CHAPTER-4

FSR and IMU Sensors-based Human Gait Phase Detection and Its Correlation with EMG Signal for Different Terrain Walk

Highlights of the Chapter

- *Gait phase detection using FSR and IMU sensors*
- *Heuristics and zero-crossing algorithms detected the gait phase in 10ms using FSR and IMU sensors*
- *FSR and IMU signals were synchronized with EMG signal for the 5-terrains walk*
- *Gait phase detection using fuzzy logic implementation using Arduino Nano*

ABSTRACT

Purpose – This chapter presents gait analysis for five different terrains: (i) level ground, (ii) ramp ascent, (iii) ramp descent, (iv) stair ascent, and (v) stair descent, respectively.

Design/methodology/approach– Gait analysis has been carried out using a combination of the following sensors: force-sensitive resistor sensors fabricated in foot insole to sense foot pressure, a gyroscopic sensor to detect the angular velocity of the shank, and MyoWare electromyographic muscle sensors to detect muscle's activities. All these sensors were integrated around the Arduino nano controller board for signal acquisition and conditioning purposes. In the present scheme, the muscle activities were obtained using EMG electrodes from the tibialis anterior and medial gastrocnemius muscles. The acquired EMG signals were correlated with the simultaneously attained signals from the force-sensitive resistor and gyroscope sensors. The nRF24L01+ transceivers were used to transfer the acquired data wirelessly to the computer for further analysis. A Python-based graphical user interface (GUI) was designed to acquire, process, and analyze the

signal data. Later, in this chapter, a fuzzy logic-based algorithm is developed in the Arduino IDE to detect 7- gait phases. MATLAB's Neuro-Fuzzy Designer toolbox is utilized to obtain a rule base for these gait phases. The present work's motivation was to design and develop a reliable real-time gait phase detection technique that can be used later in developing a control scheme for the powered ankle-foot prosthesis.

Findings – The effectiveness of the gait phase detection was obtained in an open environment. Both off-line and real-time gait events and gait phase detections were accomplished for the force-sensitive resistor and gyroscopic sensors. Both sensors showed their usefulness for detecting the gait events in real-time, i.e., within ten milliseconds. The heuristic rules and a zero-crossing based-algorithm for the shank angular rate correctly identified all the gait events for the locomotion in all five terrains. In another experiment, the fuzzy logic-based system provided a prediction accuracy of 89.65 (± 3)% that is good enough, along with a 20ms prediction time to make it real-time gait phase detection.

Practical implications – This study leads to an understanding of human gait analysis for different types of terrains. A real-time standalone system has been designed and realized, which may apply to the design and development of ankle-foot prosthesis with a real-time control feature for the above five terrains.

Originality/value – The noise-free data from three sensors were collected in the same time frame from both legs using a wireless sensor network between two transmitters and a single receiver. Unlike the data collection using a treadmill in a laboratory environment, this setup is helpful for gait analysis in an open environment for different terrains. Further, the human kinematics and foot insole data were collected from each leg, and fuzzy logic

algorithms were implemented to make a standalone system for gait phase detection and current foot movement detection.

4.1 Introduction

Gait analysis is the systematic investigation of human locomotion [Tao *et al.*, 2012]. The term human gait defines the way of human walking, and it comprises of successive gait cycles. A gait cycle is well-defined as the duration from one heel strike to the subsequent heel strike on the same limb. The gait cycle divides into two main phases, the stance and swing phases [Winter, 2009]. The stance phase represents the time for which a foot is on the ground, whereas the swing phase represents the total time the foot is in the air. Figure 4.1 shows the gait cycle for the right reference lower limb. The gait cycle further splits into eight sub-phases, i.e., initial contact (IC), loading response (LR), midstance (MSt), terminal stance (TSt), pre-swing (PSw), initial swing (ISw), mid-swing (MSw), and terminal swing (TSw) [Winter, 2009; Prakash *et al.*, 2018]. Whittle (2014) used heel-contact, foot-flat, mid-stance, heel-off, toe-off, and mid-swing events to divide a gait cycle into convenient periods.

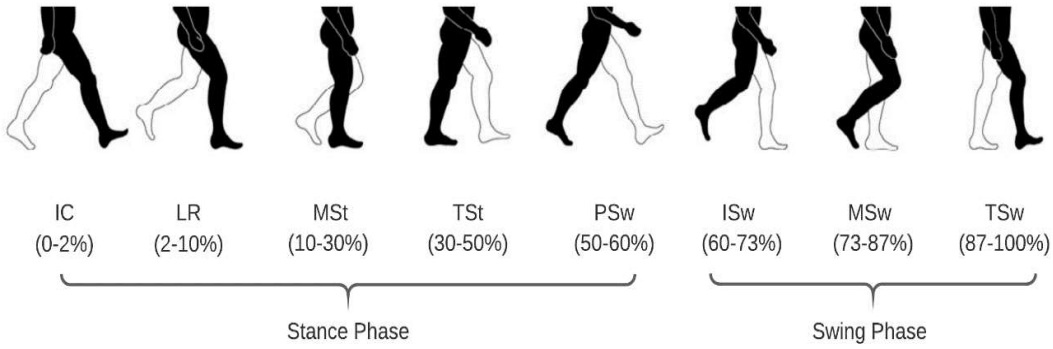


Figure 4.1 *Gait cycle (Stride)*

Available gait approaches can be categorized into vision-based, sensor-based, and hybrid approaches [Prakash *et al.*, 2018]. Non-wearable and wearable sensors-based devices are available for human gait analysis. Non-wearable sensors-based devices are most suitable for gait analysis in the laboratory environment. It includes an image processing-based system and a floor sensors-based system to capture gait. However, it is possible to capture human gait data outside the laboratory during everyday activities using wearable sensors-based devices. Wearable sensor-based gait recognition is used extensively in clinical diagnosis, rehabilitation, and sports. Tao *et al.* (2012) have presented the available wearable sensors and ambulatory gait analysis methods. The wearable sensors-based devices use sensors placed on the subject's feet, knees, and thighs. Sensors such as accelerometers, gyroscopes, force sensors, electro-goniometers, and Electromyography (EMG) are commonly used to characterize human gait [Muro *et al.*, 2014; Anwary *et al.*, 2018]. Magnetic system-based approach, electro goniometer, force myography (FMG), and imaging modalities (magnetic resonance imaging, computed tomography, and ultrasound medical imaging) are other techniques for human gait analysis. A combination of two or more approaches mentioned above is termed a hybrid approach.

Wearable sensor-based gait analysis techniques generally fall into gait kinematics, kinetics, and electromyography. The kinematics of the human gait defines the activities of the major joints and components in the lower limbs. Gait kinetics highlights the study of forces and moments that result in the human segment's movement. The human gait includes changes in the angles of each joint in the lower limbs. These angles change with changes in the bioelectrical energy of skeletal muscles. Therefore, the EMG signal generated during human gait relates to the posture angle of each joint during human gait. In the present study, Force-sensitive resistors (FSRs) and inertial measurement unit (IMU) sensors were used simultaneously for gait phase detection. Also, the EMG signals

were acquired in the same time frame from two leg muscles. The waveform patterns are explained in terms of flexion and extension of the ankle during human locomotion on different terrain. FSRs sensors of various kinds and sizes were used by many authors to be employed in smart insoles either to measure foot pressure during walking or as footswitches (Aminian *et al.*, 2002; Pappas *et al.*, 2004; Malvade *et al.*, 2018). Godiyal *et al.* (2018) have acquired FMG signals from the thigh to determine the heel-strike (HS) and toe-off (TO) events during overground and ramp walking. The FlexiForce is a kind of standard tiny thin FSR whose resistance varies with pressure. In this study, the authors used 4- FlexiForce A401 sensors placed on each foot insole for gait phase detection.

Another primarily used sensor for the gait analysis is the IMU sensor that provides the measurement of acceleration and gyroscopic signals from the body during locomotion (Sabatini *et al.*, 2005; Joshi *et al.,* 2014; Muller et al., 2015; Gouwanda *et al.,* 2015 and 2016; Maqbool *et al.*, 2016). Sabatini *et al.* (2005) have devised a gait phase segmentation technique by integrating IMU to determine off-line temporal gait parameters using MATLAB. Gouwanda *et al.* (2015, 2016) have acquired gyroscope data from InvenSense MPU 6000 to identify HS and TO gait events and determine the temporal gait parameters. The peaks greater than zero correspond to mid-swing (MS), and troughs less than zero correspond to HS and TO in the curve characteristics. The authors used algorithms based on heuristics and zero-crossing methods for gait event detection in the present paper. The first derivative of the angular rate was used for monitoring zero crossings, which correspond to a potential MS, HS, and TO. When a zero crossing or a sign change was detected, the corresponding angular rate was recorded as a potential point. Finally, heuristics were used to evaluate these possible points to determine the related gait events.

Pioneer's work by Ng *et al.* (1994) revealed that fuzzy logic better-detected gait phases than those found using non-fuzzy rules in a lookup table format. It is tough to distinguish gait sub-phases concerning a threshold value as gait parameters do not change visibly for each sub-phase. This difficulty can be solved by extending the fuzzy logic principle to detect gait phases for fuzzy membership values (Senanayake *et al.*, 2010). Researchers have used fuzzy logic techniques to control upper limbs (Tasar *et al.*,2017) and lower limb prostheses (Kadhim *et al.*, 2020). Among these studies, few researchers worked on image-based control (Prakash *et al.*, 2016).

