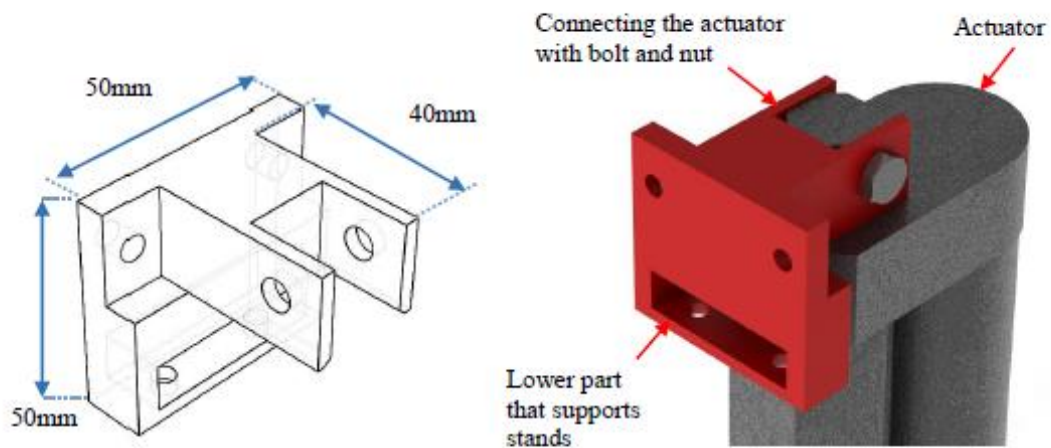


Lower limb brace



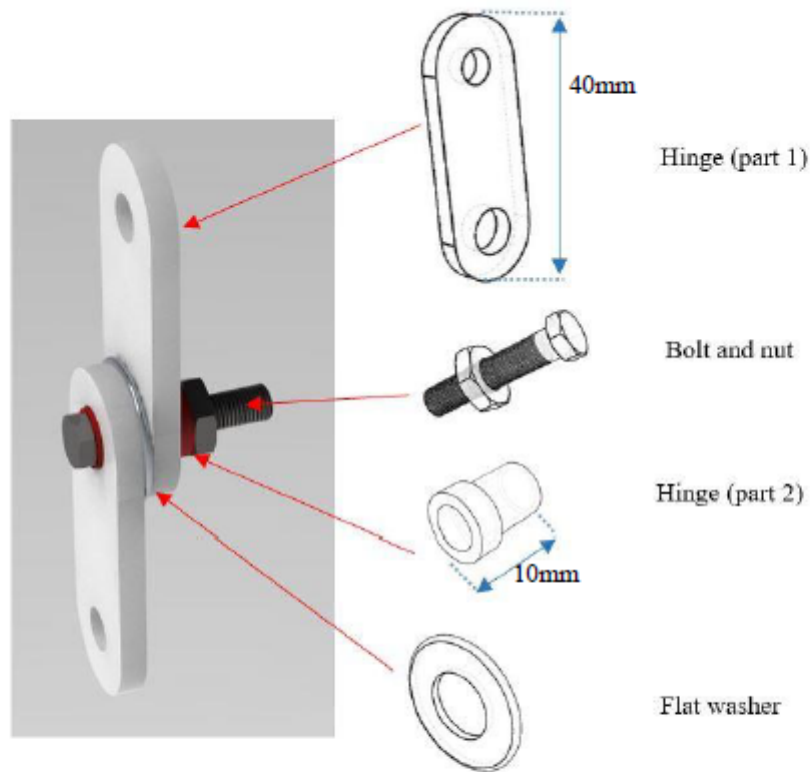
Lower limb brace with adhesive strap designed to stabilize the affected lower limb. Shape of the brace is based on the size and shape of patient's lower limb. Several holes are made for assembly to other parts such as hinge, support and stands.

