Study on Control System of Lower Extremity Exoskeleton Rehabilitation Robot Based on Neuromusculoskeletal Model

by

Shi Lei

Thesis

Nagasaki Institute of Applied Science

Doctor

Professor Liu Zhen

Abstract

The work presented here is a study on the interface and control system for rehabilitation robot system of lower-extremity, which integrates the human neuromusculoskeletal (NMS) model into the system. Combining with evaluation module of biological electrical signals, the NMS model can predict the operator movement intention that is used as the base for the development of control strategy. This thesis firstly presents a Lower-extremity Rehabilitation Robot System (LRRSym) in Chapter 2 for those people who have lower-extremity weakness or paralysis. The system can implement almost every therapy stages from total paralyzed status to normally move for lower limb patients. Its main working part is an exoskeleton robot called ERRobot (Exoskeletons Rehabilitation Robot). In the thesis, the core content is focused on the development of the soft Human Robot Interface (sHRI) for ERRobot after a simple introduction for its hard Human Robot Interface (hHRI).

We present a hybrid and hierarchical dynamic reactive control system for the ERRobot based on Finite State Machine (FSM) theory, and the model-based method is also adopted during the sHRI development. At the same time, in order to make the rehabilitation system seamlessly integrated with operator at the neuro-muscular level, and perform an intuitive control like a natural extension of the human body, the NMS biological model is adopted in the interface. For the selection of biological electrical signals, the surface Electromyography (sEMG) signals collected from the selected muscles are utilized as the main means of information transportation between the human operator and the ERRobot (in Chapter 3). Aiming to build the model of muscle-skeleton model, we first construct the muscle dynamic model (in Chapter 4), which is adopted by all selected joint muscles in the work. These muscles are decided by an experiment. Next, we add all these selected muscles to the skeletal model (in Chapter 5), which is built as a Linked Segment Digital Human Model (LS-DHM for short) of 19 segments and 49 DOFs (43 joint DOFs and 6 root DOFs) in the work. In addition, to make the interface automatically making decision and completing the therapy process, a novel quantitative assessment method (in Chapter 6) is also proposed, in which two new concepts, "Health Belt" and "Health Evaluation Index (HAI)" are introduced in the work. Based on the modules developed in previous chapters, the whole sHRI design is integrated in the Chapter 6. Several main modules are included in upper control system: (1) Module of Motion Intent Parse (MIP); (2) Rehabilitation Assessment Module (RAM) and Rehabilitation Program Planning (RPP); (3) Motion Planning Module (MPM).

Acknowledgments

I would like to express sincere appreciation to my advisor, Professor Liu Zhen. I would like to thank him for his invaluable support and continuous guidance. This thesis would never be done without his guidance, advice and encouragement. I was very fortunate to be his student. Liu encouraged me to set the highest standards and have ambitious goals in my research, and he enabled me to achieve my goals. His insights gave me lots of inspirations in my project. In addition to the study, Liu also gave me a lot of help in life. I would also like to express my gratitude for master tutor, professor Wang Qiang who is working in Xi-Hua university located in china, for his guidance and help when I started to step into this study field. In addition, without his help, I could not have this study opportunity in japan.

I thank professors of review committee for serving on my thesis committee and offering me their helpful suggestions and comments on how to improve my dissertation, which guided and challenged my thinking, substantially improving this work.

Least but not last, I shall express, from the bottom of my heart, my deepest gratitude, appreciation, and thanks to my parents and younger brother. Thanks for their belief in me all the time, for their understanding and encouragement from far away. Without their boundless love and encouragement, I would never have come to this point in my career. I would like to give my very special thanks to my wife, Wu Lijun, for her invaluable support, continuous encouragement and patience throughout these years. In particular, I want to thank her for giving birth to a lovely daughter for me.

Finally, I would also like to express my gratitude for those people and institutions for providing support during writing the dissertation, such as (1) Nagasaki University Hospital which provided device and place to let us record sEMG signal initially; (2) Changchun university which provide 3D motion capture system.

Contents

List of Figures

List of Tables

List of Acronyms/Abbreviations

CNS: Central Nervous System

COM: Center of Mass

CoP: Center of Press

CAD: Computer Assisted Design

CPG: Central Pattern Generator

CGA: Clinical Gait Analysis

CRF: Child Reference Frame

DOF: Degree of Freedom

DG: dynamic ground

ERRobot: Exoskeletons Rehabilitation Robot

FSM: Finite State Machine

GUI: Graphical User Interface

hHRI: hard Human Robot Interface

HMI: Human Machine Interface

HRI: Human Robot Interface

HAI: Health Evaluation Index

IMUs: Inertia Measurement Units

ICC: Intraclass Correlation Coefficient

LRF: Local Reference Frame

LRRSym: Lower-extremity Rehabilitation Robot System

LS-DHM: Linked Segment Digital Human Model

LS-SVM: Least Squares Support Vector Machines

MBD: Model-Based Design

MU: Motor Unit

MRF: Mom Reference Frame

MUAP: Motor Unit Action Potential

MIP: Module of Motion Intent Parse

MTC: Muscle Tendon Complex

MISO: Multiple inputs single output

MPM: Motion Planning Module

MVIC: Maximal Voluntary Isometric Contraction

NMS: neuromusculoskeletal

PT: physiotherapist

RF: Reference Frame

RA: Rehabilitation Assessment

RAS: Rehabilitation Assessment System

RAM: Rehabilitation Assessment Module

RPP: Rehabilitation Program Planning

ROM: Range of Motion

RAM: Rehabilitation Assessment Module

sEMG: surface Electromyography

sHRI: soft Human Robot Interface

SG: static ground

SVM: Support Vector Machines

SCI: Spinal Cord Injured

SJTRobot: Single-Joint Training Robot

3D-Three Dimension

TE(s): Therapeutic Exercise(s)

ZMP: Zero Moment Point

WRF: World Reference Frame

Chapter 1.　Introduction

In the chapter, we will present the study background and main content of this thesis. Through the introduction of this chapter, we can have a preliminary understanding for the research content and the organizational structure of the thesis.

1.1 Background

Rehabilitation is to restore the abilities of performing daily life such as eating and walking. Its task is to reintegrate them into social life so as to improve the patients' quality of life[1]. The restoring degree of legged mobility for individuals with paraplegia can obviously promoting quality of life. According to surveys for patients, the most prevalent several functions wanting to restore are the walk and the stand (64% and 25%, respectively). Therefore, the work will focus on the function rehabilitation of lower-extremity. The number of the patients with the need of functional rehabilitation training of lower-extremity is increasing year by year due to the following two reasons: (1) arrival of the aging society. The Figure 1.1(a) shows that the population of elderly people over the age of 65 in Japan has reached 26% in 2014 and it is estimated to reach 39.9% by 2060. In addition, it isn't only limited in Japan from now on, many other countries are being such situation, as shown in Figure 1.1(b); (2) skeletal impairment, and/or as the sequelae of many neuromuscular disorder diseases such as stroke, spinal cord injuries (SCI). Stroke is the third leading cause of death in Japan, after cancer and heart disease.
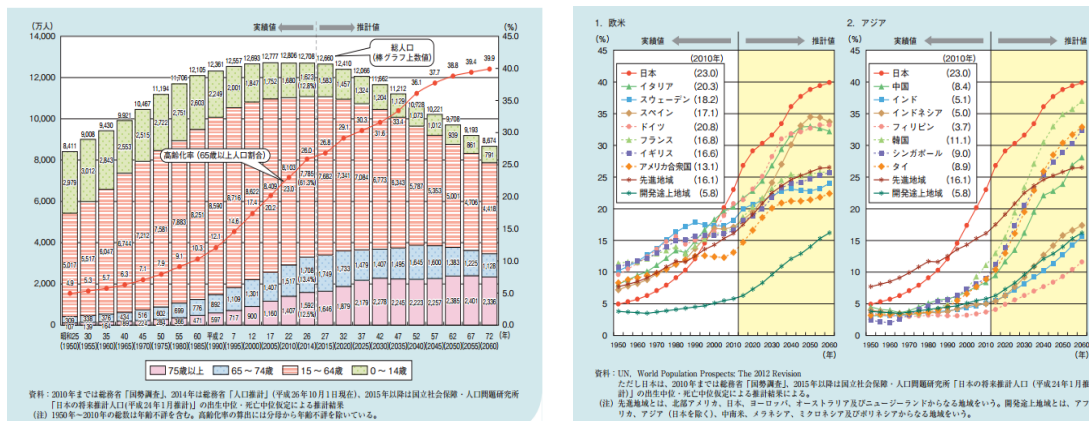
(a) Japan　　　　　　　　　　(b) Europe, America and Asia[2]

Figure 1.1 A change in aging and the future's estimation

With the increasing of the patients with physical movement disorder, the traditional rehabilitation has been unable to satisfy the needs, because it is high therapy cost, demands for occupational therapists, and mainly relies on therapist's one-by-one rehabilitation therapy, etc. And furthermore, the shortage of occupational therapists has also become a profound social problem. Therefore, the

robot of lower-extremity rehabilitation is introduced, in attempt to mitigate the aforementioned problems and cater to the needs of the society. The following presents several main advantages by adopting the system compared with traditional rehabilitation method:

- Some relevant experimental data can be simultaneously recorded[3], which can be used in many other fields by simple analysis, for example, (1) the sports-medicine research such as the study on the metabolic cost of locomotion [4]; (2) the clinical medicine such as the objective quantitative assessment[5, 6], developing muscle rehabilitation training program by studying on which exercises are most useful in activating specific muscles;

- It can be used to offer assistance to patients during rehabilitation training by guiding motions on correct trajectories, or giving force support to be able to perform certain motions. This could intensify the training and lead to better results;

- A person with movement disabilities needs generally to undergo periods of therapeutic exercise sessions spread over a long period of time. At this time, the robot can be used to assist therapists performing repetitive exercises replacing with PT, which can reduce the labor intensity of PT ,time, and cost of the whole rehabilitation process;

Besides of that ,many documents confirm that the robot rehabilitation training is able to improve the patients' mobility[7]. The "Lokomat" exoskeleton is an example of the early gait trainer[8, 9], and evidence based data shows that it can improve gait symmetry, walking ability, increases muscle strength and so on compared to conventional physical therapy in stroke patients [10, 11].

1.2 Challenges and State-of-the-art of Lower-extremity Rehabilitation Robot

There have been a number of attempts to develop automated and semi-automated robotic systems described in literature for the motor function rehabilitation of lower extremities in recent years. The section reviews these developed devices by classifying them into two groups: single joint assist training robot and multi-joint assist training robot. Among of them, the "exoskeleton" robots make up a large part, so we initially give a brief review for the "exoskeleton" robot. We focus on these aspects of movement capability (mechanical specifications) and control technologies.

1.2.1   Exoskeleton Robot

In this review, an exoskeleton is defined as an active mechanical device that can be worn around human lower limb by an operator to allow direct transfer of mechanical power and information signals. In general, the term "exoskeleton" is used to describe a device that augments the performance of an able-bodied wearer. In this application, the operator provides control signals

for the exoskeleton, while the exoskeleton actuators provide most of the power necessary for performing the task. The human becomes a part of the system and applies a scaled-down force compared with the load carried by the exoskeleton[12]. Occasionally, however, it also used to describe certain rehabilitation and functional compensation devices[13], for example, to increase the ambulatory ability of a person suffering from a leg pathology. As a kind of wearable robot[14], the distinctive, specific and singular aspect of exoskeleton is its kinematic chain maps on to the human limb anatomy. We will focus our review on exoskeleton devices for the lower limbs, and only cover devices that operate in parallel with the human legs, as opposed to devices such as the Spring Walker[15] that operate in series with the wearer[16]. From 1960 to 1971, general Electric Co. developed and built a prototype system called the Hardyman[17, 18], which is almost in parallel with a number of research groups in the United States[16]. However, the former was primarily focused on developing technologies to augment the abilities of able-bodied humans, often for military purposes, while the latter was intent on developing assistive technologies for physically challenged persons. Comparing with the wheeled vehicles, exoskeleton-based assistive devices have many advantages.

1.2.2 Single-Joint Training Robot (SJTRobot)

SJTRobot is mainly to complete single joint function recovery, such as the hip joint, knee joint or ankle joint. The main objective of recovery training is to increase tension-generating capability, flexibility and endurance of muscles. According to the rehabilitation objectives, it can do recovery training for any one or two lower limb joints and it can be performed when patients sit or stand. Fleischer and etc.[19] designed a knee joint exoskeleton controlled through the wearer's intention estimation. A musculoskeletal knee joint model is introduced to improve the estimation of the wearer's torque. Ferris[20]and his group have designed a pneumatically powered, myoelectrically controlled ankle-foot orthosis as a tool for rehabilitation. Of course, there exist many other devices such as HME(Human Muscle Enhancer)[21], NeXOS[22], Multi-Iso[23], RoboKnee[24] and so on. They can be classified and analysis from varies perspectives: 1) from the view of driven type. These drive types like electric, pneumatic, hydraulic can be applied. For example, the HME is a pneumatically actuated system and Multi-Iso adopts the brushless motor driver; 2) from the type of implemented rehabilitation exercise. NeXOS and Multi-Iso are able to perform active assistive, passive and resistive exercises using pre-training visual position information; 3) for the type and the number of rehabilitation joint. They can do recovery training for one joint (hip joint, or knee joint or ankle joint). For example, Multi-Iso is used for knee limb; 4) from the control method. Traditional (such as PID control) and modern control technology (such as fuzzy control technology) or their combination are adopted. For example, Multi-Iso uses classical force,

position and speed control methods developed with fuzzy control techniques, and the switching control is also adopted. In the following, we give a more detailed descriptions for two devices of this type rehabilitation device.

Ferris and etc.[25] proposed a powered ankle foot orthosis of single degree of freedom (DOF), as shown in Figure 1.2(a). The powered orthosis aims at rehabilitation of patients with neurological injuries. Artificial pneumatic muscles are attached to the orthoses to provide flexion and extension torque at individual joints in sagittal plane for gait rehabilitation. The control schema is based on the assisting torque estimation by using a proportional controller as a function of the sEMG activation level. It is made from a combination of carbon fiber and polypropylene. Its detailed description can be found in references[20, 26].

(a) An ankle-foot orthosis with an artificial pneumatic plantar flexor muscle

(b) Exercise movement capacity of RM (hip flexion-extension)
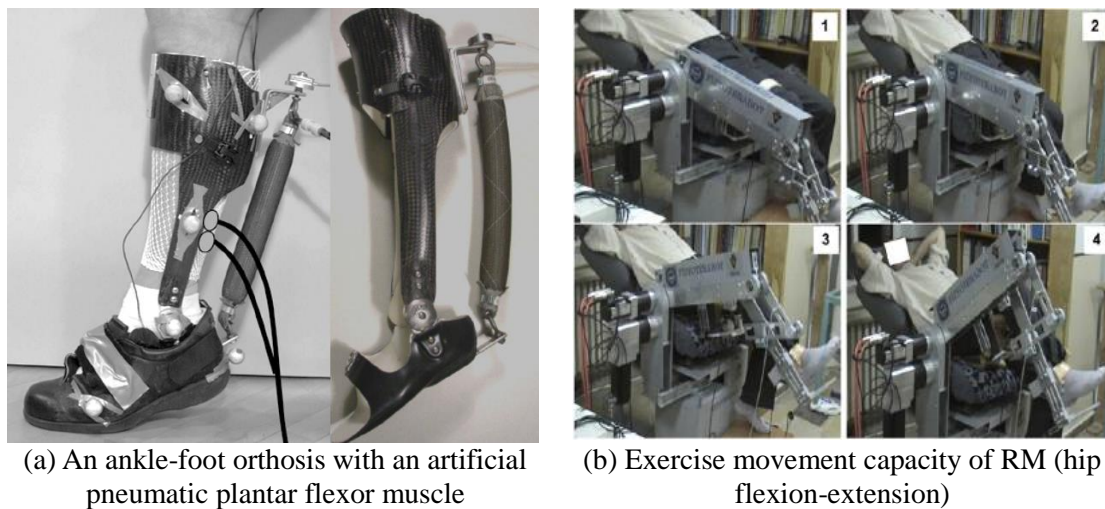
Figure 1.2 Single-Joint Training Robot

RM system(Robot Manipulator)[27] shown in Figure 1.2(b) is used for the rehabilitation of hip-knee joint rehabilitation. A parallelogram structure is used for Link 2 that allows placing the actuator. The system has two force/torque sensors to detect patients' voluntary and involuntary reaction forces during the exercise, and they are positioned at the calf and thigh locations. Position data are obtained via encoder emulation that is provided by servo-drivers. All the motors are placed on the base and are immobile so that the effect of motors' weight on the dynamics of the RM is eliminated.

1.2.3   Multi-Joint Training Robot (MJTRobot)

MJTRobot can simultaneously perform rehabilitation training for multiple joints, including muscles function recovery training and coordinated action training among multi-joint muscles. To produce normal motion such as walking, muscles corresponding joint motion must function together in a well-coordinated and efficient manner. To date, many devices of this kind of

rehabilitation have been introduced. We give a review on MJTRobot developed so far. We classify them into two kinds in the section, fixation and portability, according to whether or not they are lightweight and can move over-ground.

● **Fixation Type:**

This group of equipment can also be divided into two subclasses based on working and connecting pattern with users, that is, serial and parallel work devices. The action of the type of serial work device acts in series with the action of user's own legs to enhance their capability, such as NEUROBike[28] that employs the brushless servomotors, and LokoHelp[29] (LokoHelp Group, Germany) shown in Figure 1.3(a)[30] that is a treadmill gait trainer system and helps to move the patients' foot trajectory with a fixed step length of 400 mm, the electromechanical "Gait Trainer GT I", and HapticWalker[31], which is based on the principle of programmable footplates. For this kind of devices, it is mounted to the user's legs or feet when working, and engages the ground instead of the user's own feet with a varying length[15]. The review lays stress on the other type of assisting training device, parallel work device, which acts in parallel with users' legs to enhance their capability.

(a) LokoHelp   (b)Latest version of the LOKOMAT

Figure 1.3 Fixation type robots in series/ parallel with user's legs

To my knowledge, devices in parallel with users' legs are mainly exoskeleton robot, and treadmill gait trainer is a main part among of them. The rehabilitation robot LOKOMAT shown in Figure 1.3(b)[32] (Hocoma AG, Volketswill Switzerland) that has been developed at the Spinal Cord Injury Center of the University Hospital Balgrist Zurich, is a representative example. LOKOMAT comprises two actuated leg orthosis which are attached to the patient's legs. Each orthosis has one drive in the hip joint and one drive in the knee joint, and each joint is equipped with redundant position sensors, a force sensor to measure the joint angles and the joint torques, respectively. The direct current (DC) motor, with helical gears, is used for the actuation power of the system to

precisely control the trajectory of the hip and knee joints. Considerable control algorithms have been implemented in this system to improve its performance, such as position, adaptability, impedance controllers, etc.[30]. Besides the described LOKOMAT above, many other research groups develop many this kind of rehabilitation devices. Here gives some simple introductions for some of them.

The ALEX(Active Leg Exoskeleton) is presented in reference[33], which adopts linear actuators to actuate the hip joint thigh and knee joint shank. It integrated the AAN rehabilitation strategy into the orthosis system. Compared to other existing robotic training methods, this strategy allows the patients to actively contribute during the retraining process of gait locomotion; The LOPES[34, 35] employs the Bowden-cable driven series of elastic actuators (SEA), with the servomotors as the actuation system, to implement low weight (pure) force sources at both the posterior and anterior sides of the leg orthosis. It implements impedance control (as opposed to admittance control), which is based on a combination of position sensing with force actuation to operate the lower-limb leg orthosis. The training effect of this orthosis is enhanced by emphasizing the implementation of an assist as needed by AAN control algorithm; The ReoAmbulator robotic system (Motorika Ltd., Mount Laurel, NJ, USA), which is also known as "AutoAmbulator", is another example of existing treadmill gait trainers for lower-limb rehabilitation therapy[36]. It enables patients to contribute during the gait motion, but also provides the remaining force necessary for walking.

- **Portability Type:**

This type of rehabilitation device is over-ground gait trainers. In an effort to facilitate legged locomotion in individuals with paraplegia, many computer-controlled lower limb orthosis systems have been, and are being developed and described in the research literature, such as HAL(Hybrid Assistive Limb shown in Figure 1.4(a))[37] from University of Tsukuba, BLEEX (Berkeley Lower Extremity Exoskeleton) [38] ,EXPOS from Sogang University[39, 40] , Vanderbilt exoskeleton[41], REX Bionics[42], MINDWALKER exoskeleton[43] shown in Figure 1.4(b), Ekso[44] that is relatively lightweight (20~25kg) but rely on the help of crutches to maintain balance and compensate the power insufficiency in the exoskeleton joints. Motion Maker[45] that is a commercial lower limb rehabilitation device developed by Swortech SA, etc. Most of them can perform active assistive, passive and resistive exercises using pre-training visual position information, and can generally accomplish the co-coordinate joint training for specific function, such as walking, running, etc. For control method, traditional (such as PID control) and modern control technology (such as fuzzy control technology) or their combination are adopted. In this section, we will give a review for HAL as an example.

(a) HAL-5   (b) MINDWALKER
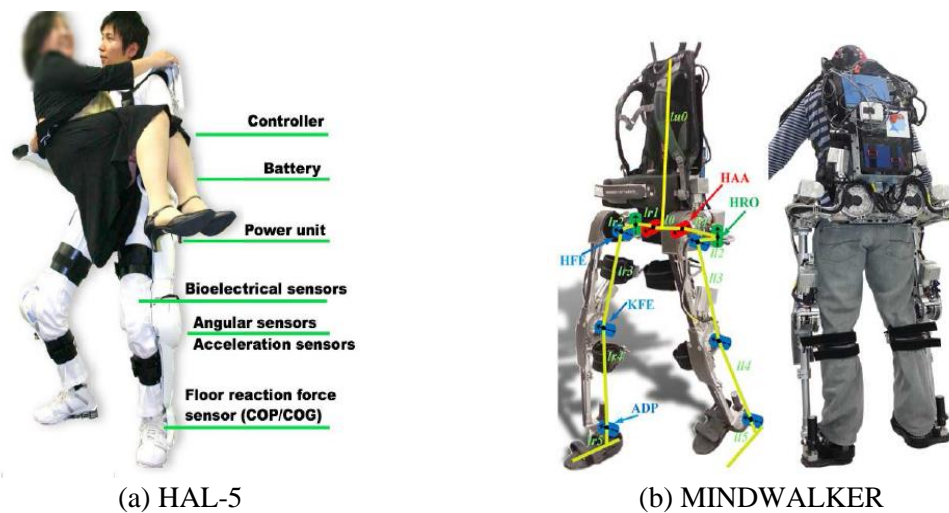
Figure 1.4 Robot of portability type

HAL-5 shown in Figure 1.4(a)[37] that is a full body version is battery powered and driven by direct current motors which receive control inputs from a hierarchical controller. The controller integrates the bioelectric information from sEMG sensors and kinematic inputs from encoders at the joints. The project focuses on supporting elderly and gait disordered people. During the course of the project from HAL-1 to HAL-5, different prototypes of the mechanical construction exist, and different control strategies have been developed with the primary interface being sEMG signals from the operator's muscles. In the first published control strategy for HAL-1, the EMG signal is picked up from the rectus femoris to initiate the movement. After reading an increased muscle activity, the EMG signal is evaluated for 300ms and the numerically integrated post-processed EMG signal is converted into a hip angle by a linear relationship that was calibrated before. This angle is used as the reference for the position controller of the system. After the 300ms interval the state machine can decide the transition based on the angle measurements, and predefined trajectories are used for the rest of the movement[46]. An improved version of this control structure with an extended state machine incorporating standing up, sitting down and walking can be found in reference [47]. But this control scheme lacks flexibility and was replaced with newer versions of HAL[48]. Four muscles responsible for flexion and extension in the hip and knee joints have been recorded, and support torques for the actuation in those joints have been computed in linear relation to the post-processed EMG signal. And besides, traditional control laws, such as PD control also used in the HAL to perform gravity compensation and balance control[49].

1.3 Scope and Summary of the Research

In all of existing lower-extremity rehabilitation systems, there is no one system that can fully

automatically complete the overall rehabilitation process. Therefore, the work will propose an integrated solution for the realization of this goal. The developed rehabilitation system for lower-extremity patients can not only automatically complete various tasks' rehabilitation such as functions of walking and climbing up the stairs, but also adapt to the patients with different impairment levels of lower-extremity. After the system scheme is put forward, the work focuses on the part that contacts directly with users, and a wearable Exoskeleton Rehabilitation Robot (ERRobot, for short) is designed to use as the contacting part. And furthermore, we pay my attention on the construction of an intelligent control system for the ERRobot. Through an overview of control technologies applying in exoskeleton robot, we present the summary of the research.

Existing exoskeleton rehabilitation robot studies have mostly focused on position and force control, and they passes through three stages of development in the past few decades. The first stage is to perform the robot control by providing position command from the operator, for example, the control of Hardyman man-amplifying exoskeleton[12].When getting into the second stage, a control interface of human machine interactive is introduced by force sensor ,which can be used to detect the direct interaction force between the human and the machine in a dynamics environment, and use as a main command signal to the exoskeleton. The control system of several developed experimental prototypes such as[50, 51] are built based on the force control . Of course, there are some robots integrate the two control methods into same system, for example, LOKOMAT adopts the control strategy ,which switches control methods between the position and force control systems based on the stages of gait[32]. A common feature of the control methods in both the first and the second stage is that the operator must apply an action, either kinematic-position command (the first stage) or dynamic-contact force command (the second generation) in order to trigger the exoskeleton response. Obviously, this will lead to time delay in the both systems. The leading idea of current research is that HMI (Human Machine Interface) is constructed at higher levels of the human physiological system hierarchy, which allows us to overcome some delay. Many approaches with different physical quantities can be used including electromyography (EMG), electroencephalography (EEG)[52].

Nowadays, many researchers adopt the sEMG signal to control some devices by deriving movement commands [53]or force commands. It is known that the sEMG is produced before the muscle attraction, that is, there is a time delay from signal emission to muscle force production. This time, called the electromechanical delay, is reported to be about 50–80ms[54, 55], mainly due to low muscle fiber conduction velocity and the chemical processes which lead to contraction. It allows the signal evaluation process to start before the force production begins, reducing the

latency of control systems if effects of muscle fatigue are not taken into account and the time between the emission and detection of the sEMG signal is neglected. As a result, the reaction time of the human/machine system should decrease, resulting in a more natural control of the task. We call this control method as the third development stage. For some time, researchers have attempted to use sEMG signals measured from the limb as control commands. However, most of these systems only provide discrete or binary levels of motion control whereas daily activities require a continuous limb movement control. For the development of the control algorithm that uses sEMG signal, the difficulty of using it as continuous control commands is the nonlinear and the nonstationary characteristics. The methods that is developed in the past to extract the desired information from sEMG control variables is mainly black-box methods, such as fuzzy-neuro[56], artificial neural network(ANN)[57], pattern recognition, proportional control [58, 59]. Proportional myoelectric controllers use the subject's muscle activation to control the magnitude of joint torque for the powered device, which may be beneficial in lower-limb control [60]. There are also reference such as [18] shows that the impedance control method is more convenient for the development of rehabilitation systems. In line with the newest control strategy, we introduce the sEMG signal into the development of our control system. In the development of the control system, the contact force sensors are still adopted between the exoskeleton and the human, however, it generates only feedback signals to correct the kinematics and the dynamics control of the exoskeleton. Moreover, we mainly use the sEMG to estimate the created muscle force moment with respect to a joint. We do not directly produce the muscle force by processed sEMG signal like many references[61],rather than combine a muscle model(i.e., Hill-type Muscle Model) to predict the muscle force. That's because when we operate an exoskeleton, the assumptions of non-isometric and non-isotonic do not hold, which are valid assumption for controlling a prosthetic devise using sEMG signals alone. Therefore, the muscle's force cannot be estimated solely on the sEMG signals. The angle of the human limb joint, coupled with the exoskeleton joint, is constantly changing during the exoskeleton operation, and as a result the muscles attached to that joint are changing their length and end points velocities. So, a muscle model has to take into account the muscle's length and velocity in addition to the sEMG signal that defines the muscle activation level for predicting the force.

In the work, we proposed a hybrid and hierarchical dynamic reactive control architecture for the developed exoskeleton rehabilitation robot system. During the developing of control algorithm, the neuromusculoskeletal model is adopted, which is helpful in developing an intuitive interactive interface between exoskeleton and operator. Compared to the traditional control by way of an external device (for example, a keypad or a wheel and so on) that has to be manipulated, intuitive

interface can reduce operator's mental load, that is, the operator can focus on fulfilling a task with the exoskeleton rather than focus on mere control of the device[46].

1.4 Organization of the Thesis

The remainder of the thesis is organized as follows. Chapter 2 gives the presentation of the overall structure of rehabilitation system, the pathogenesis of lower extremity function loss and the rehabilitation mechanism using robot. It is followed by the Chapter 3 that presents the sampling and processing of the sEMG signals, which is adopted to assess users' motion intention in the system. Chapter 4 is devoted to the math modeling of the adopted muscle, and Chapter 5 presents the biomechanical model of human body and its match model development to perform motion analysis. In Chapter 6, an intelligent control system based on the NMS model for ERRobot (Exoskeletons Rehabilitation Robot) system is proposed. Next, Chapter 7 presents our conclusions, including our contributions and recommendations for future work. This is followed by a bibliography of the references used in this study.

Chapter 2.   Physiological Foundation and Overall Framework of Lower-extremity Rehabilitation Robot System

In this chapter, we will give a comprehensive description of the LRRSym (Lower-extremity Rehabilitation Robot System) designed in our project for lower-extremity patients following the neuromuscular disorder diseases such as stroke, SCI. As the core content of the thesis, we mainly introduce the design of ERRobot system. Because the physiological theory of lower extremity rehabilitation is the foundation of the rehabilitation training and system design, the section starts from the causes to lead function loss or weakness of lower extremity, and then introduce how to make the rehabilitation program aiming to improve mobility ability.

2.1 Physiological Foundation Theory of Lower-extremity Rehabilitation

To effectively make training program, we first present the causes to lead function loss or weakness of lower-extremity, next give a presentation related with rehabilitation therapy and rehabilitation system design, including physiological rehabilitation principle, rehabilitation assessment, making rehabilitation program and so on.

- **Pathogenesis**

It is generally accepted that locomotion is based on the activity of neuronal circuits within the spinal cord (the Central Pattern Generator, CPG). Both the spinal CPG and peripheral nervous system that mediates afferent input to the spinal cord are under the control of the brainstem. Any damage within them can be followed by an impairment of pattern generation that leads to changes in leg muscle activation , consequently ,results in a movement disorder [30, 62]. From the view of process leading to lower limb disorders, we can know that a movement disorder is the consequence not only of the primary motor lesion but also of secondary processes that can be compensatory and can be supported during rehabilitation[62]. Moreover, some article has shown that the muscle weakness is the primary cause of disability in people with neuromuscular disorders and central nervous system injuries. The LRRSym is just designed for the latter lower-extremity patients due to a variety of neuromuscular disorder diseases, among of which, the two diseases of SCI and stroke [30, 43, 62] are mainly considered. According to many articles, the primary outcomes of the two diseases are lower-extremity paralysis or weakness.

- **Methods and Physiological Principle of Rehabilitation**

The regulation of human walking requires a close coordination of muscle activation between the two legs which seems to be achieved by a flexible neuronal coupling at a spinal level[63]. According to pathogenesis of lower-extremity function loss described above, we know we can

restore the motion function by re-building neural map to movement control, which can be completed by physiotherapy taking advantage of the plasticity of spinal nervous circuits [64, 65]. The plasticity of neural system becomes the important medical theory basis of neural function recovery and robot assist rehabilitation[66]. For the plasticity of spinal neuronal circuits, it has several implications: (1) spinal reflex plasticity: It tells us that the improvement of motor function can be completed by taking advantage of the plasticity of neuronal centers, unnecessarily directed to correct the isolated clinical signs, such as the reflex excitability; (2) use-dependent plasticity: It shows that loss of motor capacity following neural injury can become enhanced when locomotor networks are no longer used, for example following a stroke [67]; (3) task-specific plasticity[68]: The plasticity can be exploited by rehabilitative purposes using specific training approaches following a neural injury. Some studies show that task-specific exercise can maintain or improve independence.

In the work, the LRRSym is designed to provide autonomous training where patients are engaged in repeated and intense task-specific practice to improve lower extremity motor function. Besides, studies have shown that active involvement for operators in the production of a motor pattern results in greater motor learning and retention than passive movement[69-71].So in our system design, in order to make the patient actively participate in the task-specific training, sEMG is adopted to decode intent of operator.

2.2 System Functions and Preliminary Considerations

The design requirements of the LRRSym are determined in accordance with the lower limb rehabilitation theory and the design criteria of human-robot system. Several main design points for the designed rehabilitation system are listed as follows:

Safety: The rehabilitation system is a patient-objected system. Ensuring safety of the subject is a very important issue. The patient should ensure the highest priority in the implementation of the system because the human is inside the control loop. The safety and reliability are ensured by both software and hardware in the proposed system. For hardware design, it includes selecting secure devices, setting emergency stop button, and reasonable mechanism design and so on. Furthermore, the robot chooses an architecture that is almost anthropomorphic for maximum safety and minimum collisions with the environment. For software design, it includes the limitation of servomotors' torque and rotation angle through servomotor drivers, adding necessary security judging statements, etc.

Multi-functionality and Adaptability: In order to perform rehabilitation training for lower-extremity rehabilitation, the developed system LRRSym must be able to realize some functions,

for example, data measuring, monitoring and saving in real time. The system should be adapted to different patients in terms of physical characteristics and should be worn and removed as easily as possible. For example, its mechanism should be adjusted according to patient limb size.

Portability: It mainly refers to the weight of the device, especially the moving parts. The weight must be light to prevent fatigue of the user. It can help in: 1) reduction in the metabolic energy expenditure of the user while wearing the device, and 2) minimization of the power requirements for actuating the system.

2.3 Overall Conceptual Design of the Rehabilitation System

The section presents a conceptual design of rehabilitation system. It consists of four basic sub-system: a stabilizing base frame system (including driving parts, sensor parts etc.), a wearable exoskeleton rehabilitation robot sub-system (i.e., ERRobot, including driving parts, sensor parts and so on), a suspension system and control system. The system structure is shown in Figure 2.1.

In a typical training process, the body weight of the patient is supported while his lower extremities are guided by two pairs of robot legs following a gait reference. The developed system is able to complete this function by way of the suspension system (two ropes positioning on two sides of operator), which is controlled by two servo motors (MR-J2S), and are connected to the patient by an easy-to-wear jacket. The system can also prevents the patient from falling.
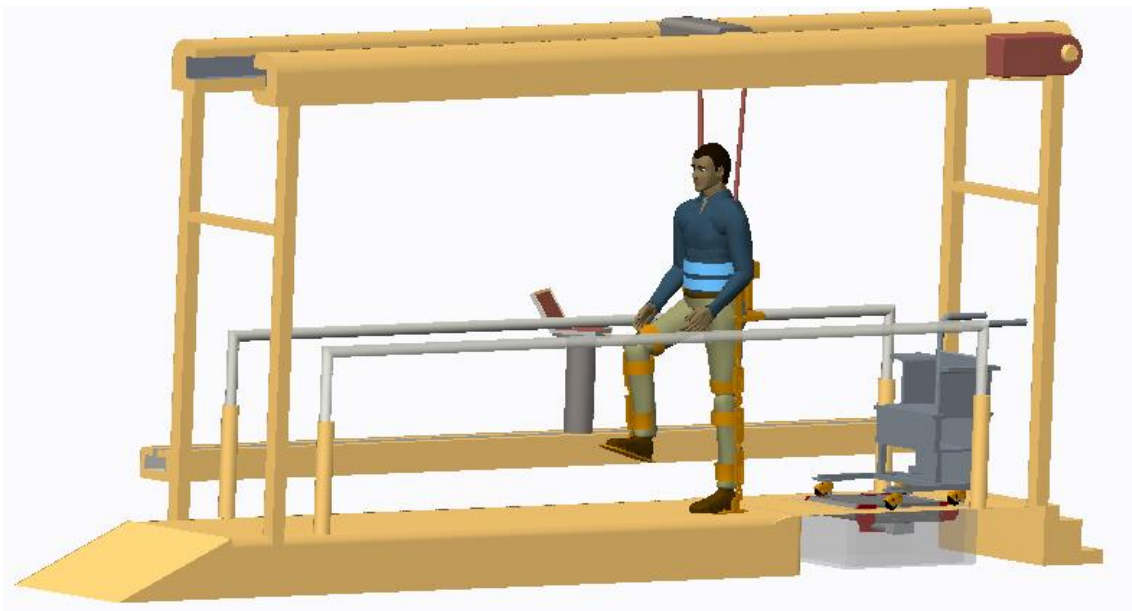
Figure 2.1 Overall structure of the LRRSym

The system can implement every stage from total paralyzed status to normally move for lower limb patients. When starting to do rehabilitation training, the ERRobot is configurable to be

coupled to people's lower limbs. Under the coordination of the total control system, LRRSym can perform rehabilitation task by sit-training and stand-training, as shown in Figure 2.1. After finishing the sit-training process, the system has the ability to transfer independently from a chair to stand up by activating the suspension system, and then the training is entered into the next training stage. The following list some characteristics of the designed system: (1) the system can provide the balance training, limb coordination training and so on; (2) it can perform sit-training and stand-training using the same system. In addition, the two training can complete the conversion by activating a cabling system mechanism; (3) the system can finish gait training by walking in situ (by mean of treadmill, just like [33, 72]) or within a distance, that is, the system is a combination of treadmill gait trainer , over-ground gait trainer and other training devices such as "KineAssist", which is a robotic over-ground gait and balance training device[73];(4) human possesses naturally developed algorithms for control of movement, but have limited muscle strength. The control system of LRRSym is just to detect user's action intention by the change of the sEMG, which may lead to seamless integration and an intuitive control of the system. That is to say, the integrated control system can benefit from the advantages offered by human and robot system by combining the strength capabilities of robotics with the navigation intelligence and adaptability of humans; (5) adjusting of lower extremity load can be finished by adjusting the rope length. Therefore, the system can perform the recovery motion exercise from weight loss to totally bearing weigh; (6) the system introduces virtual reality method to boost the enthusiasm for the patient training. As shown in Figure 2.1, a computer monitor is placed in front of the subject to provide visual feedback about his/her motion trajectory during the execution of the task.

In the whole LRRSym, the ERRobot contacts directly with users and is also the core working component, therefore, we pay my attention on the design of human-robot (ERRobot) interface (HRI) in the work. We group the HRI into two aspects, that is, hard HRI (hHRI) and soft HRI (sHRI). The hHRI design mainly contains the design of physical human-robot interface between subject and robot entity for transferring the power from robot to the subject or vice-versa, and the hardware building related to control system. For the sHRI design, it mainly refers to the software design of control system including the design of the GUI, control algorithm, rehabilitation strategy program and so on.

2.4 Core Working Part: ERRobot Design Details

The ERRobot is designed as a kind of wearable exoskeleton robot parallel to the people's lower limbs [49], which adopts an anthropomorphic architecture with similar kinematics to a human. Thus, the ERRobot consists of two identical left-right parts, and each of them has ankle, knee, and hip joints similar to human legs. To provide support and control of the lower limb, it is used

to aid the partial weight-bearing gait [6] to restore lost or weak functions of lower extremity to their natural levels. The Figure 2.2 shows its overall mechanical structure. The following paragraphs summarize the design of the ERRobot in terms of its mechanical, actuation and sensing characteristics.
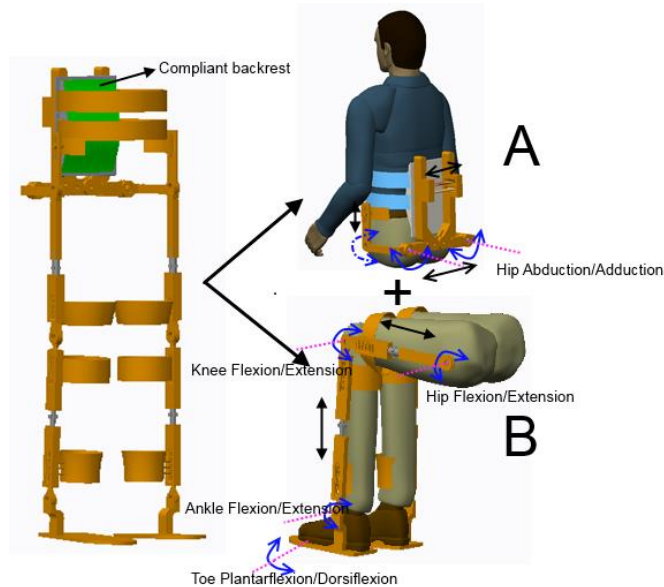
Figure 2.2 Overall mechanical structure of the ERRobot

For joints' driving methods, if every of DOFs is actuated, it will lead to unnecessarily high power consumption and control complexity. So only those joint DOFs that require substantial power when implementing specific motion should be actuated[38],that is, they are treated as active DOFs. At first step, because the LRRSym is designed for the rehabilitation of lower limbs, and gait recovery training is viewed as a primary task, the Clinical Gait Analysis (CGA) data is used to facilitate the design by providing information such as joint velocity, torque and power, for example, determining which DOF should be driven. Based on gait analyzing, flexion/extension at the ankle, knee, and hip are designed as active joint DOFs, and the other DOFs are equipped with passive impedances. For the connection part of ERRobot and lower limbs, we use elastomers to reduce the load on human muscles and increase comfort.