In contrast, others have used wearable sensors such as Electromyography (EMG), IMU sensors, FSR sensors on foot insole, knee and ankle angle sensors (Kurniawan *et al.*, 2019). During the literature survey, all authors used the Fuzzy Logic toolbox in MATLAB software to implement the fuzzy logic rule for their applications. Few researchers also used the Neuro-Fuzzy Designer toolbox to design, train, and test Adaptive Neuro-Fuzzy Inference Systems (ANFIS) using input/output training data. A study done by Senanayake *et al.* (2011) reported the implementation of the Fuzzy Inference System (FIS) in MATLAB, which was later converted into Laboratory Virtual Instrument Engineering Workbench (LabVIEW) platform to enable real-time abilities. However, there is still a scope to find a standalone system that can be carried with the amputee while walking on different terrains and provide real-time output to control prosthetic devices.

Technological improvements have led to numerous wearable devices for the physical support and restoration of human locomotion (Tucker *et al.*, 2015). Orthosis describes a device that is used for supporting a person with limb- impairment. Blaya *et al.* (2004) have presented an active ankle-foot orthosis where the orthotic joint impedance is

controlled during stride for analyzing drop-foot gait. Another boon to the amputees is prosthetic devices that replace a missing part of the body, and it plays an essential role in rehabilitation (Rietman *et al.*, 2002). There is excellent scope for improving lower-limb prostheses that require collaborative research from multidisciplinary subjects such as biomedical, mechanics, mechanical, electrical, and mechatronics (Kumar *et al.*, 2017). The ankle joint receives the body's whole weight; hence, the ankle's biomechanical analysis during gait on the ground level, slopes, and stairs plays a significant role while designing ankle-foot prosthesis. This paper presented the off-line and the real-time experimental gait analysis results using FSR, IMU, and EMG sensors for different walking terrains in an open environment. Arduino Nano-based standalone system is presented to identify the gait phases in real-time using fuzzy logic techniques. This study is useful later for the control of an ankle-foot prosthesis.

4.2 Materials and Method

4.2.1 Sensor Module

Figure 4.2 illustrates the wireless setup designed for acquiring the signals from the sensors. The sensor module used in the present study for gait analysis consists of FSR, IMU, and EMG sensors. FlexiForce A401 [Tekscan Inc.], a standard form of FSR sensors, was employed in the present study as footswitch to determine gait phase parameters. It also serves to detect the foot pressure after proper calibration. A GY-87, 10-DOF IMU sensor board, featured a 3-axis accelerometer, 3-axis gyroscope, 3-axis magnetometer, and barometric pressure sensor. EMG signals were acquired from the tibialis anterior (TA) and medial gastrocnemius (MGAS) muscles of the leg.

The surface EMG (sEMG) signal acquired from TA muscles represents dorsiflexion, whereas the signal from MGAS represents the foot's plantar-flexion. Arduino nano controller board read the FSR sensors data from the foot insole, IMU accelerometer/gyroscopic signals from the shank, and the sEMG signals from TA and MGAS muscles. The nRF24L01+ wireless modules were used to transmit data between Arduino nano and the computer.

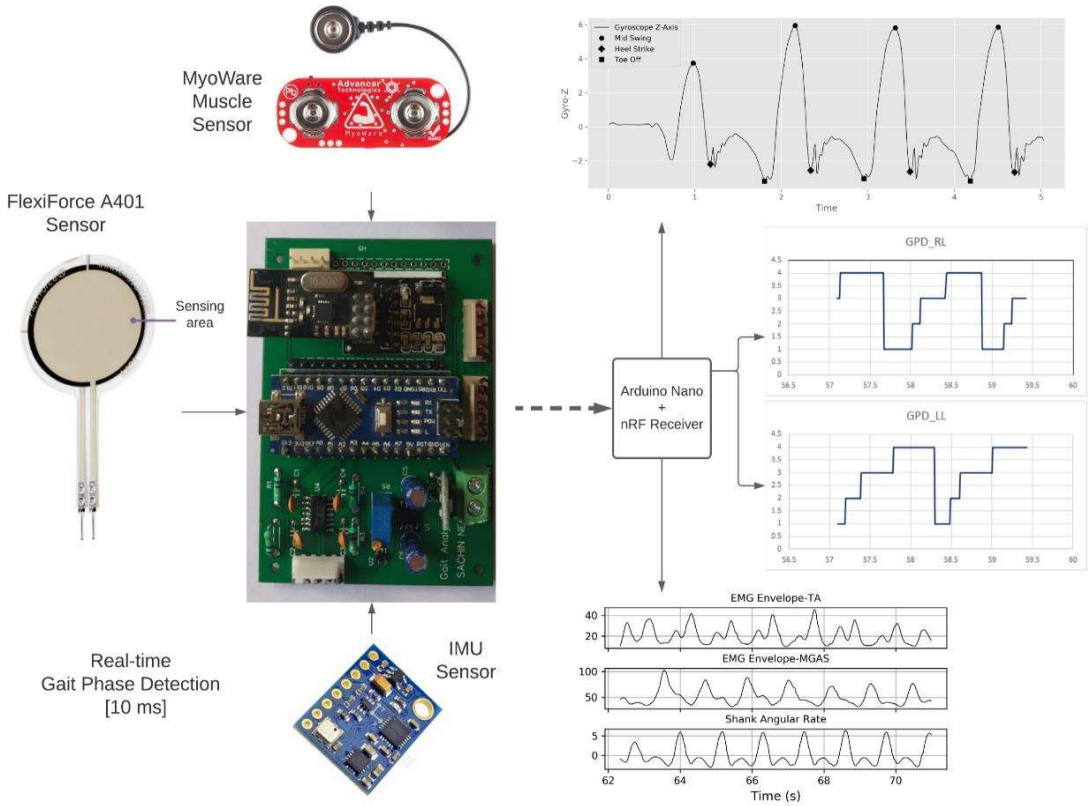


Figure 4.2 *Wireless system for gait analysis*

4.2.2 Signal Conditioning

FlexiForce A401 Sensor is a piezoresistive force sensor with a 25.4 mm sensing area. The primary circuit for the detection of the FlexiForce sensor signal is shown in Figure 4.3. To generate stable negative V_{REF}, an ICL7660 charge-pump IC was used, followed by a

voltage divider and LM321 OpAmp IC. By changing the V_{REF} voltage, one can modify the dynamic range of this force sensor. All the FlexiForce sensors were calibrated for different weights before installing them on the shoe insole. Figure 4.4 shows the *Weight versus Resistance* curve and the *Weight versus Voltage* calibration curve for four-FlexiForce sensors. The sensor's electrical output is related to the actual weight applied to the sensing area by doing the calibration.

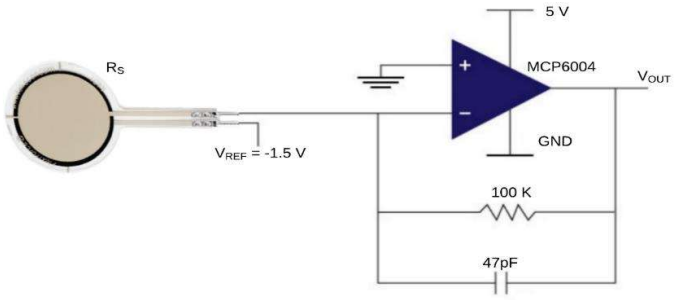
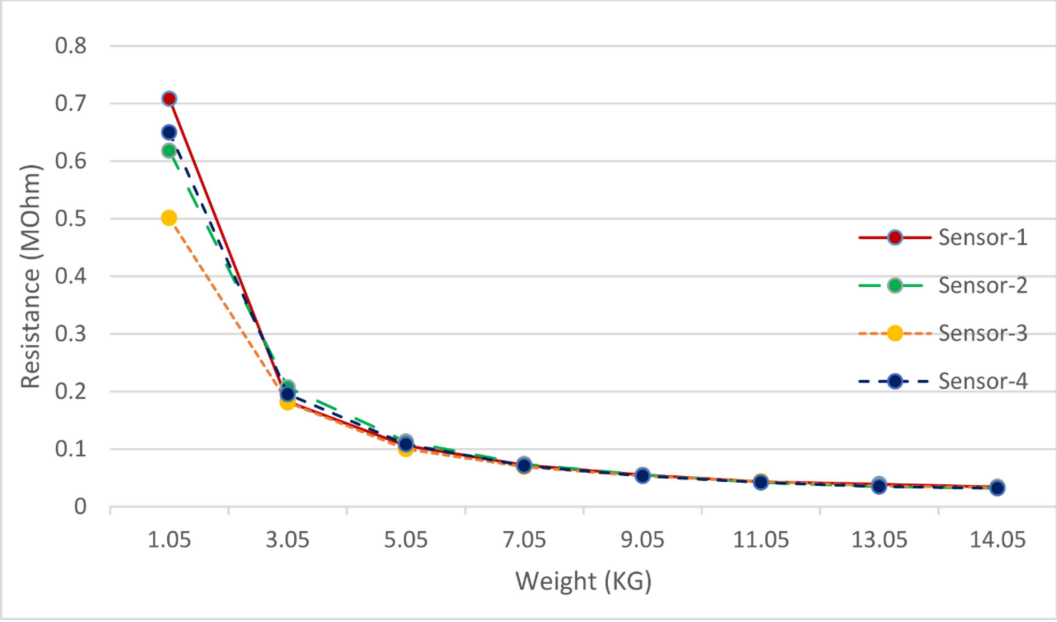
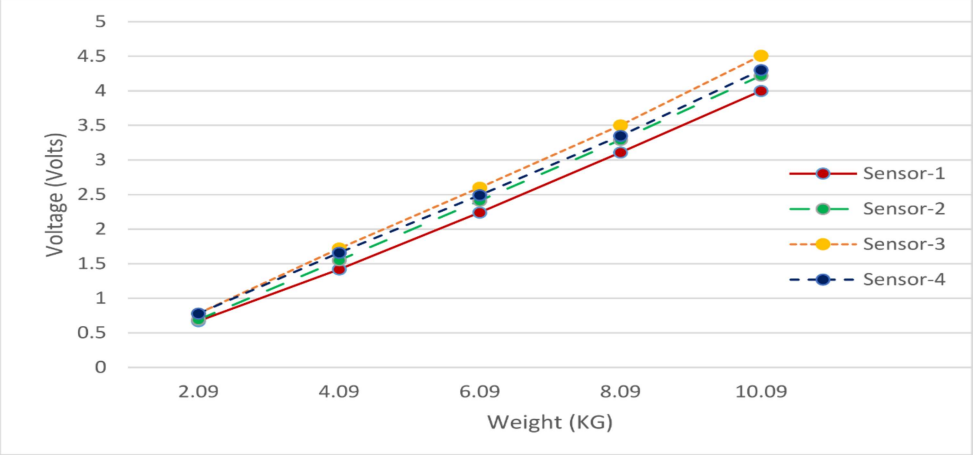