Actuator's support



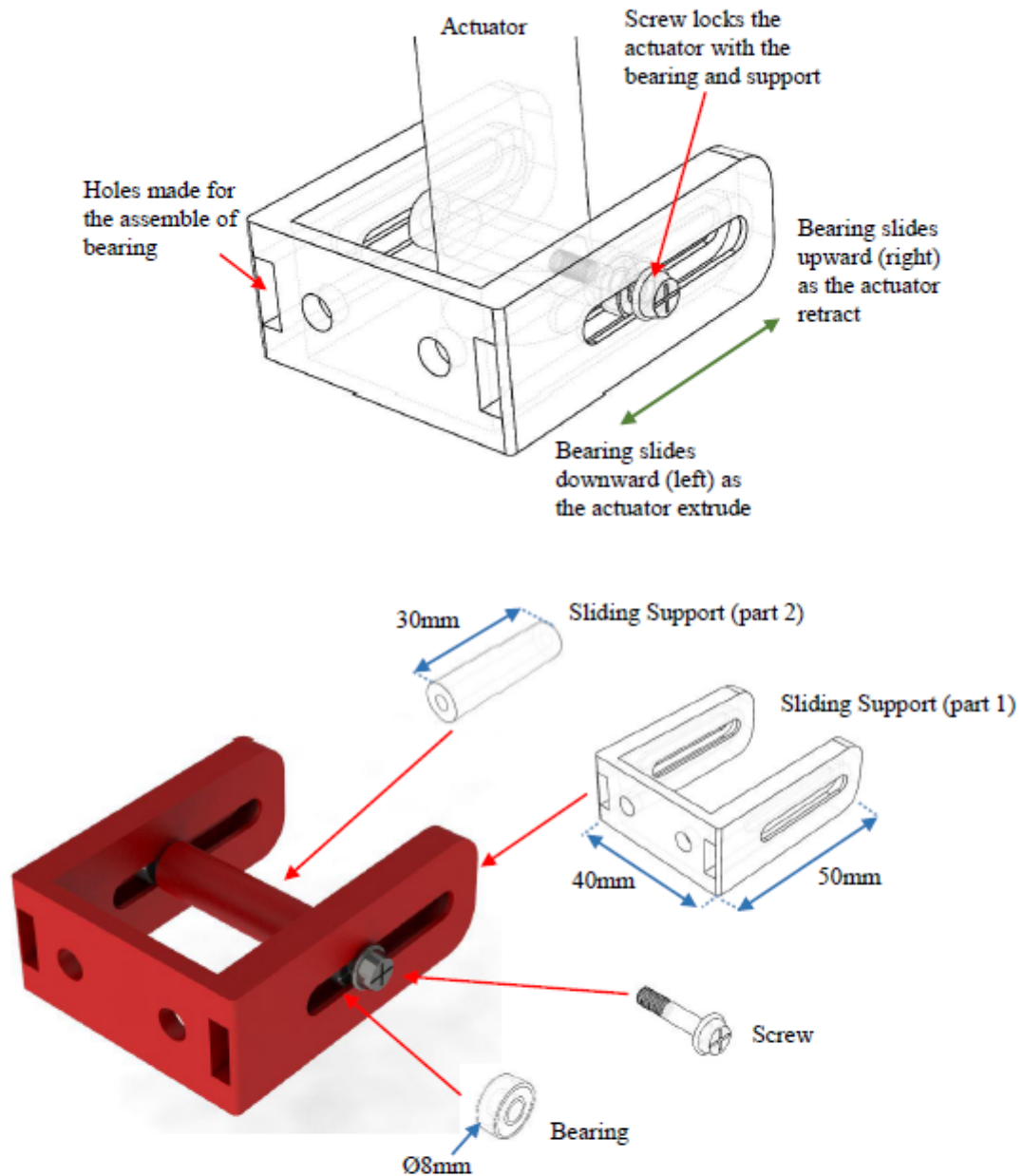
The support is designed to connect the lower limb brace and actuator as well as holding the actuator in position. Besides, an extended lower part is designed to connect the stands.

Hinge



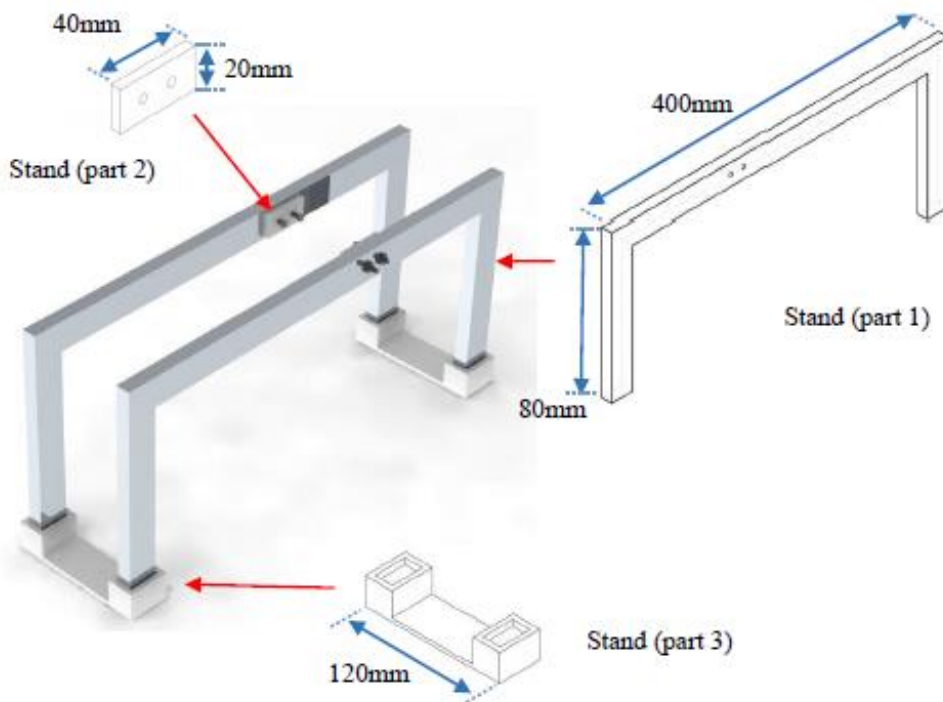
Hinge act as the pivot which located at the ankle. It provide 1 degree of freedom for the plantar-dorsiflexion. Two hinges (part 1) are used which each hinge connect 1 fixed point at the lower limb brace and foot brace respectively. Both hinges (part 1) are connected by bolt and nut that locked the hinges together and hinge (part 2) that enables the rotational hinge mechanism. A polished flat washer was used to reduce the friction between surfaces of two hinges (part 1).

Sliding mechanism



Sliding parts connect the actuator and the foot brace. It transfer force generated from actuator to the foot brace and hence complete the plantar-dorsiflexion function. Sliding support (part 1) and sliding support (part 2) connect the actuator and foot brace. Sliding support was designed with a sliding track for bearing. Bearing is necessary to reduce friction for enabling the sliding performs smoothly. Screws were used to assemble and hold the bearing on sliding track.

Stands



Stands were designed to allow the AFO could stand and locked at a straight upright position. The stands put the AFO in a fixed position to convenience the user. Stands (part 2) were used to make space for patient's lower limb. Stands (part 3) secure and connect both of the stands (part 1).