2.4.1 Joint Motion of Lower Extremity

The ERRobot kinematics are close to human kinematics, so the ROM of the ERRobot joint is determined by examining human joint ROM, and it is designed to be equal to or larger than human joint ROM in walking (see column 4,5 in Table 2.1), and less than the maximum ROM of operator's joint for safety (shown in Column 3 of Table 2.1) [22].These data can be found by examining CGA data and other study[74, 75]. Each joint angle is measured as the positive

counterclockwise displacement of the distal link from the proximal link, and relative angles are adopted, as shown in Figure 5.6. In order to provide a reference position from which joint angles can be expressed, a neutral posture is defined where the positions of all joint angles are equal to zero. The zero reference positions used in my work are shown in Figure 2.3. In the paper, under the assumption that the human gait is approximated by the motion on the sagittal plane, we consider only the motions in sagittal plane, as shown in Figure 2.3. As introduced above, the applied joint angles in the work are defined as relative angles, for example, the minimum angles of knee joint correspond to full knee joint extension while maximum values correspond to maximum knee joint flexion.

Table 2.1 The ROM of lower limb and ERRobot/ (degree)[76, 77]

Joint	Movement	ROM of Limb	Human Walking Maximum	ROM of ERRobot
Hip	Flexion/Extension	110/-45	32.2/-22.5	90/-30
	Abduction/Adduction	-50/30	-7.9/6.4	-30/30
	External/Internal	-20/20	-13.2/1.6	---------
Knee	Flexion/Extension	-110/0	-73.5/0	-115/0
Ankle	plantarflexion /dorsiflexion	-30/20	-20.6/4.1	-30/10
	Abduction/Adduction	-------	not available	---------

| lateral view | Front view | Vertical view |
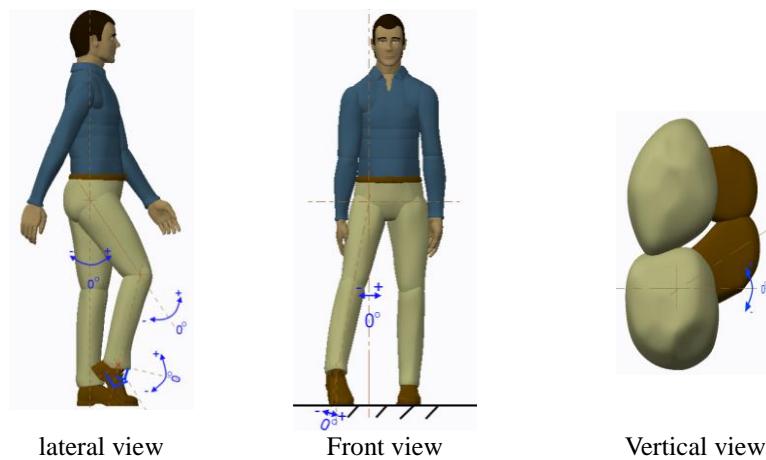
Figure 2.3 The definition of zero reference position of interested joint angles

2.4.2 Mechanical Part Design

As indicated above, the hHRI mainly refers to the design of hardware part. In the section, it is divided into two parts, i.e., mechanical and control system hardware parts, to be described. In the section, we do not give a very detailed design process, but give a description for the main function structure, components and design parameters.

- Design of Main Mechanical Structure and Components

In this work a lightweight, under-actuated exoskeleton, ERRobot, has been developed that runs in parallel to the human. The Aluminum alloy is selected as the main material for its structure so that it would be lightweight but have sufficient rigidity to support the strains. This light material: (1) is available in a variety of shapes; (2) it is readily welded and brazed, and (3) can be heat treated to desired properties, either before or after fabrication[17].

As shown in Figure 2.2, the mechanical composition can be grouped into 3 parts: a back-mounted frame attached to the upper body (Part A in Figure 2.2), right and left lower limb mechanics which are mounted on that frame mechanics (Part B in Figure 2.2). We also design some necessary attachment devices where the ERRobot can be attached to the wearer at five main locations: footplate (no showing in model), shank, thigh, pelvis, and torso. The back-mounted frame is a rigid spine that serves as a payload attachment point and an exoskeleton-to-human attachment point through a compliant backrest. The control PC, battery for electronics and some other load are located in the frame. For the core driving part of lower limb, it is actually a three links' mechanics (for single leg mechanics), including thigh part, thank part and foot part, all of which hold the subject's leg in a defined position. Based on human anatomy and joint ROM, the desired DOFs for the ERRobot is specified to allow sitting, standing, and walking. As introduced above, the ERRobot is designed to include three joints and five DOFs per leg (the joint of Hip: 2DOFs, Knee: 1DOF, Ankle: 1 DOF, and toe: 1 DOF, respectively). Among them, the DOFs of abduction/adduction at the hip, plantarflexion/dorsiflexion at ankle joint, plantarflexion/dorsiflexion at toe joint are designed as unpowered passive joint DOF whose passive impedances (created by steel springs and elastomers) are experimentally chosen, i.e., 180 Nm/rad at ankle joint, and the other DOFs are designed as active joints. It is obvious that ERRobot
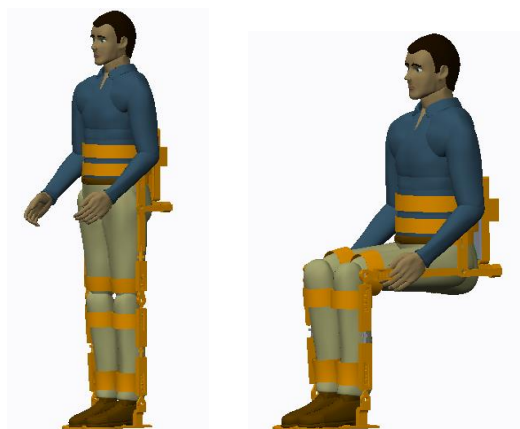
Figure 2.4 Two main training poses (standing position and sitting position)

mainly drives lower limb to move in the sagittal plane. In order to be adapted to different operators

in terms of physical characteristics, increase level of comfort and also enable a quick donning and doffing capability, the mechanism is adjustable for different body and limb lengths, and cushioned interface is adopted. In the Figure 2.2, the black double arrows represent a part of regulating mechanism (many others are not marked, for example, the gasket thickness in back is adjustable), blue arrows indicate DOFs.

The subject wearing the ERRobot is shown in Figure 2.4. It shows two important training modes based on different training stages.

- Selection of Actuators and Sensors

Specifying the actuation is a key step in the design process of the ERRobot shown in Figure 2.2. It must provide the necessary assistance if the subject is not capable of correctly performing the movement. For the considerations of actuators' selection, these include power or force density, efficiency, size and weight, and cost. In addition, since the ability of walking is the primary rehabilitation task in the thesis, many parameters can be determined by gait analysis. The Figure 2.5 shows the information of kinematics and kinetics of human gait from a normal, healthy individual (70 kg, 0.90 m leg-length) walking at 1 m/s, including joint angle, torque, and power for hip, knee, and ankle flexion/extension motions during level-ground walking. They are plotted through some modifications for the original data that can be found in reference [74].The data of y-axis is relative to body weight.
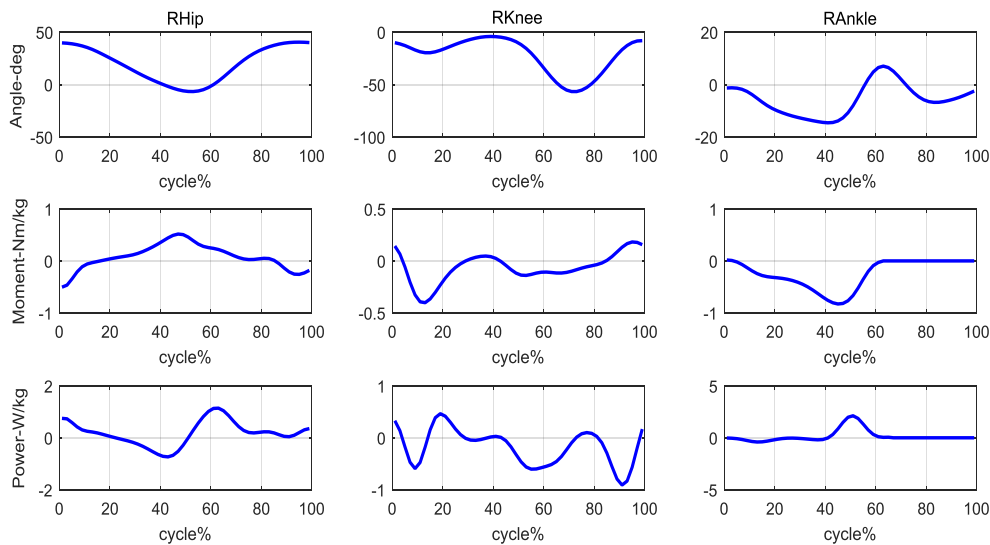
Figure 2.5 Kinematics and kinetics of human gait from a normal gait (walking speed 1m/s)

Through data processing for original data, the max torques of three joints can be obtained easily. Take a user with a weight of 80 kg for an example, these joint torque values are listed in Table 2.2

and the table also lists the corresponding positions in gait cycle, which located in the parentheses. From the table, we can see that the torque of ankle joint is maximum of three joints and the torque value increases with the increase of the walking speed for same person. For instance, the torque of ankle joint is around 70 N m during plantar flexion for push-off (around 47 % of gait cycle) for the same wearer. However, other work also present a bigger torque. According to these reference data described above, the appropriate motor and reducer can be determined. The reference [78] indicates that the permanent magnetic brushless DC motors generally have relatively higher torque density and efficiency. In the work, six brushless DC motors with the gear reducers are used in the work due to its excellent controllability , ease of supplying power, and so on compared with other actuators such as hydraulic and pneumatic actuators[79].It imposes sagittal plane torques at hip, knee and ankle joints. As seen in the figure, ERRobot contains seven segments, which are: two shank segments, two thigh segments, two foot segments and one pelvis segment. These motors are used to drive the hip, knee and ankle articulations through a speed-reduction transmission. As a safety measure, both knee joints include normally locked brakes.

Table 2.2 Max torque of three joint in the sagittal plane at three different walking velocity

Walking velocity(m/s)	Hip(Nm)	Knee(Nm)	Ankle(Nm)
0.5	38.4(2%)	17.0(14%)	62.9(45%)
1.0	41.4(47%)	32.4(12%)	66.4(45%)
1.5	49.3(48%)	48.2(12%)	70.3(47%)

The selection of sensor depends heavily on the specifics of the target application when the system is developed. According to the control requirements, encoders are adopted to measure the rotational joint angles[80] and velocities at active joints. Inertia Measurement Units (IMUs) at the segments of thigh, shank and foot are also tested to measure the acceleration, velocity and orientation of the corresponding segments in the world frame. In order to record the force and torque applied by the human, ATI Gamma force/torque sensors are used at active joint. These sensors require careful design to securely fasten them to the human. To ensure safe, each major axis (joints 1, 2 and 3) is equipped with electromagnetic brake, which is activated when power is removed from the motors.

2.4.3 Control System Building

The design of control system can also be classified into hardware and software designs, each of which belongs to hHRI and sHRI designs, respectively. The structure and the elements of the system are shown in Figure 2.6. According to this figure, the system is composed of PT, patient, rehabilitation robot and cybernetic interface. The cybernetic interface (i.e., Soft/Hard Human-Robot Interface) is the main control center of the system. It evaluates the patient's status and

analyzes the feedback information coming from the Human-ERRobot system. Then it produces the torque command which is required for the robot actuators. Regarding this torque command, the rehabilitation robot assists the patient's movement when it is needed. More detailed information about the rehabilitation robot and the cybernetic interface are given below.
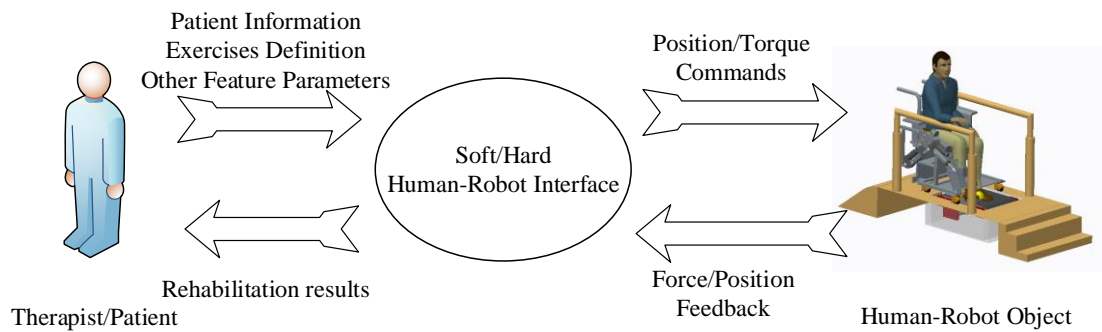
Figure 2.6 Structure and elements of the control system

● Development Procedure of Control System

In the development of the ERRobot control system, we adopt Model-Based Design (MBD) throughout the development process instead of relying on and working with actual physical systems. We use engineering software Matlab/Simulink/Simmechanics/Stateflow (MathWorks 2015, 64-bit, windows) to implement mathematically complex control strategies in real time, which also provides tools for visual modeling, simulation(of hybrid systems that may include continuous-time, discrete-time, and event-driven dynamics), system's functionality verification and automatic code generation, as shown in the following four steps. (1) Model construction of Human-ERRobot system: Control system design starts with an accurate plant model(a Human-ERRobot system in the rehabilitation system), whose complex dynamics model can be described using data-driven(i.e., system identification) or based on first principles[81]. A type of first-principles based modeling is physical modeling, where a model consists in connected blocks that represent the physical elements of the actual plant. In the thesis, most of the Human-ERRobot modules are built by the method of physical modeling. (2) Development of the model controllers: After creating the Human-ERRobot model, we continue to construct and analyze controllers based on the model. The proposed control system framework basically consisted of three distinct parts: a master upper-level control module, intermediate control module, and a low-level control module. The work uses the method of FSM (Finite State Machine) to build the supervisory logic part of upper-level controller shown in Figure 6.2 and create the corresponding commands. (3) Simulation of controller and Human-ERRobot system: Desktop simulation is performed to

demonstrate the efficacy of the proposed control architecture, which can also find design errors before building hardware and creating production software. Such early-verification testing reduces the need for expensive and difficult-to-access prototypes. To perform early-verification testing, we combine the controlled-object model, environment model (i.e., gravity, contact force, etc.), related sensors' signal, and control algorithm into one model and simulate the closed-loop system behavior. This method enables us to evaluate control laws rapidly, modify our control system, and immediately see the effects during simulation. (4) Deploying the developed and validated controllers: After the control system has been designed and tested, it is deployed onto a targeting microcontrollers through automatic code generation. With this approach we can generate highly efficient code for controllers and avoid errors that can happen when your design must be reinterpreted during manual coding. The whole control system runs and develops in Matlab environment.

To conveniently operate and build interactive simulation environment by persons (doctor, therapist and patient), a GUI system, window-based application program, is designed in Matlab environment, which also gives the user an easy access to all functions of the sHRI. Therefore, all functional modules are organized and called by the GUI program. Generally speaking, in order to activate some control modules, some necessary parameters such as task parameters (types of the exercise, the therapy period, the number of repetitions of exercise, desired reaching distance, desired speed of motion etc.), operator's identification (name, date of birth, kind of illness, physical properties of the operator, etc.) should be imported by means of GUI. Moreover, some useful data are shown on GUI to provide visual feedback in our system, where patients can visually see their movement and target location. The visual feedback is used not only to inform the participants of how closely they are tracking the desired motion but also as a motivational factor to keep them focused on the task. For some training programs, we make patients execute them in the form of the game. Rehabilitation training games that can activate collaboration between brain and limb are necessary for developing effective recovery processes. There is now sufficient evidence that using a rehabilitation protocol involving motor imagery practice in conjunction with physical practice of goal-directed rehabilitation tasks leads to enhanced functional recovery of paralyzed limbs [82].

- Design of the sHRI

The design of sHRI is referred to the software part design of control system. The proposed sHRI is based on the biomechanical model of NMS system. To develop the system, the biological engineering model must be built firstly. Therefore, some necessary knowledge including

neuromotor control mechanism, anatomy structure and physiology of functional parts, must be understood. For details, you can refer to relevant books and literatures such as [6, 83, 84]. The developed sHRI contains these component modules: GUI, module of Motion Intent Parse (MIP), Rehabilitation Assessment Module (RAM), and Motion Planning Module (MPM), etc. The Chapter 6 presents its detailed design process. The core task of the system is to capture the generic sequential control requirements related to recovery stage. As the decision-making center of the whole system, the developed control system should be able to carry out easily different training configurations to comply with the diversity of requirements such as applying for different patients. In addition, the control system is also required to handle other tasks such as: initialization, data calibration, management of user's safety, as well as other specific features.

● Control Related Hardware Design (hHRI)

The high-level control architecture of the LRRSym is implemented in a standard personal computer (PC) because extensive calculations are required due to the model based control algorithms. It hosts the program and performs the calculations for a large portion of the control scheme, including the organization and the coordination of the activities involved in the training sessions, the controlling of signal acquisition, storage and display, core algorithms executing, data analysis and report generation, and so on. According to design requirement, the PC must have the environment of real time computer simulation (Matlab/Simulink + RT Toolbox + Arduino Mega 2560 lower control board) to allow fast and easy system development. The Arduino Mega 2560 is a microcontroller board based on the ATmega2560. It is responsible for communicating with the sensors and motors, and for data preprocessing. Each active DOF has its own low-level control hardware in order to control the torque so as to generate a desired motion. An overview of this hardware and its function is given in Figure 2.7. The figure only shows a connection between
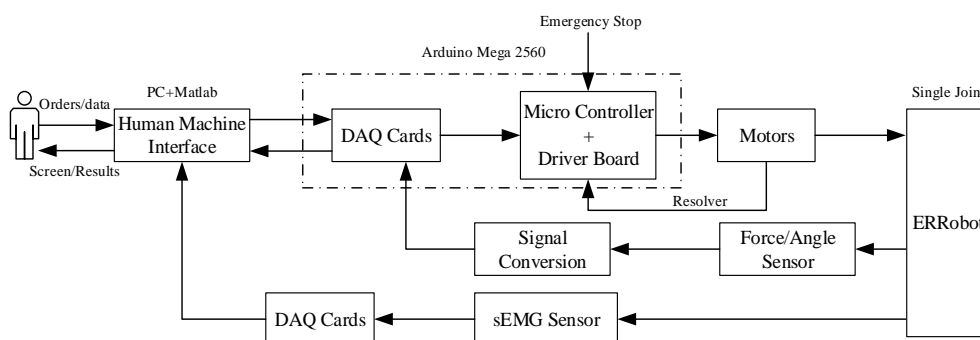
Figure 2.7 Relationship between hardware compositions of the control system

upper controller and an active joint controller. Other joint controllers adopt the same structure. The micro-controller board performs the local control loop and communicates with a central

PC. A fast communication line between PC and the lower control board is provided. Since USB is a widely used standard, which is available on all modern computers, and the work uses USB2.0 high speed serial bus (up to 480Mbit/s) is chosen as communication interface. These controllers update at 2KHZ.

2.5 System Working Flow

This section presents the working flow of the control system in the context of the gait rehabilitation task. When the control system is started, the main window is activated at the same time. At this time, users are asked to input patient's information (i.e., age, sex, height, weight, disease-related information) and select working modes ("automatic" or "manual (Debug)"). These patient's information can be used to create medical record data base, construct Human-ERRobot dynamic model, build and search the "Health Belt" etc. For the system's work mode, the 'manual mode' can complete module functional debugging and single step functional rehabilitation training. The "automatic mode" is to make the system automatically finish the whole rehabilitation process. We'll make an illustration by taking an example of "automatic mode" in the following.

Prior to running the system, we must select the rehabilitation task (for example, "Walking" task is selected) and set up some module parameters. Then click the "Start" button to make system run automatically. At the starting, the system turns into the stage of "data acquiring", when the subject is asked to repeat walking at his maximum effort according to the prompt information during pre-specified time, as shown in Figure 0.4. At that time, some necessary variable values are recorded, which mainly contain joint angles, and sEMG time sequence. After time over of the data acquiring, the system will turn into Rehabilitation Assessment and Recovery Program Planning (RA-RPP) module which acts as therapies to assess health level of patients' lower limbs and arrange the appropriate rehabilitation training. The outputs of RAS is used as the inputs of the RPP module, which is implemented by FSM whose state or event information can be from RAS or directly from GUI so as to make system run automatically. During the executing process of rehabilitation training, we ask the participants to follow a visually presented desired motion trajectory that is likely to command their concentration. The participants receive visual feedback of both their actual position and the desired position trajectories on a computer screen, which is placed in front of them. There are now sufficient evidences that using a rehabilitation protocol involving motor imagery practice in conjunction with physical practice of goal-directed rehabilitation tasks leads to enhanced functional recovery of paralyzed limbs among stroke sufferers[82]

2.6 Conclusion

As the preparations for further study for control system, this chapter firstly presented the system of LRRSym aiming to those people who have lower-extremity weakness or paralysis. It can implement almost every therapy stages from total paralyzed status to normally move for lower limb patients. Its main working part is the exoskeleton robot (ERRobot), whose 3D mechanical model is developed by CREO2.0 software. Next, we pay attention on the design of sHRI in the work. The details design process will be presented in subsequent chapters.

Chapter 3.   Surface Electromyography and Processing

In robot-assisted therapies, the action of rehabilitation robot (i.e., ERRobot in the work) is mainly decided by the patients i.e., current states, the next motion intention. To develop an autonomous controller for ERRobot system in the work, the sEMG have been used as nerve control signal to identify voluntary movement from a patient. In the work, combining with human musculoskeletal model, we estimated the joint moment of operators by analyzing the sampling sEMG data from lower limb muscles. The joint moment can be used as the indicator of human movement intention. And furthermore, during the process of sEMG processing, the work also develops a novel algorithm to perform sEMG (or other biological electrical signals) signal segment.

3.1 Application Merits and Properties of sEMG

There are two types of EMG: surface EMG (sEMG) and intramuscular EMG (iEMG). Both of them are the detection of the electrical activity associated with muscle contraction, and they can be obtained by sensor during muscle contraction. Therefore, EMG signals are directly linked to the desire of movement of the person. In my proposed control scheme, the sEMG has been used as nerve control signal to identify voluntary movement from operators. Raw surface EMG has an amplitude range of 10 mV, from -5 mV to +5 mV. Surface EMGs can be detected from 0 to 500 Hz, but the dominant frequency range is from 50 to 150 Hz[85]. The sEMG is the summation of MUAPs evoked at the same time and can be directly measured invasively with surface electrodes located on top of the skin. The acquired sEMG signal is usually considered to be random. Many models have been proposed for sEMG, e.g. the EMGsim model[86], the RRDsim model[87] and the SiMyo model [88]. However, these models do not include all the factors that affect sEMG signals, such as electrode configuration and location, fiber type. By a decoding procedure, these signals can be used for many aspects: biofeedback, robot control[89], ergonomics, sport medicine, painless assessment and evaluation of neuromuscular functions[90] and so on. Advantages of using sEMG signals in general are: (1) A main advantage by using sEMG is that the user's intention detecting prior to the actual contraction of the muscle. If the time between the emission and detection of the sEMG signal can be neglected in the context of this work, there is also a time between emission and force production. This time, called the electromechanical delay, is reported to be about 50–80ms[54, 55], mainly due to low muscle fiber conduction velocity and the chemical processes which lead to contraction. It allows the signal evaluation process to start before the force production begins, reducing the latency of control systems coupled to EMG signals; (2) It directly links with the movement intention of the person and it is emitted unconsciously: the property of sEMG emitted unconsciously makes operators no additional

mental load needed; (3) It can also be used to detect movement intention in the context of no limb movement: If the muscles are too weak or the external load too heavy, the intention of movement can still be detected, although no movement is performed[46]; (4) The measurement of the signals is not influenced by temporary external influences like contact forces, in contrast to force sensors; (5) It provides a safe, easy, and noninvasive method that make it readily obtainable even by nonmedical personnel; it is more suitable to provide insights into the neural motor control mechanism[90], and allows objective quantification of the energy of the muscle[91].

And besides, researchers also find some merits during the process using sEMG. For example, exoskeletons under sEMG control may be able to achieve lower metabolic cost than those under kinematic control[25].

3.2 Preparation and Signal Acquisition

● Subject Groups

Some references show that many factors can influence the sEMG signal which mainly include the age, gender, gait velocity [92, 93] and so on. Therefore, in order to eliminate the influence of these factors on sEMG signal as far as possible, we will initially group the subjects, that is , 12 subject groups by gender (male and female) and age (i.e., A:[20,29];B:[30，39], C: [40,49], D: [50,59], E: [60,69],and F: [70,79]) are adopted.

● Real Time Signal Acquisition of sEMG

From the generation of sEMG signal to the controlled object, sEMG signal processing generally
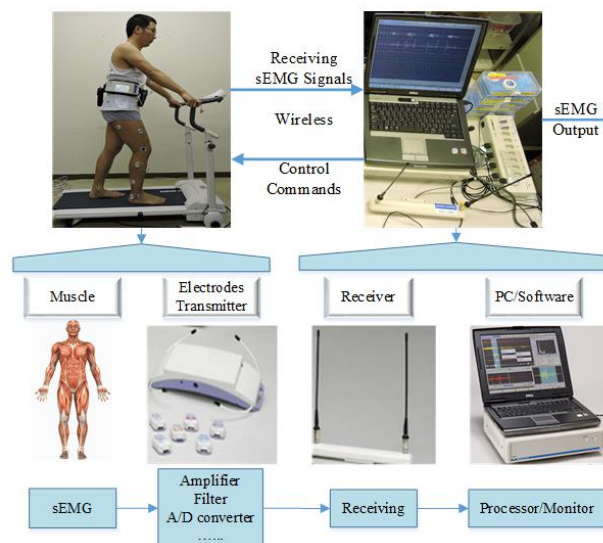
Figure 3.1 The system composition of the sEMG signal acquisition

passes through two processes, analog and digital signal processing processes. Nowadays, virtually

all contemporary analyses and applications of the sEMG are finished by digital signal processing that is accomplished with algorithms implemented on computers. It is necessary that signals are expressed as numerical sequences. At the beginning of the study, we adopted a wire sEMG acquiring system MEB-2312 to collect data, as shown in Figure 0.5.Howere, our study involves the long distance signal acquisition such as signal collecting during gait movement, so the wireless sEMG acquiring system is adopted. The Figure 3.1 shows the sEMG signal acquisition system adopted in our experiment, which shows the main hardware composition of the sEMG signal sampling.

The wireless system of sEMG signal acquisition adopted in the experiment is a telemetry multichannel surface electromyography system WEB-7000® QP-700H (NIHON KOHDEN Inc., Tokyo, Japan) shown in Figure 0.6. The Figure 3.2 shows a signal sampling window. The differential surface electrodes (ZB-150H: size 25x34.5x12mm; weight 10g; power: 3.7v Lithium rechargeable batteries) are connected wirelessly to main receiver sub-system (ZR700H: size 330x75x300; weight 5.5kg), which receive the signals and transfer them to PC through USB interface for off-line analysis. Following data transfer, the signals are displayed on a computer monitor. For the soft configuration during the acquisition stage of signals, several main parameter settings are included, that is, (1) the sampling rate was set at 1,000 Hz per channel and analog-to-digital converters is 16-bit; (2) the band pass filter is from 15 to 500 Hz; (3) the inter-electrode distance is an important parameters and you should make sure that this distance is consistent throughout all subjects and trials. In my experiment, this step is skipped because the electrodes has fixed electrode geometries. Before starting to obtain signal, a lot of preparations must be done, which is very important if a good quality signal is to be obtained. In my experiment, it includes (1) shaving the excess hair , using alcohol to removal of dirt, oil, and dead skin in order to reduce any skin resistance to below 5kΩ and allow electrodes to be attached without coming loose; (2) a part of subjects whose skin surface are dry use electrode gel (Elefix Z-181BE made in NIHON KOHDEN corporation in japan) rubbed into the skin, which can dramatically reduce the skin impedance and improve signal to noise ratio for the electromyography sensors; (3)In the decision of the specific site for the electrode, in order to assure repeatability for different subjects, various bony landmarks are used as a reference .After finding the position for one subject, marks are made so as to assure that the electrodes are over the same muscle fibers in different trials; (4)The inter-electrode distance is a important parameters and you should make sure that this distance is consistent throughout all subjects and trials. In my experiment, this step is skipped because the electrodes has fixed electrode geometries.
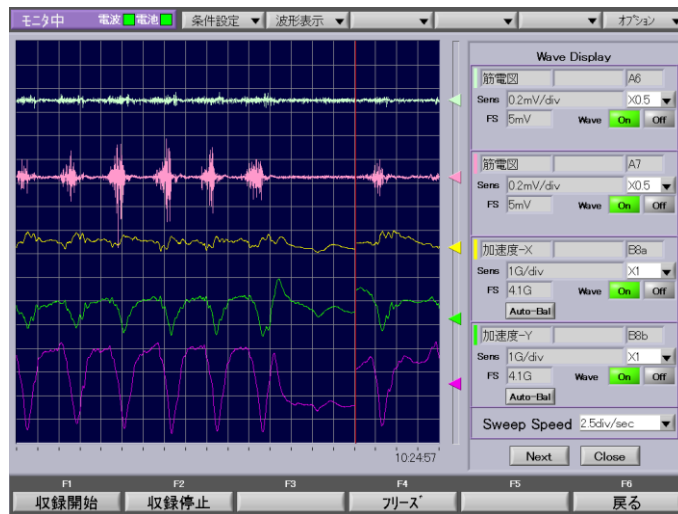
Figure 3.2 Waveform display window

3.3 Conditioning Methods of sEMG

When the sEMG signal is adopted to do some analysis, the data that is clean and keep enough useful information is necessary to obtain correct results. Therefore, according to practical application, we must implement some data processing operations so as to remove noise and strengthen the useful information. These operations can be divided into the common processing and particular processing according to the difference of application fields. It's main content and result have been published in [94]. In the system, in order to be able to easily modify the processing parameters and complete the process, we have developed a sEMG signal processing toolbox in MATLAB.

3.3.1 Common Signal Conditioning Process

(a) Removing the mean

The sample mean of sEMG waveform is subtracted from the data so as to remove any DC offset of the signal. These adjusted waveforms are then used to feed into the filter.

(b) Filtering operation

During the process of signal sampling, several major types of noise, artifact and interference are electrode noise, electrode and cable motion artifact, alternating current power line interference, and other noise sources such as a broad band noise from electronic instrument[95, 96]. The first three types of noise can be removed using typical filtering procedures such as band-pass filter, band-stop filter, or the use of well electrode and instrument [95, 96]. So before other signal digital processing, we first carried out the filtering operation. The filters have many types, such as low-pass filter, high-pass filter, band-pass filter, band stop and so on. It is necessary to know the bandwidth of the signal of interest. The SENIAM (Surface ElectroMyoGraphy for the Non-

Invasive Assessment of Muscles) recommendations for sEMG: high pass with 10-20 Hz cutoff, low pass "near 500 Hz" cutoff, in most cases. The paper mainly adopts the "Band pass" filter to remove the lower and higher frequencies signal in given cutoff frequencies. Low end cutoff removes electrical noise associated with wire sway and biological artifacts, and high end cutoff eliminates tissue noise at the electrode site. The main filter parameters include filter type, filter order and the frequency band to filter.

(c) Rectification

The rectification step is essential for getting the shape or "envelope" of the sEMG signal. The rectification operation usually has two kinds, which are "Half-wave" and "Full-wave" rectification. Half-wave rectification is that all negative data is discarded, positive data is kept. Full-wave rectification is that the absolute value of each data point is used.

(d) Smoothing(or Linear envelope)

The result, the envelope of the sEMG (denoted by "EnsEMG"), is a measure of the contraction level of the muscle. The spectral components of human movements are low-frequency, ranging from 1 or 2 Hz to 3–4 Hz for voluntary contractions and from 3–4 to 20 Hz for involuntary movements. For voluntary isometric contractions a 2 Hz cutoff frequency should be enough. For a more general detector, 6–8 Hz cutoff frequencies are used[14]. One advantage of the smoothing processing is that it can be used in real-time applications. There are two kinds of methods to finish the task of data smoothing in the toolbox developed in the work. They are "Moving-window average" and "Low-pass filter". The former method is a digital smoothing technique. Values are averaged over a specified moving time window. For moving windows the smaller the time window the less smooth the data will be. The latter is finished by a low pass filter. After finishing the signal processing process of the section, you can perform particular processing based on various applications, for example, signal segmentation described in next.

3.3.2 A Novel Method of sEMG Signal Segmentation

In the sEMG application, the basic challenge is signal processing. The signal segmentation is a key step of signal processing for many applications. The accuracy of signal segmentation result directly affects the precision of feature extraction and classification. In the present literature, the work is almost finished based on action signal onset and termination that are specified based on certain rules. They are performed either completely manually by moving a cursor across a display of the sEMG signal[97], or partially manually by letting the computer operator by trial and error decide on the threshold level. A disadvantage of these methods is that they are time and labor consuming, because it requires the assistance of a computer operator. Furthermore, the

interference of an operator ultimately means that subjective criteria are used to determine burst delimiters. In the section, to make up the deficiency of the conventional method of biological signal segmentation, we propose an automatic and online sEMG analysis method P&WND that is based on the analysis of peak information and non-equidistant window function. Compared with traditional methods, this method proposed in the paper greatly improves the data processing speed and accuracy, and reduce the storage memory, calculating time and the human's subjectivity. The implementation of the method mainly contains four steps.

Step1.Removing the signal segments whose derivatives are negative

In the method, the reference position of signal segmentation is designed to locate on the signal rising along. Therefore, the first step initially remove the signal segments whose derivatives are negative. Here, because the sampling signal x is discrete, derivative is approximated by first order finite difference. The expression is shown in Equation (3.1).

$$\frac{dx}{dt} \approx diff(k) = x(k+1) - x(k), k = 1,2,3\cdots \tag{3.1}$$

Where, k is the index of sampling signal.

A differential expression of discrete signal is defined as a vector shown in the Equation (3.2):

$$Diff = [0, diff(1), diff(2), \cdots, diff(k), \cdots diff(n)] \tag{3.2}$$

Next, in order to better illustrate this problem, define a "find (vector)" function that is to find the index of negative value in the Diff vector. Then assign the value of x(index) as 0.where,the "index" is the result of "find(vector) " function. The process is represented by Equation (3.3)

$$x\big(find(Diff < 0)\big) = 0 \tag{3.3}$$

The analyzed data greatly is reduced by the processing of the step and is also helpful to improve the accuracy of the subsequent data segmentation.

Step2.Getting the reference position to add window functions

The step is to get the reference position based on the information of signal peak and tries information, which is carried out recursively.

1) Assuming that the numbers of same action tries in the analyzing data session is TR, and then firstly finds out (TR+N) different peaks positions .Here, N is a positive integer. Considering the calculation time and accuracy of the extraction results, this work suggested that selection of N is 5 through our experiments. After the stage, get N1 signal positions. In order to better illustrate the relationship between initial intercept fragment numbers (TR+N) and the number of the actually

getting segments N1 after the calculation of the stage, an example is depicted in Figure 3.3.

(a)   (b)
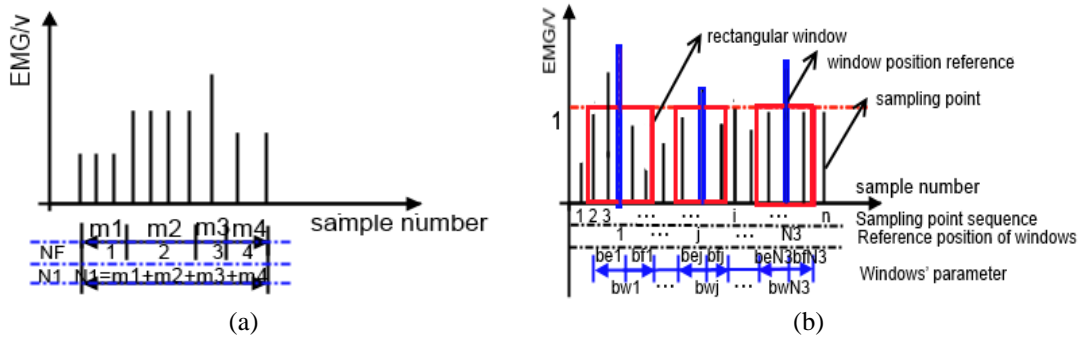
Figure 3.3 (a) the relationship between NF and N1; (b) Diagram of method to add window

In Figure 3.3(a), *mi* (i=1,2,3,4) is the corresponding sampling number of owing same peak value. Based on the previous illustration, if the number of first cutting pieces NF (=TR+N) equals to 4, then in the step finally get the number of action pieces N1 is equal to 10(=m1+m2+m3+m4). In the same way, if NF is set to 3, then N1 is 7 (=m2+m3+m4) and so on. From the above, we can see that if the peak has the same value, then N1 is greater than NF value, if it has no identical value, N1 is equal to NF.

2) Given the time interval among adjacent motion try is TI, then use this parameter to filter unqualified signal from the numbers of signal positions which are obtained in the previous step 1. That is to say, the step would remove those signal positions that are far less than TI. After the step, finally getting the number of signal slice is recorded as N2. In data processing, the sampling number SN is adopted to replace the sampling time TI to finish the work. It is known that the relationship between SN and TI is as: SN=FS×TI, Where, FS is the sampling frequency. The meaning of the other two parameters is as described above.

3) If the number N2 of selected signal position in stage 2 is less than the tries TR, then increase the number of N in stage 1 and repeat to perform the 1-3 stages until the size relationship between TR and N2 is met. After the step, finally getting the number of motion positions is denoted as N3.

Step3. Adding window function and get final wanted signal

Before getting the final analyzed signal data and extracting features vectors, the step adds window function to the signal positions obtained from step 1. An example of the adding window is depicted in Figure 3.3(b). Where, 1 to N3 corresponds to the reference location selected in the previous step. And bwj (j=1,2…N3) is the width of the window added to the position of j, which is equal to the sum of bej and bfj, which are forward and backward offset distance relative to reference position of adding window, respectively. As shown in Figure 3.3(b), rectangular window is adopted in this paper, whose expression is shown in Equation (3.4).

$$R(j) = \begin{cases} 1, j - bej \leq j \leq j + bfj \\ 0, other \end{cases} \tag{3.4}$$

In the step ,in order to get final analyzed signal, mean absolute value (MAV) is calculated in each window and filter out the signals based on the deviations from the mean. The MAV definition is shown in following formula (3.5).

$$MAV = \frac{1}{bwj} \sum_{i=j-bej}^{j+bfj} |S_i| \tag{3.5}$$

Where, S_i is the i-th sample. bwj is the width of j-th window and also is the number of samples in the j-th window. The meaning of the other parameters is as described above.

Step4.Feature extraction and selection or other data processing for each part of data

The last step is to calculate and select the feature based on added window segments obtained in previous step. Especially, this technology is used in the pattern recognition. Here the application in classification field is used as an example to illustrate the step content. For a classification algorithm, choosing discriminating and independent features is a key step to be successful[98]. A feature is an individual measurable heuristic property of a phenomenon being observed. And features extraction is also a good idea to reduce the complexity of signal to an appropriate size and reject noise and unimportant sEMG signal. There are a number of signal features that are possible to extract from a myoelectric signal, such as estimation of the amplitude, zero crossings or mean absolute value and so on, which are divided into time domain and frequency domain features. The temporal approaches to feature extraction are generally easy to implement and efficiently calculated, which makes them well suited for real-time feature extraction. So in order to verify the effectiveness of the proposed algorithm in this paper, we select the time domain features. Data with a high number of features require inevitably large processing time. So, for analyzing data, it is necessary to reduce the data dimensionality by selecting a subset of features that are relevant for classification. After the step, and then be able to perform the classification.

3.3.3 Normalization

sEMG normalization is a technique that permits to access the relative level of activation of a given muscle by expressing the absolute amplitude of the signal measured during the exercise as a percentage of a meaningful reference sEMG value[99]. Take the amplitude normalization for an example. The most compelling reason for using the method is to allow comparisons of sEMG activity between subject, between day and between-muscle, whose reference values can be a MVIC or the peak EMG during a dynamic activity. Using MVIC as a reference value has some

advantages such as having a physiological meaning. However, the measuring of MVIC reference value depends on subject`s subjective nature, for example, sincerity, motivation, or the degree of pain during the exertion. Therefore, to make up these limitations at some extent, sub-MVIC is used as a predetermined reference value [100].

In the work, the maximum sEMG value is used as a reference value to do amplitude normalization before performing time normalization at the experimrnt exploring the relationship of joint motion and muscle activity in Chapter 4.

● **Time Normalization**

In almost every practical measurement situation, each movement cycle or EMG measurement (or data collection) period lasts for a slightly different length of time. As a result, we cannot compare the EMG levels from sample "n" in one measurement period with the same sample "n" in another period if the lengths of the measurement periods (gait cycles) are different then the two samples may occur at very different points in the sEMG contraction cycle. Therefore, in order to make a valid comparison of sEMG data across various gait cycles we must find a way of making each cycle or measurement period identical. This process is called "Time Normalization".
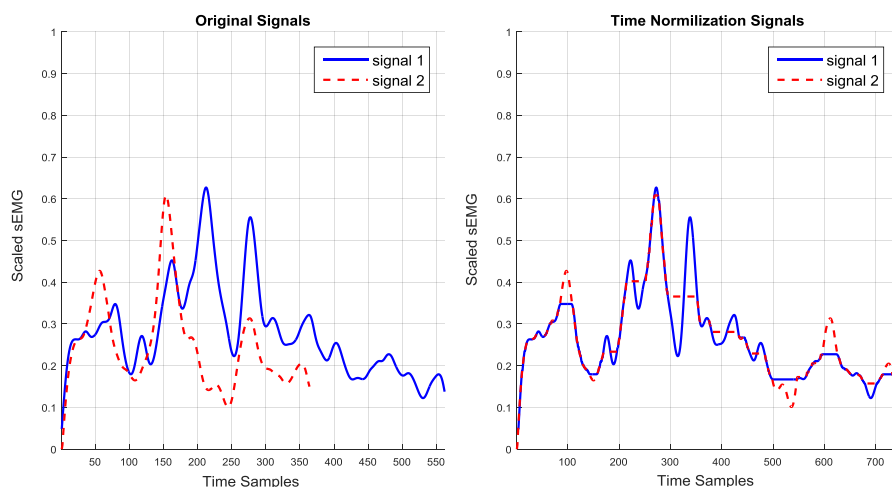
Figure 3.4 Time normalized for muscle "Tensor Fasciae Latae"

Continuous Dynamic Time Warping (CDTW) algorithm is used to execute the process, which uses a linear interpolation model. We select the first action stage signal to be a reference template and then time-warp all the data to that template. After the signal normalization process, all the signals have the same temporal length[101]. The Figure 3.4 shows a processed example, which is processed for muscle "Tensor Fasciae Latae" sampling in the action stage "FromStart2Max" when right knee is moving. The detailed is introduced in Chapter 4.