Figure 4.3 *Signal conditioning for FlexiForce piezoresistive force sensor*



(a)



(b)

Figure 4.4 *(a)Weight versus Resistance, and (b)Weight versus Voltage curves for FlexiForce sensors*

In the GY-87 board, the MPU6050 device integrates a three-axis gyroscope and a three-axis accelerometer on the same silicon chip collectively with an onboard Digital Motion Processor capable of processing complicated nine-axis Motion Fusion algorithms. The sEMG signals were acquired using wet surface electrodes. Where raw signals were fed to a low-power instrumentation amplifier that offers excellent accuracy with very low offset voltage, drift, and high common-mode rejection. After that, the signal is passed through a 20-500Hz bandpass filter to remove the motion artifacts, external noises of low frequencies, and noises occurring at high frequencies (De Luca *et al.*, 2002; Hermens *et al.*, 2000). An envelope detector containing a precision rectifier circuit and a low pass RC circuit was employed to yield a linear envelope of amplified and filtered EMG signal (Prakash *et al.*, 2019).

4.2.3 Heuristic Rules and Zero-Crossing Algorithm-based Gait Phase Detection

4.2.3.1 Gait phase detection using FSR sensors

Stance and swing phases can be detected easily using two FSR sensors. One FSR was attached below the heel and another one below the toe position. Figure 4.5 shows the flow

chart for the detection of these gait phases. The method consisted of a threshold depending on the weight of the subject and a set of heuristics. The code was implemented in the Python language. Here FF_{heel} and FF_{toe} denote the output of FSR sensors attached at heel and toe positions, respectively. The algorithm first detects the HS event, whose timing and magnitude are represented by hs_ptt and hs_ptm. Once the HS point is detected, the algorithm searches for the TO point, characterized by to_ptt and to_ptm. By assuming the HS occurred before the TO, the swing and stance phases were obtained using the following equations:

$$t_{gait}[j] = hs\_ptt\ [j+1, 0] - hs\_ptt\ [j, 0] \tag{4.1}$$

$$t_{swing}[j] = hs\_ptt\ [j+1, 0] - to\_ptt\ [j, 0] \tag{4.2}$$

$$t_{stance}[j] = t_{gait}[j] - t_{swing}[j] \tag{4.3}$$

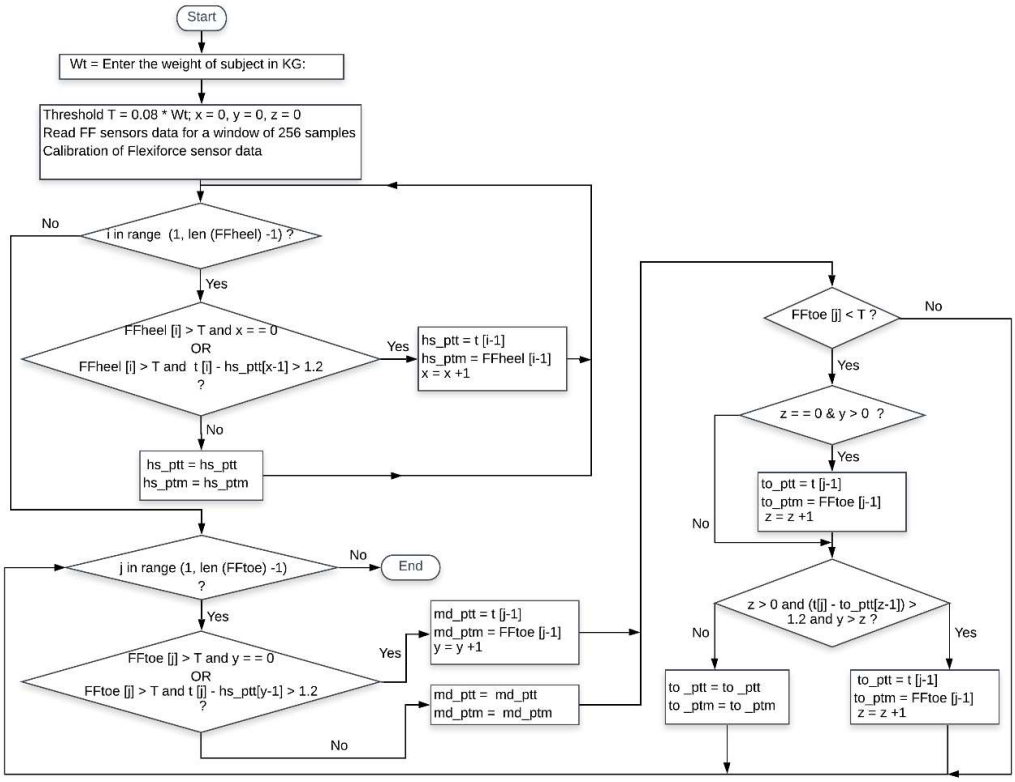


Figure 4.5 *Flow chart of the gait phase detection algorithm using the FSR sensor*

Two sensors can detect only stance and swing phases. Au *et al.* (2009) used four phases in a gait cycle: controlled plantar-flexion (CP), controlled dorsiflexion (CD), powered plantar-flexion (PP), and swing phase (SW). The ankle undergoes a CP intending to lower the foot to the ground. It starts with HS and ends with a flat foot. The ankle begins to dorsiflex (CD) from the toe-strike, and it stays till maximum dorsiflexion. Later in the stance phase, PP begins at maximum dorsiflexion. The ankle releases the stored elastic energy for the forward propulsion of the body, and it terminates at TO. SW starts at TO and ends at HS for a level ground walk. For these four phases, a real-time algorithm was realized to run on the Arduino nano board. Here four FSR sensors were used beneath the heel, toe, metatarsal-1 (MT-1), and metatarsal-5 (MT-5) region of both feet. Algorithm 4.1 highlights basic steps of real-time gait phase detection.

Algorithm 4.1: *Real-time gait phase detection*

1. Initialize the Arduino ports A0 to A3 as input ports.
2. Read FSR data from the heel, toe, MT-1, and MT-5 positions.
3. Apply the filter equation corresponding to a low pass digital filter with a cut-off frequency of 1 Hz and a sampling rate of 100 samples/s.

$$\boldsymbol{y(n+1) = 0.9391 \times y(n) + 0.0609 \times u(n)}$$

4. Assign each FSR sensor either 1 or 0 depending on the threshold as below
   if *(FSR_n >= threshold)*:
   $FSR_n = 1$
   else:
   $FSR_n = 0$
5. Write the set of rules for gait phase detection (GPD) in real-time, where the value of GPD is 1 to 4, corresponding to the above four phases.
   if ((*Heel == 1*) **and** (*MT1 == 0*) **and** (*MT5 == 0*) **and** (*Toe == 0*)):
   GPD = 1
   else if ((*Heel == 1*) **and** (*MT5 == 1*)):
   GPD = 2
   else if ((*Heel == 0*) **and** ((*MT1 == 1*) **or** (*MT5 == 1*) **or** (*Toe == 1*))):
   GPD = 3
   else if ((*Heel == 0*) **and** (*MT1 == 0*) **and** (*MT5 == 0*) **and** (*Toe == 0*)):
   GPD = 4
6. Go to step 2.

4.2.3.2 Gait phase detection using IMU sensors

A single IMU sensor was also placed at the shank to detect stance and swing phases by acquiring the lower limb's angular velocity. The sensor was placed on the right leg's lateral side (medially to left leg) by pointing the z-axis outwards perpendicular to the

sensor module's surface. The angular velocity waveform shows one maxima and two minima; these correspond to MS, HS, and TO, respectively. Figure 4.6 shows a detailed flow chart to detect gait events. Algorithm 4.2 highlights the steps for gait analysis.

Algorithm 4.2: *Real-time gait phase detection*

1: Read the shank angular rate (ω_z).
2: Obtain the potential points where the slope changes its sign in the waveform.
3: Identify the potential points where the signal's amplitude is greater than a threshold (2 rad/s). Only one maximum point is allowed between the threshold interval (0.8 s).
4: Mark these points as MS points.
5: Search for two minima (HS points) points after detecting an MS. Only one HS point is allowed between the threshold interval (0.8 s).
6: Identify the second minima after which ω_z crosses zero points.
7: Obtain the stance and swing phases using equations 1 to 3.
8: Go to step 1.

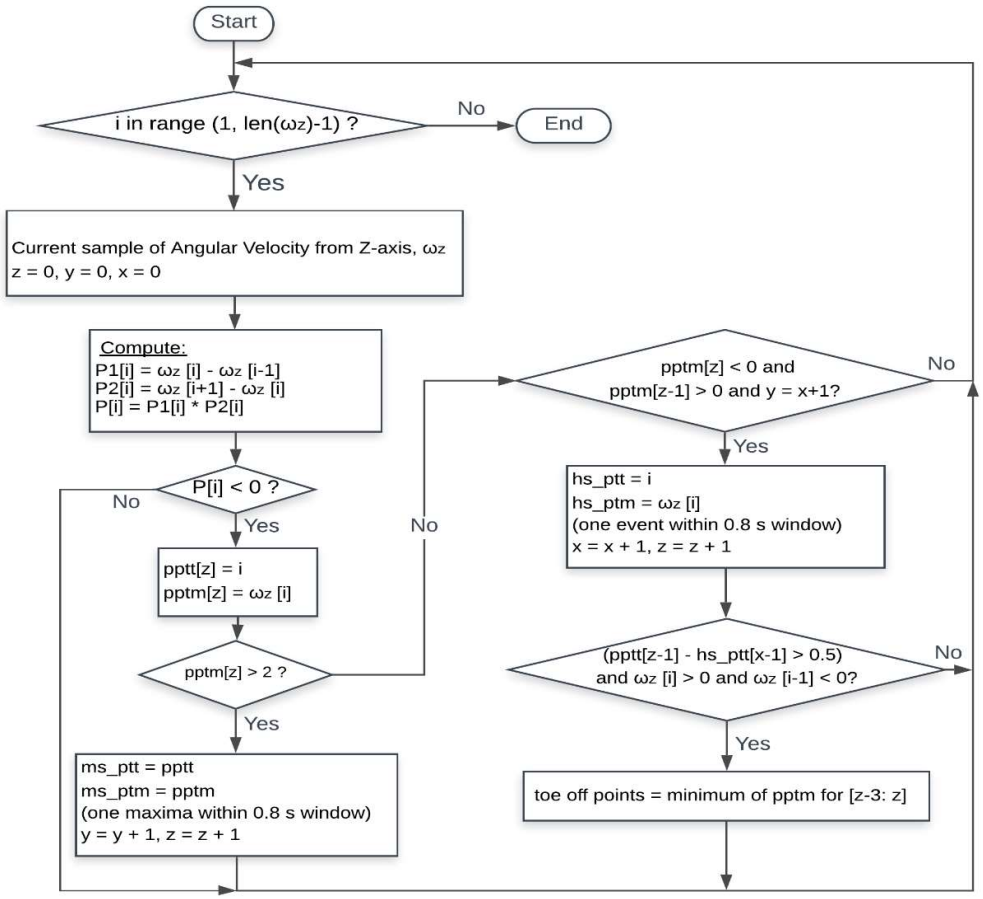


Figure 4.6 *Flow chart of the gait phase detection algorithm using the IMU sensor*

4.2.4 Gait Sub-phase Detection using Fuzzy Logic Implementation in Arduino

4.2.4.1 Fuzzy logic for gait phase detection

Fuzzy logic is applied in scenarios where there is no exact answer. It is primarily used to give an approximate classification in a vague dataset. Fuzzy logic is a branch of machine intelligence that allows computers to see the world more humanistic. Fuzzy logic is based on the form "if…then," which maps the input to the output. A typical fuzzy logic system consists of 3 parts: the fuzzifier, the FIS, and the defuzzifier, as shown in Figure 4.7. The fuzzifier intakes the crisp input values and maps them to a range of [0-1] by applying suitable membership functions. A membership function is a curve that maps input values to membership values between 0 and 1. The fuzzified values are passed through the FIS, and compared with a rule base of the form "if…then" to give the fuzzified output values (Zadeh, 1988). These values pass through a defuzzifier to give crisp output values.

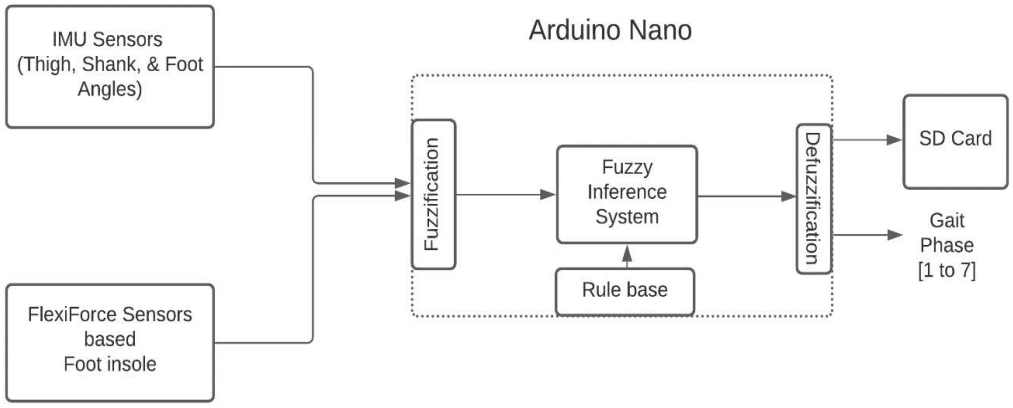


Figure 4.7 *Arduino Nano-based data acquisition system*

Two most prominently used fuzzy inference systems are the Mamdani fuzzy inference and Takagi-Sugeno-Kang fuzzy inference. In the Mamdani system, the output for each rule is in the form of a fuzzy set. While the Mamdani system is easy to understand by a layman, it falls short on accuracy and computational efficiency. The Sugeno model makes

up for this shortcoming using output membership functions that are either constant or linear. Defuzzification is done by taking a weighted average of a few data points (Sugeno, 1985). In the present study, the Sugeno FIS performs gait phase detection in real-time.

4.2.4.2 Adaptive Neuro-Fuzzy Inference Systems

ANFIS combines a FIS with a neural network. The amalgamation of neural networks with fuzzy logic has gained traction since Jang et al. (1993) proposed. An adaptive network is a feed-forward multi-layered network with each layer comprising of different nodes. Galván-Duque *et al.* (2015) compared the performance of ANFIS with various linear and nonlinear models. Such systems benefit from the neural networks' adaptability and learning, and also use fuzzy logic linguistic inference rules. Figure 4.8 shows the ANFIS model structure used in the present study.

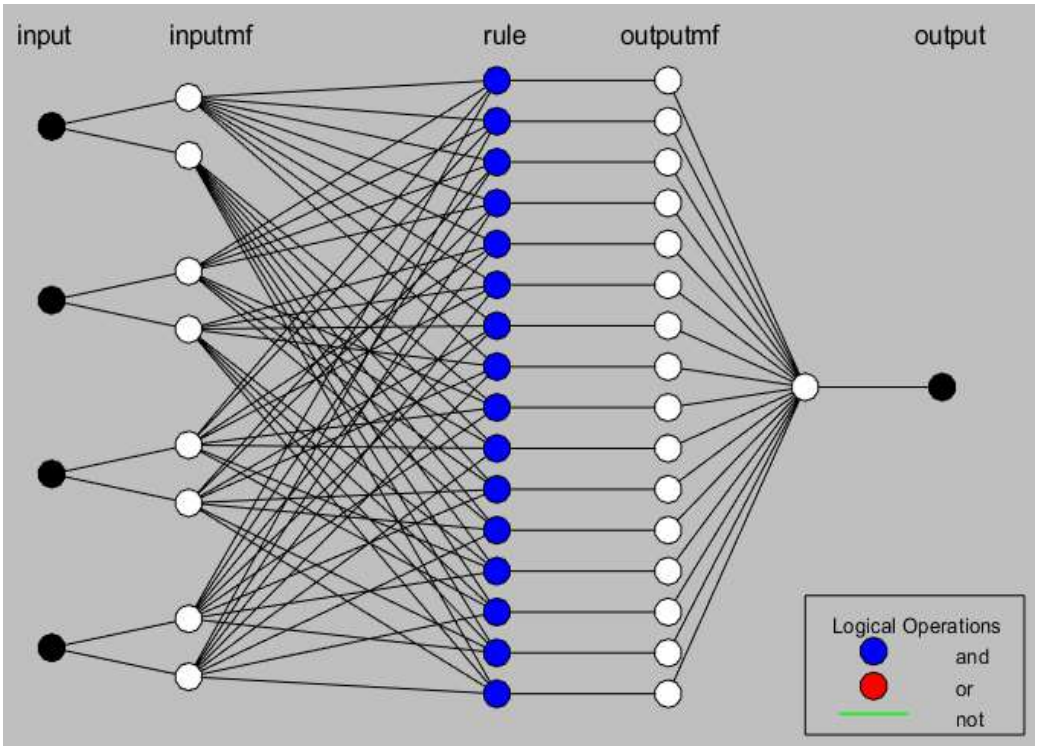


Figure 4.8 *Model Structure of ANFIS [using MATLAB]*

The authors used the Takagi Sugeno model as it can be effectively trained using ANFIS. A typical Takagi Sugeno model for two inputs, x, and y, can be seen as mentioned in equations 4.4-4.6.