- **Normalization as Muscle Excitation**

In voluntary contractions, muscle force is modulated by a combination of motor unit(MU) recruitment and changes in MU activation frequency [102], both of which also affect the electrical activity in a muscle. Thus a direct relationship between the sEMG and exerted force might be expected. Many researchers have proposed some estimate models of muscle force, for example, reference [14] estimate force (or torque) by the intensity of the muscle sEMG. As discussed in previous chapter, the estimate method is inapplicable in my situation (i.e., muscle is in an anisometric contraction) to do motion control of exoskeleton robot. Therefore, we introduced a muscle model in the prediction of the torque production, whose full description will be given out in the next chapter.

As a part of the predicting model building, the neural input must be decided previously. In the work, we use the normalization value of the sEMG envelope signal as a continuous indicator of neural excitation input $u(t)$, because the value of $u(t)$ must be in the range of [0, 1], which is defined as Equation (3.6). It maps the sEMG signal with an arbitrary amplitude into a normalized signal in the range of [0 1], where a value of 1 stands for maximal voluntary muscle activation.

$$u_i(t) = \frac{\left|sEMG_i(t) - sEMG_i^{rest}\right|}{sEMG_i^{max} - sEMG_i^{rest}} \tag{3.6}$$

In this equation, $sEMG_i^{rest}$ represents the sEMG signal at limb rest and $sEMG_i^{max}$ represents the sEMG signal when the maximum muscle contraction occurs, which is calculated by mean value of $sEMG_i(t)$.

3.4 Validation

In order to verify the segmentation result accuracy of the new proposed method and effectiveness for action classifier, two experiments are designed .One is to directly view the segmentation result and evaluate the accuracy by experienced operator using visual inspection. The other experiment is to construct a LS-SVM classifier model so as to observe the signal segmentation method on the result of classification.

3.4.1   sEMG Signal and Initial Processing

The sEMG signals[103] collected from four sites on the hamstring and thigh muscles of lower limbs from four healthy young subjects (20-30 years old) are adopted. Table 3.1 list the electrode site placement. The symbols listed in the last "label" column are abbreviations for the channel signal. For example, in the symbols of "Ch5_Rleg_Thi", "Ch5" and "Rleg" represent 1st sampling channel and sampling from right leg, and "Thi" is the abbreviation of sampling muscle name

"thigh". The detailed description can refer to [94, 104, 105].

Table 3.1 Electrodes site placement

Site	Muscle(s)	Leg		Label
		left	right	
Ch5	thigh		◎	Ch5_Rleg_Thi
Ch6	hamstring		◎	Ch6_Rleg_Ham
Ch7	thigh	◎		Ch7_Lleg_Thi
Ch8	hamstring	◎		Ch8_Lleg_Ham

After getting the needed experimental data, some common and particular signal conditioning methods are applied to the data. To illustrate the process, the section presents the detailed process of data processing and parameter selection utilized in the experiment by taking the analyzing signal of Ch5_Rleg_Thi signal for an example. Its original waveform and processed signal waveforms are shown in Figure 3.5. After loading the original signal showed in Figure 3.5(a) into Matlab workspace, we first implemented these operations of removing mean value and filter, whose final result shows in Figure 3.5(b). In the processing, the digital band-pass filter is used and band-pass frequency is set to [20,450] Hz, the filter order 6 and so on. Next, we did the half-wave rectification operation for the signal data and the processing result is shown in Figure 3.5(c). At the last step of common processing, data smoothing operation is performed, and we showed the result in Figure 3.5(d). We used the smoothing method of "Moving-window average". We selected the width of the moving time window was 250ms, other parameters used the default values. The sEMG signal of other channels is also able to be processed in accordance with the above steps. At this time, we can continue to do the next particular processing such as segmentation and other further operations.

(a)Raw signal

(b) Removing mean value and filtering

(c)Rectification

(d)Smoothing
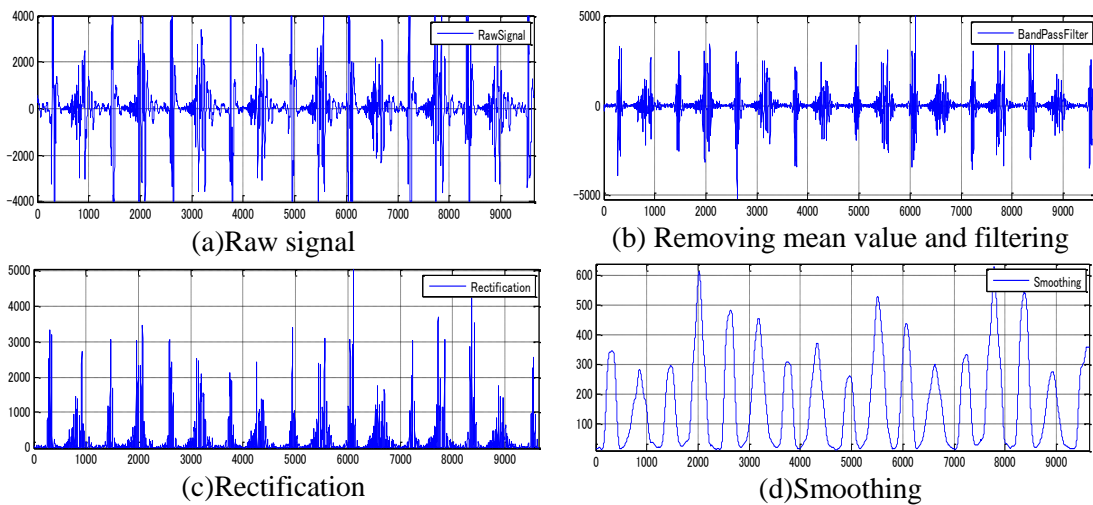
Figure 3.5 The main process of Ch5_Rleg_Thi signal processing

3.4.2 Visual Evaluation of sEMG Segmentation Results

Di Fabio's method is compared with the result obtained from the proposed method in the paper. Di Fabio's method is a good reference to benchmark a new technique. It is combinations of two variables that are threshold and duration[106]. The threshold is calculated by adding standard deviation (SD) factors to a mean value of resting state of sEMG signals. The duration is a period of one rectified pulse. A pulse beyond the threshold is tested within its period, or not. The two conditions are selected experimentally as the conditions yield the best output. For the signal of Ch5_Rleg_Thi, when using the Di Fabio's method, the onset is set at the first point where the sEMG signal exceeds the average value of the signal of the baseline for 350 consecutive milliseconds. The Figure 3.6(a) shows the result of segmentation for the signal of Ch5_Rleg_Thi when walking.

(a) The result of Di Fabio's method   (b) The result of the P&WND method
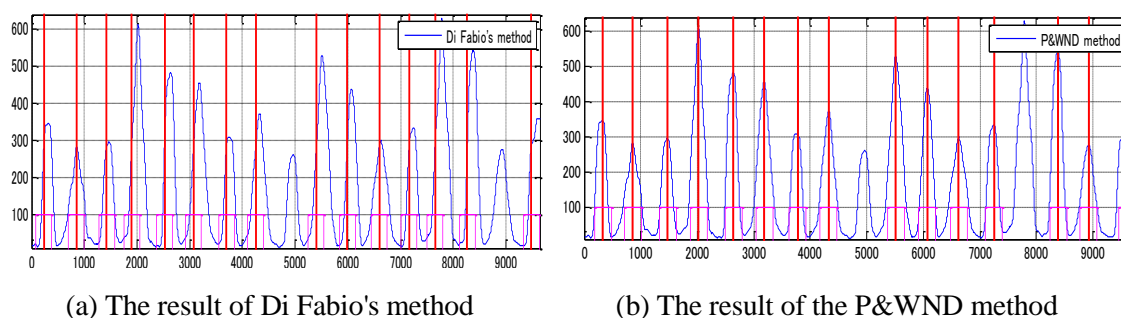
Figure 3.6 Results of data segmentation for Ch5_Rleg_Thi signal

To do the segmentation using the P&WND method proposed in the work, firstly need to decide the initial parameters. First of all, the action times TR is set to 15 and the number of different peaks NF is 20(=15+5). In the experiment, all the width of the window added is assigned to 400ms, and the front and rear half window width is set to equal. Finally, according to the interval between consecutive sampling actions, this experiment make the threshold of peak distance set to 480. The Figure 3.6(b) shows segmentation result of walking Ch5_Rleg_Thi signal by the algorithm proposed in the paper. From the division result, we can see that the segmentation result is very similar, which shows that the new method can fulfill the signal segmentation task.

In order to further quantitatively analysis the accuracy of the proposed method and Di Fabio's method for segmentation result, experienced operator is asked to judge the division result. Table 3.2 tabulates the comparison of the results of two segmentation techniques. The method proposed in the paper yields better results.

In the Table 3.2, 5-RT, 6-RH, 7-LT, 8-LH are the abbreviations of Ch5_Rleg_Thi, Ch6_Rleg_Ham, Ch7_Lleg_Thi, Ch8_Lleg_Ham, respectively. And we can get that the average success rate of the new method can reach 95.06%. However, the average success rate of Di Fabio's method only is

87.7%. Obviously, the new method has higher accuracy.

Table 3.2 Comparison of success rate of segmentation result

Segmentation Technique		Success Rate (%)				Total success rate (%)
		5-RT	6-RH	7-LT	8-LH	
Di Fabio's method	walking	93.3	86.7	93.3	100	93.32
	jumping	86.7	80.0	80.0	86.7	83.35
	runing	85.7	80.0	86.7	93.3	86.43
P&WND method	walking	93.3	100	93.3	93.3	94.98
	jumping	100	86.7	86.7	100	93.35
	runing	100	86.7	100	100	96.68

3.4.3　Validation by Using in Classification System

The objective of classification is to identify three different lower limb actions (walking, jumping and running) in the experiment. In order to generate feature vectors, four channels' sampled signals and four feature values, namely Mean Absolute Value (MAV), Root Mean Square (RMS), four-order autoregressive (AR) model coefficients and waveform length-WL, are selected. Therefore, the dimensionalities of feature vector are 28 columns (4 channels $\times$ 7features/channel). Before feature vectors are fed into classifier, principal component analysis (PCA) that is an unsupervised learning method is adopted to synthesize numerous criteria, eliminate information overlapping of the sample, and reduce the input dimension to 5.

For classification method, LS-SVM is employed in the experiment. It is a reformulation to standard Support Vector Machines (SVM)[107, 108]. Its working process is introduced briefly. When the data is linearly separable, it computes the hyper-plane that maximizes the margin between the training examples and the class boundary. When the data are not linearly separable, the examples are mapped to a high dimensional space where such a separating hyper-plane can be found. The mechanism that defines this mapping process is called the kernel function, whose choice is one of the key elements. In the experiment, Gaussian RBF kernel is adopted.

Table 3.3 Several core parameter values of LS-SVM Classifier

Segment Technique	Kernel	Coding schemes	Decode schemes	Dimension reduction	γ	σ^2	Accuracy (%)
Di Fabio's	RBF	one-versus-one	Hamming distance	PCA	10.736	0.25	79.78
P&WND					1.762	1.96	96.32

In order to make an LS-SVM model (with Gaussian RBF kernel), we need two tuning parameters: γ and σ^2. The γ is the regularization parameter, determining the trade-off between the training error minimization and smoothness. In the common case of the Gaussian RBF kernel,

σ^2 is the squared bandwidth. Although SVM is initially developed for binary classification problem, it can be adapted to deal with multi-class problems that are decomposed into multiple binary classification tasks. And several coding schemes can be used at this point: one-versus-one, one-versus-all and so on. To decode a given result, the Hamming distance, loss function distance and Bayesian decoding can be applied. Table 3.3 lists several main parameter values selected in LS-SVM classifier and the result of average testing accuracy. From the table, we can see that the average accuracy of Di Fabio's method is about 79.78% whereas the average accuracy of the P&WND segmentation method reached 96.32%. The LS-SVM results in the environment of the training data is shown in the Fig6.

(a)The result of Di Fabio's method (b)The result of P&WND method
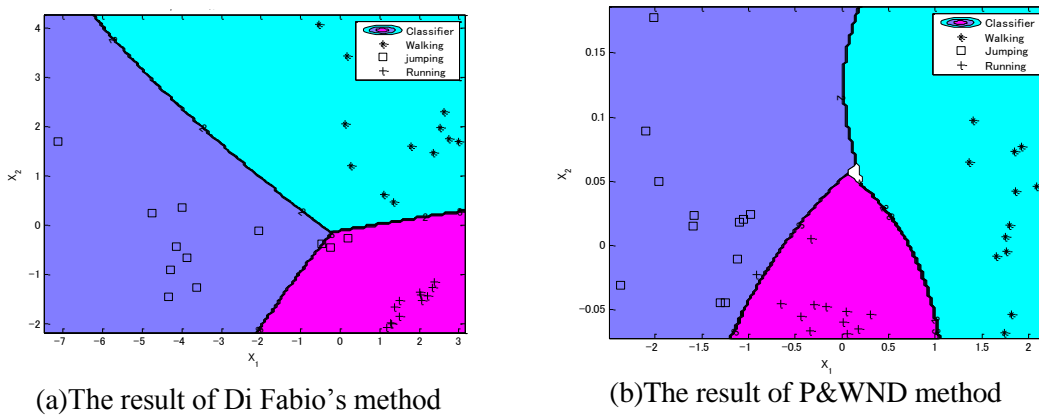
Figure 3.7 The result of LS-SVM classification

The experiment result shows that the new method P&WND greatly improves the accuracy of signal segmentation and the correctness of action classification when compared with Di Fabio's method. At the same time, it greatly reduces the human's subjectivity and time consumption.

3.5 Conclusion

In the system, the muscular activation is obtained by sEMG signals' processing, so some necessary process methods were developed in the chapter. Aiming to the particular processing, we proposed a novel and real-time sEMG signal segmentation method named after "P&WND". To validate the method, two experiments are designed. The first part is intuitively evaluation the results of signal segmentation, which is accomplished by comparing with the division result of the classic signal segmentation method, Di Fabio's method. Through judging the result by experienced operator, the new method has higher accuracy. The other experiment is to observe the influence of signal segmentation method on classification results. We can see that the average accuracy of Di Fabio's method is about 79.78% whereas the average accuracy of the P&WND segmentation method reached 96.32%. That is to say, the proposed method can improve the classification accuracy.

Chapter 4.   Engineering Model of Muscle Tendon Complex

Movements of the body are brought about by the harmonious contraction and relaxation of selected skeletal muscles [109]. To build a control scheme for a rehabilitation robot based on NMS system, muscle engineering model is needed to identify how a neural input to a mechanical output, i.e. a contraction force. Nowadays, there are many muscle models of varying complexity can be found in the literature[110], many of which are built by modifying the models proposed by Hill[111]and Huxley[112]. In my work, a type of Hill-type[111] muscle model is used, which modified based on the work of Thelen(2003)[113]. The content of the chapter has been partially published in the IEEE international conference on DASC2014 [114].

4.1 Dynamic Modeling of Muscle Tendon Complex

The electrical state of a muscle (i.e., activation) is modulated by neural inputs from the CNS[57], therefore, in the paper, the processed sEMG is adopted and represents the neural input $u(t)$ to the muscle, which has been described in Section 3.3.3. The sEMG provides a useful estimate of a muscle effort. Clinicians may infer muscle force directly from the magnitude of the rectified sEMG signals, but sEMG alone does not give an accurate assessment of muscle force [115]. Muscle has nonlinear properties that are not reflected in sEMG[85]. Therefore, the MTC model is introduced to predict muscle force, as shown in Figure 4.1. The dynamics of MTC includes activation dynamic and contraction dynamics, both of which can be viewed as two processes acting in series depicted in Figure 4.1. Neural excitation input $u(t)$ drives the activation dynamics and outputs the muscle activation $a(t)$, and then muscle activation feeds into the module of contraction dynamics, which outputs single muscle force. These muscle forces then drive the linked segment dynamic which affects the length and velocity of the MTC.
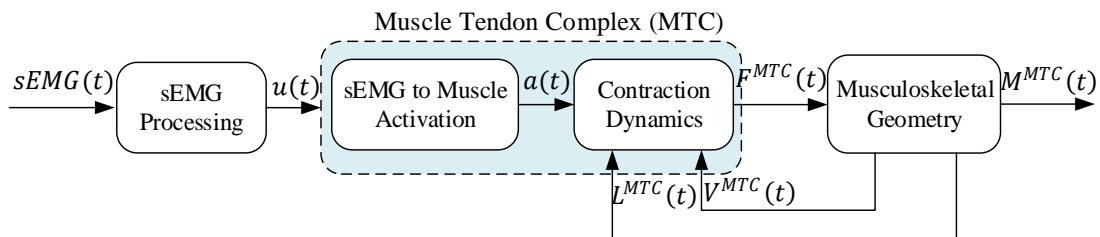
Figure 4.1 The input-output and components of the MTC dynamic module

4.1.1   Activation Dynamics

The module corresponds to the transformation of neural (or artificial) excitation $u(t)$ to activation $a(t)$. In this case, the work adopts the post-processed sEMG signal as $u(t)$, which is regarded as a control dimensionless variable and varies between 0 and 1.The $u(t)$ is used to

quantify the net motor signal to a muscle, and 1 means full excitation, 0 means no excitation. Just like the value of $u(t)$, muscle activation a (t) is also a dimensionless quantity [0, 1], and 1 means full activation, 0 means no activation. Determining a serial of $u(t)$ to produce a desired movement is one of the major challenges in creating a forward dynamic simulation. One approach is to use dynamic optimization to determine it that generates a simulation that best reproduces experimental data[116].

The activation of a muscle does not happen instantaneously. It takes some time until the muscle force is generated and ceases. In essence, the process of activation dynamics gives a description of the interaction between neural input $u(t)$ and the mechanical activation of muscle $a(t)$. The transformation has been shown to obey the differential equation .Some models such as developed model in [52, 113, 117, 118] have been presented. We adopted a model [76, 119, 120] that is defined in the Equation (4.1) in the paper, which is a modified non-linear first-order dynamic model

$$\frac{da(t)}{dt} + \frac{1}{\tau_{act}}\left[\frac{\tau_{act}}{\tau_{deact}} + \left(1 - \frac{\tau_{act}}{\tau_{deact}}\right)u(t)\right]a(t) = \frac{1}{\tau_{act}}u(t) \qquad (4.1)$$

Where, u is excitation, a is activation, τ_{act} is the activation time constant and τ_{deact} is the deactivation time constant. They are different for slow and fast muscles. Typical values for activation τ_{act} and deactivation τ_{deact} time constants are 0.01 s and 0.04 s, respectively.

The general form of Equation (4.1) indicates that the resulting activation function is a smooth function, ideal for feedback control analysis. In system modeling ,we always make the value of muscle activation greater than 0 (we assigned the minimum value of 0.001), which is based on the fact that in a whole muscle with it's vast number of contractile proteins some cross bridges will always generate force even in the absence of neural stimulation[110]. Once we have obtained the activation of the muscle, we can compute the resulting force exerted by the muscle using a muscle model that will be presented in next section.

4.1.2   Contraction Dynamics

As depicted in Figure 4.1, in the module of MTC dynamics, muscle length, or more precisely MTC length, MTC contraction velocity, and neural muscle activation are inputs, and MTC force is output. The section will construct their relationships. Before the construction, several important assumptions are presented. For details, please refer to literatures [121] and [122]. (1) Fibers are straight, parallel, of equal length, and coplanar; (2) The area and height of the parallelogram remains constant; (3) The muscle and tendon units are massless even though there are some stability issues which arise if muscle mass is absent in the dynamics.

(1) Model Structure

As depicted in Figure 4.2, this type of model consists of three components: an active contractile element (CE), a passive parallel element (PE) and a series element (SE)[123]. The contractile element CE is an active force producing component, in which the generated force f_{norm}^{ce} is a function of the activation $a(t)$, non-linear force-length $f_l^{ce}(l_{norm}^m(t))$ and force-velocity $f_v^{ce}(v_{norm}^m(t))$ relationships. We give their relationships as following Equation (4.2) applied and described by Zajac [121], that is, the total active force is given by the product of the force-length and force-velocity dependences .

$$
\begin{aligned}
f_{norm}^{ce} &= a(t)f_l^{ce}\left(l_{norm}^m(t)\right)f_v^{ce}\left(v_{norm}^m(t)\right) \\
&= af_l^{ce}(l_{norm}^m)f_v^{ce}(v_{norm}^m)
\end{aligned}
\tag{4.2}
$$

Here, the subscript string "norm" means that the corresponding variable is processed to a normalization quantity. The main advantage of the no dimensional method is that it can perform analysis for different persons and muscles conveniently. In the whole model building, besides of the non-dimension quantities $f_{norm}^{ce}, l_{norm}^m, v_{norm}^m$ introduced above, almost all other forces , length and velocity quantities are normalized to maximum isometric muscle force (F_0^M) , optimal muscle fiber length (L_0^M), tendon slack length (L_s^T) and maximum muscle fiber contraction velocity(V_{max}^M)resreactively. If we use the $F^{CE}, L^{CE}, V^{CE}, F^{PE}, L^{PE}, V^{PE}, F^{SE}, L^{SE}, V^{SE}$ notations to represent produced forces and their length, velocity of elements CE, PE, SE, respectively, then their corresponding normalization quantities are $f_{norm}^{ce} = F^{CE}/F_0^M$, $f_{norm}^{pe} = F^{PE}/F_0^M$, $f_{norm}^{se} = F^{SE}/F_0^M$ and $l_{norm}^{ce} = L^{CE}/L_0^M, l_{norm}^{pe} = L^{PE}/L_0^M, l_{norm}^{se} = L^{SE}/L_s^T$,and $v_{norm}^{ce} = V^{CE}/V_{max}^M$, $v_{norm}^{pe} = V^{PE}/V_{max}^M$, $v_{norm}^{se} = V^{SE}/V_{max}^M$. The details of no dimensional method can refer to [76, 114, 120, 121].According to model in Figure 4.2, the lengths, velocities of element of CE and PE are equal, so both of the quantities l_{norm}^{ce} and $l_{norm}^{pe}$are replaced with l_{norm}^m ,and v_{norm}^{ce} , v_{norm}^{pe} are replaced with $v_{norm}^m$to derive the muscle contraction dynamic equation in the thesis.

Using Newton's third law (under the assumption that the muscle and tendon units are massless), it is obvious that the muscle-tendon elements fulfill the force equilibrium Equation (4.3) in which the force dependencies are also specified.

$$
\left(F_0^M f_{norm}^{ce}(a, l_{norm}^m, v_{norm}^m) + F_0^M f_{norm}^{pe}(l_{norm}^m)\right)cos\alpha - F_0^M f_{norm}^{se}(l_{norm}^t) = 0 \tag{4.3}
$$

From Equation (4.2) and (4.3), we know that the force of tendon F^T can be expressed as Equation (4.4).The three equations are used as the auxiliary equations to derive Equation (4.25).

$$F^T = F_0^M f_{norm}^{se}(l_{norm}^t)$$

$$= F_0^M \left(a f_l^{ce}(l_{norm}^m) f_v^{ce}(v_{norm}^m) + f_l^{pe}(l_{norm}^m) \right) cos\alpha \qquad (4.4)$$

The elements' forces in (4.4) will be explicitly formulated in the following paragraphs.
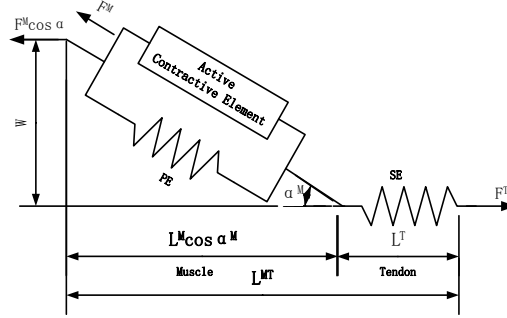
Figure 4.2 Architecture of adopted Hill-type muscle-tendon actuator model

(2) Muscle and Tendon Properties

To define the dynamic of a particular muscle, we first give some attributes of a generic muscle, then be scaled with appropriate parameters to reflect the dynamics of a particular muscle, as discussed above. For this purpose, we must supply five parameters $(F_0^M, L_0^M, L_s^T, V_{max}^M, \alpha_0)$ .In addition, because there exist the relation of $V_{max}^M = L_o^M/\tau_{max}$, there are four scaling parameters $(F_0^M, L_0^M, L_s^T, \alpha_0)$ needed in the model in fact. Four general curves (related parameters are listed in Table 0.1) are: (1) force-length function for passive element PE; (2) force-length function of active muscle CE; (3) force-velocity function for active muscle and (4) the force-length relation for tendons.

(1) Force-length function for passive element PE

The curve produced when muscle is not stimulated is referred to as the passive force, $f_l^{pe}(l_{norm}^m)$, which is only related with muscle fiber length. The relation between them is plotted in Figure 4.3(PE, solid line), which is represented by exponential function as Equation (4.5):

$$f_l^{pe}(l_{norm}^m) = \frac{e^{k^{PE}(l_{norm}^m-1)/s_0^m} - 1}{e^{k^{PE}} - 1} \qquad (4.5)$$

Where, k^{PE} is an exponential shape factor, $k^{PE} = 5$ and s_0^m is the passive muscle strain due to maximum isometric force, which is set differently for young and old adults as described in literature [113].Here, $s_0^m = 0.6$.

(2) Force-length function of active muscle CE

The contractile element CE represents the active fiber bundles in the muscle. Its force depends on

the current length of the muscle fibers. The force-length relationships can be easily obtained by force measuring at various lengths under isometric conditions. When muscle is fully activated, i.e., a(t)=1, the resulted curve is called the tetanized curve referred to as the active force-length relation $f_l(l_{norm}^m)$ .It is modeled as a Gaussian function[110, 113, 124]as Equation (4.6).When $w = 0.56$ and $c = 0.05$,the force-length relationships of active muscle CE is plotted in Figure 4.3(CE, blue dashed line), which is in good agreement with results[76] and the conclusion of force is only generated in a nominal region, $0.5 < l_{norm}^m < 1.5$ given in literatures [113, 125].A theoretical explanation for the active muscle force production is called the Sliding Filament Theory developed simultaneously by H. Huxley[112] and A. Huxley[76, 126].

$$f_l^{ce}(l_{norm}^m) = e^{c\left(\frac{|l_{norm}^m - 1|}{w}\right)^3} \tag{4.6}$$

Where w depicts the width of the normalized bell curve ,$w = 0.56, c = \ln(0.05)$,fulfilling $f_l^{ce}(1 \pm w) = 0.05$[127].
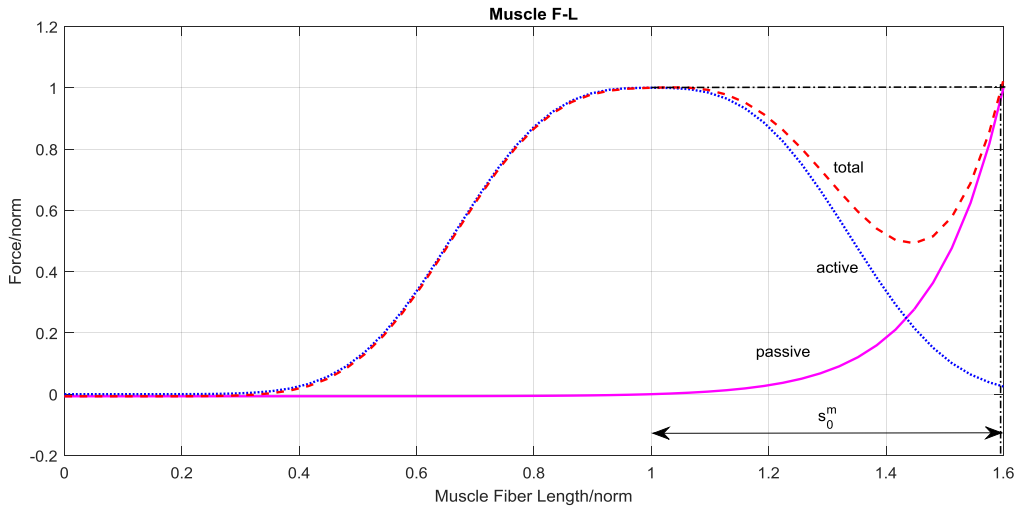
Figure 4.3 Isometric Force-Length Relation of the contractile element (CE, blue dashed line), and Force-length relation of the parallel elastic element (PE, solid line), and total force of muscle fiber (PE+CE, dashed line) at activation a (t) =1

(3) Force-velocity function for active muscle

Active muscle force is also dependent on muscle velocity, denoted by $f_v^{ce}(v_{norm}^m)$. When a muscle actively shortens (concentric contraction, $v_{norm}^m < 0$), it produces less force than it would under isometric conditions, that is, the active force reaches the maximal value F_0^M when the velocity, v_{norm}^m,is zero. That is to say, when muscle actively shortens, force $f_v^{ce} < 1$. For the force-velocity relation of the active muscle fiber is modeled by Equation (4.7) which is obtained by modifying model described in literature[113] . For the stabilization in simulation, when muscle force f_v^{ce} is equal to or larger than $\bar{F}_{len}^M$ (out the normal range of muscle fiber can be extended,

so it fulfills the relation $|f_v^{ce}| < \bar{F}_{len}^M$ in normal situation), we assign the value of $\bar{F}_{len}^M - f_0$ to input f_v^{ce}, and f_0 is assigned to 0.001 in my work. The advantages of this model are the possibility to use similar equations for concentric and eccentric force velocity relation and the good approximation of the experimental data.

$$v_{norm}^{ce} = \frac{(0.25 + 0.75a)(f_v^{ce} - 1)}{k} \tag{4.7}$$

Where parameter k is computed differently by Equation (4.8) depending on whether the muscle fiber is shortening ($f_v^{ce} \leq 1$) or lengthening ($f_v^{ce} > 1$)

$$k = \begin{cases} \left(1 + \dfrac{f_v^{ce}}{A_f}\right), |f_v^{ce}| \leq 1 \\[2ex] \dfrac{2\left(1 + \dfrac{1}{A_f}\right)\left(\bar{F}_{len}^M - f_v^{ce}\right)}{\bar{F}_{len}^M - 1}, 1 < |f_v^{ce}| < \bar{F}_{len}^M \\[2ex] \dfrac{2\left(1 + \dfrac{1}{A_f}\right)\left(\bar{F}_{len}^M - f_v^{ce}\right)}{\bar{F}_{len}^M - 1}, \bar{F}_{len}^M \leq |f_v^{ce}| \ and \ f_v^{ce} = \bar{F}_{len}^M - f_0 \end{cases} \tag{4.8}$$

Where, f_v^{ce} is the force-velocity relationship of active muscle CE element, which can be computed by Equation (4.3) or (4.4); $\bar{F}_{len}^M$ is the maximum normalized muscle force achievable when the fiber is lengthening ,whose value is different for young and old adult[113]. We set 1.8 as a default value; A_f is a force-velocity shape factor, which was set to 0.25[117]. Using these default parameter values, we can get the force-velocity relation curve shown in Figure 4.4.
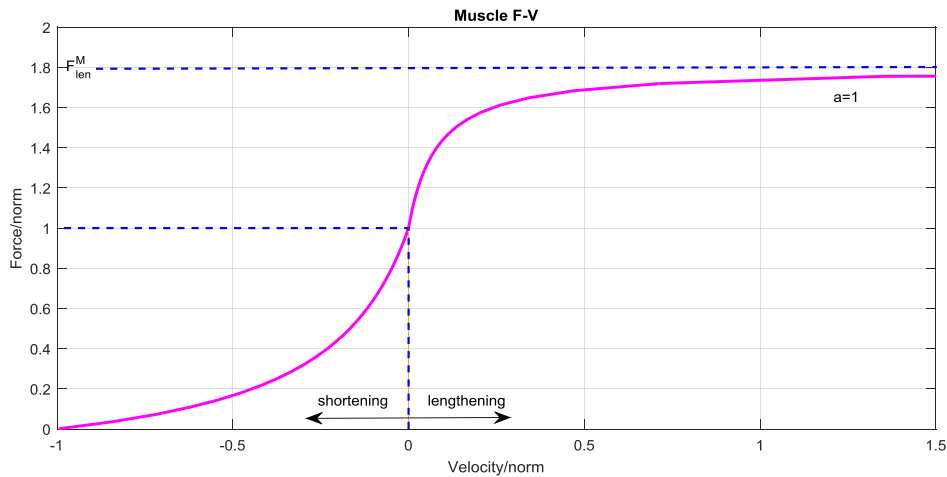
Figure 4.4 Force-Velocity Relation of the muscle contractile element (CE), Full activation

(4) Force-length relation for tendons.

The tendon tangent of modulus of elasticity increases with strain in the "toe" region ε_{toe}^T and then

becomes linear at higher strains until the tendon fails. Tendon failure occurs around 10% strain and 100 MPa[121]. Use notation ε^T to represent the tendon strain and it is defined as Equation (4.9):

$$\varepsilon^T = \frac{L^T - L_s^t}{L_s^t} \tag{4.9}$$

Where the L^t, L_s^t represent the tendon length and its slack length.

The force-strain relationship of tendon is represented by an exponential function [113]:

$$f_{norm}^t = \begin{cases} \dfrac{\bar{F}_{toe}^T}{e^{K_{toe}} - 1}\left(e^{\frac{K_{toe}\varepsilon^T}{\varepsilon_{toe}^T}} - 1\right), 0 \le \varepsilon^T \le \varepsilon_{toe}^T \\ K_{lin}(\varepsilon^T - \varepsilon_{toe}^T) + \bar{F}_{toe}^T, \varepsilon_{toe}^T < \varepsilon^T \end{cases} \tag{4.10}$$

Let $m = \bar{F}_{toe}^T/(e^{K_{toe}} - 1)$, $n = K_{toe}/\varepsilon_{toe}^T$, $k_l = K_{lin}$, $g = -\varepsilon_{toe}^T K_{lin} + \bar{F}_{toe}^T$, at this time, the Equation (4.10) above can be rewritten into the following Equation (4.11). Here, K_{toe} (1.5 as a default value) is an exponential shape factor and K_{lin} is a linear scale factor. The transition from nonlinear to linear behavior was prescribed to occur for normalized tendon forces greater than $\bar{F}_{toe}^T = 0.5$. Requiring continuity of slopes at the transition resulted in $\varepsilon_{toe}^T = 0.609\varepsilon_0^T$ and $K_{lin} = 37.5$. When ε_0^T is assigned to 0.033, the stress-strain has the following relationship depicted in Figure 4.5.

$$f_{norm}^t = \begin{cases} m\left(e^{n\varepsilon^T} - 1\right), 0 \le \varepsilon^T \le \varepsilon_{toe}^T \\ k_l\varepsilon^T + g, \varepsilon_{toe}^T < \varepsilon^T \end{cases} \tag{4.11}$$

Then

$$f_{norm}^t = \begin{cases} 0.1436\left(e^{74.638\varepsilon^T} - 1\right), 0 \le \varepsilon^T \le 0.0201 \\ 37.5\varepsilon^T - 0.2536, 0.0201 < \varepsilon^T \end{cases} \tag{4.12}$$
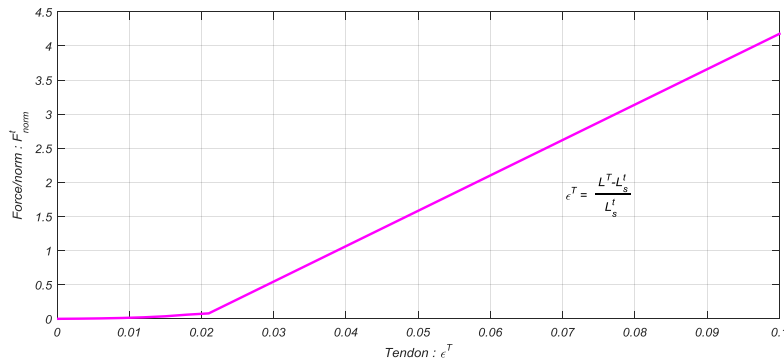
Figure 4.5 Tendon standardized force-strain relationship

If we multiply (4.11) by F_0^M, we have an expression for F^T

$$F^T = F_0^M f_{norm}^t = F_0^M \cdot \begin{cases} m\left(e^{n\varepsilon^T} - 1\right), 0 \leq \varepsilon^T \leq \varepsilon_{toe}^T \\ k_l\varepsilon^T + g, \varepsilon_{toe}^T < \varepsilon^T \end{cases} \tag{4.13}$$

Based on Equation (4.9)(4.13),we can derive an equation for L^T in term of F^T.

$$L^T = L_s^t \cdot \begin{cases} \dfrac{ln\left(\dfrac{f_{norm}^t}{m} + 1\right)}{n} + 1, 0 \leq f_{norm}^t \leq m\left(e^{n\varepsilon_{toe}^T} - 1\right) \\ \dfrac{f_{norm}^t - g}{k} + 1, (k\varepsilon_{toe}^T + g) < f_{norm}^t \end{cases} \tag{4.14}$$

Then

$$L^T = L_s^t \cdot \begin{cases} 0.0134\,ln(6.9638 f_{norm}^t + 1) + 1, 0 \leq f_{norm}^t \leq 0.5 \\ \dfrac{f_{norm}^t + 0.2536}{37.5} + 1, 0.5 < f_{norm}^t \end{cases} \tag{4.15}$$

Now we derived from Equation (4.11) (4.9) the absolute tendon stiffness which defined by $K^T \equiv dF^T/dL^T$, which is shown in Equation (4.16)

$$K^T(F^T) = \frac{dF^T}{dL^T} = \frac{dF^T}{df_{norm}^t}\frac{df_{norm}^t}{d\varepsilon^T}\frac{d\varepsilon^T}{dL^T}$$

$$= \frac{F_0^M}{L_s^t}\begin{cases} (nf_{norm}^t + mn), 0 \leq f_{norm}^t \leq m\left(e^{n\varepsilon_{toe}^T} - 1\right) \\ k, m\left(e^{n\varepsilon_{toe}^T} - 1\right) < f_{norm}^t \end{cases} \tag{4.16}$$

Then

$$K^T(F^T) = \frac{F_0^M}{L_s^t}\begin{cases} (74.638 f_{norm}^t + 10.7186), 0 \leq f_{norm}^t \leq 0.5 \\ 37.5, 0.5 < f_{norm}^t \end{cases} \tag{4.17}$$

(3) Derivation of Contraction Dynamics

There are a couple of reasons why the muscle and tendon should be grouped together and treated as the musculotendon unit in gait analysis. One of them is that tendon and muscle act in series and when the tendon stretches an amount approaching L^M, the force-length characteristics of the grouped actuator differ significantly from the muscle alone[121]. Muscle-tendon contraction dynamics accounts for the interaction of the activation-force-length-velocity properties of muscle and the elastic properties of tendon. Next, we will derive the muscle-tendon unit dynamics depicted in Figure 4.2 by utilizing the curves defined in the previous section. It can be expressed by Equation (4.18).Here, we denote tendon velocity by V^T,that is, $V^T = dL^T/dt$.

$$\frac{dF^T}{dt} = \frac{dF^T}{dL^T}\frac{dL^T}{dt} = K^T \cdot V^T \tag{4.18}$$

According to Figure 4.2, four physical quantities (α, L^M, L^T, L^{MT}) satisfy the Equation (4.19).Among of them, L^{MT} is the length of the muscle-tendon that is defined to be the distance from origin to insertion and it is a function of the segmental orientations.

$$L^{MT} = L^T + L^M cos\alpha \tag{4.19}$$

Now an expression for V^T can be formulated by differentiating Equation (4.19),

$$\dot{L}^T = \dot{L^{MT}} - \dot{L}^M cos\alpha + \dot{\alpha}L^M sin\alpha \tag{4.20}$$

The pennation angle changes as the length of the muscle changes. According to the assumption above, that is, the width of the muscle W remains constant, so that the following Equation (4.21) is always true. So that we can compute how α changes.

$$W = L^M sin\alpha = L_0^M sin\alpha_0 \tag{4.21}$$

Where, L_0^M is the optimal fiber length and α_0 is the optimal pennation angle. They are input parameters and are typically obtained from the literature for the particular muscle.

The first time derivative of Equation (4.21) yields the components of the angular velocities:

$$\dot{\alpha} = -\frac{\dot{L}^M}{L^M} tan\alpha \tag{4.22}$$

Substitution of $\dot{\alpha}$ into Equation (4.20) gives:

$$V^T = \dot{L}^T = \dot{L^{MT}} - \frac{\dot{L}^M}{cos\alpha} \tag{4.23}$$

In the Equation (4.23), muscle fiber velocity $\dot{L}^M$(it is also $V^M = v_{norm}^m v_{max}$) can be calculated by Equation (4.24).

$$v_{norm}^m = \frac{V^M}{v_{max}} = f_v^{-1}(f_v^{ce}) $$

And so

$$V^M = v_{max}f_v^{-1}(f_v^{ce}(v_{norm}^m)) \tag{4.24}$$

In the Equation (4.24), $f_v^{ce}(v_{norm}^m)$ can be derived by combining with Equation (4.2)(4.3)(4.4). It shows in Equation (4.25)

$$f_v^{ce}(v_{norm}^m) = \frac{(F^T/\cos\alpha) - F_0^M f_l^{pe}(l_{norm}^m)}{a \cdot F_0^M \cdot f_l^{ce}(l_{norm}^m)} \tag{4.25}$$

Substituting the expressions given by Equation (4.24) (4.25) into Equation (4.23), we obtain the following relation:

$$V^T = L^{\dot{M}T} - \frac{v_{max}}{\cos\alpha} f_v^{-1}\left(\frac{(F^T/\cos\alpha) - F_0^M f_l^{pe}(l_{norm}^m)}{a \cdot F_0^M \cdot f_l^{ce}(l_{norm}^m)}\right) \tag{4.26}$$

Thus, substitute of expression given by Equation (4.26) into Equation (4.18), the differential equation is:

$$\frac{dF^T}{dt} = K^T \cdot \left(L^{\dot{M}T} - \frac{v_{max}}{\cos\alpha} f_v^{-1}\left(\frac{(F^T/\cos\alpha) - F_0^M f_l^{pe}(l_{norm}^m)}{a \cdot F_0^M \cdot f_l^{ce}(l_{norm}^m)}\right)\right) \tag{4.27}$$

There are different choices of state that can be used for the muscle-tendon model. In this model building, we will use MTC length, velocity and muscles activation.

4.2 Computation of the Moment

The torque contribution on the skeletal system from each muscle is computed as the product of force and corresponding moment arm. Therefore, to prepare for the coupling the MTC system with skeletal system, the section presents the computing method of MTC joint moment. The net effect of muscle activity at a joint can be calculated in terms of net muscle moments, denoted by $\boldsymbol{M}_k(t)$. Combined with muscle dynamic model, the joint torque exerted by internal muscles can be exported by Equation (4.28), which is a net algebraic sum of the cross product of all force vectors and moment arm vectors relative to the axis of the joint(s) the muscle spans. Here, muscle moment arm d is the measure of the effectiveness with which the contraction force of a given muscle can generate a torque about a "joint of interest".

$$\boldsymbol{M}_k(t) = \sum_{i=1}^{Ne} \boldsymbol{F}_e^i(t) \times \boldsymbol{d}_e^i(t) - \sum_{j=1}^{Nf} \boldsymbol{F}_f^i(t) \times \boldsymbol{d}_f^i(t) \tag{4.28}$$

Where, k is the joint identifier. For lower extremity, k could be hip, knee, ankle and other joint identifiers. Ne is the number of extensor muscles for k joint; Nf is the number of flexor muscles for k joint; $F_e^i(t)$ is the force produced by $i - th$ extensor muscle at time t ; $F_e^j(t)$ is force produced by $j - th$ flexor muscle at time t; $d_e^i(t)$ and $d_f^j(t)$ are moment arm of $i - th$ extensor and $j - th$ flexor muscle at time t, respectively.

Combining with the Equation (4.28) and considering the fact that: (1) we didn't measure all

muscles responsible for joint flexion and extension; (2) we implemented model simplifications; and (3) any friction effects at the joints or within the muscle cannot be separated from this net value, we introduce a adjusting gain G and finally get the Equation (4.29) to evaluate the moment of joint k from MTC $i = 1,2 \ldots N_k$. In general, the muscle contractile elements create actually moments higher than that calculated values, so the gain G is greater than 1, that is, $G > 1$.

$$M_k(t) = G \cdot \left[\sum_{i=1}^{N_k} (\pm F_k^i(t)) \times d_k^i(t) \right] \qquad (4.29)$$

The muscles' net moment is the sum of its components with the corresponding sign [+] for agonist muscles and [-] sign for antagonist muscles. In the Equation (4.29), N_k is the numbers of muscles that dominate over joint k ; $F_k^i(t)$ is the force produced by $i - th$ muscle acting on the joint k at time t and $d_k^i(t)$ is the moment arm of $i - th$ muscle acting on the joint k at time t; The sign in equation depends on whether the MTC i acts to extend or flex the joint k it spans; G is a adjusting gain.

According to the activation and contraction dynamics model of the muscle-tendon complex, we can know that the muscle force $F_k^i(t)$ is a function of the total length length L^{MT} of MTC, total velocity $V^{MT}(i.e., L^{\dot{MT}})$ of MTC and sEMG signal, which is represented by Equation (4.30)

$$F_k^i(t) = f\left(sEMG_k^i, {}^{MT}L_k^i, {}^{MT}V_k^i\right) \qquad (4.30)$$

Where $sEMG_k^i$ is the sEMG signal from muscle i actuating the joint k ; ${}^{MT}L_k^i, {}^{MT}V_k^i$ are the length and velocity of the MTC.

To derive ${}^{MT}L_k^i$ and ${}^{MT}V_k^i$ as a functions of the generalized coordinates (i.e., Joint Angle), it is necessary to place the muscles geometrically on the body segment. A human rigid model of Linked Segment (LS) Digital Human Model (DHM), denoted by LS-DHM is constructed in next chapter. To represent the coordinate of muscle linking point on body, we constructed LRFs (Local Reference Frames) in each of rigid body such as thigh body. These reference frames are used to specify the attachments of muscles and ligaments. Given any free joint k comprised of skeleton A, B, there is a muscle i spans the joint. Let coordinate point S_1 through S_n define the muscle path. Points S_1 through S_m are fixed in body A, and S_{m+1} through S_n are fixed in body B. Thus, the line segment of $S_m S_{m+1}$ is the only muscle segment that changes length as the joint movement. A minimum of two points is required to define the muscle path. Muscle-tendon total length (L^{MT}) is given by Equation (4.31). The velocity of the MTC is then just the time derivative of the length.

$$L^{MT} = \sum_{i=1}^{n-1} | S_{i+1} - S_i | \tag{4.31}$$

As shown in Equation (4.29), the moment arm $\boldsymbol{d}_k^i(t)$ described as a function of the generalized coordinates (i.e., Joint Angle) has to be determined. It depends only on the corresponding joint angle θ between the articulating bones, that is $d = d(\theta)$. According to Delp, etc. [128, 129], using the principle of virtual work, the moment arm equals the derivative of the length $S_m S_{m+1}$ with respect to joint angle, that is, we can calculate moment arms of the three orientation angles in three planes by equation $d_i = \partial |S_m S_{m+1}| / \partial \theta_i \ (i = x, y, z)$.In my work, we adopted the following Equation (4.32) to define the value of $\boldsymbol{d}_k^i(t)$.

$$\boldsymbol{d}_k^i(t) = \left| {}_k^i P \times \frac{{}_k^i S_{m+1} - {}_k^i S_m}{\left| {}_k^i S_{m+1} - {}_k^i S_m \right|} \right| \tag{4.32}$$

In the Equation (4.32), k is joint indicator and i is the muscle index spanning joint k. Points ${}_k^i S_{m+1}$ and ${}_k^i S_m$ are effective origin and insertion coordinates of MTC i spanning joint k. ${}_k^i P$ is a vector from the joint to a point that is on the line of action of the MTC i.

4.3 Model Validation

Below, the performance of the developed MTC model is tested in a subset of individual muscle tasks[130]. For the selected tasks, the outputs of MTC unit are compared to the results presented in paper[85] that uses a virtual muscle model to validate its results. In the work, as stated previously, the developed muscle model is a MISO system of three inputs (neural excitation u, MTC length of L_{MTC}, MTC velocity of V_{MTC}) and one output (MTC unit force). Moreover, because the MTC velocity is the differential of its length, the muscle model is processed to be a system of two inputs and one output, that is, MTC velocity is directly calculated by its length differential.

In order to start the simulations, first of all, the model must be provided with all required parameters. The simulation takes the parameters of muscle "Vastus Lateralis (VL)" as an example, which are listed in the Table 0.1, Table 0.2, and Table 0.3, and they also present other needed parameters. For the module of activation dynamic, we let the minimum of the input u be 0.001 and set the initial output activation value of integration 0.001. And in addition, the time constants of ramping up and down of muscle activation are 0.01s and 0.04s, respectively.

Next, we import signals into the MTC dynamic system, and view the system response. For the selection of neuromotor inputs, there are two classic types[130]: (1) A 2 ms duration electric pulse that caused action potentials. The corresponding muscle response is called a twitch response by

physiologists, or an impulse response by engineers; (2) A high stimulation frequency (e.g., 100 Hz) that resulted in a maximal (saturating) ability to generate muscle force. The corresponding muscle response is called tetanus by physiologists, or a maximal step response by engineers). According to the introduction above, the work can executes three tests of six classic muscle tests defined in literature[130], whose outputs are MTC force, that is, (1) isometric twitch:input a = impulse, $V_{MTC} = 0$, $output\ F_{MTC}$ measured; (2) isometric tetanus: input a = max $step$, $V_{MTC} = 0$, $output\ F_{MTC}$ measured ; (3) max isokinetic: input a = max, $V_{MTC} =$ const, $output\ F_{MTC}$ measured.

The following will execute the three simulation experiments. The used models are constructed and implemented by manually changing the switches of the model shown in Figure 4.6.
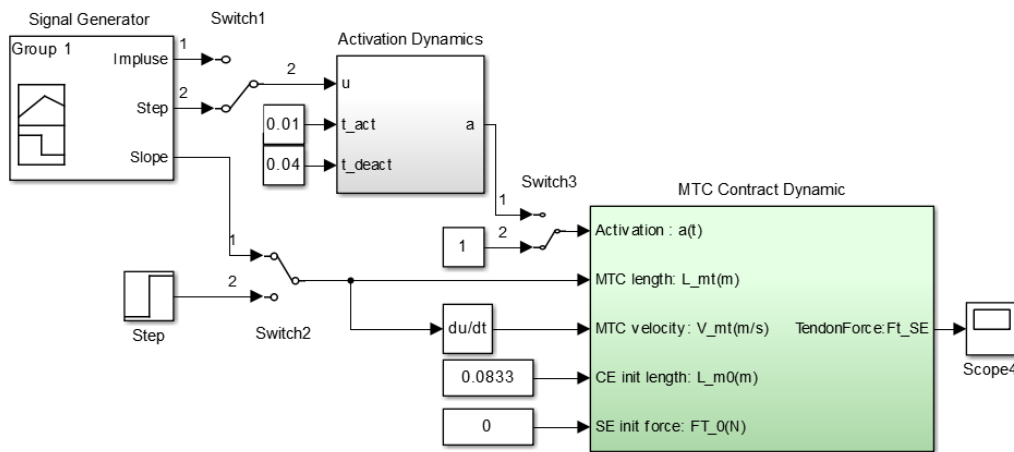
Figure 4.6 MTC dynamic simulation model

(1) Isometric twitch

Isometric is a special case of isokinetic[130] , namely the prescribed velocity is zero. In the test, the input u is an impulse signal and L_{MTC} is a constant. In order to run the simulation, we change the switch position of the model shown in Figure 4.6 according to Table 4.1.

Table 4.1 Switch position for every muscle test type

Test types	Switch position		
	Switch1	Switch2	Switch3
Isometric twitch	Up(1)	Down(2)	Up(1)
Isometric tetanus	Down(2)	Down(2)	Up(1)
Max isokinetic	Up(1) or Down(2)	Up(1)	Down(2)

As introduced above, the simulation adopts the muscle parameters of muscle "Vastus Lateralis (VL) ". According to Table 0.2, we know that its optimal muscle length is 0.084m and slack length is 0.157. Their sum is 0.241m. Therefore, the simulation set initially the length of MTC 0.24m, and make the length of tendon equal to slack length 0.157m so as to warrant the zero initial value

for tendon force. For the other initial input, i.e., active muscle fiber CE length, its value is found by evaluating the muscle-tendon path length equation (4.19). So we assign the initial CE element value as 0.0833 and initial tendon force as 0 N. The inputs and the results for the muscle task are shown in Figure 4.7. The isometric twitch task represents the force response of the muscle to a single neural activation impulse.
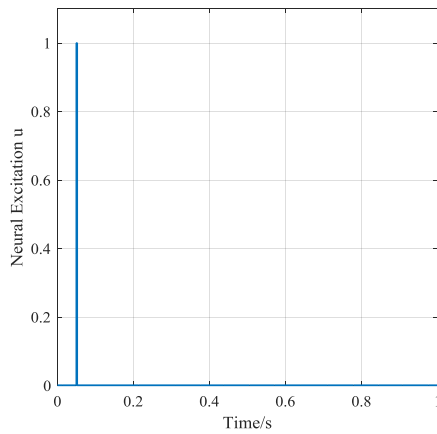
(a) Input signal of neural excitation that is a 2ms duration impulse signal that occurs at 0.05 seconds
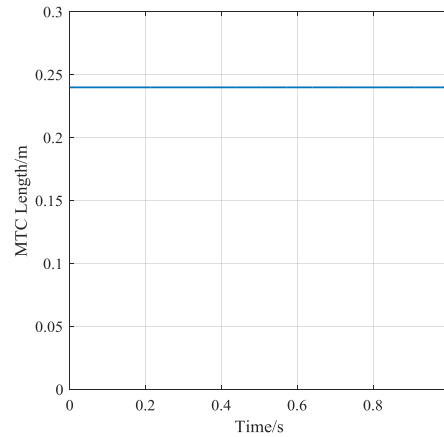
(b) Input signal of MTC length
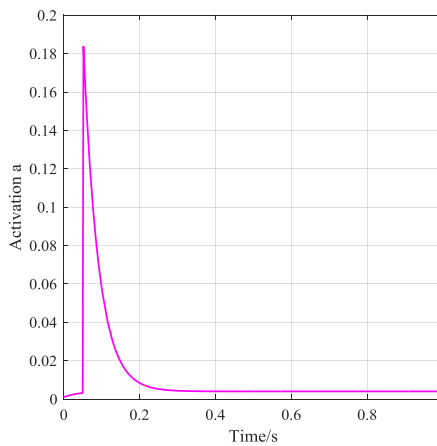
(c) A mediate state variable, activation signal that is the output of the activation dynamic module and is directly used as the input of the MTC contract dynamic module
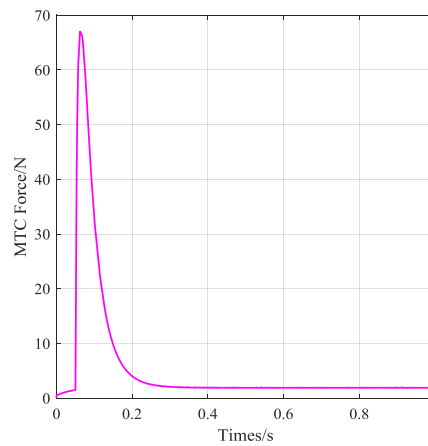
(d) Output of estimated MTC unit force

Figure 4.7 Input and output in isometric twitch test

(2) Isometric tetanus

The isometric tetanus task represents the ability of muscle to maintain a constant length despite producing a sharp increase in stiffness (i.e., output force)[85].In order to run this simulation, i.e., isometric tetanus task, we continue to change the switch position of the model shown in Figure 4.6 according to Table 4.1. For the isometric tetanus task, the neural activation input is the step

function that makes the transition from 0 to 1 at 0.05 seconds. Other parameters are same with implementing the isometric twitch task. Its input and output signals are depicted in Figure 4.8.

(a) Input signal of neural excitation that is a step signal

(b) Input signal of MTC length

(c) A mediate state variable, activation signal that is the output of the activation dynamic module and is directly used as the input of the MTC contract dynamic module

(d) Output of estimated MTC unit force
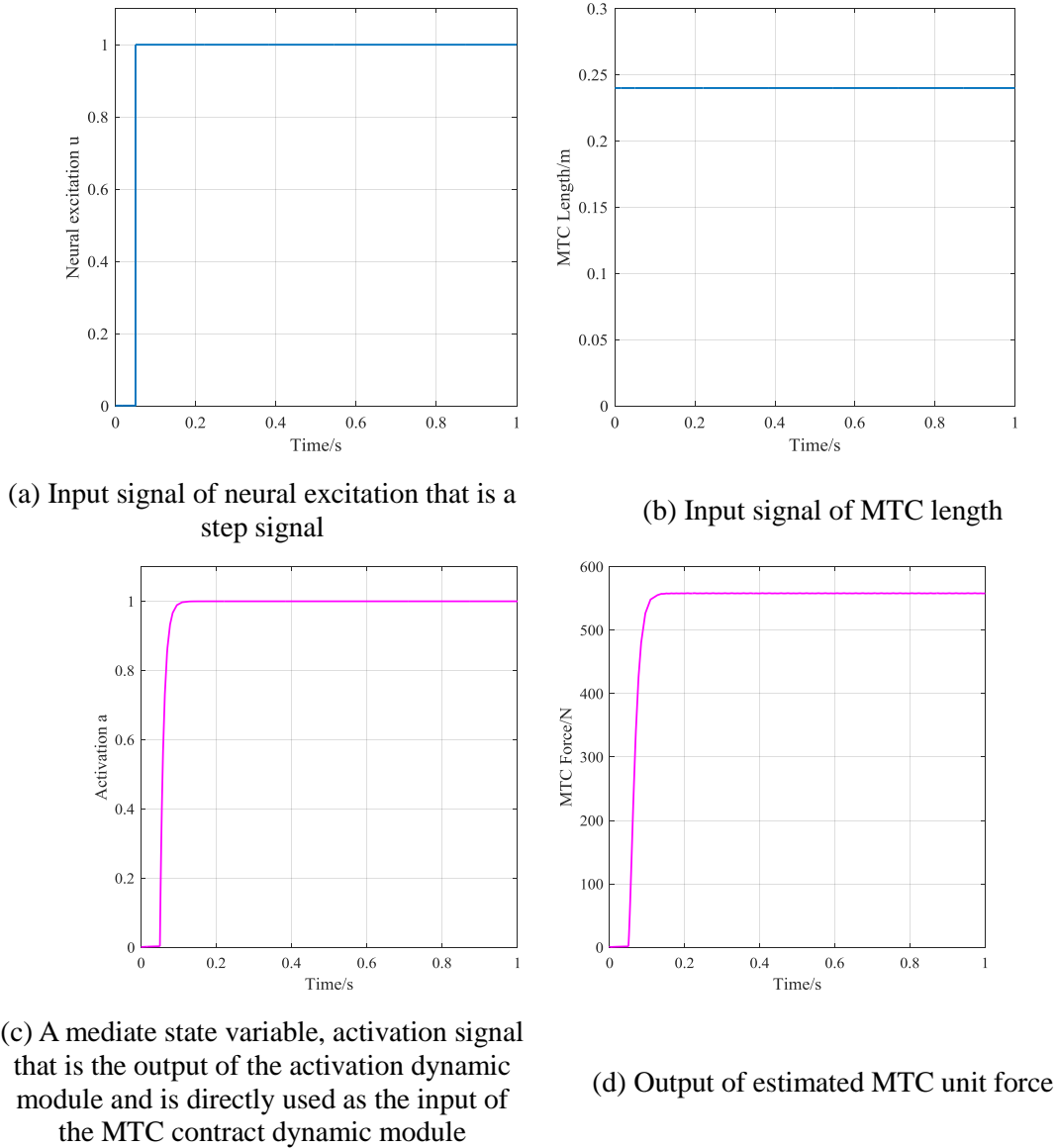
Figure 4.8 Input and output in isometric tetanus test

(3) Max isokinetic

The maximum isokinetic task represents the ability of muscle to change length smoothly throughout a movement task. The simulation results are depicted in Figure 4.9. The length input is the decreasing ramp; the neural activation input is set to unity throughout the task.

These derived results indicate smooth motion and MTC output forces. They are similar with the results presented in other reference such as reference [85] except for some differences that are maybe caused by adopting different simulation parameter values.
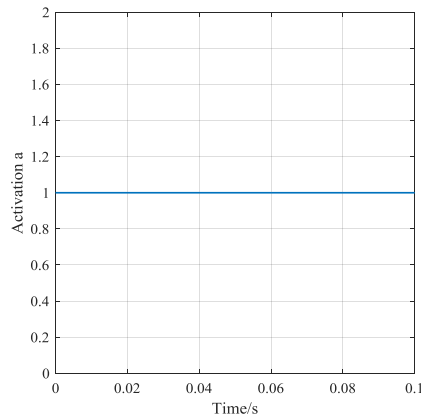
(a) Input signal of activation
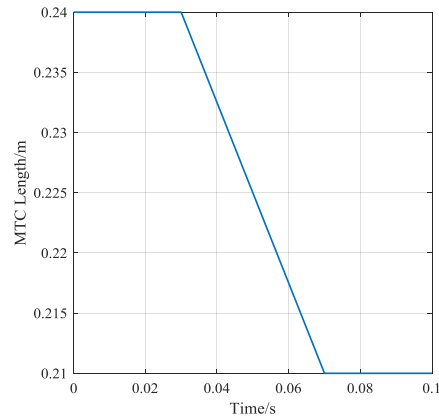
(b) Input signal of MTC length
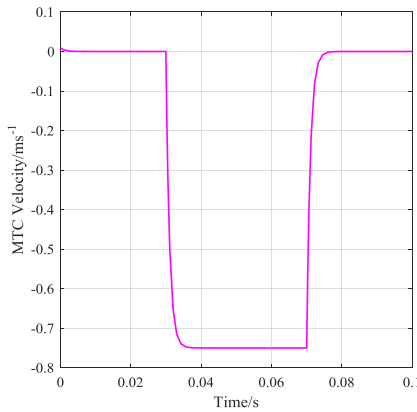
(c) A mediate state variable, MTC velocity that is derived from the MTC length and is directly imported into the MTC contract dynamic module
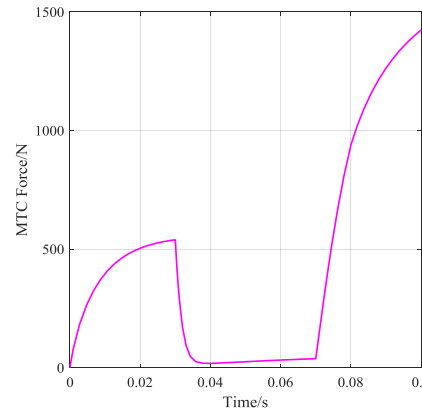
(d) Output of estimated MTC unit force

Figure 4.9 Input and output in max isokinetic test

4.4 Experiment

In the section, we mainly describe a fundamental experiment study in detail, which is to explore the relation between muscle activity and lower limb movement. The study adopts the sEMG to compare the relative activity intensity of muscle of interest during a particular single joint movement. To accomplish this task, the videos of joint motion are recorded to represent as the time sequence of the particular motion or position of joint. Through analysis for these data, we can explore their relationship between joint movement and muscular activity.

4.4.1   Experiment Preparation

(1) Selected Muscles and Position

To explore the relationship between the muscle activity and lower limb joint movement in the work, based on some pilot studies[131, 132], 21 muscles of interest of the right lower-extremity

as shown in Table 4.2 and Table 4.3, are included in my protocol to collect sEMG signals. They are grouped by muscle type, that is, mono-articular muscles and bi-articular muscles. Sites used for the placement (location and orientation) of the electrodes for each muscle are identified following the recommendations of the literature[49, 133]. Simultaneously, we record some corresponding motion physics quantities describing the studied motion.

Table 4.2 Investigated mono-articular muscles list

Joint Movement	Muscles	Abb	Channel	Action
Hip Joint	Tensor Fasciae Latae	TFL	CH01	Flexion
	Gluteus maximus	GMAX	CH02	Extension
	Gluteus medius	GMED	CH03	Extension
	Iliopsoas	IL	CH04	Flexion
	Sartorius	SART	CH05	Flexion
	Adductor Brevis	ADDB	CH06	Adduction
	Adductor Longus	ADDL	CH07	Adduction
	Adductor Magnus	ADDM	CH08	Adduction
Knee Joint	Gracilis	GRA	CH09	
	Vastus Lateralis	VL	CH10	Extension
	Vastus Medialis	VM	CH11	Extension
Ankle Joint	Soleus	SOL	CH12	Extension
	Tibialis Posterior	TIBP	CH13	
	Peroneus Longus	PL	CH14	
	Tibialis Anterior	TA	CH15	Flexion

Note: 1. Quadriceps comprised of the vastus lateralis, the vastus intermedius, the vastus medialis, and the rectus femoris are located at the front of the leg above the knee;

Table 4.3 Investigated bi-articular muscles list

Muscles	Abbreviations	Channel	Action(s)		
			Hip	Knee	Ankle
Rectus femoris	RF	CH16	Flexion	Extension	---------------
Biceps femoris long head	BFL	CH17	Extension	Flexion	---------------
Semitendinosus	ST	CH18	Extension	Flexion	-----------------
Semimembranosus	SM	CH19	Extension	Flexion	-----------------
Gastrocnemius medial	GSM	CH20	-----------	Flexion	Extension
Gastrocnemius Lateral	GSL	CH21	-----------	Flexion	Extension

Note: 1.Hamstrings(HAM) is a group of Biceps femoris long head, Semitendinosus and Semimembranosus;

    2.RF, electrodes were placed midway along a line between the anterior superior iliac spine and superior border of the patella

    3. BFL, electrodes were placed midway laterally on the posterior part of the thigh after having differentiated short and long heads by palpation of muscle contraction at the ischial attachment[99].

(2) Subject and Preparation

As described in Chapter 3, in order to eliminate some factors' influence on sEMG signal acquiring

as far as possible, the subjects are firstly grouped into several groups. For the signal sampling in the studies of the work, all testing was done at the Center of Intelligent Information of Nagasaki Institute of Applied Science. Five healthy males' students volunteered as subjects, and they are in the same group division ranged from 20 to 29 years (average age 26.8 years), which was determined by time limitations and number of willing volunteers. Prior to participation, informed consent was obtained from all subjects. For the preparations of sEMG data sampling, in addition to the content presented in section 3.2, some other preparations must be done for subjects in advance of the experiment. For example, we must verify electrode placement and adjust amplifier gain settings by monitoring the sEMG signals on a monitor. For the preparation of video signals acquisition used in the first study, a total of 6 reflective markers of an optical camera system were used to collect time histories for selected segment points which were used to determine sagittal-plane motion of the lower extremity. Up to now, all equipment were ready to collect data. Make sure all switches of all data acquisition equipment are turned on and each system working properly. Then begin to collect data by experimental protocol of particular study content as follows.

(3) Equipment and Configuration

According to the objective of this study, the experiment needs to record synchronized joint movement projected into the sagittal or walking plane and sEMG signals. In the experiment, they were collected by different acquisition systems shown in Figure 4.10, so a common time event has to be identified on the recorded signals in order to synchronize and divide them. The acceleration signal is also recorded as an auxiliary reference signal. One of main drawbacks of this method may be existence of time delay at each of the systems. Thus it should be able to adjust finely the delay time manually to make the signals as synchronous as possible. In our system, in order to adjust finely the delay time manually during data processing of original signals, several other referred physical signals were also collected at the same time in the experiment, that is, the force plate pressure data and acceleration signal.

The Figure 4.10 is the composition diagram of the whole experimental system. The figure shows the sensors configuration of the lower part of the body projected into the sagittal plane. The system numbered ③ shows the sEMG acquisition part, and the figure above line is a screenshots when sEMG is sampling. The system numbered ① that is developed in Micro visual studio 2010 using C++ language, can be used to capture forces and acceleration signals. Part numbered ② is the system to capture video. The instrumentation recording the movement video adopted optical camera system. Markers were located at the greater trochanter (hip), lateral femoral condyle (knee), and lateral malleolus (ankle) on leg. Videos of the experiments were captured using a PC

installed SkillCapture (v2.1.0, published by Video4Coach) software. The software is an automated system which does video capture, display, compression and upload. The video data was captured at 30 Hz and directly recorded. For main settings of SkillCapture software, it includes: (1) the prepare time was set to 5 sec that is a waiting time from pressing toggle key to automatically activating video capturing; (2) the initial capture time was set to 30 sec that is video capture time, however, the video captures ware sometimes terminated manually based on studied action.
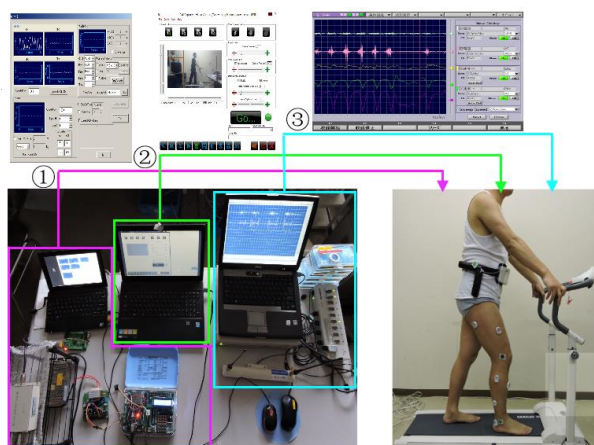
Figure 4.10 Experimental apparatus of our motion capture system

4.4.2   Protocol of Data Collection

The whole process of data collecting was implemented under the control of an examiner denoted by "CmdSender". Following an explanation of the experimental procedure, the subject participating in the data sampling was instructed to hold erect in the neutral posture against the gravitational field and completely shifted his weight onto the left lower extremity prior to each repetition so that the right foot was free to move and was just in contact with the floor without bearing weight. The exercises chosen in the study were flexion-extension exercise of single hip joint, knee joint and ankle joint, which are labeled as Hip-FE, Knee-FE and Ankle-FE. The all data collection was performed with subjects standing, and seven same action repetitions of each exercise were contained and called an experimental session. For example, when measuring the data of Hip-FE exercise, the subject was required to move hip joint in the sagittal plane while maintaining parallel with the plane.

Subjects were given verbal commands to execute one try among three exercises, and the start and end of the capture were also indicated through voice commands by the investigator. The subject keep static state while the "CmdSender" saw the sEMG displayed on a computer monitor in real time. When recordings kept steady for 3s, the "CmdSender" sends a starting command in verbally

guided fashion and presses simultaneously shortcut key set to "A" to activate "SkillCapture" software to run the countdown. With the predefined prepare time 5-second countdown to the end, the software produce a sound, and then video acquisition system is run automatically, while the recording operation of sEMG signal and acceleration data, and the action video of the subject was also executed at the same time. In the stage, if sEMG or acceleration signal artifact was detected during a trial, attempts were made to improve signal quality and the trial was repeated. Typically, the aberrant signals were easily identified as randomly occurring, large amplitude, low-frequency waveforms presumably resulting from movement artifact. If the problem could not be corrected, this experiment was invalidated and did again. Subject goes through each of exercises for ten times, followed by a return to the neutral position. In this way, the sEMG tracings of the corresponding movement can be easily recorded. All data were stored on disk for later analysis. The format of video is "*.avi" and all other signals are saved in "*.txt" file in the experiment.

4.4.3   Data Analysis and Results

(1) Initial Data Processing

The main task of the section is initially to process the data collected before, so as to get the simultaneous data of kinematic data and sEMG signal. As was previously introduced, the measurement system of human joint motion adopts optical camera system and the software exported the video data as "*.avi" format file. In the study, the joint motion is mainly represented by joint angle time serials. So one of the main tasks is to derive joint-angle trajectories from motion capture video. The sagittal plane is selected to calculate joint angle in the study, because the joint angles in the plane are more robust to derive from marker data than complete, full degree of freedom joint angle specifications. To complete the task, a video-based motion analysis program, "DigitizingTools" Matlab Tools (version updated on March 3 in 2015) [134], is adopted to export markers' position coordinate and save them in "*.xlsx" format files. Then a script was written in Matlab (MathWorks, 2012a, 64-bit, windows) to calculate angles from markers' coordinate. Their definitions of zero, positive and negative degree position are illustrated in Figure 2.3 (lateral view).

Next, Mokka0.6.2 tool was adopted to visualize and process data, including the adding event to signal, adjusting synchronization time among signals and so on. It is an open-source and free motion analysis GUI application and supports many biomechanical data files (i.e. *.C3D files). In the study, the data processing procedure included: (1) create a *.C3D format files and all data including sampled raw data and processed data were saved into the file, which was implemented with the aid of Biomechanical ToolKit (BTK) 0.3 toolbox in Matlab[135], which is an open-source and cross-platform library for biomechanical analysis; (2) import the created c3d file into

Mokka.6.2 to edit signal by visual method, which is shown in Figure 4.11. When finishing the process, the modified file was exported and prepared for using in the next step; (3) the remainder procedures were done by writing Matlab scripts, which included the data extracting by events, sEMG signal processing (mainly including operations of removing DC, rectification, filtering, smoothing data), normalization and so on.
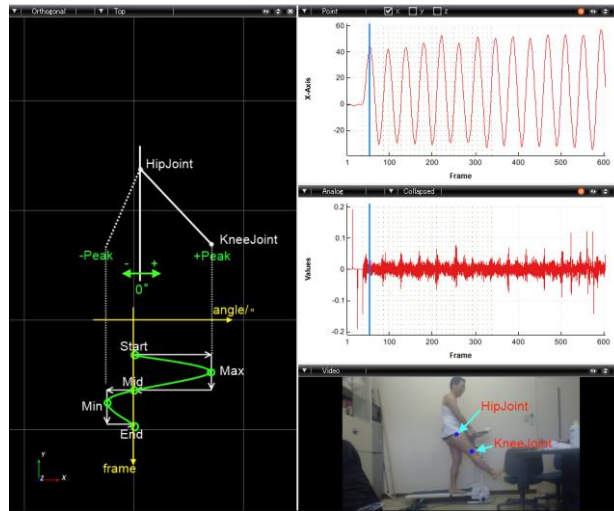
Figure 4.11 Processing of signal collected during Hip-FE exercise in Mokka0.6.2

(2) Data Reduction

The following takes the processing of signal collected in Hip-FE exercise as an example to illustrate the method presented in the study in detail. The Figure 4.11 is a part of processing interface of Mokka0.6.2 software. The figure at the upper right shows the hip joint angle curve with respect to time (or frame), the figure at middle right is original sEMG signal of a muscle, and a real-time video is also displayed at he same time. The entire process was executed in Matlab script, which is illustrated in Figure 4.12. Five valid cycle phases of the exercise were chosen based on signal quality and consistency of performance, which was indicated by total ROM and velocity. Events were marked on the basis of joint angle and its position in each cycle. For every cycle of movements, five events were selected and added to the signal that was used as the signage to divide signal. The naming of event was to combine the signage that is five critical points illustrated in Figure 4.11 with the ordinal of movement cycle, just like "CycleN_Pos" (N=1, 2,3,4,5; Pos= "Start", "Max", "Mid", "Min", "End"). In the stage of signal division, the main task is just to divide signals based event names, and save the pieces of data in corresponding data field whose names are given as "FromStart2Max", "FromMax2Mid", "FromMid2Min" and "FromMin2End".

Start

Calculate angles from motion video

Creating *.c3d file and Saving data

pf=30;FreAnalog=1000;
ADCres=16;
ratio=floor(FreAnalog/pf);
afn=nImageFrames*ratio;

Import *.c3d file into Mokka0.6.2 Software

Add event markers

Adjust delay time

......

Export *.c3d file and Import it into Matlab

Divide data and get synchronous signal

1.Data dividing is based on event marks ;
2.Divided signals are stored in four structures called
   FromStart2Max,FromMax2Mid, FromMid2Min and
   FromMin2End.Data reading format are:
Example:
   FromStart2Max.Angle{j}
   FromStart2Max.AllMuscleList{i}.Original{j}
   FromStart2Max.AllMuscleList{i}.ProcessedMethod{j}
   …
   i=1,2…21;j=1,2…5
   ProcessedMethod=Method of Filter,Rectification...

sEMG signal processing (Rectification, Filtering...)

Peak and Time Normalization

End

Figure 4.12 Steps of initial data processing

As the name implies, the data saving in "FromStart2Max" field is the data segment between events "CycleN_Start" and "CycleN_Max". In a similar manner, other data fields save the according data segment. In order to save and access various stages' data easily, the work organized the data as the following structure.

SecNames. AllMuscleList{i}. ProcessedMethod

"SecNames" is one of "FromStart2Max", "FromMax2Mid", "FromMid2Min", "FromMin2End"; AllMuscleList is a cell matrix, which saves all muscles of interest and its number is 21, i.e., i=1, 2, …, 21; "ProcessedMethod" is changing according to the methods adopted during data processing; Every cycle data segment is saved in separate, i.e., j=1, 2, …, 5.

In the stage of sEMG signal processing, processing operations mainly included removing the mean, rectification, filtering, smoothing processing. In the study, "Band pass" filter is adopted with [20 450] Hz to remove the lower and higher frequencies signal of given cutoff frequencies, and filter order is 4; Signals were full wave rectified in the study. The amplitude and frequency characteristics of the raw sEMG signal have been shown to be highly variable and sensitive to many factors including extrinsic and intrinsic factors according to literature [136], so it is not easy to use sEMG to study muscle activity, especially to make comparisons between sEMG values

obtained from identical muscles in different subjects, different muscles from the same subject, or even the same muscle from the same subject on different days. To overcome these shortcomings of sEMG, the sEMG should be normalized. Each subject's sEMG values for each muscle were subsequently normalized as a percentage of the highest mean value in any three exercises .The ensemble averages were calculated in three time intervals, that is, the whole interval and the before and after a quarter part of each interval. This was done to better characterize the pattern of response of each muscle during the execution of these exercises. In our view, if between-group differences did exist, those differences would be most apparent at those times when the greatest demands were placed on the muscle. Such differences might otherwise be obscured if only data over the entire period of the exercise were analyzed.

(3) Data Analysis and Results

We first perform the study of reliability of repeated measurements, which is frequently quantified in the movement sciences literature[137]. When we speak of reliability, we refer to the "test-retest" repeatability or reproducibility of a measurement or variable. The measurements could be from two people (or two types of equipment), or the same person on two, or more, occasions. The study mainly uses the technique of Intraclass Correlation Coefficient (ICC) to explore the variability of the data to gauge reliability. The work used IBM SPSS Statistics 20 for Windows to do the calculations.

Table 4.4 Mean of hip joint angle in each of divided motion interval/degree

Index	FromStart2Max			FromMax2Mid			FromMid2Min			FromMin2End		
	Subj1	Subj2	Subj3	Subj1	Subj2	Subj3	Subj1	Subj2	Subj3	Subj1	Subj2	Subj3
1	20.8	20.4	20.0	35.1	33.1	35.7	-9.0	-9.5	-9.6	-19.1	-19.2	-19.6
2	21.0	21.5	21.4	28.6	26.2	29.6	-18.6	-16.8	-17.0	-21.1	-21.7	-22.7
3	20.4	20.4	20.2	30.6	31.2	28.7	-6.5	-6.4	-6.3	-18.7	-19.2	-19.3
4	20.6	20.1	19.8	32.8	30.6	31.4	-19.0	-17.0	-16.6	-22.3	-21.0	-21.2
5	21.4	20.8	21.2	37.7	36.1	36.3	-12.0	-13.5	-13.6	-18.2	-18.6	-19.8
6	22.2	21.8	22.0	30.1	32.2	30.1	-12.3	-12.4	-12.1	-22.3	-25.3	-21.5
7	21.3	21.2	20.8	34.4	30.6	32.2	-12.5	-12.0	-12.5	-18.3	-19.6	-19.6

In the work, joint motion is mainly described by joint angle time sequence, and the joint angle is not directly measured, but calculated by processing of motion video. Therefore, to estimate the measures' reliability for the method used, we computed the joint angle variation tested many times within subject. This measure of reliability is simply the change in the mean value between 7 trials of a test. Data from the mean of all movement joint angles during defined movement stage listed three participants, are recorded from columns 2 to 13 in Table 4.4. A comparison of the reliability of measurements from three subjects was performed. The processed data is listed in Table 4.5.

Overall, it appears that Subj1 measures slightly higher and more stable than other two subjects (see means & standard deviations in Table 4.5).

Table 4.5 Statistics of hip joint angle

		Mean	Std. Deviation	N
Subj1	1	21.100	0.6028	7
	2	32.757	3.2046	7
	3	-12.843	4.6039	7
	4	-20.000	1.8448	7
Subj2	1	20.886	0.6336	7
	2	31.429	2.9993	7
	3	-12.514	3.7905	7
	4	-20.657	2.3223	7
Subj3	1	20.771	0.8118	7
	2	32.000	2.9676	7
	3	-12.538	3.7594	7
	4	-20.522	1.2849	7

An ICC is measured on a scale of 0 to 1.1 represents perfect reliability with no measurement error, whereas 0 indicates no reliability. ICC is commonly used to assess test–retest reliability and reflects the relative reliability of a measurement. ICC >0.75 is considered excellent, 0.40~0.75 is

Table 4.6 Intraclass correlation coefficient

		Intraclass Correlation[b]	95% Confidence Interval		F Test with True Value 0			
			Lower Bound	Upper Bound	Value	df1	df2	Sig
Single Measures	1	0.854[a]	0.570	0.970	18.573	6	12	.000
	2	0.811[a]	0.476	0.961	13.873	6	12	.000
	3	0.965[a]	0.879	0.993	84.715	6	12	.000
	4	0.725[a]	0.317	0.940	8.929	6	12	.001
Average Measures	1	0.946[c]	0.799	0.990	18.573	6	12	.000
	2	0.928[c]	0.731	0.987	18.873	6	12	.000
	3	0.988[c]	0.956	0.998	84.715	6	12	.000
	4	0.888[c]	0.582	0.979	8.929	6	12	.000

Two-way mixed effects model where people effects are random and measures effects are fixed.
a. The estimator is the same, whether the interaction effect is present or not.
b. Type C intraclass correlation coefficients using a consistency definition-the between-measure variance is excluded from the denominator variance.
c. This estimate is computed assuming the interaction effect is absent, because it is not estimable otherwise.
d. The numbers of 1,2,3 and 4 represent the reliability test in according motion interval of "FromStart2Max", "FromMax2Mid", "FromMid2Min" and "FromMin2End" among Subj1, Subj2 and Subj3, respectively;

regarded as fair to good, and 0~0.4 as poor. Table 4.6 shows information relating to the ICC calculations. Use the "Single Measures" option, as individual values are collected. Our estimated

reliability is ICC (0.725, 0.965) with 95%. It shows that statistical analysis of the ROM data provided by processing for joint motion video revealed no significant differences across exercises.

To furtherly position muscle activity place during action region, we continue to do the following pre-processing for the smoothing signals obtained from sEMG processing, that is, each action region such as "FromStart2Max" are further divided into three sub-region. The three small regions were named after "1/3 before", "1/3 after" and "All data". In addition, the sEMG values for each muscle during each test were normalized as a percentage of the maximum sEMG value obtained from the processed signals during the activity in each muscle for each individual separately. Moreover, considering that the different duration of gait cycles while averaging or comparing sEMG data from several different gait cycles, time normalization is done for all muscles of interest. Finally, to reduce the amount of processed data, we then computed the feature quantity that is used to represent the muscle activity intensity at these small region. In the work, the mean value calculating were used as the feature quantity. The results related with Hip-FE exercise are listed in Table 0.4. From the computing table, we can conclude some interesting information. Table 4.7 list the relative intensity of muscle activity.