$$IF\ x\ is\ A \quad AND|OR \quad y\ is\ B \quad THEN\ z = f(x, y), \tag{4.4}$$

$$f(x, y) = a_0 + a_1 x + a_2 x^2 + \cdots + a_n x^n + b_1 y + b_2 y^2 + \cdots + b_n y^n \tag{4.5}$$

where A and B represent fuzzy sets in antecedents and z is the crisp function in the consequent (Sugeno *et al.*, 1988). If n = 0, then it is called a zero-order Sugeno model; for n=1, it is called a first-order Sugeno model, and so on. In this study, a first-order linear Sugeno model is used.

$$z = a_0 + a_1 x + b_1 y \tag{4.6}$$

The authors used the trapezoidal membership function for the fuzzification of the input data, and it is expressed as

$$f(x; a, b, c, d) = \max\left(min\left(\frac{x-a}{b-a}, 1, \frac{d-x}{d-c}\right), 0\right) \tag{4.7}$$

MATLAB's Neuro-Fuzzy Designer toolbox is used to train the ANFIS model for the trapezoidal membership function. The model is trained for 5000 epochs and optimized using the hybrid optimization system.

4.2.5 Wireless Sensor Network

Wireless sensor networks (WSNs) are self-organized distributed systems built through communication with many sensor nodes in the sensory area. It incorporates recent hi-tech sensors and networks with embedded computing and distributed information processing technology. Arduino is a microcontroller which has gained significant popularity in recent years. It is an open-source platform to build an electronic prototype with high flexibility,

low price, and ease to use. Unlike wired network designs, wireless network designs offer flexibility during the acquisition of sensor data. For human gait phase detection in an open environment, wireless sensor technology gives hassle-free data recording and movement freedom compared to wired connections. So, the authors have implemented this for real-time gait phase detection. The core of the WSN consists of the base node (coordinator) and end devices.

Bluetooth, ZigBee, and nRF24L01+ are commonly used wireless protocols for WSNs. The nRF24L01+ transceiver module is intended to function in a 2.4 GHz ISM frequency band and listen to six other devices. The data transfer rate ranges from 250kbps to 2Mbps. A star topology was used in the present study, where a sensor module was attached to each leg. Each module consists of four- FSRs, an IMU sensor, two- EMG sensors, an Arduino nano, and an nRF24L01+ transceiver. Both devices act as transmitters. A set of an Arduino nano and nRF24L01+ transceiver acts as the receiver, connected to the computer by using a serial port. For off-line gait analysis, the Tera Term software was used to save the serial data from the receiver module in text format on the computer, which can then be extracted for later analysis in Python. For real-time gait phase detection, Arduino-based code was used for FSR sensor data; however, the algorithm was coded in Python for IMU data.

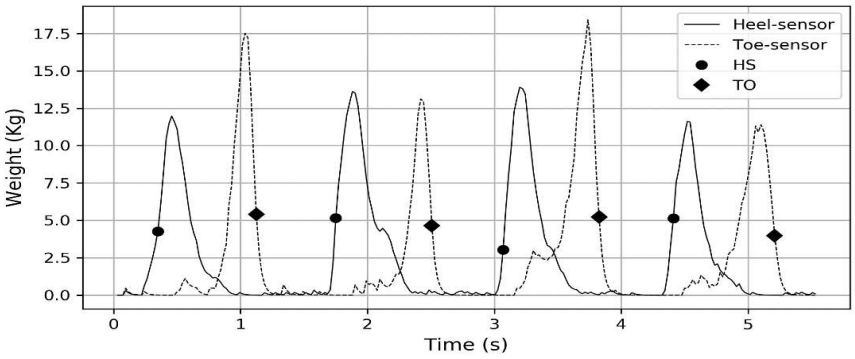
4.3 RESULTS AND DISCUSSION

Data were recorded from six healthy subjects (five male and one female), where the subject's mean (± STD) age, height, and weight were 27 (± 9.2) years, 166 (± 8.8) cm, and 70.5 (± 12) kg respectively. The length of the level ground, ramp, and stair terrains used in this study were 34, 8, and 9 feet, respectively. The height of each stair was 20 cm, whereas the ramp slope was 8 degrees. A subject-specific foot insole was designed to

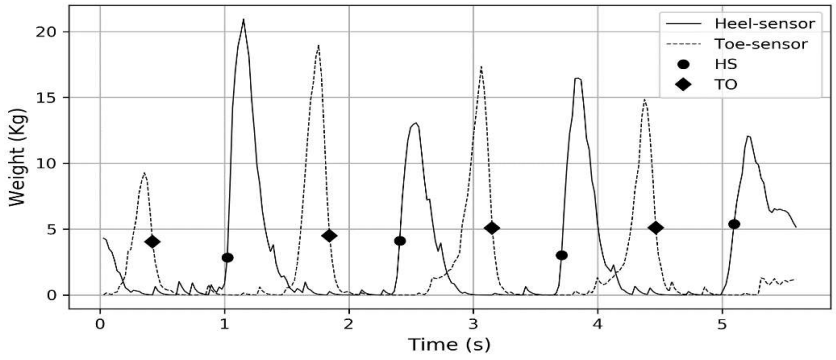
record the FSR sensors data, where 30- trials data of a subject were mentioned for the analysis.

4.3.1 Gait Phase Detection using FSR Sensors

Figure 4.9 shows the two FSR sensors' responses placed on each leg's heel and toe sites. The results show that all the events, i.e., the MS, HS, and TO, are correctly identified. Using Equations 4.1 to 4.3, the swing and stance phases were obtained, which are mentioned in Table 4.1. For the level ground walk, the swing phase was detected 41% of the gait cycle, and the stance phase was found to be 59%.



(a)



(b)

Figure 4.9 *Gait phase detection using 2-FSR sensors* (a) Right leg, (b) Left leg

Table 4.1 *Gait parameters using two- FSR sensors*

Leg	t_gait (s)	t_swing (s)	t_stance (s)
Right	1.298 (±0.046)	0.535 (±0.032)	0.763 (±0.029)
Left	1.290 (±0.049)	0.532 (±0.032)	0.758 (±0.029)

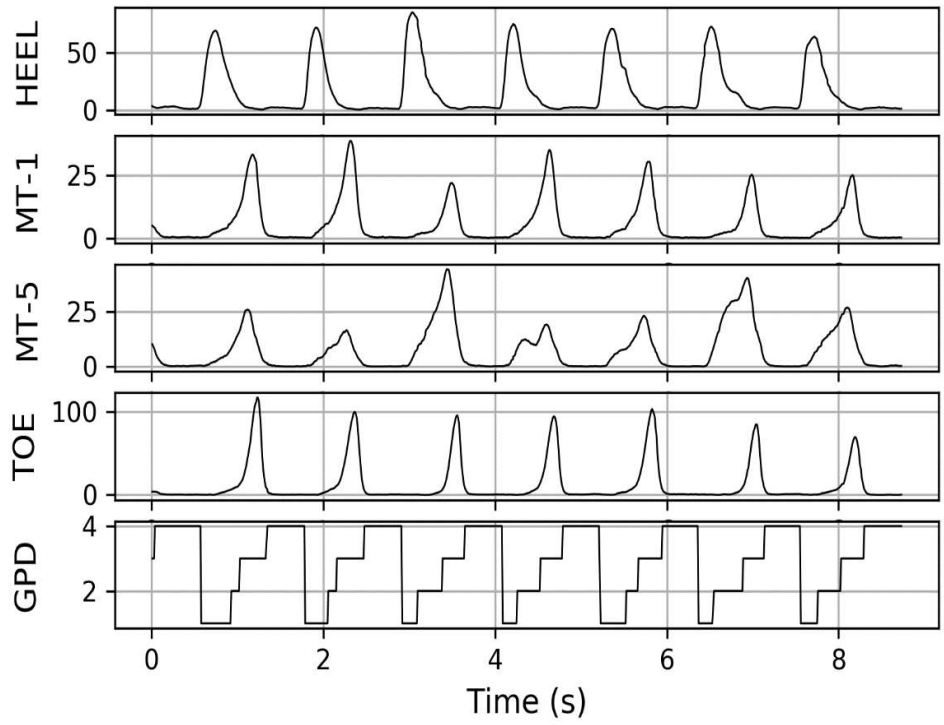


Figure 4.10 *Real-time gait phase detection using 4-FSR sensors*

Later by placing the four FSR sensors at the heel, MT-1, MT-5, and toe positions on the insole, a real-time gait phase detection technique was implemented on the Arduino code itself. The results are shown in Figure 4.10 for the right leg, where GPD values 1, 2, 3, and 4 correspond to CP, CD, PP, and SW phases, respectively. The gait phases were identified with 90% accuracy for the right leg for a 30-trials level ground walk. The algorithm took approximately 10 ms time using Arduino nano for the detection of gait phases. Gait phase detection within 10 ms is well-thought-out real-time detection for the control of the prosthetic leg.

4.3.2 Gait Phase Detection using IMU sensors

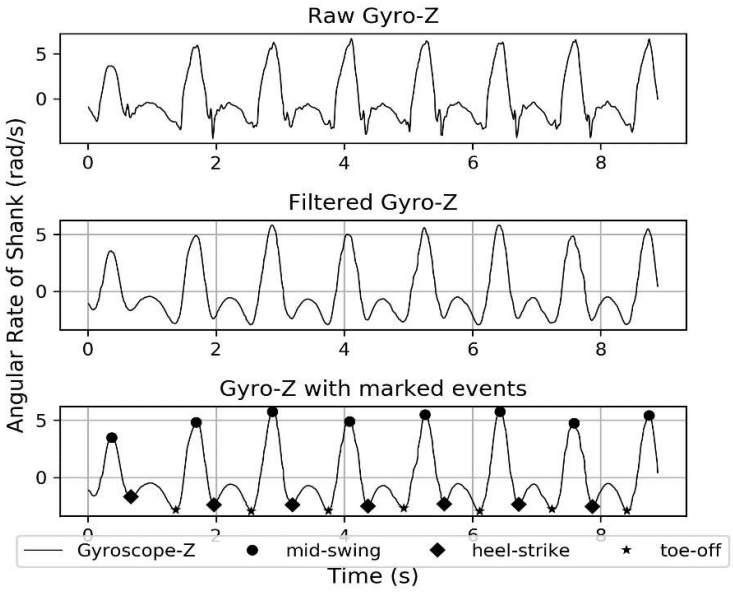
This chapter also enlightens the utilization of IMU sensors to record stance and swing phases. Table 4.2 shows the swing phase detected was 40% of the gait cycle for the level ground walk, and the stance phase was found 60% of the gait cycle. In (Negi *et al.*, 2020a), the authors used acceleration signal-based identification of gait events, where the hybrid sensors (EMG + 3-axis acceleration) were placed at TA and MGAS muscles to record EMG and acceleration. A single IMU sensor was placed at the shank's lateral side (for right leg) in the present work and measured its angular rate. Figure 4.11 shows the shank's angular rate for walking on five different terrains. The first subplots represent the raw gyroscope signal from the z-axis of the IMU sensor, the second subplots are the low pass filtered signal at 3Hz cut-off, and the last subplots show the gait events marked on the signal. All the events were detected correctly, and these findings agree with [Gouwanda *et al.*, 2015].

Table 4.2 *Gait parameters using IMU sensors*

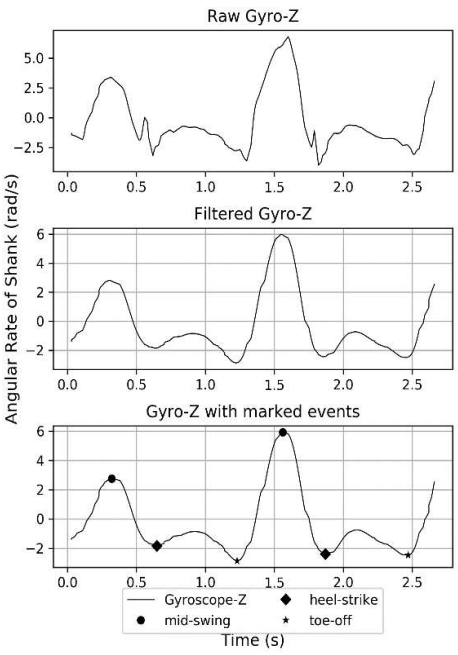
Leg	t_gait (s)	t_swing (s)	t_stance (s)
Right	1.284 (±0.035)	0.526 (±0.025)	0.758 (±0.032)
Left	1.286 (±0.029)	0.504 (±0.018)	0.782 (±0.027)

IMU sensor was also checked for real-time gait phase detection. The total time required to detect any gait event/ phase is the sum of code processing time in Python and the time taken to acquire the serial monitor data. Figure 4.12 shows the gyroscope waveform, where the notable events were obtained within an average of 10 ms. For a 30- trials data set from the right leg, all the events were identified correctly for a level ground walk, which shows that using a single IMU attached to the shank is more favorable to collect gait data in real-time. The authors also correctly identified all the events for six subjects on all the five terrains. Our findings are much better than 100 ms reported by the authors

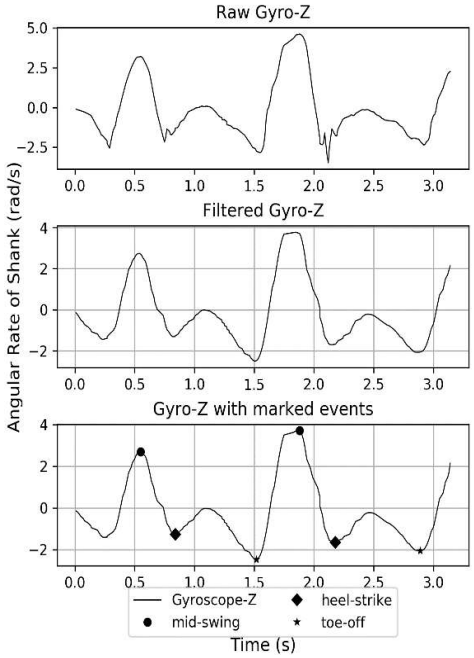
(Gouwanda *et al.*, 2015). The IMUs are advantageous over FSRs in terms of cost as a single IMU sensor can detect gait- phases in real-time, and it does not require wearing a subject-specific foot insole.



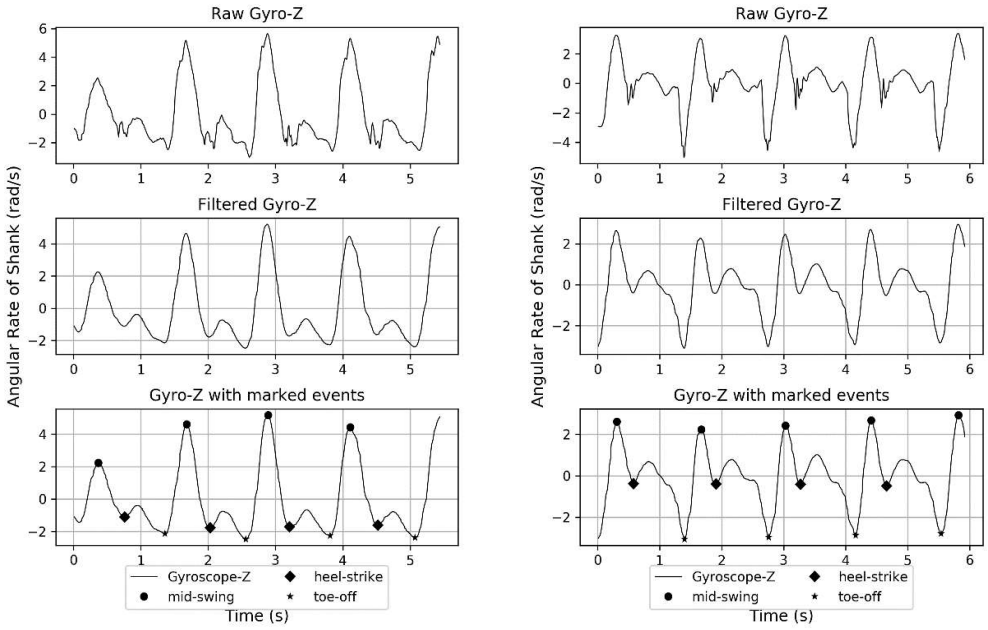
(a)



(b)



(c)



(d) (e)

Figure 4.11 *Gyroscope readings of IMU sensor attached at the shank for different gait patterns: (a)LG, (b)RD, (c)RA, (d)SD, (e)SA*