Table 4.7 The sequence of muscle activity intensity in Hip-FE exercise

Hip flexion	FromStart2Max	TensorFasciaeLatae>GluteusMaximus>Iliopsoas> Rectusfemoris> Semimembranosus> Semitendinosus>Bicepsfemorislonghead>GluteusMedius>AdductorLongus >AdductorMagnus>AdductorBrevis>Sartorius
	FromMin2End	Gluteus Maximus> Rectusfemoris> TensorFasciaeLatae> Iliopsoas> Semimembranosus> Semitendinosus>Bicepsfemorislonghead>AdductorLongus>GluteusMedius > AdductorLongus> GluteusMedius> Sartorius> Adductor Brevis
Hip extension	FromMax2Mid	Rectusfemoris> GluteusMaximus> Semimembranosus> Semitendinosus> Iliopsoas> TensorFasciaeLatae >Bicepsfemorislonghead> AdductorLongus> GluteusMedius> AdductorBrevis> AdductorMagnus> Sartorius
	FromMid2Min	GluteusMaximus> Rectusfemoris> TensorFasciaeLatae> Iliopsoas> Semimembranosus >Semitendinosus> AdductorLongus> Bicepsfemorislonghead >GluteusMedius> AdductorBrevis> AdductorMagnus> Sartorius

For our work, to simplify the complexity in the construction of control model, we only adopted several muscles with largest activity intensity. And they will selected based on joint and Flexion-Extension movement, not continue to go to smaller joint movement, such as "FromStart2Max", which will be considered in the future. For the muscles selection related with the other two joints, we can accomplish the process in the similar way above. Finally, the work selected the following muscles to study lower limb motion, as shown in Table 4.8. In some analysis, because of the consideration that the TFL muscle is related with the muscle GMAX in function and structure and is continuous with the iliotibial tract, which attaches to the tibia, TFL is often eliminated.

According to Wolkotte [84], these selected muscles are several main muscles during walking.

Table 4.8 Selected muscles of interest

Joint	Adopted Muscles		
	Mono-articular	Bi-articular	
Hip	Gluteus Maximus (GMAX) Tensor Fasciae Latae (TFL)	Rectus Femoris(RF) Biceps Femoris Long head (BFL)	-------------------------------
Knee	Vastus Lateralis(VL)		Gastrocnemius Medial(GSM)
Ankle	Tibialis Anterior(TA), Soleus(SOL)	----------------------------	

4.5 Conclusion

To build a control scheme for a rehabilitation robot based on NMS system, the chapter derived the dynamic model of a Hill muscle structure, which was based on but different from the original Hill-type model. It included activation dynamic and contraction dynamic. Simultaneously, we also studied some muscle quantitative properties by the simulation method, which could easily study the relation between parameters. To validate the correctness of developed model, the performance of the developed dynamic model was tested in a subset of individual muscle tasks, and by comparing with the results presented in other references, the correctness is validated. At the end of the chapter, an experiment is performed to study the relationship of muscle and movement. The fundamental study is helpful in reducing the computational effort for future embedded applications so as to optimize the system executing.

Chapter 5. Modeling of Human-Robot System

In this chapter, we build a comprehensive motion analysis model including kinematic and kinetic analysis of Human-ERRobot system. The chapter mainly gives a full description for the modeling of biomechanical model of the human body, and the ERRobot is processed as a force provider. The biomechanical model here mainly refers to the musculoskeletal model comprised the skeleton model and the MTC model .The MTC model has been described in Chapter 4 and the created moment is added into relevant skeleton model that is modeled as an articulated multibody system. Figure 5.1(a) shows three major planes dividing the human body in certain areas, and Figure 5.1(b) explains the most important medical directions used within this work. The sagittal plane is the only plane of symmetry.

(a) Three major planes   (b) Anatomical directions
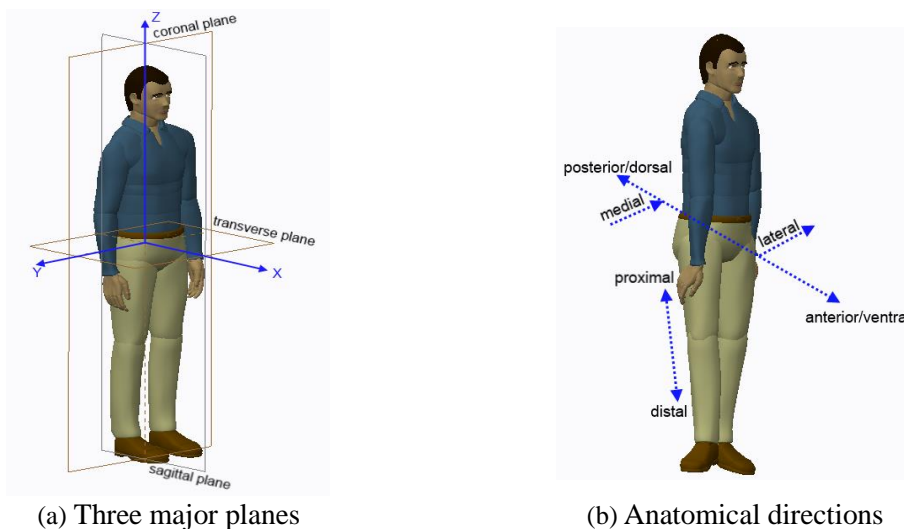
Figure 5.1 Anatomy planes and directions within the human body

5.1 Constructing Linked Segment Digital Human Model (LS-DHM)

In many studies, most of body simulation models are modeled as a mechanical system that includes link lengths, mass moments of inertia, joint torques, and external forces and so on. Given certain reasonable assumptions, the biomechanical analyzing of human can be completed by the methods used to study multi-body system[14].Therefore, we also simplify the human model as a collection of rigid bodies (segments) linked together, called LS-DHM. During the modeling, for the sake of generality, we firstly construct a template human model including 19 segments and 46 DOFs (43 joint DOFs and 6 root DOFs), and then configure it to a specific analyzing model when processing a specific study.

5.1.1   Reference Frame and System State Vector

The work mainly focuses on processing the multi-body system of tree-structure, because the

topological structure of human model presented in the work is a tree-structure, as shown in Figure 5.4. Before presenting the construction method of Reference Frames (RFs), we introduce firstly two kinds of RFs for analyzed rigid system.

- World Reference Frame (WRF): It can describe the relationship between rigid-system and around environment. The frame is an inertial reference system, where we measure the absolute motion. In general, it is fixed with "Ground".

- Local Reference Frame (LRF): Any rigid body motion description can be represented by the motion of RF fixed in the body that is called body-fixed RF, which is a special LRF. The motion we measured in LRF is referred to as the relative motion.

All RFs can be represented in the form of $Name(index)$. The value of "$Name$" can be RF, WRF, or LRF. For the "$index$", it can be following two possible values in the paper: (1) a character (or string) identifying a specific frame; (2) a number, when it may express the index-th RF or a special reference system with $<index>$ identifier.

Both of WRF and LRF are defined as an orthonormal cartesian coordinate system. In the paper, we denote three unit orthonormal basis coordinate vectors of RF as [$^i e_x$, $^i e_y$, $^i e_z$], where $^i e_x = [1\ 0\ 0]^T$, $^i e_y = [0\ 1\ 0]^T$ and $^i e_z = [0\ 0\ 1]^T$. The left upper superscript shows which reference frame it belongs to. For example, the orthonormal basis of WRF is denoted by $^w e_x$, $^w e_y$, $^w e_z$ or e_x, e_y, e_z and a LRF, for example $\sum A$, is expressed in $^A e_x$, $^A e_y$, $^A e_z$. As a general processing, the following describes the principles for coordinate setting used in the work, as shown in Figure 5.2. It is a general method to be applicable to general multi-body system analysis. Combined with the Figure 5.2, there are several main notes:

(a) In order to model and process data conveniently, the ground is processed as the segment same with other rod segments in the work, and is broken into static ground (SG) and virtual dynamic ground (DG). The segment-fixed RF with SG and DG are expressed as $\sum SG^i$ and $\sum DG^i$ are, and right upper index i is used to differentiate different RFs in the same segment;

(b) Each segment is given a unique Id that is used to refer to the particular body, and the Id of SG segment must be 1.That is because we are to judge whether or not a RF is the global coordinate system, i.e., WRF based on the Id value;

(c) The closed curve plotted by dotted line expresses segments and dot is the joints connecting segments. Both of their index numbers are from 1 and make the serial number of joint always equal to its "Child Segment Index" that is the segments far away from the base. This treatment is helpful in segment or joint retrieving, and programming;
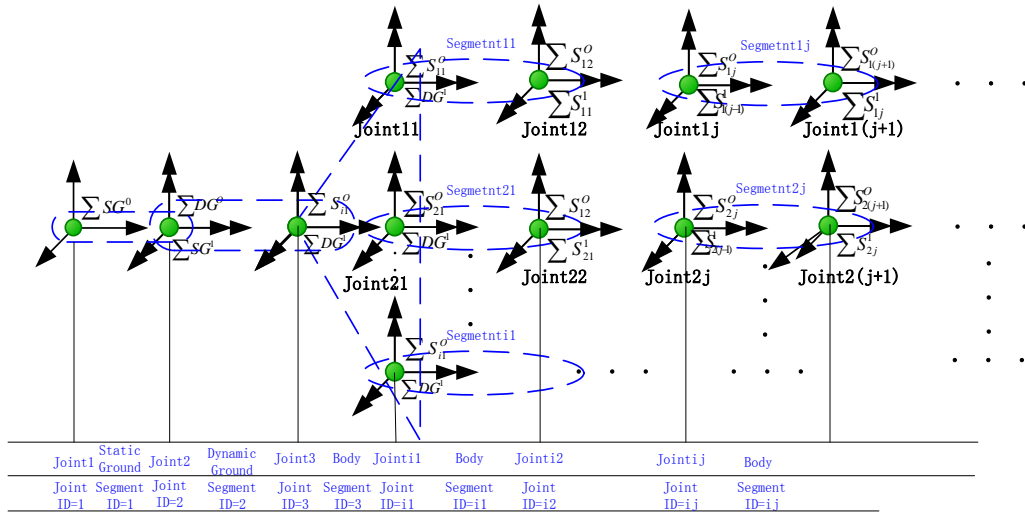
Figure 5.2 Method to set the reference system fixed in segment

Please note another aspects of joint and segment for id setting. Like the naming method shown in the Figure 5.2, we can calculate the numbers of sub-branch, and identify them. Besides of them listed above, we can also standardize some other rules based on different needs, such as, the setting of RF origin positions. To derive the systematic analysis method for multi-body system, definitions of two other types of RF are presented firstly, both of which are considerable in my work.

- Mom Reference Frame (MRF): It is a special body-fixed RF that is defined as the relatively forward position in a movement transformation sequence. And call the rigid body in which MRF is fixed as "Mom Segment". Both of them are relative to "Child Reference Frame (CRF)" and "Child Segment" defined in the next. In equation, it is denoted by $\sum Mom$ $(or\ \sum M)$, respectively.

- Child Reference Frame (CRF): It is also a special body-fixed RF that is located at the relatively rear position in a movement sequence. And also call the rigid body in which the moving RF is fixed "Child Segment". Both of them are also relative to MRF and "Mom Segment" defined above. In equation, it is denoted by $\sum Child$ $(or\ \sum C)$, respectively.

For the sake of generality, consider a pair of adjacent Mom-Child body-fixed frame, whose orthonormal basis are denoted by $\left[{}^{M}e_x,\ {}^{M}e_y,\ {}^{M}e_z\right]$ and $\left[{}^{C}e_x,\ {}^{C}e_y,\ {}^{C}e_z\right]$, respectively. The whole systematic analysis is based on the studying of the transformation between two systems of $\sum Mom$ and $\sum Child$, which is given next. In the coordinate system, each joint i can have as many as six DOFs (i.e., possible coordinates), as shown in Equation (5.1). Its first three elements are translational DOFs and the other three elements express rotational DOFs. And

besides, the method to set RFs in the work defines three special joints. They have DOFs specified in advance, that is, SG-SG (0 DOF), SG-DG (3 translational DOFs) and DG-RS (RS is used to denote "Root Segment"; 3 rotational DOFs) joints, as shown in Table 5.3.

$$[Dof]_i = [Tx\ Ty\ Tz\ Rx\ Ry\ Rz]_i \tag{5.1}$$

In my work, the joint DOF is configured by locking (assigning 0 at corresponding DOF position) or opening (assigning 1 at corresponding DOF position) some DOFs, so the three special joints above have the following DOF configuring Table 5.1. Other joints can use the same method to get needing DOFs.

Table 5.1 DOF configuring table of three special joints

Joints	Tx	Ty	Tz	Rx	Ry	Rz
SG-SG	0	0	0	0	0	0
SG-DG	1	1	1	0	0	0
DG-RS	0	0	0	1	1	1

If we use notation $q_i(t)$ to represent the joint-i displacement (translation and rotation) at time t and $u_i(t)$ is the velocity at the time. Furthermore, given that the distances between two origins Mom RF and Child RF along three axis are Tx, Ty, Tz, and rotational angles around three axis are Rx Ry Rz, then they will be expressed as the following Equations (5.2) (5.3).

$$q_i(t) = [Tx\ Ty\ Tz\ Rx\ Ry\ Rz]_i(t) \tag{5.2}$$

$$u_i(t) = \dot{q}_i(t) \tag{5.3}$$

In order to process conveniently, the concatenated list of displacement and velocity is defined as the state vector, as shown in Equation (5.4).

$$S_i(t) = [q_i(t), u_i(t)]^T_{\ i}(t) \tag{5.4}$$

For the whole system, the state vector $S(t)$ is the concatenated list of displacement and velocity of all single joints, and all translational DOFs precede any rotational DOFs. Besides of the information of displacement and velocity, we can also add some other additional useful variables, such as variables saving intermediate results of interest. For simplicity sake, the dependence on t is omitted from these equations below.

5.1.2 Human Model Simplification and Modeling

(1) Creating a Template Model of Total Body

In this section, a 3D whole body LS-DHM is constructed to be able to simulate human motion as many as possible,

● Modeling Motion Linkage Between Stone and Joint

The modeling is based on the model presented in literatures [138-140]. Total body LS-DHM is presented and illustrated in Figure 5.3. Their corresponding joint type and DOFs are shown in the figure with WRF, that is, the X-axes points anteriorly (progression direction), the Y-axes points
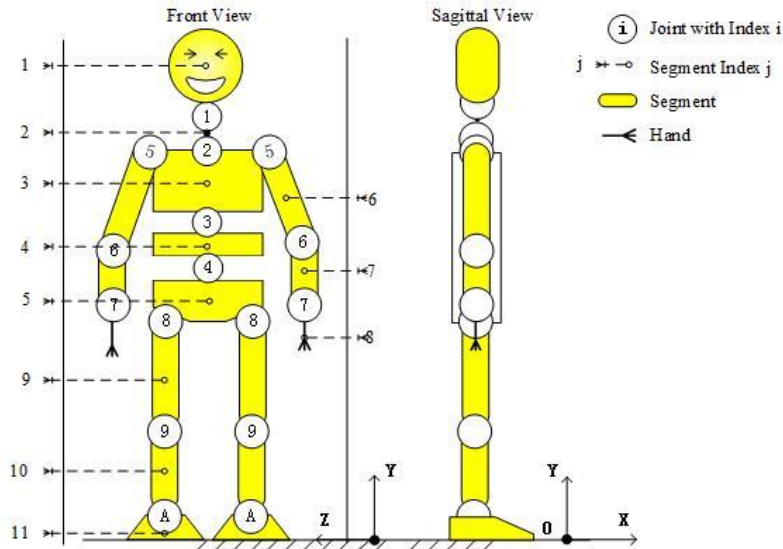
Figure 5.3 The method of human body decomposition and labeling: Circles with numbers represent kinematic joints. Their corresponding joint type and DOFs are shown in Table 5.2

superiorly (vertical direction), and the Z-axes points to the right (laterally for the right leg). The model is used as a template model to configure the specific analyzed model. In Figure 5.3, circles with numbers represent kinematic joints. Their joint type and DOFs are shown in Table 5.3. Figure 5.4 shows the model's logical structure. It is obvious that LS-DHM is a tree structure.
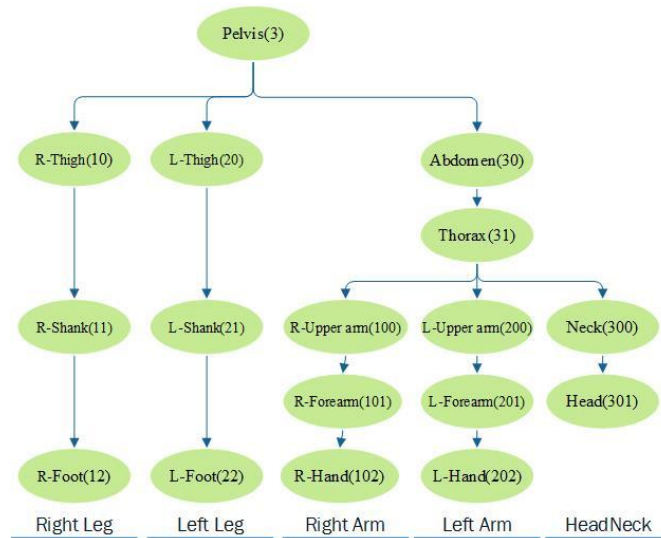
Figure 5.4 The method of human body decomposition and labeling: the logic structure diagram (the numbers in parentheses are index Id used in computer modeling and searching)

During the modeling, we select pelvis as the root segment whose parent is DG and the parent of

DG is SD that is the position to fix WRF. From Table 5.3, we can know that pelvis has 6 DOFs relative to WRF.

It is obvious that human model is comprised of three main branches, and some of them include some child branches in the body frame. Every segment is labeled in Figure 5.3 and its corresponding index Id used in searching by computer is listed in Table 5.2. The first branch is

Table 5.2 Segment information

Name	Segment Label	Index Id in program
Static ground(SG)	S0	1
DG	D0	2
Head	1	301
Neck	2	300
Thorax	3	31
Abdomen	4	30
Pelvis	5	3
Upper arm	6R/6L	100/200
Forearm	7R/7L	101/201
L/R Hand	8R/8L	102/202
L/R Thigh	9R/9L	10/20
L/R Shank	10R/10L	11/21
L/R Foot	11R/11L	12/22

Notes:
1) HAT=:(HeadNeck=1+2,Arms=6+7+8,Trunk=3+4+5)
2) L/R : left and right

the right leg labeled in 1N (N=0, 1, 2), the second is the left leg labeled in 2N (N=0, 1, 2), and the third is the spine labeled in 3N (N=0, 1). In the spine branch, there are three child branches that are named after "Right arm" labeled starting from 100, "Left arm" labeled starting from 200, and "Neck, Head branch" labeled starting from 300, respectively, as can be seen in Figure 5.4. Each branch has a starting local frame that differs from its parent branch. Based on Table 5.2and Table 5.3, it is known that the template LS-DHM comprises 19 segments and 46 DOFs.

For the joint index processing, we start joint Id from 1 in programming .This joint is a special joint formed from SG self and its DOF is zero called as weld joint. The benefits of this treatment is that it can make the serial number of joint always equal to its "Child Segment Index", which is helpful in segment or joint retrieving, and programming.

Let us assume that the model is characterized by a vector q of m dependent coordinates. According to Table 5.3, the total LS-DHM system having 18 joints of nonzero DOF and 46 DOFs that would correspondingly have 46 generalized coordinates expressed in q vector whose

expression is Equation (5.5).

$$q = \left[(q^1)_3, (q^2)_3, \cdots, (q^k)_{dof(k)}, \cdots, (q^{18})_3 \right]^T, k \in [0,18], dof(k) \in [0,6] \qquad (5.5)$$

Table 5.3 Joint information

Index	Joint Label	Joint Type (Name In Program)	DOFs	Mom Segment Index	Child Segment Index	Id In Program
0	S0	Weld(SG-SG)	0	S0	S0	1
1	S0	Translation(SG-DG)	3	S0	D0	2
2	D0	Spherical(DG-Pelvis)	3	D0	5	3
3	1	Spherical (Neck -Head)	3	2	1	301
4	2	Spherical (Thorax- Neck)	3	3	2	300
5	3	Spherical (Abdomen- Thorax)	3	4	3	31
6	4	Spherical (Pelvis -Abdomen)	3	5	4	30
7/8	5R/5L	Spherical(Shoulder)	3×2	3/3	6R/6L	100/200
9/10	6R/6L	Revolute(Elbow)	1×2	6R/6L	7R/7L	101/201
11/12	7R/7L	Spherical(Hand)	3×2	7R/7L	8R/8L	102/202
13/14	8R/8L	Spherical(Hip)	3×2	5/5	9R/9L	10/20
15/16	9R/9L	Revolute(Knee)	1×2	9R/9L	10R/10L	11/21
17/18	AR/AL	Spherical(Ankle)	3×2	10R/10L	11R/11L	12/22

Note: SG-static ground; DG-dynamic ground; R-right; L-left

In the Equation(5.5), k is the joint index listed in the first column of Table 5.3 and $dof(k)$ expresses the number of DOFs in joint k, for example, $dof(0) = 0 \; and \; dof(1) = 3$, whose values are listed in the fourth column of Table 5.3. It is obvious that $\sum_{k=0}^{18} dof(k) = 46$. From the equation, we can know the main compositions of q is:

(1)Three translational DOFs $(q_1)_3$ between DG and SG, which is the starting portion of q vector. The three DOFs are also the displacement along x, y and z axis between SG (or WRF) and the Root Segment (RS, for short). The RS is "Pelvis" segment in model construction of LS-DHM;

(2)Three rotational DOFs between RS and DG, which is the item $(q_2)_3$;

(3)The DOFs between the adjacent joints, which maybe include linear and rotational displacement. In the model, this portion only includes the rotation displacement. If the linear displacement is existed, we should process all translational DOFs to precede any rotational DOFs.

For purposes of convenience, we will write Equation (5.5) as the following Equation (5.6). All

elements in Equation (5.6) is the sequential expansion of Equation (5.5).In the work, we specified the expanding sequence is along the x-y-z. From the description above, the vector $\left(q^k\right)_{dof(k)}$ has $dof(k)$ elements, so the overall generalized coordinates vector $\boldsymbol{q}$ is their sequential arrangement.

$$q = [q_1, q_2, \cdots, q_m]^T \quad and \quad \text{m} = \sum_{k=0}^{18} dof(k) = 46 \tag{5.6}$$

According to Equation (5.4), the basal state space is the 2*DOFs-dimensional space spanned by the generalized coordinates and generalized velocities, that is, the basal state space S is Equation (5.7) that is a 92x1 vector. As described above, besides of the information of displacement and velocity, we can also add some other additional useful variables to save mediate result.

$$S = [q, \dot{q}]^{\mathrm{T}} \tag{5.7}$$

- Modeling Moment $\boldsymbol{M_n}$

In my work, the net moment $\boldsymbol{M_n}$ actuating on a joint of the LS-DHM divided into two parts, i.e., internal moments and external moments. The internal moments include internally generated muscle force moment $\boldsymbol{M_m}$, ligament passive tissue force torque $\boldsymbol{M_p}$, and external torques comprise of gravitational force $\boldsymbol{M_g}$, robot assisted moment $\boldsymbol{M_r}$ and environment constraint moment $\boldsymbol{M_e}$ (such as external perturbations, the ground reaction forces (or moment), and so on).

In our model, the influences of the muscles and other factors where realized as torques entered about the joints. The $\boldsymbol{M_n}$ can be written as:

$$\boldsymbol{M_n} = I\boldsymbol{\alpha} = r \times F = [M_m + M_p]_{\text{in}} + [M_g + M_r + M_o]_{\text{ext}} \tag{5.8}$$

Where $\boldsymbol{\alpha}$ is determined from motion analysis, $\boldsymbol{I}$ is calculated from anthropometric data.

The next step is to differentiate to calculate the torques for each joint:

- Gravitational Force $\boldsymbol{G}$

Gravity is exerted on all the masses. It is a conservative force and directly integrate it into the constructed motion of equation in the work. A gravity vector of $[0 \ -9.81 \ 0]^T \ m/s^2$ is exerted on all the segments.

- Muscle Torques $\boldsymbol{M_m}$

This part of content has been given full description in Chapter5.The related equations for calculating the muscle moment are listed in Equations (4.29) (4.30) and (4.31).

● Passive Joint Moment $\boldsymbol{M}_p$

All the soft tissues, cartilages, ligaments and muscles wrapping the joints exert resistance to joint rotation and dissipate energy[141]. This resistance (and then the resultant moment) limits the range of motion to normal ranges, so we included passive damping and stiffness in the model. It is worth noting that these passive generalized forces are used extensively in natural locomotion to reduce energy consumption (energy temporarily stored and released), increase stability and simplify the control, thus increasing the efficiency of locomotion. If the joint angle stays within a nominal range, then the effects of the passive structure are negligible, but when the joint angle is near the boundary ,the passive torques blows up exponentially in a positive or negative direction that depends on which boundary of the state space the angle approaches. The general form of the passive moments is given by Equation (5.9) [76, 142, 143]:

$$M_p(\theta, \dot\theta) = k_1 e^{-k_2(\theta - \theta_{min})} - k_3 e^{-k_4(\theta_{max} - \theta)} - c\dot\theta \qquad (5.9)$$

where θ denotes the joint angle in radians , $\dot\theta$ is the joint angular velocity given in radians per second, and M_p is measured in Newton-meters . The $\theta_{min}, \theta_{max}$ are two boundaries of joint dynamic motion, and we consider $\theta_{min} < \theta < \theta_{max}$ .The other parameters $k_i (i = 1,2,3,4)$ and c determine the shape of the curve and the amount of damping required for smooth movement. They are taken from literature[76] listed in Table 5.4. The inclusion of the damping term is vital since the passive $c\dot\theta$ is excluded from the musculotendon model. These parameters we chosen so that the curve gets very large near boundaries of joint movement.

Table 5.4 Coefficients for Passive Joint Moments

Hip	$k_1 = 2.6, k_2 = 5.8,$ $k_3 = 8.7, k_4 = 1.3$	$c_1 = 1.09,$ $\theta_{max} = 1.92, (110^\circ\ flexion)$ $\theta_{min} = -0.1744\ (-10^\circ\ extension)$
Knee	$k_1 = 3.1, k_2 = 5.9,$ $k_3 = 10.5, k_4 = 11.8$	$c_1 = 3.17,$ $\theta_{max} = 0.00, (full\ extension)$ $\theta_{min} = -1.92, (110^\circ\ flexion)$
Ankle	$k_1 = 2.0, k_2 = 5.0,$ $k_3 = 9.0, k_4 = 5.0$	$c_1 = 9.43,$ $\theta_{max} = 1.92, (20^\circ\ dorsiflexion)$ $\theta_{min} = 1.047, (-30^\circ\ plantarflexion)$

Based on the given parameter values listed in Table 5.4, we plot the following passive joint torques curves shown in Figure 5.5 under static conditions, i.e., neglecting the damping term. Here we assign moment positive direction in counter-clockwise. It is obvious that the value of passive moment get larger in positive direction near θ_{min} and smaller in negative direction near θ_{max}.
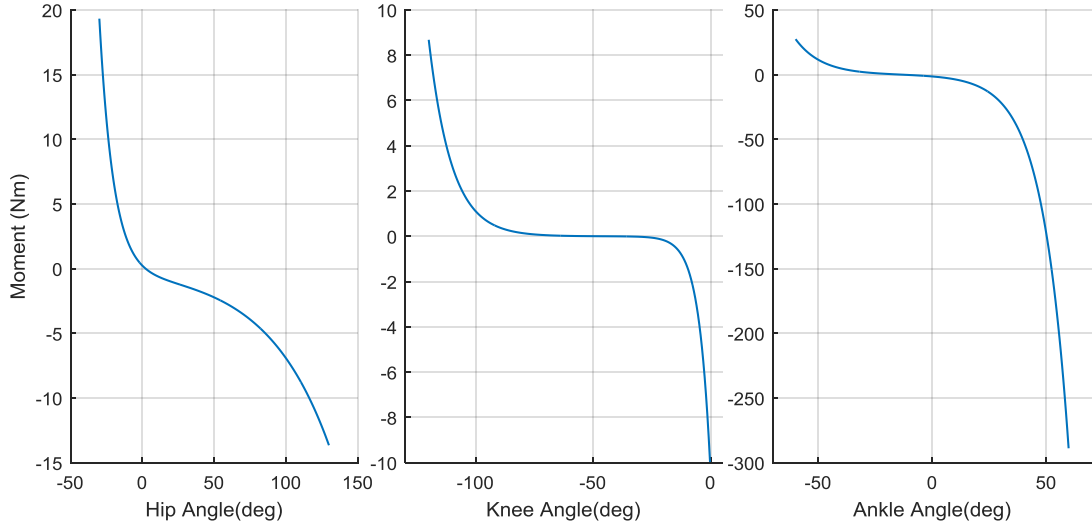
Figure 5.5 Passive joint moment under static conditions

● Environment constraint moment $\boldsymbol{M}_e$ or other external torques, such as $\boldsymbol{M}_r$

In the work, this part of forces mainly comprised of the robot assisted moment $\boldsymbol{M}_r$. In addition, according to the state of Human-ERRobot system, other forces or torque can be brought in, for example, when the system is in walking, ground reaction force and friction force can be introduced, but the force will be disappeared when in sitting. For the modeling to describe the foot-ground contacts, accurate models adopt the finite element models[144]. The modelling of the foot contacts can be divided in two parts: first the search for the contact points and secondly the force generation.

● Building Human-ERRobot System Model

The work mainly involves lower limb movement, so a simplified seven-segment human model (two feet, two legs, two thighs, and an HAT segment) is obtained by DOF configuration for the LS-DHM of total body, as shown in Figure 5.6. It is designed to include three joints and four DOFs per leg (the joint of Hip: 2DOFs, Knee: 1DOF, and Ankle: 1 DOF, respectively). According to Equation (5.5) (5.6) and Table 5.3, the generalized coordinate's vector $\boldsymbol{q}$ of the simplified model is Equation (5.10) or Equation (5.11), which are also expressed in Figure 5.6. These coordinates are the relative angles of each link of the robot, and are measured with respect to the coordinate frame fixed with proximal segments. For the definition of zero joint angle and joint angle positive direction, you can refer to Figure 2.3.

$$\boldsymbol{q} = [(\boldsymbol{q}^1)_3, (\boldsymbol{q}^2)_3, (\boldsymbol{q}^{13})_2, (\boldsymbol{q}^{14})_2, (\boldsymbol{q}^{15})_1, (\boldsymbol{q}^{16})_1, (\boldsymbol{q}^{17})_1, (\boldsymbol{q}^{18})_1]^T \qquad (5.10)$$

Or

$$\boldsymbol{q} = [q_1, q_2, \cdots, q_{14}]^T \tag{5.11}$$
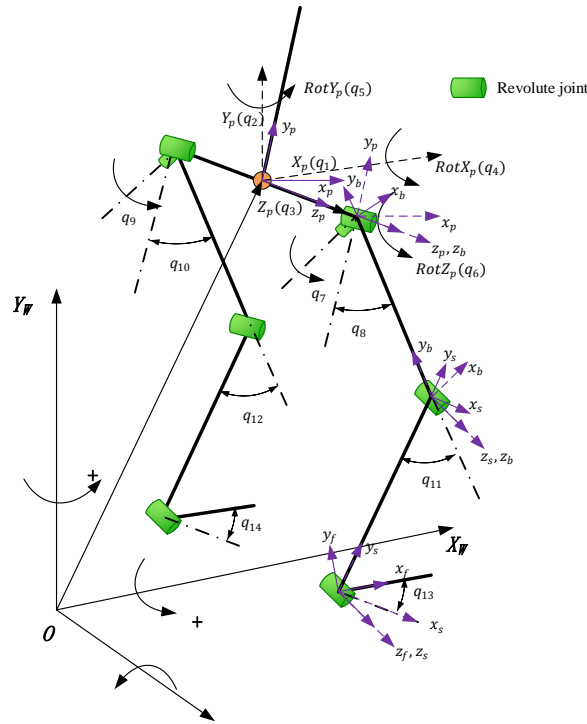
Figure 5.6 Simplified LS-DHM of 7-segment human model with: (1) generalized coordinates $\boldsymbol{q}_i (i = 1,2 \dots 14)$; (2) local RFs fixed on Segments. Only the part of right leg is shown in the figure above

Moreover, the Figure 5.6 also shows the definition of the system's global frame and local frames fixed on segments. As described previously, the setting of the coordinate system is performed according to the method shown in Figure 5.2. In order to construct the governing equations by using Lagrangian formulation, the force diagram of the model is presented, as shown in Figure 5.7(taking the right leg as an example) .In addition, the segment mass, inertial and geometrical parameters of the simulation model are also illustrated in the figure at the same time. The details of the forces and torques applied on the human body have been described in Section (1).In the Figure 5.7, the applied forces and torques are expressed in red line. The $M_m^{RHip}, M_m^{RKnee}, M_m^{RAnkle}$ represent the muscle net torques acting on joint of right leg Hip, knee and ankle respectively, F_r^{RT}, F_r^{RS} represent the assistive force from ERRobot robot, and superscript "RT" "RS" are the abbreviations of "thigh of right leg" and "shank of right leg", respectively. As been showed in Figure 2.4, we can know that the ERRobot is designed to have two shank attachment devices. In the work, we consider their influences on human by introducing a resultant force, namely, F_r^{RS}.
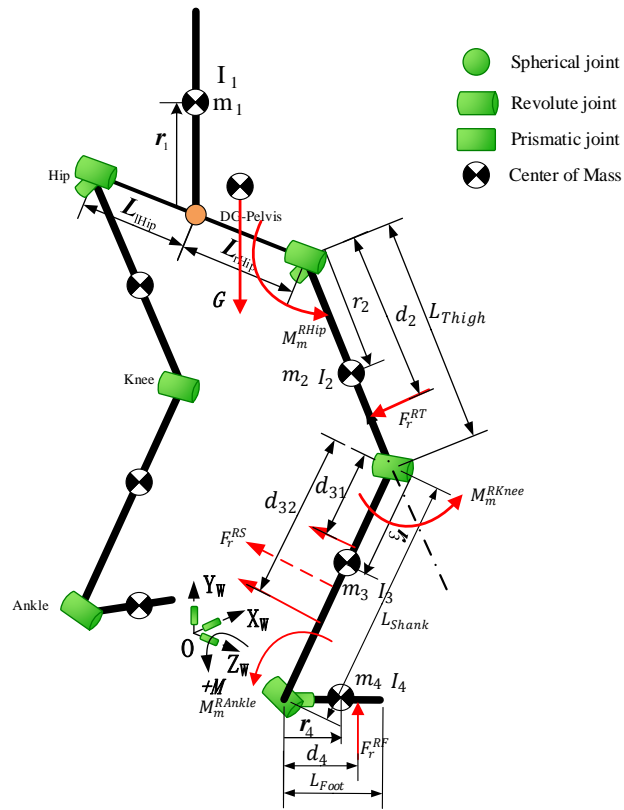
Figure 5.7 The schematic diagram of the simplified LS-DHM with forces acting on the part of the body and parameters (taking the right leg as an example);

In the work, we will continue to do further simplification on the simplified lower limb model shown in Figure 5.6 based on the studied problem in the following, and the generalized coordinates will also be re-numbered for make the subscripts of generalized coordinate consecutive.

5.2 Mathematical Model Formulation of LS-DHM Kinematic Analysis

In the chapter, the skeleton system of human is modeled as a linked rigid segment system connected by revolute joints.

5.2.1   Systematic Kinematic Analysis Method

In kinematic research of multi-body system, the rigid body kinematic description includes linear and angular displacement (position), velocity and acceleration components, respectively. Kinematics is the study of the movements of rigid structures, independent of the forces that might be involved. Regardless of the system being modeled, it is necessary to describe systematically. In the section, we shall develop the techniques. We will derive the iteration kinematics equations for multi-body system analysis, which is a systematic method that allows analysis large, complex mechanical systems. The work in the section provides a good foundation for the dynamic analysis

and control of rehabilitation robot.

In this section, we shall relate the coordinate frames of $\sum Mom$ and $\sum Child$. In particular, we shall study the transformation between the components of a vector in the two systems.
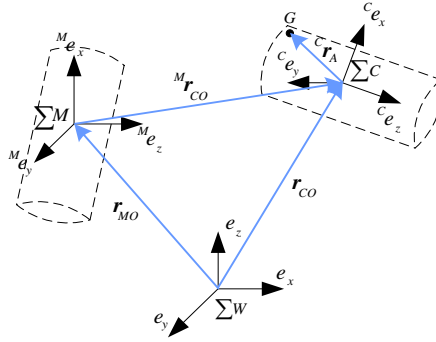
Figure 5.8 Transformation of Coordinate Systems between Child and Mom RF

The rotation transformation is based on the rotation sequence of $z - x' - y''$, that is, starting with $\sum M$, we rotate about the z axis α to create the first intermediate system $\sum X_1$, about new x axis β to create the second intermediate $\sum X_2$, and about the new z axis γ to get the desired $\sum C$ last. A kinematics table, shown in Table 5.5, presents a method for efficiently managing the mathematics involved in analyzing multibody and multiple coordinate system problems. A schematic diagram like Figure 5.8, which defines a WRF or body-fixed LRF, must accompany every kinematics table. The purpose of the schematic is to identify the relative position between body-fixed LRFs and point on the body at which the absolute velocity and acceleration is to be determined. In creating human model, points can represent the origin of a reference system, a segment's COM, a point where an external force is applied, a joint center, a muscle insertion, and other points of interest. In the analysis of experimental joint mechanics data, the transformation of point coordinates from one coordinate system to another is a frequent task.

Table 5.5 Kinematics table

	$^i\boldsymbol{e}_x$	$^i\boldsymbol{e}_y$	$^i\boldsymbol{e}_z$
a. $^M\boldsymbol{r}_G = {}^M\boldsymbol{r}_{CO} + R_{M\leftarrow C}\,{}^C\boldsymbol{r}_G$			
$^M\boldsymbol{r}_{CO}$	$^M\boldsymbol{r}_x^{CO}$	$^M\boldsymbol{r}_y^{CO}$	$^M\boldsymbol{r}_z^{CO}$
$^C\boldsymbol{r}_G$	$^C\boldsymbol{r}_x^G$	$^C\boldsymbol{r}_y^G$	$^C\boldsymbol{r}_z^G$
b. $^M\boldsymbol{v}_G = {}^M\boldsymbol{v}_{CO} + [R_{M\leftarrow C}]\,{}^C\boldsymbol{v}_G + [R_{M\leftarrow C}]\,{}^C\widehat{\omega}_C\,{}^C\boldsymbol{r}_G$			
$^M\boldsymbol{v}_{CO}$	$^M\boldsymbol{v}_{CO}^x$	$^M\boldsymbol{v}_{CO}^y$	$^M\boldsymbol{v}_{CO}^z$
$^C\boldsymbol{v}_G$	$^C\boldsymbol{v}_G^x$	$^C\boldsymbol{v}_G^y$	$^C\boldsymbol{v}_G^z$
$^C\boldsymbol{\omega}_C$	$\omega_2 c\gamma - \omega_1 c\beta s\gamma$	$\omega_3 + \omega_1 s\beta$	$\omega_2 s\gamma + \omega_1 c\beta c\gamma$
c. $^M\boldsymbol{a}_G = {}^M\boldsymbol{a}_{CO} + [R_{M\leftarrow C}]\,{}^C\boldsymbol{a}_G + 2[R_{M\leftarrow C}]\,{}^C\widehat{\omega}_C\,{}^C\boldsymbol{v}_G + [R_{M\leftarrow C}]\,{}^C\widehat{\omega}_C\,{}^C\widehat{\omega}_C\,{}^C\boldsymbol{r}_G + [R_{M\leftarrow C}]\,{}^C\hat{\xi}_C\,{}^C\boldsymbol{r}_G$			
$^M\boldsymbol{a}_{CO}$	$^M\boldsymbol{a}_{CO}^x$	$^M\boldsymbol{a}_{CO}^y$	$^M\boldsymbol{a}_{CO}^z$
$^C\boldsymbol{a}_G$	$^C\boldsymbol{a}_G^x$	$^C\boldsymbol{a}_G^y$	$^C\boldsymbol{a}_G^z$
$^C\xi_C$	$\xi_2 c\gamma - \xi_1 c\beta s\gamma$	$\xi_2 + \xi_1 s\beta$	$\xi_2 s\gamma + \xi_1 c\beta c\gamma$

The kinematics table is divided into three sections (a) (b) and (c). They are sequentially related to each other and a previous one is the precondition of the next, that is, they must be calculated in order a → b → c, because positions, are required for finding velocities, and both positions and velocities are required for finding accelerations. The following gives an illustration for each of three sections.

a. Calculate pose of point G in $\sum C$ with respect to $\sum M$: $^{M}\boldsymbol{r}_{G}$

(1)Origin O of $\sum C$ in $\sum M$: $^{M}\mathrm{r}_{CO}$ (three components in x-y-z axis: $^{M}\boldsymbol{r}_{CO}^{x}$, $^{M}\boldsymbol{r}_{CO}^{y}$, $^{M}\boldsymbol{r}_{CO}^{z}$)

(2)Any point G in frame C: $^{C}\mathrm{r}_{G}$ (three components in x-y-z axis: $^{C}\boldsymbol{r}_{G}^{x}$, $^{C}\boldsymbol{r}_{G}^{y}$, $^{C}\boldsymbol{r}_{G}^{z}$)

(3)Transformation matrix from $\sum C$ to $\sum M$:

The rotation matrix R shown in Equation(5.12)(5.13) is obtained by combining the effect of the three rotations.

$$\begin{cases} R_{M \leftarrow C}(\alpha,\beta,\gamma) = R_z(\alpha)R_x(\beta)R_y(\gamma) \\ R_{C \leftarrow M} = [R_{M \leftarrow C}]^{-1} = [R_{M \leftarrow C}]^T = R_y(-\gamma)R_x(-\beta)R_z(-\alpha) \end{cases} \tag{5.12}$$

In expanded form,

$$R_{M \leftarrow C}(\alpha,\beta,\gamma) = \begin{bmatrix} c\alpha c\gamma - s\alpha s\beta s\gamma & -c\beta s\alpha & c\alpha s\gamma + c\gamma s\alpha s\beta \\ c\gamma s\alpha + c\alpha s\beta s\gamma & c\alpha c\beta & s\alpha s\gamma - c\alpha c\gamma s\beta \\ -c\beta s\gamma & s\beta & c\beta c\gamma \end{bmatrix} \tag{5.13}$$

b. Calculate velocity of point G in $\sum C$ with respect to $\sum M$: $^{M}\boldsymbol{v}_{G}$

The velocity vector of the same point is obtained by taking the time derivative of position vector $^{M}\boldsymbol{r}_{G}$.