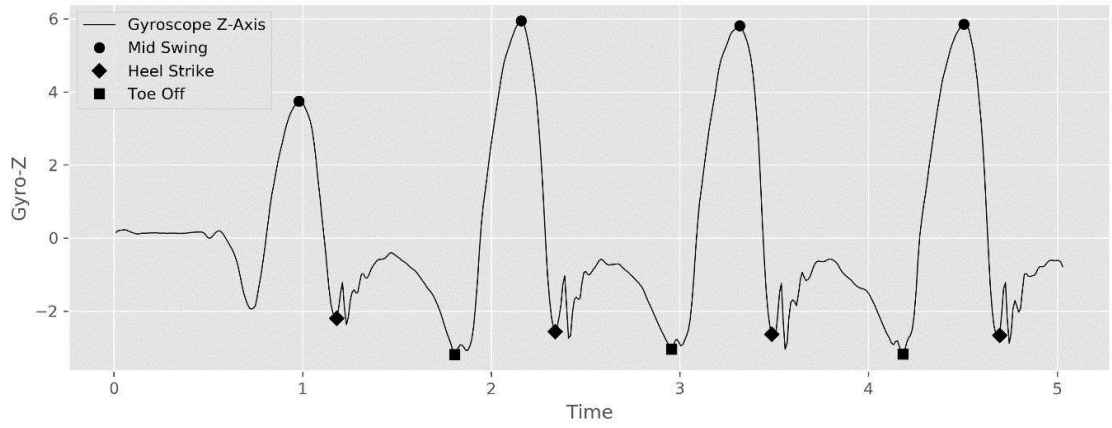
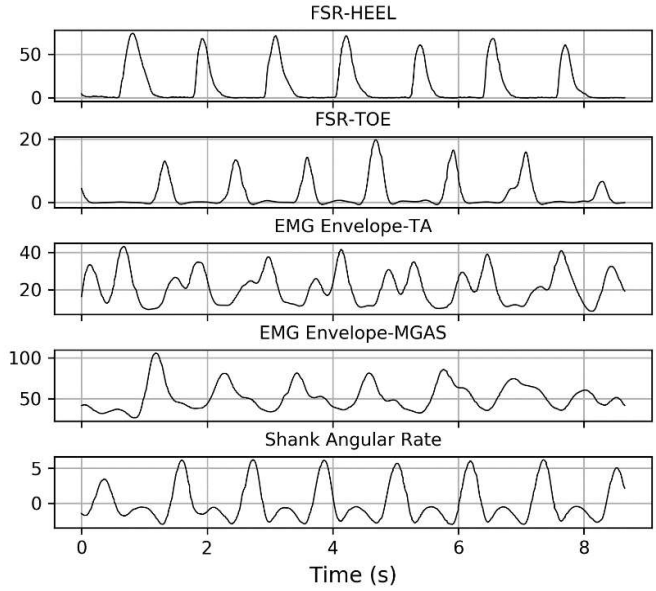


Figure 4.12 *Real-time gait phase detection using a single IMU sensor*

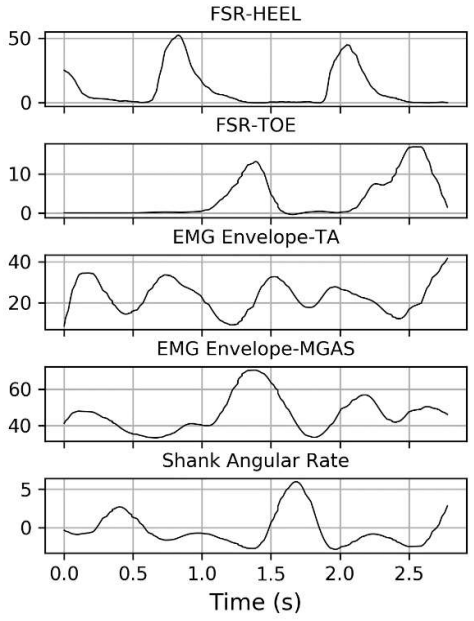
4.3.3 Synchronization of FSR, IMU, and EMG Sensors Data

The EMG sensors' response was checked simultaneously to correlate the EMG signal with the gait signal analyzed previously. Figure 4.13 shows the EMG signal envelope acquired from the TA and MGAS muscles for five terrains. The advantage of taking the

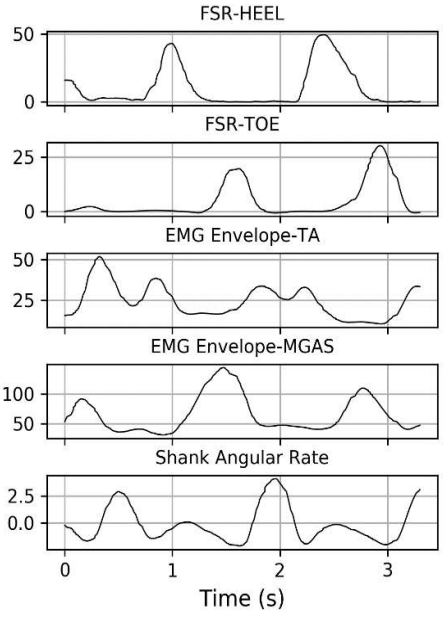
EMG signal envelope is that one can easily acquire the EMG signal using the Arduino nano controller board, whose sampling is not as good as other standard high-cost data acquisition cards.



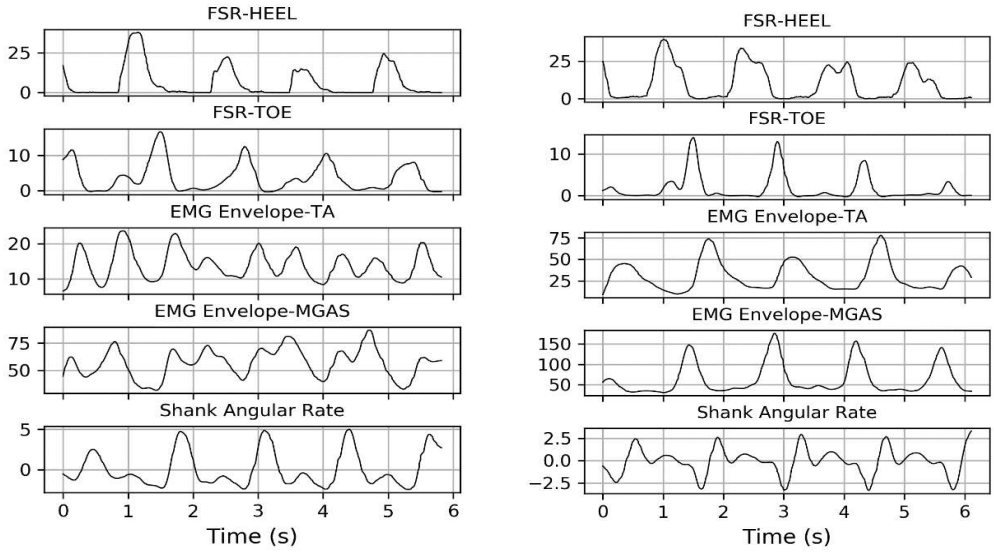
(a)



(b)



(c)



(d) (e)

Figure 4.13 *Synchronization of FSR, IMU, and EMG sensors data for different gait patterns: (a)LG, (b)RD, (c)RA, (d)SD, (e)SA*

In Figure 4.13 (a), TA muscle shows its action commencing at TO and stays all over the swing phase; but the scale is reduced during MS. TA muscle is remains active at HS until the foot completely touches the ground. The MGAS muscle is active between the initial and terminal stances. Figure 4.13 (b) and (c) show that TA muscle is more dominant for descending on the ramp surfaces, whereas MGAS muscle is more active during TO for ramp ascent. These waveforms are similar to the results (Pickle *et al.*, 2016), where the functional roles of different muscles, including TA and MGAS, are explained for various positive and negative slopes. For walking down the stairs, the MGAS muscle is in peak action after the swing phase to make proper foot landing (Figure 4.13 (d)). The TA muscle is active mainly during the initial swing phase, and remains so throughout the stance phase. These findings are in agreement with (Benedetti *et al.*, 2012). Figure 4.13 (e) represents that for stair ascent walk, the TA muscle is energetic during the swing phase, and it makes available proper foot clearance from the ground. MGAS muscle is active for

most of the stance phase; however, no activity is detected during the initial stance, as the heel is not entirely placed on the stairs during SA. All the results agree with another study done by the authors (Negi *et al.*, 2020a) using standard Delsys Trigno wireless EMG sensors. For the sensor data acquisition, a Python-based graphical user interface (GUI) has been designed to analyze and display the processed data; it is shown in Figure A.1 – A.3. The 2-D view of the designed PCB and schematic diagram is shown in Figures B.1 and B.2.

For the intent-based recognition of human locomotion, the EMG signal is beneficial. It is also helpful for the control of orthotic and prosthetic devices. The present study will aid in our further work by including the EMG sensor data acquired from the below-knee amputee's residual muscles and the FSR and IMU data collected from the artificial leg to control a power-driven ankle prosthesis. These real-time gait phase detection techniques and the real-time classifier, such as support vector machine (Negi *et al.*, 2020b) find the best scope to control the prosthetic leg on different terrains.

4.3.4 Gait Phase Detection using Fuzzy Logic Implemented in Arduino IDE

In another experiment, the different gait sub-phases are detected using a Fuzzy logic algorithm. Rules are derived using MATLAB's Neuro-Fuzzy Designer toolbox. Figures 4.14 and 4.15 show the training and testing data error for gait phase detection (GPD) using FSR and IMU sensor data. Due to the limited memory of Arduino Nano, the authors selected four sensors: heel FSR, toe FSR, thigh angle, and shank angle. Two membership functions are chosen for each input variable. Figure 4.16 shows the rule viewer for the model, which consists of 16 rules for the present study.

Once rules are obtained, these rules are written in the form of the Arduino IDE program; accordingly, the output is obtained for the given input variables. The authors observed

that the Sugeno fuzzy algorithm took on average 0.020s to predict the gait phase from a sensor data sequence in real-time. Also, for the three different trial datasets, the prediction accuracy is obtained class-wise.