(1)Velocity of $\sum C$ origin in $\sum M$: $^{M}\boldsymbol{v}_{CO} = {}^{M}\dot{\boldsymbol{r}}_{CO}$ (components in x-y-z axis: $^{M}\boldsymbol{v}_{CO}^{x}$, $^{M}\boldsymbol{v}_{CO}^{y}$, $^{M}\boldsymbol{v}_{CO}^{z}$)

(2)Relative velocity of point G in $\sum C$: $^{C}\boldsymbol{v}_{G} = {}^{C}\dot{\boldsymbol{r}}_{G}$ (components in x-y-z axis: $^{C}\boldsymbol{v}_{G}^{x}$, $^{C}\boldsymbol{v}_{G}^{y}$, $^{C}\boldsymbol{v}_{G}^{z}$), if G is a fixed point, $^{C}\boldsymbol{v}_{G}$ becomes a zero vector.

(3)Angle velocity of $\sum C$ that is transformed into CRF: $^{C}\boldsymbol{\omega}_{C}$ (three components in CRF axis are $^{C}\omega_{C}^{x}$, $^{C}\omega_{C}^{y}$, $^{C}\omega_{C}^{z}$, in following equation, $\omega_1 = \dot{\alpha}$, $\omega_2 = \dot{\beta}$, $\omega_3 = \dot{\gamma}$

$$^{C}\boldsymbol{\omega}_{C} = \begin{bmatrix} ^{C}\omega_{C}^{x} \\ ^{C}\omega_{C}^{y} \\ ^{C}\omega_{C}^{z} \end{bmatrix} = \begin{bmatrix} \omega_2 c\gamma - \omega_1 c\beta s\gamma \\ \omega_3 + \omega_1 s\beta \\ \omega_2 s\gamma + \omega_1 c\beta c\gamma \end{bmatrix} \quad ^{C}\widehat{\omega}_{C} = \begin{bmatrix} 0 & -^{C}\omega_{C}^{z} & ^{C}\omega_{C}^{y} \\ ^{C}\omega_{C}^{z} & 0 & -^{C}\omega_{C}^{x} \\ -^{C}\omega_{C}^{y} & ^{C}\omega_{C}^{x} & 0 \end{bmatrix}$$

c. Calculate acceleration of point G in $\sum C$ with respect to $\sum M$: $^M\boldsymbol{a}_G$

The acceleration vector of the same point is obtained by taking the time derivative of velocity vector $^M\boldsymbol{v}_G$.

(1)Acceleration of $\sum C$ origin in $\sum M$: $^M\boldsymbol{a}_{CO} = {}^M\ddot{r}_{CO}$ (components in axis: $^Ma_{CO}^x$, $^Ma_{CO}^y$, $^Ma_{CO}^z$)

(2)Relative acceleration of point G in $\sum C$: $^C\boldsymbol{a}_G = {}^C\ddot{r}_G$ (components in x-y-z axis: $^Ca_G^x$, $^Ca_G^y$, $^Ca_G^z$), if G is a fixed point, $^C\boldsymbol{a}_G$ becomes a zero vector.

(3)Angle acceleration of $\sum C$: $^C\boldsymbol{\xi}_C = {}^C\dot{\boldsymbol{\omega}}_C$ (three components in CRF axis are $^C\xi_C^x$, $^C\xi_C^y$, $^C\xi_C^z$, in following equation, $\xi_1 = \ddot{\alpha}$, $\xi_2 = \ddot{\beta}$, $\xi_3 = \ddot{\gamma}$

$$
^C\boldsymbol{\xi}_C = \begin{bmatrix} ^C\xi_C^x \\ ^C\xi_C^y \\ ^C\xi_C^z \end{bmatrix} = \begin{bmatrix} \xi_2 c\gamma - \xi_1 c\beta s\gamma \\ \xi_2 + \xi_1 s\beta \\ \xi_2 s\gamma + \xi_1 c\beta c\gamma \end{bmatrix} \quad ^C\hat{\boldsymbol{\xi}}_C = \begin{bmatrix} 0 & -^C\xi_z & ^C\xi_y \\ ^C\xi_z & 0 & -^C\xi_x \\ -^C\xi_y & ^C\xi_x & 0 \end{bmatrix}
$$

Once these lists are established, with the aid of the first element in each list, the displacement and velocity vectors of any point in the system can be derived easily, and the system is completely specified when the mass and the COM of the two blocks are given. This is done using the lists:

5.2.2 Mathematical Model Formulation of LS-DHM Kinematic Analysis

The kinematic analysis is regardless of the forces in the system, and only constraint equations are considered. Prior to the dynamic analysis, the kinematic analysis is necessary. In addition, the section only presents one of two transformations, that is, the developed transformation is from coordinate systems that are far away from the base coordinates to those coordinate systems that are near to base framework. That is because this type of transformation is commonly used.

We create two object classes, "SEGMENT" and "JOINT", and they encapsulate some member variables and functions. According to whether or not the variables are changing on time t, we also classify all variables into two kinds. Those variables not changing with time t are called configuration parameters, which can be assigned by programme coding or modifying configuration files or GUI. The others changing with time t are called as state variables. Notations of these variables expressed in equations and programme are shown in Table 5.6, and their descriptions are also presented below. Before establishing the following formulas, we present several equations first.

a) The mother segment ID of Segment 1 is 1, that is, Equation (5.14) is to hold.

$$1\text{m} = 1 \tag{5.14}$$

b) The rotation matrix transforming from RF(i) to itself, i.e., $R_{i \leftarrow i}$, is a unit matrix. The equation is shown in Equation (5.15).

$$R_{i \leftarrow i} = E_{3 \times 3} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{5.15}$$

According to the values' assignment principle for joint Id and segment Id, the Id of joint is always equal to the Id of its child segment, that is, there is the following relationship:

If conditions are as following:

$$jm = i \ and \ jc = ic$$

Table 5.6 Main class members' variables

Class Name	Variable Types	Main Notations in equation	Corresponding expressions in programme
<SEGMENT> Class Member	Configuration Parameters	i	ID
		im	MotherID
		ic	ChildID
		C_i	Centroid
		m_i	m
		I_i^c	Inertia
	State Parameters	$^w P_i^c$	CentroidPoseT4x4_InWorldCord
<JOINT> Class Member	Configuration Parameters	j	ID
		jm	MotherSegmentID
		jc	ChildSegmentID
		m_j	m
		$[^{jm}r_{jo}]_{3 \times 1}$	Joint_Point_In_Mother(1:3)
		$[^{jm}A_{jo}]_{3 \times 1}$	Joint_Point_In_Mother(4:6)
		$[S_j]_{6 \times 1}$	SequenceOfFreedom
		$[q_j^0]_{6 \times 1}$	displacement0
	State Parameters	t	time
		$[q_j^t]_{6 \times 1}$	displacement
		$[Q_j^t]_{6 \times 1}$	displacement_All_InMotherCoord
		$[^{jm}v_{jo}^t]_{3 \times 1}$	JointVelocity(1:3)
		$[^{k=1,2,3}\omega_j^t]_{3 \times 1}$	JointVelocity(4:6)
		$^j v_G$	v
		$[^{k=1,2,3}\xi_j^t]_{3 \times 1}$	dw
		$^j a_G$	dv

Then, there is Equation (5.16):

$$j = ic \qquad (5.16)$$

This processing is very important, because the movement of LRF fixed in joint j is viewed as the same with the movement of segment that is the child segment of joint j at that time. That is to say, the Id of joint and segment can be shared.

(1) Rotation Matrix

The calculation of rotation transformation matrix between any two RFs is the foundation of all other calculations such as position analysis, velocity analysis and so on. Therefore, here derives its calculation formula first, as shown in Equation (5.17), here assuming that we need to calculate the rotation matrix from $RF(j)$ to $RF(i)$. When i is equal to 1, the transformation matrix, $R_{i \leftarrow j}$, is the transformation from frame j to WRF. Let the $RF(i)$ be closer to the base coordinate system WRF than $RF(j)$, that is, the id value of i belongs to any one of j's mother segment sequences starting from ground (Id=1).

- Known conditions:
$R_{1 \leftarrow 1} = E_{3 \times 3}$ $\boldsymbol{Q}_j^t$ $^{km}\boldsymbol{A}_{kO}$ (k∈ Any id value between i and j)

- Calculating Formula:
$$\begin{cases} R_{i \leftarrow j} = E_{3 \times 3} \, , i = j \\ R_{i \leftarrow j} = \prod_{k=i}^{j} \left[R\left(^{km}\boldsymbol{A}_{kO}\right) R\left(\boldsymbol{Q}_j^t\right) \right], i \in Id \ value \ between \ 1 \ and \ j \end{cases} \qquad (5.17)$$

Where, because the work adopts the rotation sequence of $z0(\alpha) \rightarrow x1(\beta) \rightarrow y2(\gamma)$, according to Equation (5.13), the Equation (5.18) is to hold. Because both of $^{km}\boldsymbol{A}_{kO}$ and $\boldsymbol{Q}_j^t$ vector values are assigned based on the sequence of x → y → z, so we need to note the corresponding value relations. Taking $\boldsymbol{Q}_j^t$ for an example, the substitution relations are: $\alpha = Q_j^t(3), \beta = Q_j^t(1), \gamma = Q_j^t(2)$

$$R(\alpha, \beta, \gamma) = R_{M \leftarrow C} = \begin{bmatrix} c\alpha c\gamma - s\alpha s\beta s\gamma & -c\beta s\alpha & c\alpha s\gamma + c\gamma s\alpha s\beta \\ c\gamma s\alpha + c\alpha s\beta s\gamma & c\alpha c\beta & s\alpha s\gamma - c\alpha c\gamma s\beta \\ -c\beta s\gamma & s\beta & c\beta c\gamma \end{bmatrix} \qquad (5.18)$$

(2) Position Analysis

To perform the transformation between any two RFs, the work derives the calculation formula shown in Equation (5.19), which can translate any point coordinate or 3×1 vector in $RF(j)$ into another coordinate system $RF(i)$. As mentioned above, here only gives out one transformation from coordinate systems that are far away from the base coordinates to those coordinate systems that are near to base framework. Assuming any a point G in $RF(j)$ whose position vector is

denoted by $^j\boldsymbol{r}_G$, now we can change it into $RF(i)$ by Equation (5.19).

- Known conditions:

$^j\boldsymbol{r}_G$, $R_{i\leftarrow k}$, $^{km}\boldsymbol{r}_{kO}$(k∈ Any id value between i and j)

- Calculating Formula:

$$
\begin{cases}
^i\boldsymbol{r}_G = {}^j\boldsymbol{r}_G, j = i \\
^i\boldsymbol{r}_G = R_{i\leftarrow j}\,{}^j\boldsymbol{r}_G + \sum_{k=i}^{j}[R_{i\leftarrow km}]^{km}\boldsymbol{r}_{kO}, i \in [1,j]
\end{cases}
\tag{5.19}
$$

When needing to calculate the absolute position vector of a point positioned in $RF(j)$, we only need to assign i to 1.Here stipulates that $^1\boldsymbol{r}_{10}$ is equal to $[0\ 0\ 0]'$ .Then we can get the following calculating formula shown in Equation (5.20).

$$
^1\boldsymbol{r}_G = R_{1\leftarrow j}\,{}^j\boldsymbol{r}_G + \sum_{k=1}^{j}[R_{1\leftarrow km}]^{km}\boldsymbol{r}_{kO}, j \geq 1
\tag{5.20}
$$

(3) Velocity Analysis

First, we stipulate that $^1\boldsymbol{v}_{10}$ is equal to $[0\ 0\ 0]'$, and because all the adjacent joint links are only have rotational DOFs, every of the items $^{km}\boldsymbol{v}_{kO}$ is zero vector. Combining with the known conditions below, we can obtain the velocity analysis by application of recursive formula shown in Equation (5.21).

- Known conditions:

$^j\boldsymbol{r}_G$, $^j\boldsymbol{v}_G$, $^j\widehat{\omega}_j, R_{km\leftarrow k}$, $^{km}\boldsymbol{v}_{kO} = \mathbf{0}(k \in$ Any id value between i and $j)$

- Calculating Formula:

$$
^{jm}\boldsymbol{v}_G = {}^{jm}\boldsymbol{v}_{jO} + [R_{jm\leftarrow j}]\,{}^j\boldsymbol{v}_G + [R_{jm\leftarrow j}]\,{}^j\widehat{\omega}_j\,{}^j\boldsymbol{r}_G
\tag{5.21}
$$

Where, $^j\widehat{\omega}_j$ can be obtained by the following equation. Saving the derived calculation formula rather than deriving it when it is needed will save considerable time, especially in the case of complex systems.

$$
^j\widehat{\omega}_j = \begin{bmatrix}
0 & -(^2\omega_j^t s\gamma + {}^1\omega_j^t c\beta c\gamma) & ^3\omega_j^t + {}^1\omega_j^t s\beta \\
^2\omega_j^t s\gamma + {}^1\omega_j^t c\beta c\gamma & 0 & ^1\omega_j^t c\beta s\gamma - {}^2\omega_j^t c\gamma) \\
-(^3\omega_j^t + {}^1\omega_j^t s\beta) & ^2\omega_j^t c\gamma - {}^1\omega_j^t c\beta s\gamma & 0
\end{bmatrix}
$$

(4) Acceleration Analysis

Finally, combining with the following known conditions, the acceleration analysis is obtained by application of Equation (5.22)

● Known conditions:

$${}^{j}\boldsymbol{r}_{G}, \; {}^{j}\boldsymbol{v}_{G}, \; {}^{j}\boldsymbol{a}_{G}, \; {}^{j}\widehat{\omega}_{j}, \; {}^{j}\hat{\xi}_{j}, R_{km\leftarrow k}, \; {}^{km}\boldsymbol{a}_{kO} = \boldsymbol{0}(k \in \text{Any id value between } i \text{ and } j)$$

● Calculating Formula:

$${}^{jm}\boldsymbol{a}_{G} = {}^{jm}\boldsymbol{a}_{jO} + [R_{jm\leftarrow j}]({}^{j}\boldsymbol{a}_{G} + 2\,{}^{j}\widehat{\omega}_{j}\,{}^{j}\boldsymbol{v}_{G} + {}^{j}\widehat{\omega}_{j}\,{}^{j}\widehat{\omega}_{j}\,{}^{j}\boldsymbol{r}_{G} + {}^{j}\hat{\xi}_{j}\,{}^{j}\boldsymbol{r}_{G}) \qquad (5.22)$$

Where, ${}^{j}\hat{\xi}_{j}$ can be obtained by the following equation. As the ${}^{j}\widehat{\omega}_{j}$, in order to save running time, ${}^{j}\hat{\xi}_{j}$ is also derived and saved in advance, not calculated in real time when it is needed.

$${}^{j}\hat{\xi}_{j} = \begin{bmatrix} 0 & -({}^{2}\xi_{j}^{t}s\gamma + {}^{1}\xi_{j}^{t}c\beta c\gamma) & {}^{3}\xi_{j}^{t} + {}^{1}\xi_{j}^{t}s\beta \\ {}^{2}\xi_{j}^{t}s\gamma + {}^{1}\xi_{j}^{t}c\beta c\gamma & 0 & {}^{1}\xi_{j}^{t}c\beta s\gamma - {}^{2}\xi_{j}^{t}c\gamma) \\ -({}^{3}\xi_{j}^{t} + {}^{1}\xi_{j}^{t}s\beta) & {}^{2}\xi_{j}^{t}c\gamma - {}^{1}\xi_{j}^{t}c\beta s\gamma & 0 \end{bmatrix}$$

In the section, a set of recursive formulas to calculate kinematic variables is presented. It is a systematic method for kinematic model formulation and we can code recursive program to implement the calculation by using computer. The calculation process starts with a seed value, and repeats until all the points and LRFs in the model are represented.

5.2.3 Mathematical Model Formulation of LS-DHM Kinetic Analysis

In the previous chapter, we presented a 3D simplified model (LS-DHM) of human skeleton system and derived a systematically kinematic analysis method. On the basis of them, we continue to do a dynamic analysis for the model. It does so by taking into account the mechanical properties of the muscles, the mass and inertial properties of the body segments, and the external forces arising from contact with the environment. It is extremely valuable to consider in the design of robots, in simulation and animation of robot motion, in the design of control algorithm and other studies such as investigating human motion characteristics and so on. In this section, we will develop an analytical dynamic mathematical model of human lower limb musculoskeletal system, in which Lagrange approach is adopted. The simplification model can be obtained by developed GUI program shown in Figure 0.3 for achieving better user-friendship and modularization. As the first step of kinetic analysis, the work formulated the equations of motion (kinetics equation) relating the motions of the body segments to the forces and moments that are causing those motions.

(1) Calculating Jacobian Matrix

The Jacobian can be defined as the relationship between joint velocity and foot end velocity.

$$\boldsymbol{\eta} = J(\theta)\dot{\boldsymbol{\theta}} \tag{5.23}$$

where η is the velocity of end of the model and $\dot{\theta}$ is a vector of joint velocities, and $J(\theta)$ represents the Jacobian matrix, which is set out by visualizing a set of equations as:

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \\ \omega_x \\ \omega_y \\ \omega_z \end{bmatrix} = \begin{bmatrix} J_{v1} & J_{v2} & \cdots & J_{vn} \\ J_{\omega 1} & J_{\omega 2} & \cdots & J_{\omega n} \end{bmatrix} \begin{bmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \\ \vdots \\ \dot{\theta}_n \end{bmatrix} \tag{5.24}$$

where each J_v and J_ω are 3x1 column vectors. One can see in the matrix that each column is associated with a different joint, each of which is broken into what we call the upper and lower Jacobian. Each is also calculated differently depending if the joint is prismatic or revolute. For revolute joint, J_{v_i} can be calculated by:

$$\boldsymbol{J}_{v_i} = \boldsymbol{a}_i \times (\boldsymbol{p}_n - \boldsymbol{p}_i) \tag{5.25}$$

In which $\boldsymbol{a}_i$ is orientation vector represented in WRF for the unit rotation axis of joint i. $\boldsymbol{p}_n$ is global position vector of joint n; $\boldsymbol{p}_i$ is the 3x1 global coordinate vector relative to WRF for joint i position. As opposed to J_{v_i}, the upper Jacobian which calculates the linear velocities of the joint, $J_{\omega i}$ calculates angular velocities, and for each:

$$\boldsymbol{J}_{\omega i} = \boldsymbol{a}_i \tag{5.26}$$

One valuable use of the Jacobian is for calculating joint torques necessary in robot leg to maintain configuration in response to external forces applied to the foot (or other segments). This is done using the transpose of the Jacobian matrix, as shown in the equation below:

$$\boldsymbol{\tau} = [\boldsymbol{J}_{v_i}]^T \boldsymbol{F} \tag{5.27}$$

where $\boldsymbol{\tau}$ is a vector of joint torques, and F is a three dimensional vector of force on the robot foot, arranged in the form $\boldsymbol{F} = [F_x \quad F_y \quad F_z]^T$.

(2) Motion Equations Formulation

In my studies, Lagrangian formulation that is a classic approach to expressing the equations of motion is adopted. Lagrange's equations need to employ system energy and work as functions of the generalized coordinates to obtain a set of second-order differential equations of motion. In the

following, a systematic method for getting the equations of motion for LS-DHM presented above is explained. The method can be adapted easily for writing computer programs using symbolic computer languages to generate the equations of motion. The equations of motion describe how the segments will move (subject to a set of contributory muscle, passive tissue, and external force trajectories), given that the inertial parameters of the body segments are known. For a system with n degrees of freedom, the Lagrange's equation is given as Equation (5.28)[145]

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{q}_k}\right) - \frac{\partial L}{\partial q_k} = Q_k, k = 1,2,\cdots n \tag{5.28}$$

Where, L is the Lagrangian and is defined as the difference between system kinetic energy terms KE and system potential energy PE throughout the moving body(s), that is , $L = KE - PE$. The $\dot{q}_k = \partial q_k/\partial t$ is the generalized velocity and $Q_k = \partial W/\partial q_k$ is the nonconservative generalized force corresponding to the generalized coordinate q_k, where W is the external work done on the system.

Next, we will describe the step of the Lagrange equation derivation in detail. The human model shown in Figure 5.9, which is simplified further for the proposed model in Section □, is adopted to illustrate the process, and the model and the constructed Equations will also be used in simulation experiments. The motion of the human model is restricted to the sagittal plane in order to avoid unnecessary complexity, and for simplification, the ERRobot is supposed to be rigidly attached to the wearer's leg such that the whole system rotates synchronously about the three joints. In addition, we suppose that the right leg is being used as a stand leg and in swing phase of the gait cycle.
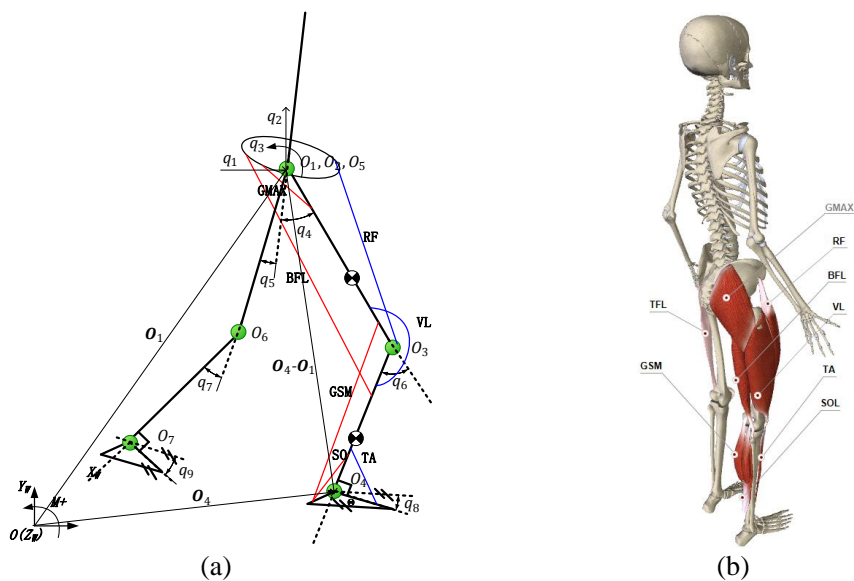

(a)   (b)

Figure 5.9 Simplified model and adopted muscles

Step1: Draw body diagram and select independent generalized coordinates

In the stage, the primary work includes the definition of system's frame (global frame and segment local frames), expressing of forces (or torque), and so on. For the analyzed model shown in Figure 5.9, its RF construction and adding acting forces to model can be implemented like doing in Figure 5.6, and Figure 5.7. The selected number of DOFs that is the minimum number of independent coordinates to describe the dynamical system is nine and the generalized coordinate of the model is given in Equation (5.30).

$$\boldsymbol{q} = [q_1, q_2, \cdots, q_9]^T \tag{5.29}$$

Step2: Express the Lagrangian L in the generalized coordinates and velocities

From the definition formula of the Lagrangian L, we know that it is the difference between system kinetic energy terms KE and system potential energy PE throughout the moving body(s). The kinetic energy for a segment i is defined as the work done on the segment to increase its linear or angular velocity from rest to some value $\boldsymbol{v}(or\ \boldsymbol{\omega})$, where $\boldsymbol{v}(or\ \boldsymbol{\omega})$ are measured relative to a global (inertial) reference system. Let inertia tensor be $[I]$ and we select an appropriate LRS to make it become a diagonal matrix. Assuming that a segment has a mass M, the formula for calculating the kinetic energy of the segment has the form:

$$[KE]_i = \frac{1}{2}M[\boldsymbol{v}]^T[\boldsymbol{v}] + M[\boldsymbol{v}]^T[\hat{\boldsymbol{\omega}}][\ ^i\boldsymbol{r}_C] + \frac{1}{2}[\boldsymbol{\omega}]^T[I][\boldsymbol{\omega}] \tag{5.30}$$

The second term reflects the selection of the LRS origin at a point rather than the COM of the segment. When the two point coincide, the second term vanishes, that is, it becomes the sum of two terms: the translational energy obtained by concentrating the entire mass of the object at the center of mass, and the rotational kinetic energy of the body about the COM, as shown in Equation (5.31).

$$[KE]_i = \frac{1}{2}[\boldsymbol{\omega}_i]^T[I_{Gi}][\boldsymbol{\omega}_i] + \frac{1}{2}m_i[\boldsymbol{V}_{Gi}]^T \cdot \boldsymbol{V}_{Gi} \tag{5.31}$$

Where I_{Gi} is the component for the mass moment of inertia about COM point G in the Gi local reference frame; $\boldsymbol{V}_{G1}$ is the velocity vector of segment taken at the COM. For the open chain system, Equation (5.31) can further be written as (5.32), in which $M(p)$ is the inertia matrix.

$$[KE]_i = \frac{1}{2}\boldsymbol{p}_i^T M(p)\boldsymbol{p}_i \tag{5.32}$$

Segment potential energy is defined as the energy possessed by virtue of a segment (or particle) position in a gravity field relative to a selected datum level (usually ground level) in the system.

In the work, the Y axis of the WRS is selected in the direction of the gravity field and its origin is taken as the zero level of the system, then the potential energy of a segment i is defined as

$$[PE]_i = M_i g \ {}^i H_G \tag{5.33}$$

Where, g is the gravity constant, and ${}^i H_G$ provides the coordinates from the base to the COM of segment i.

According to the definition formula of the Lagrangian L, it can be calculated as Equation (5.34).

$$L = \sum_{i=1}^{7} [KE]_i - \sum_{i=1}^{7} [PE]_i \tag{5.34}$$

Step3: Determine the generalized forces on each rigid body that perform (virtual) work.

According to the description in section (1), the external work done on LS-DHM system comprised two parts, i.e., internal moments and external moments. Please note that the applied forces on segment are various according to simulated model. For example, when we study human body model being sitting, the ground reaction forces can be neglected, but when the walking model is studied, for stand leg, nearly all forces (or torques) involved above may be included into model. The section will present the computing process of generalized torques by introducing transformation matrix. As supposed above, the ERRobot is rigidly attached to the wearer's leg. Therefore, most parts of Human-ERRobot system is controlled by both the human internal torque, the robot generated torques, gravity, and ground reaction force is applied on right foot. We next describe the generalized forces in detail. Gravity can be viewed as a constant force acting on the COM of each segment part, which has been included in the Lagrangian L. For joint internal moments and the robot generated torques, they are applied on each of independent coordinates, so they can be computed as:

$$[\boldsymbol{Q}_{N1}]_{9\times1} = [T_{in}]_{9\times7}[M_{in}]_{7\times1} \tag{5.35}$$

And

$$[\boldsymbol{Q}_{N2}]_{9\times1} = [T_r]_{9\times6}[M_r]_{6\times1} \tag{5.36}$$

Where, $[T_{in}] \ and \ [T_r]$ are two coefficient matrix that transfer the joint internal moments (muscles moment M_m + passive torques M_p) and robot motor applied torque to each of generalized coordinates, respectively. All of them are listed in the following Equation (5.37). Since there are no joint internal moments that apply forces directly to the root DOFs (i.e., q_1, q_2), thus the matrix comprised of the first two rows of T_{in} is a $[O]_{2\times7}$, which says that the global motion of the model is completely determined by the other external forces, such as ground

reaction forces. Similarly, there are no robot driven moments applied on the first three DOFs, so the matrix of the first three rows is $[O]_{3\times6}$ in T_r.

$$[T_{in}]_{9\times7} = \begin{bmatrix} [O]_{2\times7} \\ [E]_{7\times7} \end{bmatrix}, \quad [T_r]_{9\times6} = \begin{bmatrix} [O]_{3\times6} \\ [E]_{6\times6} \end{bmatrix}$$

$$M_{in} = M_m + M_p$$

$$M_m = \left[M_m^{HAT}, M_m^{RHip}, M_m^{LHip}, M_m^{RKnee}, M_m^{LKnee}, M_m^{RAnkle}, M_m^{LAnkle} \right]^T \qquad (5.37)$$

$$M_p = \left[M_p^{HAT}, M_p^{RHip}, M_p^{LHip}, M_p^{RKnee}, M_p^{LKnee}, M_p^{RAnkle}, M_p^{LAnkle} \right]^T$$

$$M_r = \left[M_r^{RHip}, M_r^{LHip}, M_r^{RKnee}, M_r^{LKnee}, M_r^{RAnkle}, M_r^{LAnkle} \right]^T$$

For ground reaction force and friction force, we can use Jacobian matrix, i.e., Equation (5.27) to translate it to joint torques, which is denoted by $\boldsymbol{Q}_{N3}$.

The generalized force $\boldsymbol{Q}_N$ acting on DOFs is a sum of generalized forces:

$$\boldsymbol{Q}_N = \boldsymbol{Q}_{N1} + \boldsymbol{Q}_{N2} + \boldsymbol{Q}_{N3} \qquad (5.38)$$

The equation represent the forces at a specific time instant t; for brevity, the dependence on t is omitted from these equations.

Step4: Derive the equations of motion

Let us begin to derive the equation of motion according to the Equation (5.28). First, the partial derivative of the Lagrangian with respect to the joint velocity of joint k:

$$\frac{\partial L}{\partial \dot{q}_k} = \sum_j d_{kj} \dot{q}_j \qquad (5.39)$$

Here the j subscript notes columns and rows of the corresponding matrices. d_{kj} represents the jth row of the M matrix for the k-th joint. We then take the full derivative of the same by doing the following:

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{q}_k}\right) = \sum_j d_{kj} \ddot{q}_j + \sum_{i,j} \frac{\partial d_{kj}}{\partial q_i} \dot{q}_i \dot{q}_j \qquad (5.40)$$

We also take the partial derivative of the Lagrangian with respect to the k-th joint position:

$$\frac{\partial L}{\partial q_k} = \frac{1}{2} \sum_{i,j} \frac{\partial d_{ij}}{\partial q_k} \dot{q}_i \dot{q}_j - \frac{\partial P}{\partial q_k} \qquad (5.41)$$

These four components can then be laid out in the following way:

$$\sum_j d_{kj}\ddot{q}_j + \sum_{i,j}\left\{\frac{\partial d_{kj}}{\partial q_i} - \frac{1}{2}\frac{\partial d_{ij}}{\partial q_k}\right\}\dot{q}_i\dot{q}_j + \frac{\partial P}{\partial q_k} = Q_{nc} \tag{5.42}$$

If we take the first summation to equal M, the two expressions in the brackets as C, and the last part as G, the dynamic model of the model is then described by Lagrange's equation (5.43):

$$M(q)\ddot{\boldsymbol{q}} + C(q,\dot{q})\dot{\boldsymbol{q}} + G(\boldsymbol{q}) = \boldsymbol{Q}_{nc} \tag{5.43}$$

The $\boldsymbol{Q}_{nc}$ is non-conservative force vector shown in Equation (5.38); $\boldsymbol{q}, \dot{\boldsymbol{q}}, \ddot{\boldsymbol{q}},$ are 9×1 vectors of displacement, velocity and acceleration; M is a pose-dependent, symmetric and positive definite 9×9 inertia matrix of the skeleton, and comes from taking the second derivative the kinetic energy. It uses the input of joint acceleration, thus this factor represents torque based on angular force. The next factor C represents what are called the Coriolis and centrifugal or coupling effects of the manipulator system on joint torques, which is a 9×1 vector. Note it includes both angular position and velocity terms. The 9×1 G matrix includes forces based on the influence of gravity. In order to process data conveniently for subsequent chapter, we rewrite Equation (5.43) as the following:

$$M(q)\ddot{\boldsymbol{q}} = \boldsymbol{b} + \boldsymbol{Q}_{nc}$$
$$\text{And } \boldsymbol{b} = -[C(q,\dot{q})\dot{\boldsymbol{q}} + G(\boldsymbol{q})] \tag{5.44}$$

Step 6. Solve for the equations of motion.

Since there are 9 generalized coordinates in the simplified model, i.e., $q_i(k = 1,2\dots9)$, 9 differential equations are expected. They are generated with the symbolic calculation of Matlab, resulting in a system of nine equations. We need to change original equation to different formulations depending on the different applications. In all, there are several approaches to use the derived equations of motion: (1) Forward dynamic; (2) Inverse dynamic and (3) Mixed dynamics.

5.2.4 Model Validation

The section will valid whether or not the developed mathematical models are correct.

● Validation of Kinematical Model

The part validates the kinematical model by gait study of a normal, healthy individual (70 kg, 0.90 m leg-length) who walks at 1 m/s. Gait is defined as alternating phases of stance and swing, with the strictly monotonic and horizontal displacement of the human COM[146]. Here we adopt the data found in literature[74] to drive the model to perform walking. The human data for a

typical walking cycle were obtained in the form of independently collected CGA data. After getting the original data, then we change the data to adapt to the model developed in the work, for example, the joint angle definitions (knee and ankle joint angles) are different from our model. The definition of joint angle adopted in our model is shown in Figure 5.6. The gait simulation result is presented in Figure 5.10 .It shows a simplified diagram of human walking gait in a whole cycle. The results demonstrate the validity of the kinematical mathematical model well.
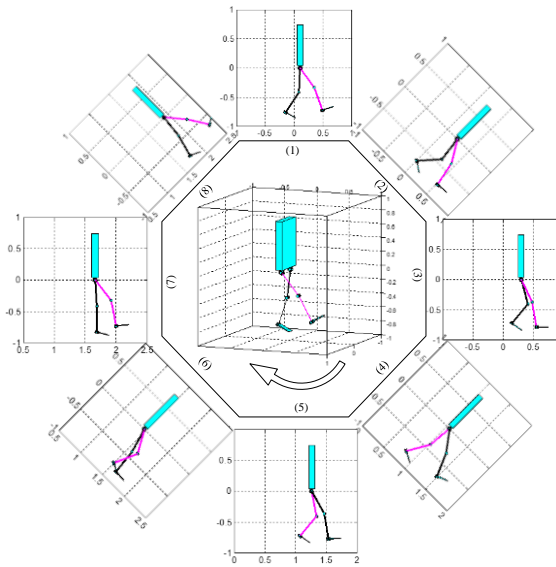

Figure 5.10 Simulation result of a gait cycle typically represented as starting (0%) and ending (100%) at the point of heel strike on the same right foot. (Pink line is right leg and black line is left leg)

● Validation of Kinetic Model

The validation of dynamic model is carried out by calculating an example that gets from reference[147] and comparing the calculated result with that calculation result given in the literature. The Figure 5.11 presents a schematic of two rigid segments connected to a non-translating cylinder, which is the same with "FIGURE 7.5 in literature[147]". The following gives a simple description for the instance. To get the further detailed descriptions of the example, you can refer to original literature above.

In the Figure 5.11, Segment1 of length L_3 is connected to a non-translating cylinder denoted by Segment0 at point B by way of a hinge joint having only one rotational axis along the z axis, and another additional segment2 of length L_4 is added to the segment1 at point C. Segment0 is free to rotate about the vertical axis, i.e., y axis, at some angular speed $\boldsymbol{\omega}_0$.
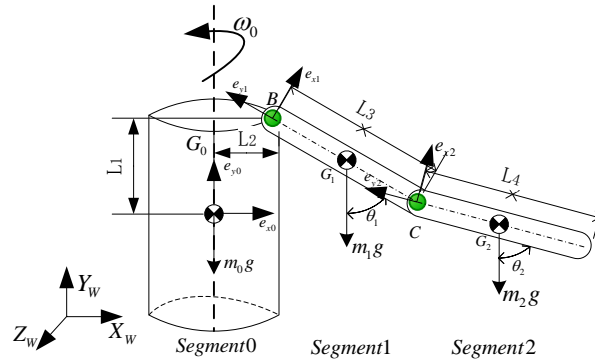
Figure 5.11 A schematic of two rigid segments connected to a non-translating cylinder

Where θ_1 is the angle between the segment1 and the vertical and $\dot{\theta}_1$ is the time rate of change of the angle. Also, the components for the mass moment of inertia about point G_1 in the $\boldsymbol{e}_{x1}$, $\boldsymbol{e}_{y1}$, $\boldsymbol{e}_{z1}$ frame of reference are

$$I_{xx} = I_{zz} = \frac{1}{12} m_i l_i^{\,2} (i = 1, 2), I_{yy} = 0 \tag{5.45}$$

To solve the problem using the method presented in the work, we need to start with the adjustment of changing the given variables θ_1, θ_2 to correspond with the generalized coordinates adopted in my work. Their changing equations are shown in Equation (5.46) and its schematic is shown in Figure 5.12.

$$q_1 = \theta_0, q_2 = \theta_1 \text{ , and } q_3 = \theta_2 - \theta_1 \tag{5.46}$$

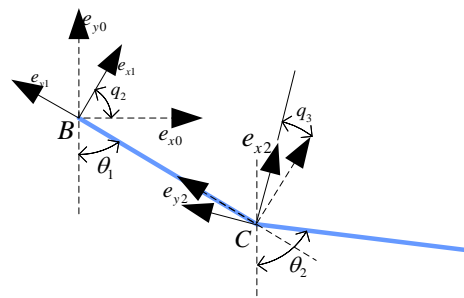Figure 5.12 Schematic to adjust the variables' corresponding relation

The following presents the results obtained by using the developed dynamic analysis model in the work. As indicated previously, the whole processing process including modeling and solving the differential equation is implemented in MATLAB. For the first generalized coordinate, $q_2 = \theta_1$, the equation of motion is solved to be Equation(5.47).

$$\left(\frac{1}{3}m_1 + m_2L_2\right)L_3^2\ddot{\theta}_1 + \frac{1}{2}m_2L_3L_4\cos(\theta_1 - \theta_2)\ddot{\theta}_2$$

$$-\left(\frac{1}{6}m_1 + \frac{1}{2}m_2\right)\omega_0^2L_3^2\sin2\theta_1 - \left(\frac{1}{2}m_1 + m_2\right)L_2L_3\omega_0^2\cos\theta_1$$

$$+\frac{1}{2}m_2L_3L_4\dot{\theta}_2^2\sin\theta_1\cos\theta_2 - \frac{1}{2}m_2L_3L_4\left(\omega_0^2 + \dot{\theta}_2^2\right)\cos\theta_1\sin\theta_2 \qquad (5.47)$$

$$= 0$$

For the second generalized coordinate $q_3 = \theta_2$, the equation of motion is solved to be Equation (5.48)

$$\frac{1}{2}L_4\ddot{\theta}_2 + \frac{1}{2}L_3\cos(\theta_1 - \theta_2)\ddot{\theta}_1 + \frac{1}{2}L_3\dot{\theta}_1^2\cos\theta_1\sin\theta_2$$

$$-\frac{1}{2}L_3\left(\omega_0^2 + \dot{\theta}_1^2\right)\sin\theta_1\cos\theta_2 - \frac{1}{6}L_4\omega_0^2\sin2\theta_2 \qquad (5.48)$$

$$-\frac{1}{2}L_2\omega_0^2\cos\theta_2 + \frac{1}{2}g\sin\theta_2 = 0$$

At this time ,you can comparing the obtained results with the results , "Equation 7.75 and 7.76 in chapter 7", presented in literature[147]. For Equation (5.48) above, we did a simplified processing by dividing by $(-m_2L_4)$ on both sides of the equation, which did not do it in original literature. And we should also note that the two results are exactly the same except that there are about three places existing difference. However, we have repeatedly solved the problem several times according to the steps presented in original document by hand, and finally proved us correct. Therefore, by using different methods to solve the same problem, and finally get the same results, we prove the kinetic analysis model developed in the work is correct.

5.3 Nomenclature

- ***Main member variables of <SEGMENT> Class:***

    i : Segment id, which is used to identify specific segment, a scalar starting from 1;

    im : Mother segment id of segment i ;

    ic : Child segment id of segment i ;

    C_i : COM coordinate of segment i ,3×1 vector;

    m_i : Mass of segment i ;

    I_i^c :Inertia of segment i about its COM;

    $^wP_i^c$:Position and Pose matrix of COM point in WRF;

- ***Main member variables of <JOINT> Class:***

    t : Running time(s);

    j : Joint id, which is used to identify specific segment, a scalar starting from 1;

jm ：Mother segment id of joint j ;

jc ：Child segment id of joint j ;

m_j ：Mass of joint j ;

S_j: DOF configuration for joint j, 6×1 vector (0 or 1 values in the vector);

q_j^t ：Linear and rotational displacement 6×1 vector of joint j at time t; the first three elements are translation variables along the sequence of x y and z axis of RF fixed in mother and positioned in origin; the others are rotation angle sequence around x y and z axis of the same RF above.

$^{j}v_G$ ：The velocity of any a point G in coordinate frame j, 3×1 vector;

$^{jm}r_{jO}$: Position vector of joint j point in mother RF, 3×1 vector;

$^{jm}A_{jO}$: Rotation angle (degree) sequence about x, y and z axis of RF fixed in mother segment and positioned in joint j, 3×1 vector;

q_j^0: Initial displacement vector of joint j at initial time (t=0), referencing to q_j^t;

Q_j^t: Total displacement vector of joint j at time t, which is the vector addition of q_j^0 and q_j^t.Meantime, need to take S_j into consideration, that is, the corresponding elements' value in Q_j^t are not change upon time at positions that are zero values in S_j;

$^{jm}v_{jO}^t$: Velocity vector of joint j point in mother RF (jm) at time t, 3×1 vector;

$^{k}\omega_j^t$: Angle velocity vector about x (k=1), y (k=2) and z (k=3) axis of RF(j) at time t with respect to it mother RF(jm), 3×1 vector $\begin{bmatrix} ^{1}\omega_j^t & ^{2}\omega_j^t & ^{3}\omega_j^t \end{bmatrix}'$;

$^{k}\xi_j^t$: Angle acceleration velocity vector about x (k=1), y (k=2) and z (k=3) axis of RF (j) at time t with respect to it mother RF (jm), 3×1 vector$\begin{bmatrix} ^{1}\xi_j^t & ^{2}\xi_j^t & ^{3}\xi_j^t \end{bmatrix}'$;

$^{j}a_G$ ：The acceleration velocity of any a point G in coordinate frame j, 3×1 vector;

5.4 Conclusion

In this chapter, we initially presented an anthropomorphic Linked Segment Digital Human Model (LS-DHM for short) of 19 segments and 46 DOFs (43 joint DOFs and 6 root DOFs), which was analogous to actual anatomic systems and can result gross motion events of the different simulations. After getting the LS-DHM, to prepare for the development of control system, the kinematical and kinetic analysis model of human-machine system were derived in the chapter, and they were coded by using recursive method in Matlab. Finally, the validation for the proposed model is performed.

Chapter 6.   Design of sHRI Based on NMS Model

The design of sHRI is the software/logical design of control system. In traditional training methods, therapists administer the therapy where they monitor the whole progress of the therapy exercise. The sHRI is just designed to accomplish the task autonomously so as to help the therapists to use their time more productively in higher-level decision making, decrease their workload and reduce the cost of treatment simultaneously.

6.1 Overview of Control Architecture

The work developed a cognitive Human-ERRobot interaction that allows a closed loop interaction schema. The cognitive interaction mode is biologically inspired and is done through neural excitation of given skeletal muscles. This approach takes full advantage of the human natural control mechanisms[14], and it is organized as the combination of an upper-level control system and a low-level control system. The low-level assistive controller performs rehabilitation tasks under the coordination of the upper-level controller. The upper-level controller processes events or tasks in the discrete domain, and the low-level controllers run in continuous domain. So this system is a hybrid system, which exhibits both continuous-time and discrete-event dynamics[148]. The system adopts FSM to coordinate the two systems. Moreover, in order to dynamically change different lower control laws at any time according to various control requirements, the switch system is adopted.

6.1.1   Lower Assistive Control System

Patients with function impairment of lower-extremity may not be able to track the desired motion trajectories that are predefined in executing an action task. Therefore, low-level assistive controllers are introduced to provide robotic assistance to patients' leg movement as and when needed to help him/her to complete the specified task. The design and selection of the lower controller depend on the type of training task (i.e., passive, active-assistive, active and resistive exercises), and the selected training task is decided based on rehabilitation assessment results. References such as [23] shows that kinematical control(i.e. position, velocity) law and force control law are required for performing exercises task. For example, the passive exercises require position control, active-assistive exercises require force and position control. The designed low level control block diagram in the work is shown in Figure 6.1, which comprises of motion controller (position, velocity control and they are applied or removed from the system by switch "Sw1" adjusting) and force controller (force control and it is applied or removed from the system by switch "Sw2" changing).
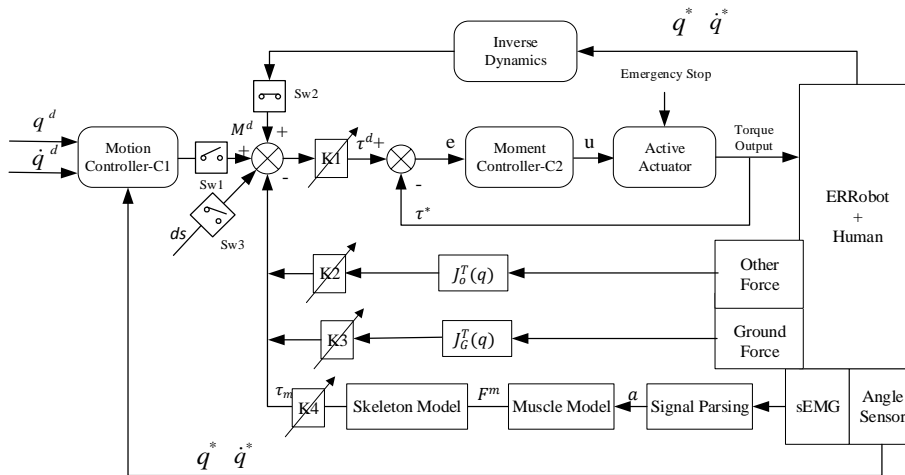
Figure 6.1 Overall architecture of the lower control system. Scaling factors K1, K2, K3 and K4 are in range of [0, 1]

According to the Figure 6.1, the functions supported by lower control system include the driving of joint actuators, reading of sensors data and etc. Their executions are organized and cooperated by the upper control system shown in Figure 6.2. In this system, an outer position loop is designed around an inner force feedback loop. The joint kinematical quantities such as joint angle, angular velocity are measured (or calculated), and then are sent to the motion controllers denoted by C1 or "Inverse Dynamics" module to compute the needed moment M^d for producing the given motion. Then drive torque τ^d of the ERRobot joint can be derived. To make actuators provide the torques precisely, we utilize the torques τ^* of actuators output axis as feedback signals. Furthermore, some scaling factors (i.e., K1, K2, K3 and K4, and all of them are in [0, 1]) are introduced in order to adjust dynamically the influence of corresponding factor. The work designs them as a single-input/single-output (SISO) system in joint-space (not in task space), and any coupling effects due to the motion of the other segments is treated as a disturbance denoted as ds in Figure 6.1. The $J_c(\boldsymbol{q})$ is a Jacobian matrix of the ground reaction force to transform it to every joint DOF; The τ_m is muscle moment, and $J_O(\boldsymbol{q})$ is transformation matrix of other force f_O to joints.

6.1.2 Upper Control System

The low-level assistive controller performs rehabilitation tasks under the coordination of the upper-level controller. The proposed overall architecture of the upper control system is shown in Figure 6.2, which contains several function modules.
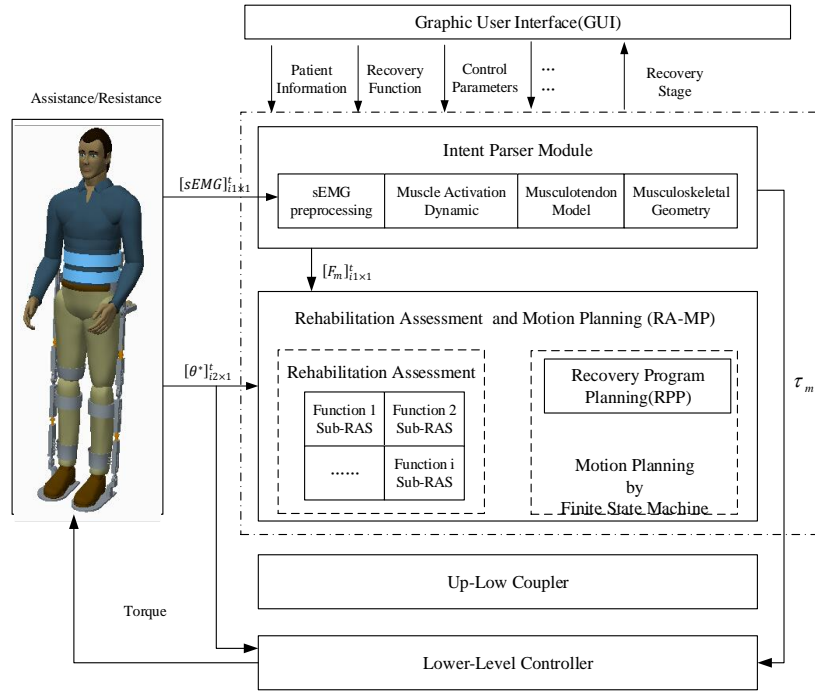
Figure 6.2 The Hierarchical control architecture of the ERRobot

In the figure, the $[\boldsymbol{sEMG}]_{i1\times1}^{t}$ is the sEMG signal of the $i1$ muscle at time t; the $[\boldsymbol{\theta}^*]_{i2\times1}^{t}$ is the joint angle of $i2$ joint at time t and the $[\boldsymbol{F_m}]_{i1\times1}^{t}$ is the muscle force of $i1$ muscle at time t; the $[\boldsymbol{\tau}]_{i2\times1}^{t}$ is the estimated torque of the user applying on the exoskeleton joint. The design details of these modules will be introduced in subsequent sections. To our knowledge, there has been no work on designing such a hybrid system for rehabilitation purposes. The upper-level controller plays a role of a PT, which monitors the task and assess whether the task needs to be updated, and creates the control commands used finally by lower-level controller. The presented upper-level controller is general for a large class of leg-related rehabilitation tasks. The work will take the rehabilitation of right lower-extremity in the walking as an example to explain the design details of the upper-level controller.

6.2 Lower-level Assistive Controller Design

In the work, we assume that the connection between leg and robot link is rigid, so the integrated system of lower limbs and robot rods is studied. According to Equation (5.38) and (5.43), the dynamics of Human-ERRobot system can be written as Equation (6.1). In the model construction, we neglect the flexibility[149]of each active DOF joint connection between rotor and segment.

$$M(q)\ddot{q} + C(q,\dot{q})\dot{q} + G(q) = \boldsymbol{Q}_{N1} + \boldsymbol{Q}_{N2} + \boldsymbol{Q}_{N3} \qquad (6.1)$$

Where the vectors $\boldsymbol{q}$ contains the leg segment positions; $\boldsymbol{Q}_{N1}$：joint internal moments including

muscle and passive joint torques; $\boldsymbol{Q}_{N2}$: the added joint torques by robot; $\boldsymbol{Q}_{N3}$:generalized torques due to ground reaction force; If other non-conservation generalized forces are existed, they are also added here. We also introduce another item $\boldsymbol{Q}_d$ to represent other disturbance factors, and in order to distinguish from other symbols, here we replace the symbol $\boldsymbol{Q}_{N2}$ with $\boldsymbol{\tau}(t)$ to represent the motor driven torque for operator active joints. So, we rewrite system dynamic equation as Equation (6.2):

$$M(q)\ddot{\boldsymbol{q}} + C(q,\dot{q})\dot{\boldsymbol{q}} + G(\boldsymbol{q}) = \boldsymbol{Q}_{N1}(q,\dot{q}) + \boldsymbol{Q}_{N3}(q) + \boldsymbol{Q}_d + \boldsymbol{\tau}(t) \qquad (6.2)$$

According to descriptions in previous chapter, the $\boldsymbol{Q}_{N1}$ includes joint muscle force (related with $q, \dot{q}$) and joint passive force (related with $q, \dot{q}$); the $\boldsymbol{Q}_{N3}$ is the joint generalized torque of ground reaction forces that is transformed by the transpose of Jacobian matrix J (related with q,). For simplification, we continue to simplify the Equation (6.2) as Equation (6.3).

$$M(q)\ddot{\boldsymbol{q}} + \boldsymbol{N}(q,\dot{q}) = \boldsymbol{Q}_d + \boldsymbol{\tau}(t) \qquad (6.3)$$

Where

$$\boldsymbol{N}(q,\dot{q}) = C(q,\dot{q})\dot{\boldsymbol{q}} + G(\boldsymbol{q}) - \boldsymbol{Q}_{N1}(q,\dot{q}) - \boldsymbol{Q}_{N3}(q) \qquad (6.4)$$

Which represents a vector of the nonlinear terms. The objective of ERRobot control is generally to select the control torques $\tau(t)$ so that the robot follows a desired prescribed motion trajectory or exerts a desired force. It should be noted that the right items of Equation (6.2) is various according to the system state. For walking control, the work divides gait into two phases in order to model the dynamics of the lower limbs: the swing phase and the stance phase. During swing phase of gait, the item $\boldsymbol{Q}_{N3}$ of the ground reaction forces is removed.

6.2.1 Motion Controller

The control objective for kinematical quantities, for example, position tracking controlling can be achieved by first defining a desired trajectory $\boldsymbol{q}_d(t)$, which is given in terms of the time sequence of the joint motion, that is, joint space control schemes are adopted in the system. It is from upper controller in the work, based on a high-level task decomposition. This section discusses the real-time motion control problem assuming that $\boldsymbol{q}_d(t)$, is given. The state-space form of the system can be obtained as Equation (6.5) according to Equation (6.2).Here we define the state vector $\dot{x} = [\boldsymbol{q}^T \quad \dot{\boldsymbol{q}}^T]^T$, as shown in Equation (5.4) (5.7).

$$\dot{x} = \begin{bmatrix} \dot{\boldsymbol{q}} \\ \ddot{\boldsymbol{q}} \end{bmatrix} = \begin{bmatrix} \dot{\boldsymbol{q}} \\ M(q)^{-1}(-N(q,\dot{q}) + \boldsymbol{Q}_d) \end{bmatrix} + \begin{bmatrix} 0 \\ M(q)^{-1} \end{bmatrix} \boldsymbol{\tau}(t) \qquad (6.5)$$

Where the input vector $\boldsymbol{u}(t) = \boldsymbol{\tau}(t)$. The equation is very convenient for many applications,

including the derivation of suitable control laws and computer simulation. For the matrix inversion $M(q)^{-1}$, it is calculated analytically off-line, to reduce the on-line computational burden. An alternative linear state-space equation in the form $\dot{x} = Ax + Bu$ can be defined as:

$$\dot{x} = \begin{bmatrix} \dot{q} \\ \ddot{q} \end{bmatrix} = \begin{bmatrix} 0 & I \\ 0 & 0 \end{bmatrix} x + \begin{bmatrix} 0 \\ I \end{bmatrix} u \tag{6.6}$$

With $u(t) = M(q)^{-1}(-N(q, \dot{q}) + Q_d + \tau(t))$. This is known as the Brunovsky Canonical Form. It is assumed that the desired motion trajectory for each joint $q^d(t)$ $\dot{q}^d(t)$, as indicated above, is prescribed by upper control system. Define the tracking error as

$$e(t) = q^d - q^*$$
$$\dot{e}(t) = \dot{q}^d - \dot{q}^* \tag{6.7}$$

According to Equation (6.6), we can get the following the flowing set of equations:

$$\begin{cases} \dot{x}^d = \dfrac{d}{dt}\begin{bmatrix} q^d \\ \dot{q}^d \end{bmatrix} = \begin{bmatrix} 0 & I \\ 0 & 0 \end{bmatrix}\begin{bmatrix} q^d \\ \dot{q}^d \end{bmatrix} + \begin{bmatrix} 0 \\ I \end{bmatrix}\ddot{q}^d & (1) \\[4mm] \dot{x}^* = \dfrac{d}{dt}\begin{bmatrix} q^* \\ \dot{q}^* \end{bmatrix} = \begin{bmatrix} 0 & I \\ 0 & 0 \end{bmatrix}\begin{bmatrix} q^* \\ \dot{q}^* \end{bmatrix} + \begin{bmatrix} 0 \\ I \end{bmatrix}\ddot{q}^* & (2) \end{cases} \tag{6.8}$$

Through Equation (1)-(2)

$$\frac{d}{dt}\begin{bmatrix} e(t) \\ \dot{e}(t) \end{bmatrix} = \begin{bmatrix} 0 & I \\ 0 & 0 \end{bmatrix}\begin{bmatrix} e(t) \\ \dot{e}(t) \end{bmatrix} + \begin{bmatrix} 0 \\ I \end{bmatrix} u \tag{6.9}$$

Where

$$u = \ddot{q}^d - \ddot{q}^* = \ddot{q}^d - M(q^*)^{-1}(-N(q^*, \dot{q}^*) + Q_d + \tau(t)) \tag{6.10}$$

We define the input control variable:

$$u(t) \equiv \ddot{q}^d + M(q^*)^{-1}(N(q^*, \dot{q}^*) - Q_d - \tau(t)) \tag{6.11}$$

A two-step design procedure now suggests itself. First, use linear system design techniques to select a feedback control u(t) that stabilizes the tracking error system in Equation(6.9), then compute the required torques using the inverse of Equation(6.10), namely,

$$\tau(t) = N(q^*, \dot{q}^*) - Q_d - M(q^*)(\ddot{q}^d - u) \tag{6.12}$$

This is a nonlinear feedback control law that guarantees tracking of the desired trajectory. It relies on computing the torque τ that makes the nonlinear dynamics of Equation (6.5), equivalent to the linear dynamics of Equation (6.6), which is termed feedback linearization. If a proportional-derivative (PD) feedback for $u(t)$ is selected, that is,

$$\boldsymbol{u}(t) = k_p \text{e} + k_d \dot{\text{e}} \tag{6.13}$$

Then the PD computed-torque controller is:

$$\boldsymbol{\tau}(t) = N(q^*, \dot{q}^*) - Q_d + M(q^*)\big(k_p \text{e} + k_d \dot{\text{e}} - \ddot{q}^d\big) \tag{6.14}$$

According to Equation (6.10), it yields the tracking error dynamics $\ddot{e} = k_p e + k_d \dot{e}$, which is stable as long as the derivative gain matrix k_d and the proportional gain matrix k_p are selected positive definite. It is common to select the gain matrices diagonal, so that stability is ensured as long as all gains are selected positive. The PD computed-torque controller is shown in Figure 6.3, which has a multi-loop structure, with a nonlinear inner feedback linearization loop and an outer unity-gain tracking loop. Note that there are actually n outer loops, one for each joint.
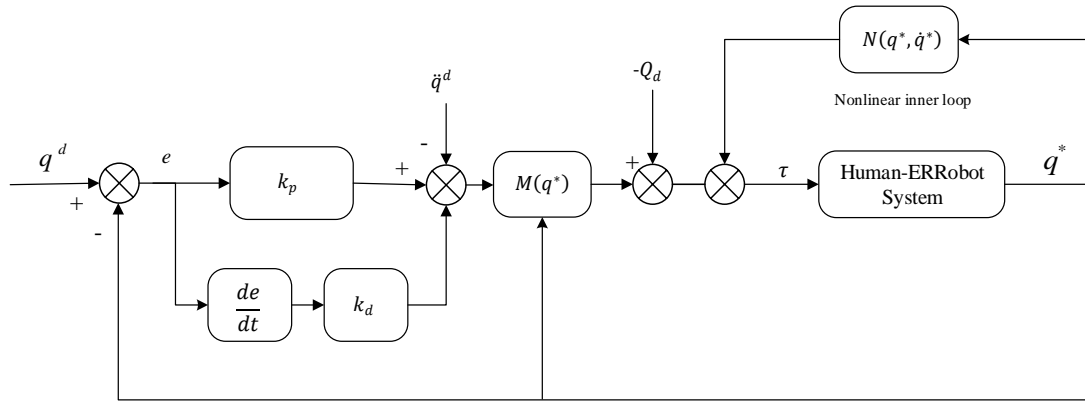
Figure 6.3 PD computed-torque controller

If an integral term is now added to each joint controller to achieve PID control, which can improve steady-state tracking accuracy. It is worth noting that the implementation of computed torque control requires that parameters of the dynamical model are accurately known and the control input is computed in real time. In order to avoid those problems, several variations of this control scheme have been proposed, for example, computed-torque-like control.

$$\boldsymbol{\tau}(t) = \hat{N} - \hat{M}\big(\ddot{q}^d - k_p \text{e} - k_d \dot{\text{e}}\big) \tag{6.15}$$

Where $\hat{M}$, $\hat{N}$ are approximations, estimates, or simplified expressions for $M(q^*)$, $N(q^*, \dot{q}^*)$. Another computed torque-like controller is to neglect all nonlinearities items and one selects simply as:

$$\boldsymbol{\tau}(t) = k_p e + k_d \dot{e} \tag{6.16}$$

With the gain matrices diagonal, so that all the joints are decoupled. The simplified classical joint controller is very easy to implement, as no digital computations are needed to determine nonlinear

terms. According to the general control system framework adopted in the work, as shown in Figure 6.1, in order to perform the most simplest motion control law, switches Sw1 is closed, Sw2 is opened, and scaling factors K2,K3,K4 are set to zero to remove the influences from non-linear elements. The motion controller C1 is a PD controller, which will compute a torque that is linearly proportional to the difference between the current state and a desired state. It takes into account both how far a joint angle is from its target, as well as how fast it is currently moving. The desired torque is formulated as Equation (6.16) that is rewritten as Equation (6.17) combined with the tracking error definition Equation (6.7) . Proportional-integral-derivative (PID) control method is used as the moment controller in the work, whose output u is the torque control volt, as shown in Equation (6.18).

$$\text{C1}: M^d = K_{po}(q^d - q^*) + K_{do}(\dot{q}^d - \dot{q}^*) \tag{6.17}$$

$$\text{C2}: u = \left(K_{pu} + \frac{K_{du}}{s} + K_{iu}\frac{N \cdot s}{s + N} \right)\Delta\tau \tag{6.18}$$

Where, gains K_{po}, K_{do} equal the proportional and differential outer loop motion control values; K_{pu}, K_{du} and K_{iu} define the inner loop pressure PID control values with $\Delta\tau(= \tau^d - \tau^*)$ as the pressure error; N is a filter coefficient.

6.2.2 Force-Assisted Controller

This force-assisted based controller can make the participant be able to walk voluntarily, that is, the patient can accomplish free walking movements. Its desired supporting force M^d is calculated by the inverse dynamics of the Human-ERRobot system based on joint motion feedback, as shown in (6.19) , which is the sum of joint torques applied on corresponding joint. In order to perform the control law, switches Sw1 is opened, Sw2 is closed, and scaling factors K1, K2, K3 and K4 are assigned proper values.

$$\boldsymbol{M^d} = M(q)\ddot{\boldsymbol{q}} + C(q,\dot{q})\dot{\boldsymbol{q}} + G(\boldsymbol{q}) \tag{6.19}$$

Next, the assistive torques $\boldsymbol{\tau}(t)$ from ERRobot can be calculated by Equation (6.3), whose accuracy is greatly dependent on human parameters of lower extremity, the estimation of muscle forces and so on. Therefore, we should adopt some methods to decide their corresponding scaling factors Ki (i=1, 2, 3, 4).

6.3 Upper-Level Control System Design

The upper-level control unit is a PC-based supervisory module in the proposed system. In the system, a hierarchical dynamic reactive controller is proposed for controlling the Human-ERRobot object. Several main modules are included in the scheme: (1) Module of Motion Intent

Parse (MIP); (2) Rehabilitation Assessment Module (RAM) and Rehabilitation Program Planning (RPP); (3) Motion Planning Module (MPM) that is the integration of the decomposition of rehabilitation task, control command production and output. The MPM is implemented by FSM, which is organized as three level structures and finally creates the corresponding reference trajectories along with some other necessary parameters like the proportional and derivative gains of the feedback controller. Next, these parameters are fed into lower controller as the control commands. A switching control module is adopted to determine which controller is to be connected in closed loop with the plant. It shows three control laws [23]: position, velocity and torque. The activation of these control laws depends on the value of "Cmd" (Cmd1 $\in$ 1, 2, 3), and their execution needs the reference trajectories Ref_j ($j = 6\ joints$) and control coefficient matrix k_{jm} ($j = 6\ joints; m = 5$ coefficients) delivered by upper controller, as expressed in Equation (6.17)(6.18). For the discussions of smooth switching mechanism and keeping system stability, you can refer to [32, 68, 150-152]. The reference [148] indicates that the stability of the switched system can usually be ensured by keeping each controller in the loop long enough to allow the transient effects to dissipate.

6.3.1 MIP Module

The module of MIP is responsible for determination of the desired action of the operator. As stated above, sEMG is adopted to detect the user's movement intention in the work. According to the control law of lower control system, this module is realized by different methods. (1) When the lower controller performs motion control such as trajectory tracking control, this module mainly completes the pattern recognition task by means of related muscle sEMG, and send motion data to lower controller to be used as its reference data. The work adopts the SVM method to execute the task, whose classifying process is introduced in Chapter 3; (2) when the force control law is implemented, the module is mainly to predict the joint torque so as to recognize the patient's action intention, since force is the cause of motion production. The function module uses the sEMG as input signals to recognize the operators' motion intention. In addition, the estimation of the joint torques is completed by using a realistic musculoskeletal model. The process is: given input excitation signals that are calculated by sEMG, each muscle generates a contraction force. Finally, the skeleton produces articulated motion in response to the internal muscle forces and external environmental forces, such as gravity and applied forces, which is completed by the forward dynamic simulation of the musculoskeletal system. The signal flow from the sEMG to the output of muscle moment is shown in Figure 4.1, which mainly comprises of three distinct parts, that is, signal processing of sEMG (Equation (6.20) presented in Chapter 3), muscle dynamic model (Equation (6.21) presented in Chapter 4) and Human-ERRobot System Dynamics

model (Equation (6.22) presented in Chapter 5).

$$u_i(t) = \frac{\left|sEMG_i(t) - sEMG_i^{rest}\right|}{sEMG_i^{max} - sEMG_i^{rest}}, i = 1,2 \dots 8 \tag{6.20}$$

$$\frac{da_i(t)}{dt} + \frac{1}{\tau_{act}}\left[\frac{\tau_{act}}{\tau_{deact}} + \left(1 - \frac{\tau_{act}}{\tau_{deact}}\right)u_i(t)\right]a_i(t) = \frac{1}{\tau_{act}}u_i(t), i = 1,2 \dots 8$$

$$\frac{d\boldsymbol{F}_i^T}{dt} = f\left[\boldsymbol{F}_i^T, L_i^{MT}, V_i^{MT}, a_i(t)\right], i = 1,2 \dots 8 \tag{6.21}$$

$$M(\boldsymbol{q})\ddot{\boldsymbol{q}} + C(\boldsymbol{q}, \dot{\boldsymbol{q}})\dot{\boldsymbol{q}} + G(\boldsymbol{q}) = \boldsymbol{Q}_{N1} + \boldsymbol{Q}_{N2} + \boldsymbol{Q}_{N3}$$

$$\boldsymbol{Q}_{N1} = D(\boldsymbol{q})\boldsymbol{F}^T + M_p(\boldsymbol{q}, \dot{\boldsymbol{q}}) \tag{6.22}$$

$$\boldsymbol{Q}_{N3} = [J_v(\boldsymbol{q})]^T \boldsymbol{F}_G$$

Where,

$sEMG_i^{rest}$: Represents the signal at limb rest;

$sEMG_i^{max}$: Represents the sEMG signal when the maximum muscle contraction occurs, which is calculated by mean value of $sEMG_i(t)$;

$\boldsymbol{q}, \dot{\boldsymbol{q}}, \ddot{\boldsymbol{q}}$: Vectors of displacement, velocity and acceleration;

$M(\boldsymbol{q})$: Inertia matrix of the Human-ERRobot system;

$C(\boldsymbol{q}, \dot{\boldsymbol{q}})$: Represents what are called the Coriolis and centrifugal or coupling effects of the system on joint torques;

$G(\boldsymbol{q})$: Forces matrix based on the influence of gravity;

L^{MT}: Length of the muscle-tendon actuator;

V^{MT}: Velocity of the muscle-tendon actuator;

a : Muscle activation and $a(t) \in (0,1]$;

$\boldsymbol{Q}_{N1}$:Joint internal torques, which is the summation of joint muscle torque $M_m(a, \boldsymbol{q}, \dot{\boldsymbol{q}})$whose calculation formula is $D(\boldsymbol{q})\boldsymbol{F}^T$ and joint passive torque $M_p(\boldsymbol{q}, \dot{\boldsymbol{q}})$ whose calculation formula is given in Equation (5.9); $D(\boldsymbol{q})$ is the joint moment-arm matrix; $\boldsymbol{F}^T$: Vector of muscle-tendon actuator forces;

$\boldsymbol{Q}_{N2}$: Actuation torque for joints from ERRobot;

$\boldsymbol{Q}_{N3}$: Generalized torque due to ground reaction force $\boldsymbol{F}_G$,and their transition are completed by

Jacobean matrix $J_v(\boldsymbol{q})$, as expressed in Equation (5.27).

The Human-ERRobot system dynamics indicate that we have a second order system. We must convert it to a first-order system. Given a single leg (right side) is considered, then the state vector for the system can be written as:

$$X(t) = [\boldsymbol{q}_{1..3}, \dot{\boldsymbol{q}}_{1..3}, \boldsymbol{F}_{1..8}, \boldsymbol{L}_{1..8}^{MT}, \dot{\boldsymbol{L}}_{1..8}^{MT}, \boldsymbol{a}_{1..8}]^T \tag{6.23}$$

Where $\boldsymbol{q}_{1..3}, \dot{\boldsymbol{q}}_{1..3}$ represent the vectors of three joints' angles, angular velocity, which are the simplification of $[q_1, q_2, q_3]$ and $[\dot{q}_1, \dot{q}_2, \dot{q}_3]$; $\boldsymbol{F}_{1..8}, \boldsymbol{L}_{1..8}^{MT}, \dot{\boldsymbol{L}}_{1..8}^{MT}$ and $\boldsymbol{a}_{1..8}$ are the vector simplification of eight MTCs' forces, length, velocity and activation. If we let this state vector be denoted by $X(t) = [\boldsymbol{x}_{1..3}, \boldsymbol{x}_{4..6}, \boldsymbol{x}_{7..14}, \boldsymbol{x}_{15..22}, \boldsymbol{x}_{23..30}, \boldsymbol{a}_{31..37}]^T$, we obtain a 37-dimensional nonlinear dynamical system of the form: $\dot{x} = f(x)$. When getting this form, we integrate it by using a Runge-Kutta integrator that is provided by many existing soft packages such as software MATLAB.

6.3.2   RAS and RPP Modules

In order to decide the most suitable therapeutic exercise type for the patient, the rehabilitation level of lower limb must be determined first[153]. Rehabilitation assessment can be classified into qualitative and quantitative evaluation. So far, researchers usually capture patients' some motion quantities to make an evaluation [153] or empirical evaluation. Some defects of these rating methods are that they are not able to specify causal muscle activities, and the latter relies on therapists' experience. To make up for these shortcomings, the paper proposes a quantitative evaluation method, which is realized by introducing two new concepts, that is "Health Belt" and HAI.

Much qualitative nature remains similar during the execution of function tasks, for example, the joint angle trajectories are similar during a gait cycle among different persons[74, 154]. We assume that these similar parameters are within a stable range for normal persons with similar age, the same gender. The value of deviation from the normal range is used to represent the extent of motor function loss of lower extremity. We call the normal range as "Health Belt" and the extent of motor function loss is called "HAI". To construct the "Health Belt" , we classify subjects into 12 groups[155] by gender (Male and Female) and age (i.e., A: [20,29], B: [30，39], C: [40,49], D: [50,59], E: [60,69], and F: [70,79]), which is based on the study of Oberg, Tommy etc.[155] who studied the gait of 233 subjects, aged 10-79 years, consisting of 116 men and 117 women. Here, we assume that those people at the same group have similar motor feature, physiology and some other variables. As indicated above, "Health Belt" is used to represent these similar

characteristics in the work, so people being in the same group are viewed to have similar "Health Belt". Therefore, the obtaining of "Health Belt" need to perform a lot of experiments under the help of doctors. The work will be performed in future. The following will only present the rehabilitation assessment method based on the two concepts, part of which have been published in my paper[153].

A "Health belt" is composed of a standard mean curve L and area that is between $\pm 2\sqrt{D(X)}$ distance from the standard curve L, which includes at least 75% of the data sampling point according to Chebyshev's Inequality shown as Equation (6.24) below. The schematic diagram of health belt can see Figure 6.4.

$$P(|X - E(X)| < \varepsilon) \geq 1 - \frac{D(X)}{\varepsilon^2} \qquad (6.24)$$

The meaning of the above inequality is that the probability of random variables X being in the ε neighborhood from mathematical expectation $E(X)$ is not less than $1 - D(x)/\varepsilon^2$. It is to say, you can make at least 75% sampling data being in "Health Belt" when assigning ε to $2\sqrt{D(X)}$. Here, depending on the parameter being measured, "Health Belt" can be classified into many types. For example, during the construction of our rating system, the work adopted joint angle health belt $\theta_{i1}^o(t) \pm \lambda_{i1}$, and muscle strength health belt $F_{i2}^o(t) \pm \delta_{i2}$, to construct our RAS. As described above, $\theta_{i1}^o(t)$ and $F_{i2}^o(t)$ are standard curve and λ_{i1} , δ_{i2} are belt width, all of which are represented in Figure 6.4. To build our RAS, the work firstly builds two independent assessment sub-system models by Equation(6.25). Two outputs are denoted by $M\theta(\theta(t))$ and $MF(F(t))$ respectively, and next basing on the two outputs, the comprehensive assessment score denoted by HAI value is derived by Equation (6.26)(6.27) .

$$Value := \int_0^t \sum_{i=1}^{N \ or \ M} \alpha_i (\Delta x_i'(t))^2 \, dt \approx T \sum_{j=0}^{k} \sum_{i=1}^{N \ or \ M} \alpha_i (\Delta x_i'(jT))^2 \qquad (6.25)$$

Let $t = kT \ (k = 0,1,2,...)$, we can get its discrete expression. The T is data sampling cycle. The N is number of measured joints, and M is the interested muscle numbers adopted to construct assessment system respectively. When $Value(= M\theta(\theta(t)))$ is calculated, $\Delta x_i'(t) = \Delta x_{i1}'(t) = \tilde{\theta}_{i1}(t) - \theta_{i1}^o(t) \ i1 = 1,2,...,N,$ and when $Value(= MF(F(t)))$ is calculated, $\Delta x_{i2}'(t) = \tilde{F}_{i2}(t) - F_{i2}^o(t) \ i2 = 1,2,...,M,$ Here $\tilde{\theta}_{i1}(t)$ and $\tilde{F}_{i2}(t)$ are the time normalization data(we adopt same method with sEMG time normalization described in Chapter 3) of corresponding sampling data based on $\theta_{i1}^o(t)$ and $F_{i1}^o(t),$ so as to make them have same phase at time. The α_i is weighting coefficient of each influencing factor on the overall health

assessment of lower limb.

From the calculation formula above, the two independent indexes $M\theta$ and MF are defined as the deviation square weighted sum of patients' sampled data in pre-defined time interval. Their outputs of $M\theta(\theta(t))$ and $MF(F(t))$ reflect the extent of deviation from normal motor function, so relatively large values mean the relatively poor performance of motor function.

$$HAI = f\left(M\theta(\theta_{i1}(t)), MF(F_{i2}(t))\right), i1 = 1,2, \dots, N; i2 = 1,2, \dots, M \qquad (6.26)$$

The map function "$f()$" is a discrete function and HAI is a serial of discrete values, which is defined as:

$$HAI = \begin{cases} 1 & MF \in [0, b_0) \, or \, M\theta \in (0, a_0], loading \, or \, unloading \\ 0 & MF \in [b_0, b_1] \, and \, M\theta \in [a_0, a_1], Add \, bigger \, resistance \\ 0.5 & MF \in [b_0, b_1] \, and \, M\theta \in [a_0, a_1], Add \, smaller \, resistance \\ 1 & MF \in [b_0, b_1] \, and \, M\theta \in [a_0, a_1], unloading \\ 2 & MF \in [b_0, b_1] \, and \, M\theta \notin [a_0, a_1], unloading \\ 3 & MF \notin [b_0, b_1] \, and \, M\theta \in [a_0, a_1], unloading \\ 4 & MF \in [b_1, b_2] \, or \, M\theta \in [a_1, a_2 - \gamma], unloading \\ 5 & MF \in [b_2, b_3] \, and \, M\theta \in [a_2 - \gamma, a_2], unloading \\ 6 & MF \in [b_3, b_4] \, or \, M\theta \in [a_2 - \gamma, a_2], unloading \end{cases} \qquad (6.27)$$

According to the Figure 6.4, the boundary values $b_i (i = 0, \cdots, 4)$ are calculated. $\Delta x'_{i2}(t) = F_{i2}(t) - F_4^{i2}$, and each of them is obtained under the following given conditions:

➢ $b_0 : F_{i2}(t) = F_{avg}^{i2} + \phi_{i2}, F_4^{i2} = max(F_{i2}^o(t) + \delta_{i2})$, where F_{avg}^{i2} is the mean value of i2-muscle strength, that is, $\int_{t1}^{t2} F_{i2}^o(t)dt/(t1 - t2)$ and δ_{i2} is a belt width. The ϕ_{i2} is a human specified value to approximately represent health belt width. Operating max () is to get the maximum value for given data;

➢ $b_1 : F_{i2}(t) = F_{avg}^{i2} - \phi_{i2}, F_4^{i2} = max(F_{i2}^o(t) + \delta_{i2})$;

➢ $b_2 : F_{i2}(t) = F_2^{i2}, F_2^{i2}$ is the i2-muscle strength when lower-extremity is able to start moving;

➢ $b_3 : F_{i2}(t) = F_1^{i2}, F_4^{i2} = max(F_{i2}^o(t) + \delta_{i2})$, where F_1^{i2} is the i2-muscle strength when lower-extremity is in a fully relaxed state. $b_3 : F_{i2}(t) = 0, F_4^{i2} = max(F_{i2}^o(t) + \delta_{i2})$

The $a_i (i = 0,1,2)$ are also values of boundary points calculated, and $\Delta x'_{i1}(t) = \theta_{i1}(t) - \theta_{max}^{i1}$, where $\theta_{max}^{i1} = max(\theta_{i1}^o(t) + \lambda_{i1})$; each of them is calculated under the following given conditions:
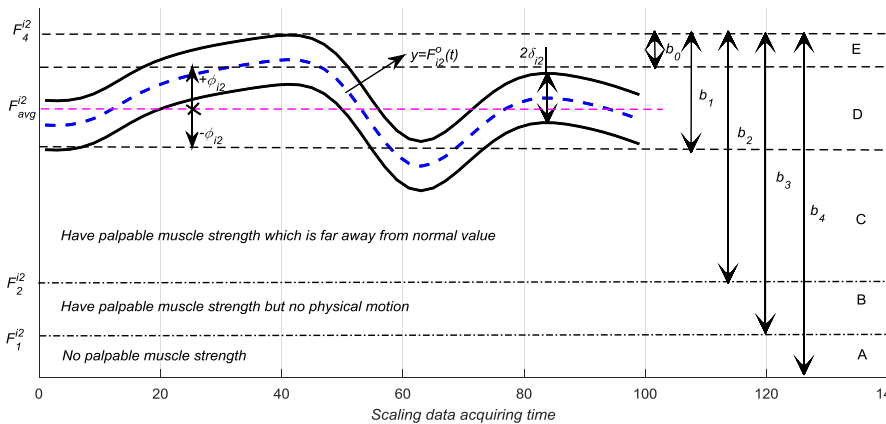
➢ $a_0 : \theta_{i1}(t) = \theta_{avg}^{i1} + \beta_{i1}, \theta_{avg}^{i1}$ is decided at same method with calculating F_{avg}^{i2}; The

meaning of β_{i1} is like ϕ_{i2};

- $a_1 : \theta_{i1}(t) = \theta_{avg}^{i1} - \beta_{i1}$;

- $a_2 : \theta_{i1}(t) = 0$. That is to say, it means no joint movement. Considering there exist sampling deviation, we introduce into an error area denoted by γ width .
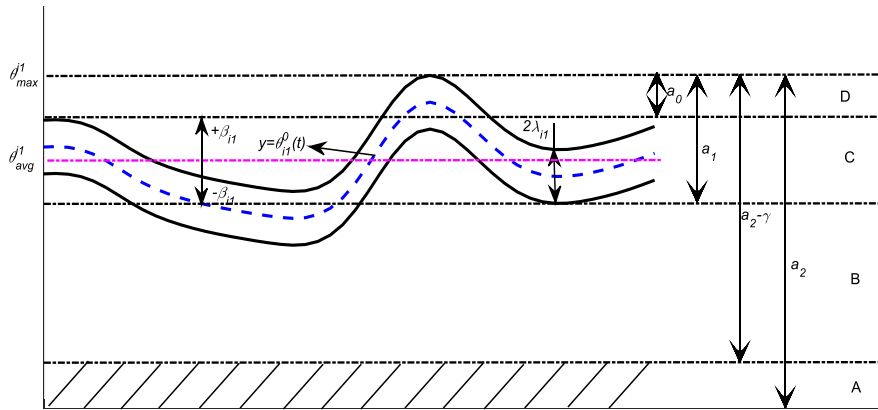
(a) Joint angle health belt

(b) Muscle force health belt

Figure 6.4 Schematic diagram of health belt and parameter specification

The RPP referred in the work is to make the executing procedure of therapeutic exercises (TEs) aiming to the specific rehabilitation task. The TEs are designed with the consultation of a physical therapist. After getting HAI values, the RPP system is activated. The function unit is responsible for the selection and allocation of therapy training, and finally produce actuator data. In traditional methods, the therapist continuously monitors the patient's progress and quickly determines if the task parameters are needed to be changed to adapt to the patients' performance. Then these task parameters are updated by the therapist. In the work, in order to make the robot automatically complete the decision-making process instead of the physicians, we introduce the unit of RPP.

The module of RPP is designed considering the need of the therapist and the patient and it can be easily modified and extended for new task requirements.

The type of rehabilitation exercise is decided according to the scores of HAI. Their relationship is shown in Table 6.1. Based on the HAI, we scale the healthy state of lower limb with seven different levels from 0 to 6, pulsing a middle stage of 0.5. Zero level represents the total health, whereas level six represents the paralysis. The type of the applied exercises changes according to this scale.