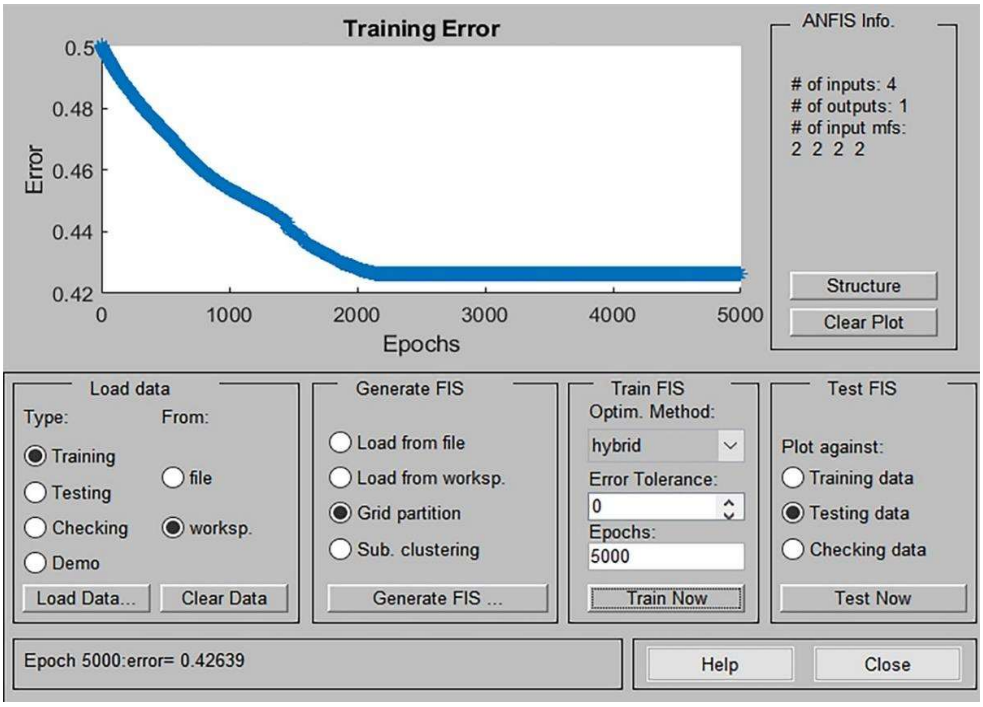


Figure 4.14 *Training data error (GPD using FSR and IMU sensor data)*

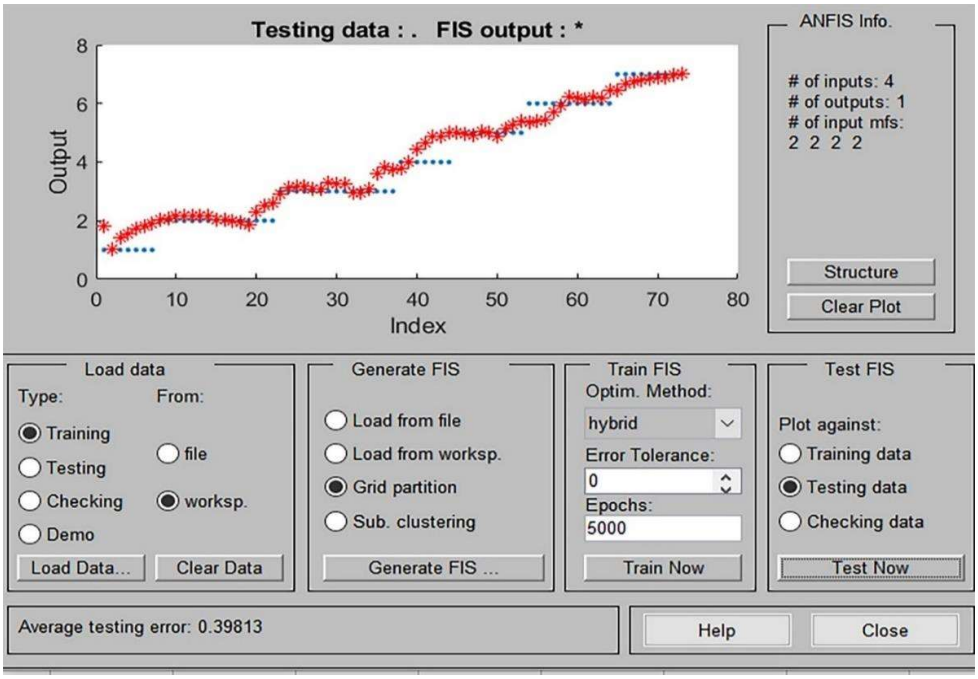


Figure 4.15 *Average Testing data error (GPD using FSR and IMU sensor data)*

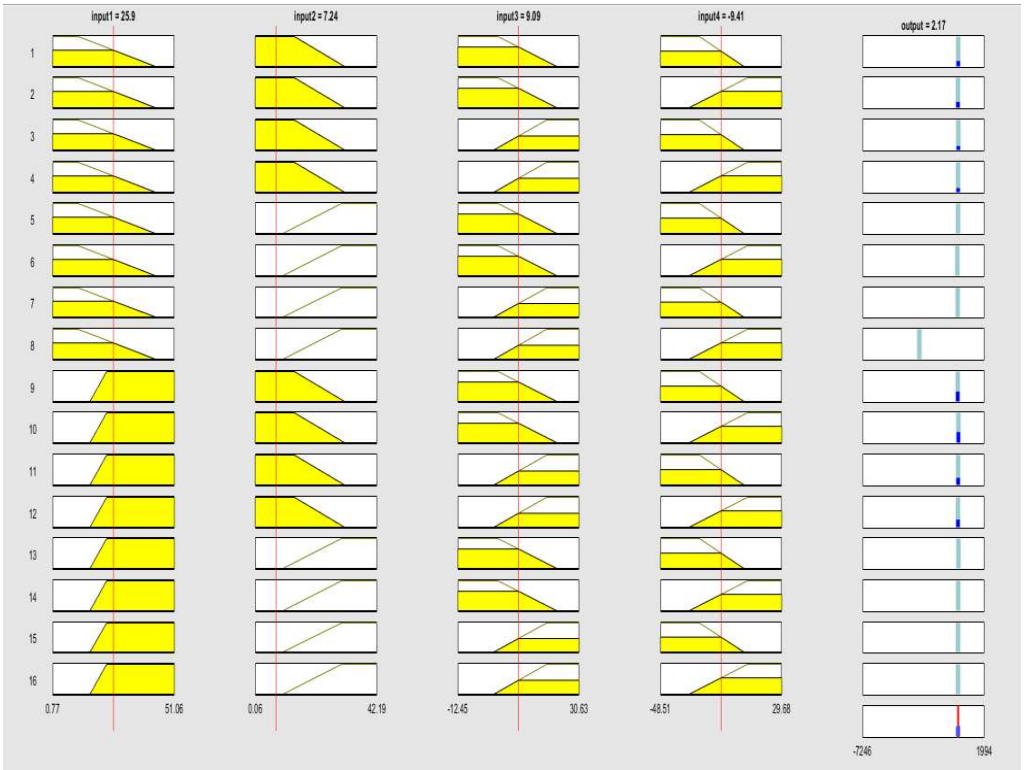


Figure 4.16 *Rule Viewer window for the trained model (FSR and IMU sensor data)*

Table 4.3 *Prediction Accuracy for Gait sub-phases*

Sub-phases	**% Accuracy on raw data (Average ± STD)**	**% Accuracy (after removing transition sample) (Average ± STD)**
LR	51.0 (± 9.2)	63.2 (± 24.2)
MSt	86.3 (± 7.8)	93.1 (± 3.5)
TSt	84.0 (± 8.2)	84.4 (± 11.0)
PSw	83.8 (± 1.6)	86.0 (± 4.9)
ISw	95.0 (± 3.3)	96.2 (± 2.6)
MSw	95.1 (± 1.8)	96.3 (± 1.2)
TSw	94.8 (± 2.7)	97.0 (± 0.9)

Table 4.3 shows the prediction accuracy for each phase. For the raw data, the overall accuracy is found at 85.83 (± 2.7)%. By removing a transition phase sample of each gait phase, the authors obtained the prediction accuracy of 89.65 (± 3)% that is good enough,

along with a 20ms prediction time to make it real-time gait phase detection. However, due to Arduino Nano's memory limitation, the authors have implemented only 16 rules for gait phase detection; rules may be increased to get better results using Arduino Mega and other controllers. Also, the prediction accuracy depends on labeling the training data; a proper method must be used while labeling the loading response phase. For all other phases, the prediction accuracy is mostly satisfactory.

4.4 Conclusion

The human gait analysis was performed using wearable and wireless sensors for five different terrains. The study was done with three wearable sensors, namely FSR, IMU, and EMG sensors. Both off-line and real-time gait events and gait phase detections were accomplished for FSR and IMU sensors. FSR and IMU sensors showed their usefulness for detecting the gait events in real-time, i.e., within ten milliseconds. The IMUs are advantageous over FSRs in terms of cost as a single IMU sensor can detect gait- phases in real-time, and it does not require wearing a subject-specific foot insole. Heuristic rules and a zero-crossing-based algorithm for the shank angular rate correctly identified all the gait events for the locomotion in all five terrains. Next, MATLAB's Neuro-Fuzzy Designer toolbox is utilized to obtain the membership functions for the input variable. Sugeno model is implemented in Arduino Nano to predict the gait phase in real-time. The overall accuracy of 85.83 (± 2.7)% is found for the unseen raw data. However, it is increased up to 89.65 (± 3)% by removing a transition sample from the dataset. The algorithm took on average 20ms to predict the gait sub-phases in real-time. This chapter explains the usefulness of the real-time gait phase-detection algorithms used in our future work to design ankle-foot prostheses for the below-knee amputees to walk on different terrains.