Table 6.1 The scores of HAI and corresponding therapeutic exercise

HAI	State	Sub-system outputs	Description
0	Total Health	$MF \in [b_0, b_1]$ and $M\theta \in [a_0, a_1]$	When maximal walking resistance (F_{max}) are applied, both joint movement and muscle forces are being in "Health Belt" at all sampling time(Active + Resistive)
0.5	Middle Health	$MF \in [b_0, b_1]$ and $M\theta \in [a_0, a_1]$	When smaller walking resistance (F_{small}) are applied, both joint movement and muscle forces are being in "Health Belt" at all sampling time(Active + Resistive)
1	Before Health	$MF \in [b_0, b_1]$ and $M\theta \in [a_0, a_1]$	In the context of unloading, joint movement and muscle forces are being in "Health Belt" at all sampling time (Active).
		$MF \in [0, b_0)$ or $M\theta \in (0, a_0]$	Sampling joint angle or muscle force is larger than given own reference of "Health Belt" with a smaller distance(Active + Resistive)
2	Mild	$MF \in [b_0, b_1]$ and $M\theta \notin [a_0, a_1]$	Muscle forces are being in "Health Belt" at all sampling time, but joint movement aren't being in "Health Belt"(Active)
3	Relative mild	$MF \notin [b_0, b_1]$ and $M\theta \in [a_0, a_1]$	Joint movement are being in "Health Belt" at all sampling time, but Muscle forces aren't being in "Health Belt"(Active).
4	Medium	$MF \in [b_1, b_2]$ or $M\theta \in [a_1, a_2 - \gamma]$	Have palpable joint movement and muscle strength without loading, but they are all being out of "Health Belt"(Active assistive)
5	Serious	$MF \in [b_2, b_3]$ and $M\theta \in [a_2 - \gamma, a_2]$	Have observable muscle contraction, but no observable joint motion(Passive)
6	Paralysis	$MF \in [b_3, b_4]$ or $M\theta \in [a_2 - \gamma, a_2]$	No palpable muscle contraction or no observable joint motion(Passive)

The rehabilitation task of walking can be comprised of two rehabilitation training sections, sitting rehabilitation training and walking rehabilitation training. So they are chosen as two discrete states, naming as "Sitting" and "Walking", respectively. Each of training section comprises several

training stages that are corresponding with the decomposition stages of movement, and also denoted by the names of state modules. In our application, only one state is active at any given time. To validate the executing logic, the Matlab/Simulink/Stateflow software is used to implement it.

6.3.3   MPM Module

The MPM is responsible for the creation of control commands, and it is accomplished by way of FSM method. For the realization of different motion controlling, we design corresponding function modules for the sake of modularization management. The organization method is conducive to an expansion of other rehabilitation tasks. In the work, the controller is organized as a three-level architecture. The first level is a data pre-processing stage, which is a preparation for the execution of the other two stages. The states included in the first level are called as "Preparation state". The second level is called as "TaskDecomposeLayer". The task-oriented structure design is adopted in FSM controller design. Aiming to every rehabilitation task, it is divided into different motion stages, and each of them is represented by a stage. For example, the walking can be divided into 9 states shown in Figure 6.5. The last level is responsible for the selection of control laws and control command production, output. For the FSM theory details, you can refer to some other textbooks or papers, such as[156].

This section will give its design process by taking the rehabilitation of walking ability as an example. The section will illustrate it by introducing the rehabilitation of the walking ability. Walking is a mixture of discrete and continuous control problems. The discrete control problem is the transition between events such as starting, stepping, and stopping. In order to link the discrete domain of upper-level controller and the continuous domain of low-level controller, the work adopts FSM method to construct the module. For simplification, we only study the gait assistance in sagittal plane. We do gait planning for the movement in the sagittal plane by using FSM, denoted by FSM-Walk.

In the work, 9 states are defined for walking control in sagittal plane, shown in Figure 6.5. For more detailed state division, you can refer to paper[43]. The work doesn't further design these intermediate states such as the shifting between right leg and left leg, because the balance control make sure by activating the cabling system. The work groups the 9 states into two classes, static states and transition states[157]. The static states consist of standing on two legs (S1), double support with right-leg-forward (RLF) (S2), and double support with left-leg-forward (LLF) (S3). The remaining 6 states are the transition between the three static states, include stand-to-walk with right half step (S4), stand-to-walk with left half step (S5), walk-to-stand with left half step

(S6), walk-to-stand with right half step (S7), right step (S8), and left step (S9). Each state in the FSM-Walk is fully defined by the combination of a set of trajectories and a set of joint feedback gains. These trajectories can be predefined based on offline simulations or captured gait data of healthy subjects[40] or online generated based on balance indicators such as ZMP. The work adopts the former two methods to provide the trajectories, which are saved in the data base and called the appropriate curves by the system of RPP. According to the state division, we divided these predefined trajectories into four parts and they corresponded to the different states, that is, the trajectories of stance (S1), double stance (S2, S3), starting gait (S4, S5) and ending gait (S6, S7), and normal swing (S8, S9). Many references such as [155] indicate that the complete gait cycle can be reconstructed under the assumption of bilateral symmetry. The phase difference between the left and right legs is half a gait cycle T, as follows:

$$\theta_{left}(t) = \theta_{right}(t + (2j+1)T/2), j = 0,1,2 \dots$$

Therefore, we simplify the gait data saving, that is, we only save these gait reference data of State S1, S2, S4, S7 and S8 into data base, and other reference trajectories of states (i.e., S3, S5, S6 and S9) are created in real time.
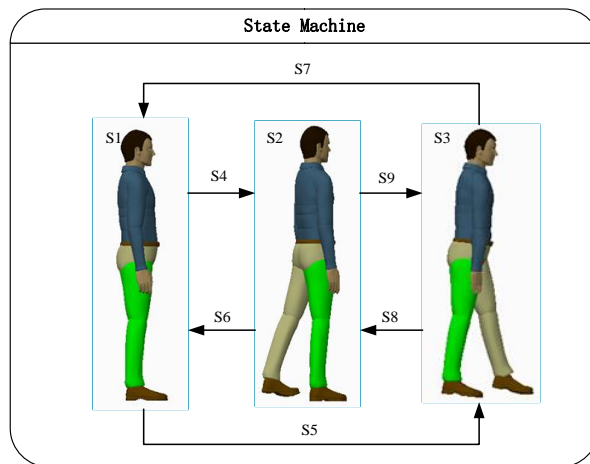

Figure 6.5 Gait state designs for the stance and gait assistance

For the transition states, they are designed to execute transition by two modes, manual and automatic transitions[43]. Manual transitions are initiated from the user (either the wearer or the operator) by control devices. For example, patients can start, stop walking and stepping by push-button interface. Automatic transitions allow to automatically switch from one state to another through some information such as Center of Press(CoP) positions [43, 157] in sagittal and frontal planes of the Human-ERRobot system which are estimated based on sensor data, the geometry and mass property of the ERRobot and the human anatomical data. Exactly speaking ,when values of the upper body orientation that is estimated by using the IMUs and relative joint angles between

segments that can be get through encoders installed in the lower extremities of the ERRobot are obtained, the pose of the virtual LS-DHM is decided and we can get the CoP value. As described above, a FSM method is used to make a transition from one state (mode) to another, if the condition defining the change is true.

6.4 Validation Experiment

In order to evaluate control strategies before implementing them in the real robot, we have designed several simulation experiments to evaluate the proposed control system sHRI, which consists of the validations of upper and lower control system.

6.4.1   Development of Simulation Framework

Simulation is a powerful approach for many aspects, so the work develops a simulation system to evaluate the proposed control system before applying them in the real robot, which is the economical and most usual selection for researchers. As introduced in Chapter 2, the simulation system is developed in Matlab/Simulink/Simmechanics/Stateflow (MathWorks 2015, 64-bit, windows) software environment, and two different methods are used to develop the simulation framework in the work, which are called "Code-based" and "Model-based" modelling methods, respectively. This first method to develop simulation system is finished by manual coding, which could be easily connected to other MATLAB toolbox or module like MATLAB/Simulink. It is flexible enough to build simulation model and do system analysis. At same time, it can also help in developing control algorithms. The latter method is done by using blocks provided by MATLAB/Simulink/SimMechanics. In the process of model developing, to get the kinetic quantities (mass and moments of inertia, etc.) of each segment, we build the detailed Human-ERRobot model using the Creo2.0 (3D CAD software, PTC Inc.) software package. Furthermore, to develop and validate model-based controller, we import the developed model into the Matlab environment by way of the data interchanging interface between them, which can simultaneously import into the mass, inertial, properties of different components and coordinate configuration, therefore, the integrated Human-ERRobot simulation model is not only a mathematical model but a physical model to some extent. When doing control experiments by using it, it is quite straightforward to add control functionalities to the model without needing additional mathematical model for the implementation. Based on experiments to be simulated, we always develop the corresponding control GUIs. Figure 6.6 shows a window manually operating a simplified human model and a real-time showing window.
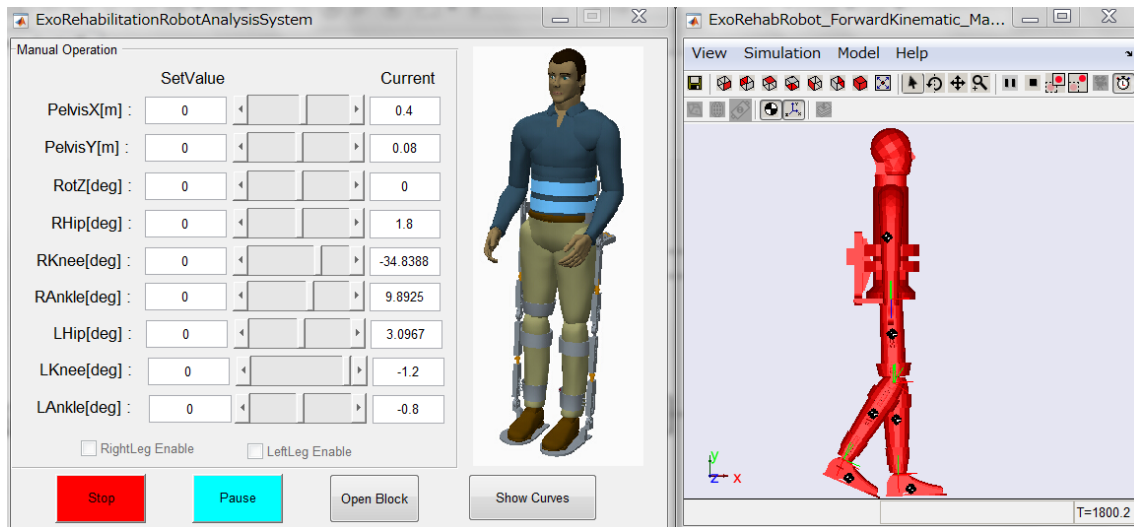
Figure 6.6 A window manually operating a simplified model and a real-time showing window

The modelled individual in the simulation is characterized by a height of about 1.7 m and a weight of 64.5 kg. The kinetic quantities (mass and moments of inertia, etc.) of each segment are summarized in Table 6.2. In the table, some parameters are given double values: the value placed

Table 6.2 Body segment anthropometrical data
-Length unit (m); Principal moment of inertia (kg.m2); Mass (Kg)

Segment	Notation	Value
HAT	L_{HAT}	0.890
	L_{RHip}	0.210
	r_1	0.356(0.322)
	m_1	47.8(55.9)
	I_1	3.56,0.947, 2.81(4.14, 1.18, 3.45)
r.Thigh	d_2	0.36
	L_{Thigh}	0.434
	r_2	0.202(0.180)
	m_2	8.51(9.7)
	I_2	0.149,0.0297, 0.137(0.183,0.0417,0.160)
r.shank	d_{31}	0.096
	d_{32}	0.384
	L_{Shank}	0.475
	r_3	0.224(0.192)
	m_3	6.02(8.38)
	I_3	0.111,0.0122, 0.111(0.160,0.0287,0.150)
r.foot	d_4	$L_{Foot}/2 = 0.120$
	L_{Foot}	0.240
	r_4	0.068(0.0775)
	m_4	2.16(3.76)
	I_4	0.00315,0.0146,0.0144(0.00984,0.0313,0.0286)

in parentheses is sum value of the ERRobot and human model segment, and the other value is the measured value of human lower limb. Here we illustrate the segment specifications by taking

right lower limb for an example, and the left part is obtained by symmetry. In software development, the all parameters are written to a configuration file, and are loaded as needed.

6.4.2   Validation of Upper Level Control System

The section conducts mainly some logical validation for the proposed upper control system. The MPM is the main running line, and it is constructed by FSM method based on the state decomposing of rehabilitation movement (or task) such as walking, climbing stairs. Here we
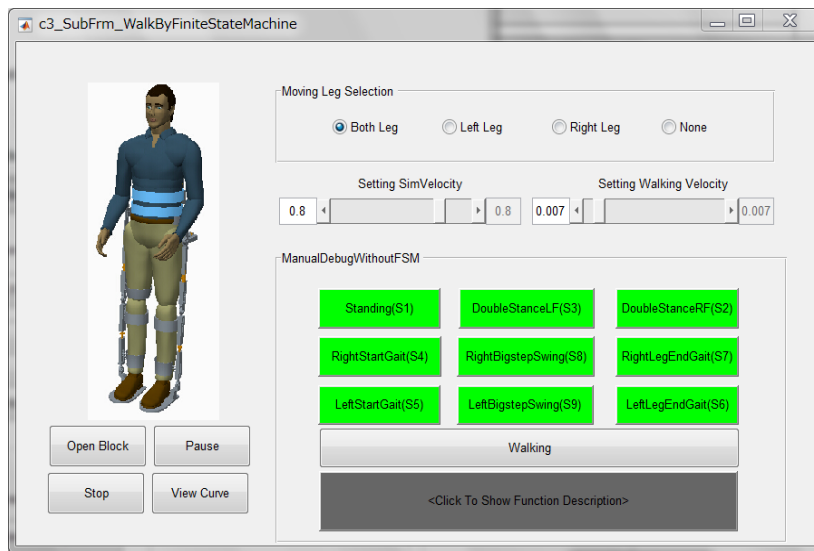
(a) A GUI designed for gait control by FSM
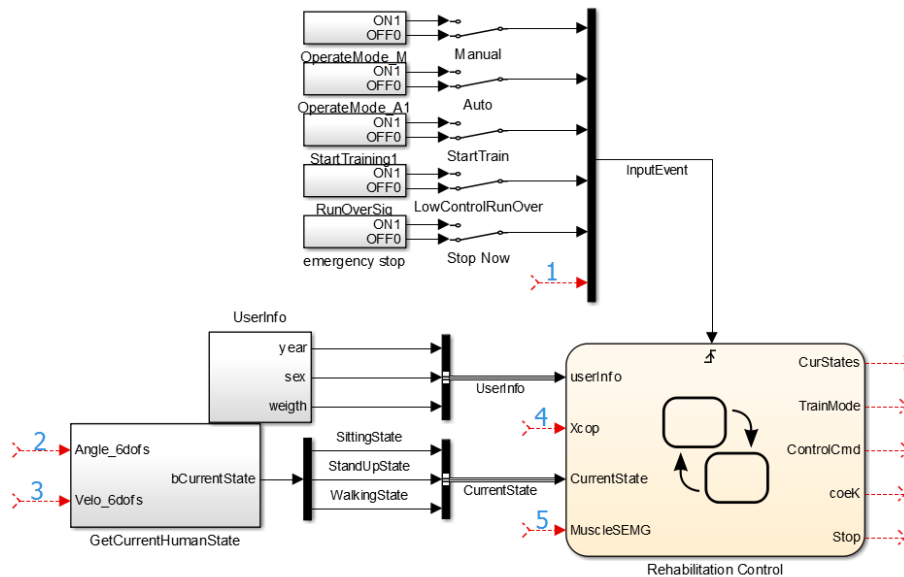
(b) Part of simulink control logic blocks that can be easily extended
Figure 6.7 StateFlow designed in Matlab/Simulink

validate the executing logic [24] by means of walking rehabilitation, most of which is

implemented by an environment of building graphical FSM. Moreover, a GUI (see Figure 6.7 (a)) is also developed in the experiment so as to better complete the interaction with users, whose daemon is accomplished by Simulink/Simmechanics blocks, as shown in Figure 6.7 (b). Statechart defines the logic of execution and provides a coherent and synthetic vision of the control flow, that is, animation highlights active objects in a chart as execution progresses.

In Figure 6.7(a), the button of "Open Block" can open the corresponding daemon that is a Simulink control model. In order to conveniently develop and maintain the program, we develop different Simulink block programs for every GUI application program. "Pause" and "Stop" buttons can control the simulation process in real. We can also show some necessary curves by click the button "View Curve" after the running simulation is over. For the daemon shown in Figure 6.7 (b), It can communicate with GUI program and a target system (i.e., reserved input interface ①~⑤). As shown in the figure, the block "Rehabilitation Control" is created by way of FSM. The block reacts to input events using a triggered chart, that is, all input events need to be added in the position like ①. Its other inputs represent the current system states. During the simulation implementation in the section, we replace the real signal by the manual input by means of Simulink signal input blocks, although the model can use an external mode communication channel to animate chart states by linking with lower controller.

Combing with Figure 6.5, the walking trajectories can be processed as three parts; stance/double stance (S1, S2, and S3), half step swing (S5, S7, S4, and S6) and total step swing (S9, S8). The reference curves are obtained from normal gait data of those healthy people[40] who are being in the same group. For stance/double stance state, their reference data adopt the boundary data of those adjacent state transition curves. These reference data must make the Human-ERRobot system being in an equilibrium posture. As an example, for the individual (64.5 kg, 1.7 m) with walking velocity of 1m/s, his reference trajectories in sagittal plane of half step swing state S4, that is, state transition from S1 to S2 are plotted in Figure 6.8, which consists of swing right leg joint curves (RHip, RKnee and RAnkle) and stance left leg joint curves (LHip, LKnee and LAnkle). The definition of angle positive direction and joint angle can refer to Figure 2.3 and Figure 5.6.

In the work, we execute the validation process in debug mode, and see whether the results behave as we expect. Through the output of created human model animation, we conclude that the control structure can complete the task implementation, and the control logic also be realized by the control system.
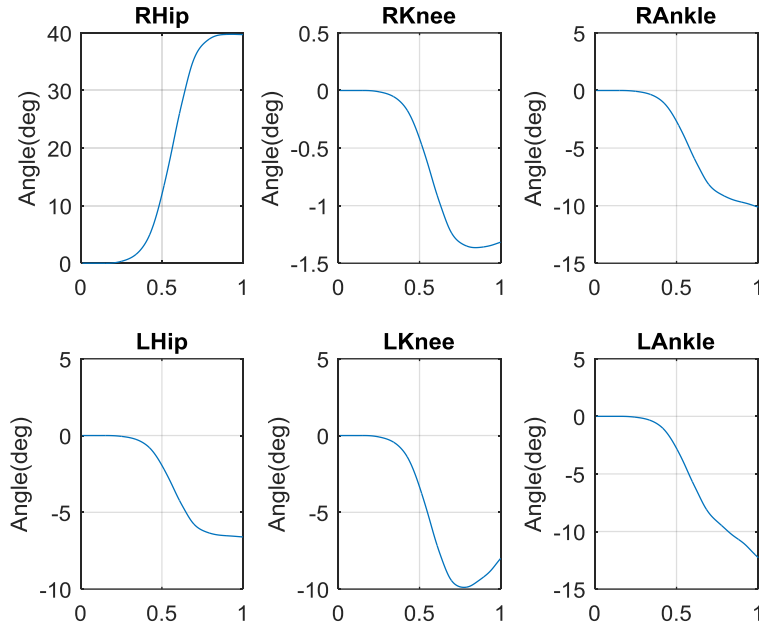
Figure 6.8 Transition from state S1 to S2 (i.e., the trajectory of state S4); The unit of Y-axes is degree. X-axes represent the scaling time: Time = 0 (0%) denotes reference data at state S1, and Time = 1 (100%) denotes the reference data at state S2

6.4.3   Validation of Lower Level Controller

This simulation research specifically addresses the system performances by assisting single-joint (the right knee joint) movements, as a preamble for the validation of the proposed multiple DOFs control scheme of the ERRobot exoskeleton. Figure 6.9 shows a diagram of a hypothetical 1-DOF exoskeleton designed to assist the motion of the knee joint. The L_r^{RS} is the length between the point of effort application and the rotational axis; M_m^{RKnee} and M_p^{RKnee} represent the muscle active torque and joint passive torque. It is a neuromuscular model and the involved motion-related muscles also show in figure. The selected 4 muscles concern the knee joint extensors (namely, the Rectus Femoris (RF) and Vastus Lateralis (VL)) and flexors (namely, Gastrocnemius Medial (GSM), the Biceps Femoris Long head (BFL)). The insertion points of the considered muscles are taken from the anatomical data model. The muscle lengths as well as the moment arms of each muscle actuating the knee joint are computed separately.

The wearer is in a sitting position with the shank freely swinging. The human lower extremity has been modelled as a rigid body swinging between $-0°$ and $-120°$. To simplify the model, we don't take into account the ground contact. According to Equation (5.38), the generalized force $\boldsymbol{Q}_N$ acting on the DOF is the sum of generalized forces of joint internal moments $\boldsymbol{Q}_{N1}$ and the robot generated torques $\boldsymbol{Q}_{N2}$. As indicated above, the model is actuated by the torques of the ERRobot $\boldsymbol{M}_r^{RS}$, the wearer's muscle torques $\boldsymbol{M}_m^{RKnee}$, joint passive torque $\boldsymbol{M}_p^{RKnee}$. They are
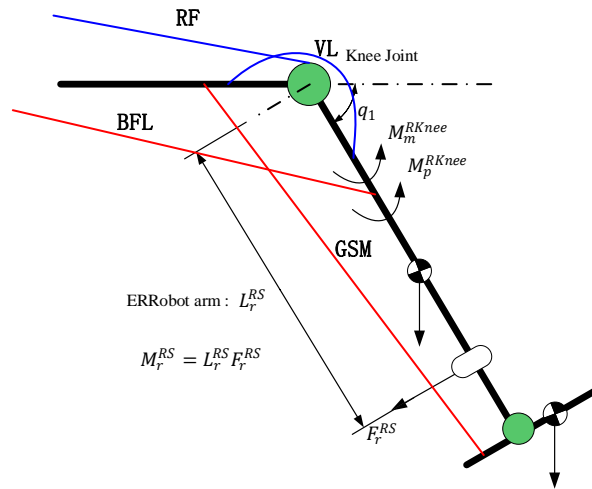
Figure 6.9 1-DOF knee joint movement rehabilitation diagram

illustrated in Figure 6.9. In the figure, F_r^{RS} is the composition of forces acting on the shank by ERRobot, which is illustrated in Figure 5.7. In addition, the moment M_r^{RS} is just the robot generated torques Q_{N2} applied on DOF q_1. Referring to the control block diagrams shown in Figure 6.2, and Figure 6.3 , we designed a control framework used in the simulation.
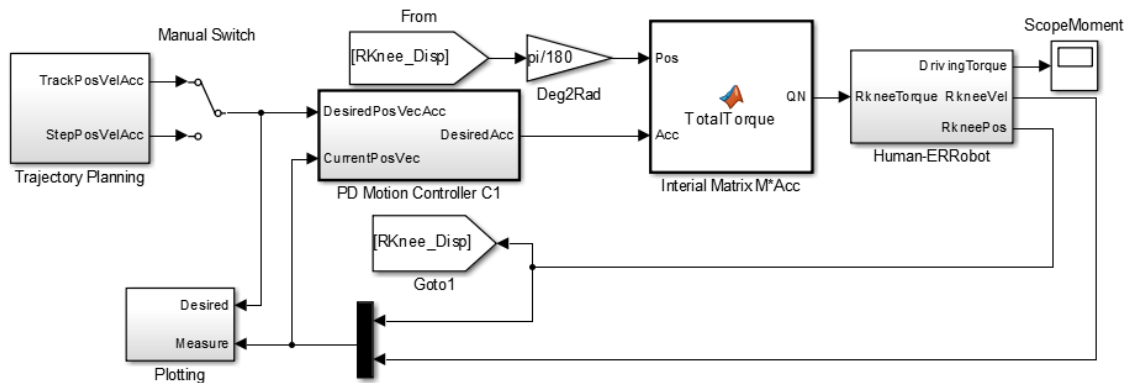

Figure 6.10 A knee joint control lower frame

Based on the Equation (6.15)(6.16) , we preliminarily designed a controller for the single joint control neglecting some nonlinearities items. The top layer of MATLAB/Simulink model is shown in Figure 6.10, and some simulation screenshots is shown in Figure 6.11 and Figure 6.12, which provide a straightforward way of proving that the model works properly.

Two trajectory tracking experiments are implemented in the section to validate the validness of the developed controller. Input signal of the first simulation experiment is several connective step values, that is, firstly steps to -20° from 0°, then steps to -60° , further steps to -32°, and finally steps to -70°,which is the blue line shown in Figure 6.11. The output result is also shown in the same figure. It is obvious that the output angle (orange dotted line) follows the input quite fast,
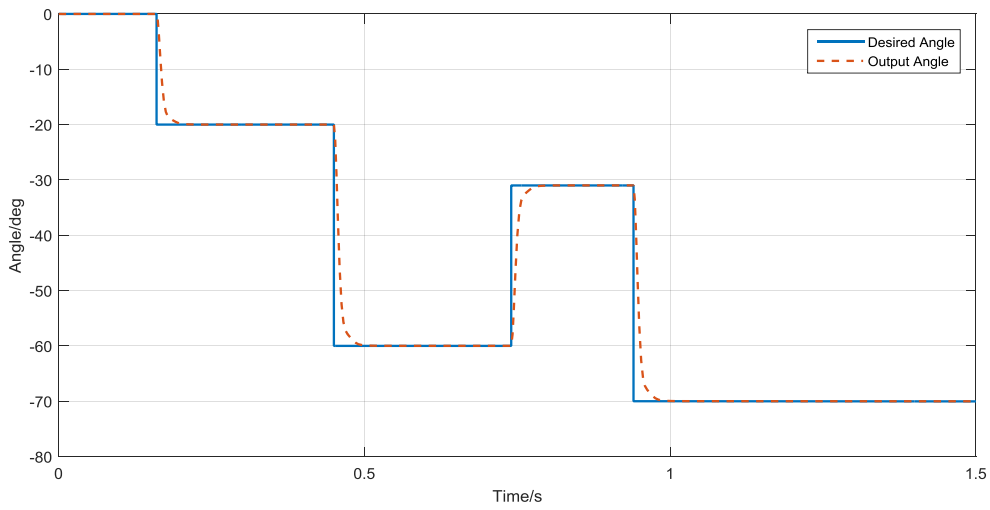
and no insignificant overshoot exists.

Figure 6.11 Trajectory tracking: Input signal is several connective step values

The other simulation experiment perform angle and velocity trajectory tracking at the same time. The input knee joint angle signal is obtained by smoothing processing for discrete joint angle sequence that is a part of gait walking by using Equation (6.28), and use the same equation to solve its differential signal.

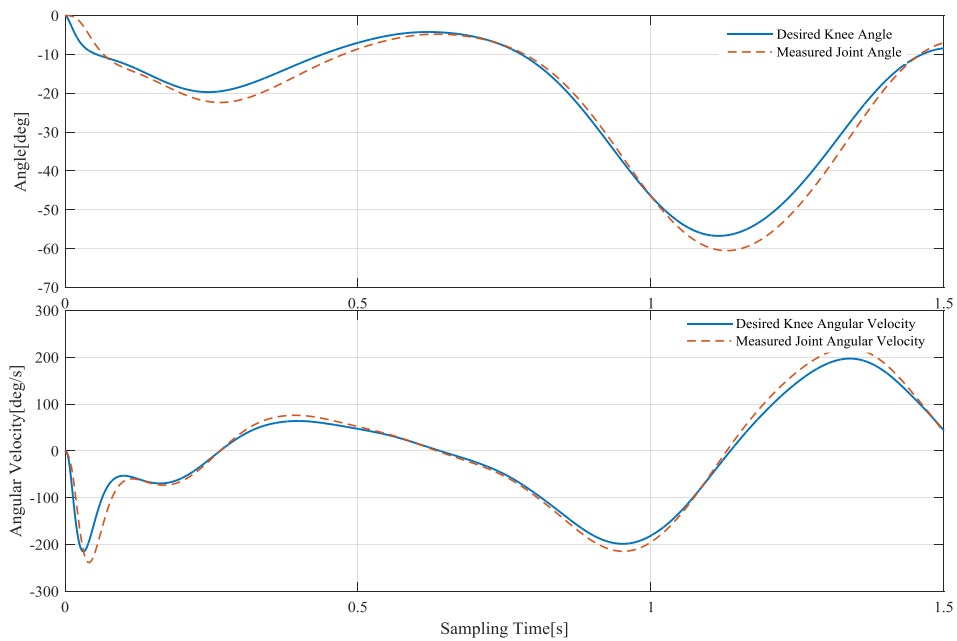$$G(s) = \frac{1}{\gamma^2 s^2 + 2\gamma s + 1} \tag{6.28}$$

Figure 6.12 Trajectory tracking: Input signal is a connective trajectory

6.5 Conclusion

In order to make the control system automatically making decision and completing the therapy process, a novel quantitative assessment method is proposed, in which two new concepts, "Health Belt" and HAI, are introduced in the work. In the meantime, both the performances of joint movement (macro level) and muscle forces (micro level) are combined in a same rating system. The proposed assessment method was included in the module of rehabilitation assessment, and on the basis of it, we also developed a module of rehabilitation program planning. Both of them are helpful in monitoring the progress of the patient and therapist's decision-making.

As a core part of the work, the chapter presents a control architecture for the exoskeleton robot (ERRobot), which is based on NMS model. The architecture incorporates FSM method and makes ERRobot offer assistance to patients during rehabilitation by guiding motions on correct training, rehabilitation trajectories, or give force support to be able to perform certain motions. Note that the presented control architecture is not specific to the given training task implementation but can be used for any other rehabilitation tasks. The proposed control system has the following several characters:

a) The control framework can provide autonomous training where patients are engaged in repeated and intense practice of goal-directed tasks leading to improvements in motor function;

b) The whole structure is a hybrid system that can process both continuous and discrete system, which is implemented by the method of FSM. In addition, the method of modularization management is adopted to be conducive to an expansion of other rehabilitation tasks;

c) The upper-level controller is a biomimetic, hierarchical control architecture which emulates the mammalian motor control architecture[158]. Combining with NMS model, the sEMG is used to estimate operators' movement intention and so on.

The sEMG is adopted as an indicator of subject's voluntary intention in the system, so it is an intuitive interactive interface between the exoskeleton and operator. Compared to the traditional control by way of external devices (a keypad or a wheel), intuitive interface can reduce operator's mental load, that is, the operator can focus on fulfilling a task with the robot rather than focus on mere control of the device. Finally, the results of numerical simulation experiments are presented to demonstrate the efficacy of the proposed control architecture.

Chapter 7. Conclusions

In this dissertation, a novel control scheme based on the human NMS model for lower-extremity rehabilitation robot (i.e., ERRobot) was developed. The work firstly proposed and built a rehabilitation robot system for lower-extremity, so as to provide a running and validation platform for proposed control framework. It was an intelligent control system framework. Or rather, it was a hybrid and hierarchical dynamic reactive control system based on FSM method. The whole system could be grouped into three levels, that is, a master upper-level control module, intermediate transformation module, and a low-level control module. As stated previously, the focus of the work was on the building of upper-level controller, although the lower control system was although developed in the work. Several main modules are included in the upper control system: (1) Module of Motion Intent Parse (MIP); (2) Rehabilitation Assessment Module (RAM) and Rehabilitation Program Planning (RPP); (3) Motion Planning Module (MPM). In order to model, simulate, and analyze biomechanics related system, a simulation system was developed in Matlab software environment.

7.1 Contributions

Comparing with those works such as literatures [1, 46, 84] that are closely related to this study, three main contributions are contained in this thesis, that is,

(1) The main contribution of this work is that an intelligent sHRI for lower-extremity human-robot rehabilitation system is proposed based on NMS model. It can provide autonomous training where patients are engaged in repeated and intense practice of goal-directed tasks leading to improvements in motor function, and it can be easily applied in a variety of potential applications of many other exoskeleton systems through slight modifications. The proposed control system has the following specialties: 1) the sHRI is a biomimetic, hierarchical control architecture, which integrates the human NMS model into the system. In the system, the muscular activation is obtained by sEMG signals' processing, so some necessary process methods are developed. For example, the work proposes a new and real-time sEMG signal segmentation method named after "P&WND". The use of sEMG can make the developed system produce a smoother and more natural movement pattern; 2) It is a hybrid system that can process both continuous and discrete system; 3) It unites two entities (the human and the exoskeleton robot) into one integrated Human-Robot system under the control of the human, which will benefit from the advantages offered by each subsystem (the ability to plan a complicated task from human, and providing requiring forces or moments to perform tasks from robot). In addition, to use the interface, it is not necessary for the operator

to learn a certain language or gestures, because the system is completely intuitive. To our knowledge, there is no this type of sHRI applied in a lower extremity exoskeleton control;

(2) The paper proposes a quantitative evaluation method for lower-extremity motion function, which is realized by introducing two new concepts, that is "Health Belt" and "Health Evaluation Index (HAI)" in the work. In the meantime, both the performances of joint movement (macro level) and muscle forces (micro level) are combined in a same rating system. The proposed method is included in the module of rehabilitation assessment, and on the basis of it, we also develop a module of rehabilitation program planning. Both of them are helpful in monitoring the progress of the patient and therapist's decision-making.

(3) During the development process of the interface, this thesis built a biomechanical model of the human body, and a human biomechanical model related simulation environment is simultaneously constructed in Matlab/Simulink. The developed simulation system is easily used to execute simulation studies and develops some control algorithms. The simulation environment can make up some defections comparing with some currently available simulation platform such as OpenSim. For example, the developed simulation system is easily to complete the development of rapid control prototype by combining with many popular controller.

7.2 Future Work and Prospect

An important direction for future development involves testing the usability of the proposed control frame with lower limb patients. In the future, we'll apply the interface in prototype machine so as to validate and make it better meet the real system requirements. In addition, considering a wider range of motion study of human body, the developed biomechanical model is incomplete. Another important objective for future work would be to extend our modeling framework to encompass more human musculoskeletal system, and integrate fully compatible.

Considering the demand for control system of rehabilitation system is higher and higher, the proposed approach in the work will be an important study direction in the field. Furthermore, the study methods, developed models, proposed methods etc. involved in the work will be useful for other researchers in the field or not.

Appendix

A: MTC-Related Muscles and Parameters

The section lists the symbols' nomination and descriptions used in formulas building in Chapter 4, and some default values for some parameters are also presented in these tables which are used in MTC-related simulations and model construction.

Table 0.1 MTC-specific parameters

Symbol	Parameter Description
	Independent of time parameters
F_0^M	Optimal muscle force(maximum isometric force).Maximum active muscle force generated by muscle fiber
L_0^M	Optimal muscle length corresponding to the length at which the muscle can produce the greatest amount of active force.
L_s^T	Tendon slack length, corresponding to the zero-force length of the tendon
V_{max}^M	Maximum muscle fiber contraction velocity, which can be approximated $6L_o^M \sim 16L_o^M$ per second [159]. It is defined as: $V_{max}^M = L_o^M/\tau_{max}$.This time quantity τ_{max} is known to vary for fast and slow muscles. Def: $\tau_{max} = 0.1s$ [76, 121].
α_0, α_{max}	α_0 is the optimal pennation angle at optimal fiber length L_o^M $\alpha_0 \in \left[0, \frac{\pi}{2}\right)$ rad. Def: $\alpha_0 = 0$; α_{max} is the maximum pennation angle, $\alpha_{max} \in \left[0, \frac{\pi}{2}\right)$ rad.Def: $\alpha_{max} =$acos(0.1) Note: the two quantities are not used as scaling parameters in my model building.
	Time-dependent Variables
α	Pennation angle between the line of action of a muscle fiber and the line of action of its tendon.
L^{MT}	Total muscle-tendon complex length
V^{MT}	Total muscle-tendon complex velocity
L^T, l_{norm}^t	Tendon length and its normalization. Their relationship: $l_{norm}^t = L^T/L_o^M$
V^T	Tendon velocity
L^M, l_{norm}^m	Muscle fiber length and its normalization. Their relationship: $l_{norm}^m = L^M/L_o^M$
V^M, v_{norm}^m	Muscle fiber velocity and its normalization. Their relationship: $v_{norm}^m = V^M/V_{max}^M$ Sometimes, V^M is written as $\dot{L}^M$ during modeling.
f_{norm}^{ce}	Active element CE normalization force of muscle fiber
a	The activation level of the muscle
f_l^{ce}	Non-linear force-length relationship $f_l^{ce}(l_{norm}^m)$
f_v^{ce}	Non-linear force-velocity relationship $f_v^{ce}(v_{norm}^m)$

Table 0.2 Parameters defining the selected MTC actuators in the work

	F_0^M/N	L_0^M/m	L_s^t/m	α_0/deg	$V_{max}^M/m/s$
Gluteus Maximus (GMAX)	1852.6	0.157	0.0643	21.93	$10L_0^M$
Tensor Fasciae Latae (TFL)	155.0[*2]	0.095[*4]	0.425[*3]	3[*4]	$10L_0^M$
Rectus Femoris(RF)	779[*6]	0.084[*6]	0.346[*3] [*5]	5[*6]	$10L_0^M$
Biceps Femoris Long head (BFL)	717.0[*2]	0.1090[*2] [*3]	0.341[*2] [*3]	0[*1]	$10L_0^M$
Vastus Lateralis(VL)	1871[*6]	0.084	0.157[*3]	5[*6]	$10L_0^M$
Tibialis Anterior(TA)	673.7[*5]	0.068[*5]	0.241[*5]	9.6[*5]	$10L_0^M$
Soleus(SOL)	3585.9[*5]	0.044[*5]	0.282[*5]	28.3[*5]	$10L_0^M$
Gastrocnemius Medial(GSM)	1308[*5]	0.051[*5]	0.401[*5]	9.9[*5]	$10L_0^M$

Note that: [*1] items are from [6];[*2] items are from [160];[*3] items are from [129];[*4] items are from [161];
[*5]-[162];[*6]-[163]

Table 0.3 Shared parameters in four general curve definition

Symbol	Parameter description
(1) CE force-length relationship	
w	Width of the normalized bell curve. In the work, we define the width is 56% of L_0^M,namely, Def: w = 0.56;
c	Remaining force at +/- width, Def:$c = Ln(0.05)$;
(2) CE force-velocity relationship	
A_f	Force-velocity shape factor, Def: $A_f = 0.25$;
F_{len}	Maximum normalized lengthening force ,Def:$F_{len}=1.8$;
(3) PE force-length relationship	
k^{PE}	An exponential shape factor. Def: $k^{PE} = 5$
s_0^m	The passive muscle strain due to maximum isometric force, , which is set differently for young and old adults as described in[113].For young, Def: $s_0^m = 0.6$.
(4) SE force-length relationship(Tendon force)	
ε^0	Def: ε^0=0.033
ε_{toe}^T	Tendon strain at maximum isometric muscle force,Def: ε_0^T =0.609ε^0
K_{lin}	A linear scale factor, Def: $K_{lin} = 37.5$;
$\bar{F}_{toe}^T$	The transition from nonlinear to linear behavior is prescribed to occur for normalized tendon forces greater than ,Def: $\bar{F}_{toe}^T = 0.5$;
K_{toe}	An exponential shape factor ,Def: $K_{toe} = 1.5$;

B: The Details of Control System Design

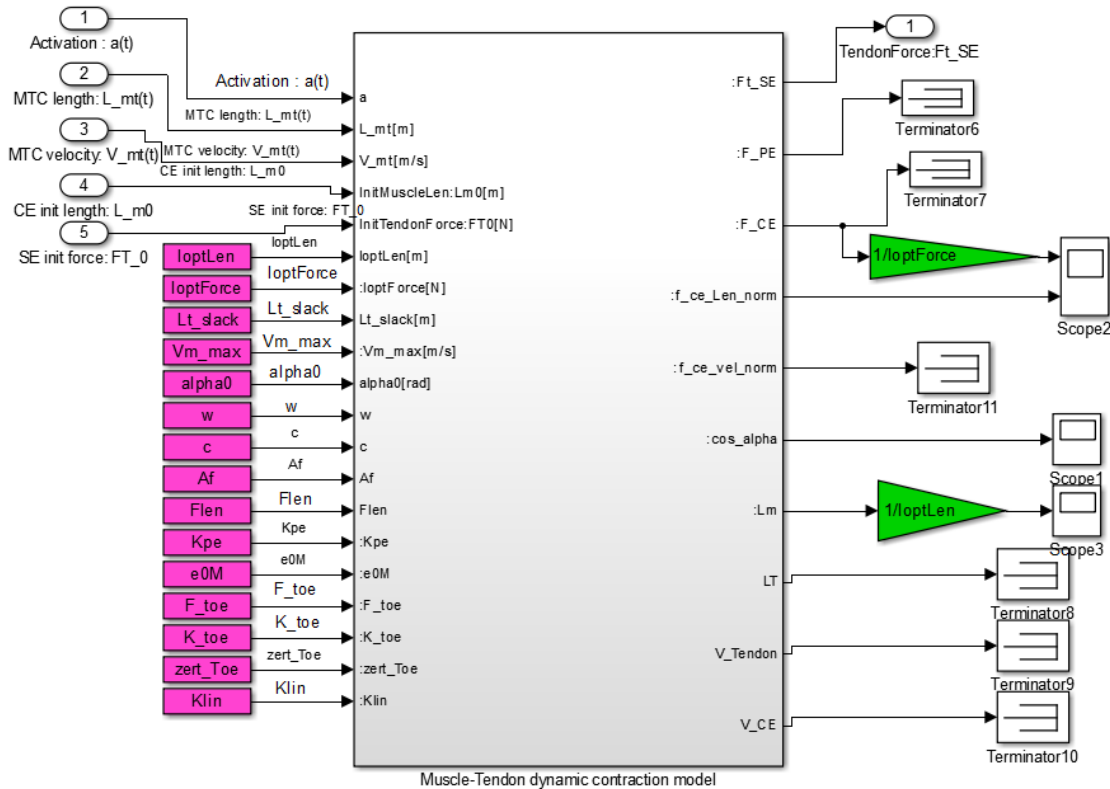## B.1 Simulink Block Set Used in the Work (part)

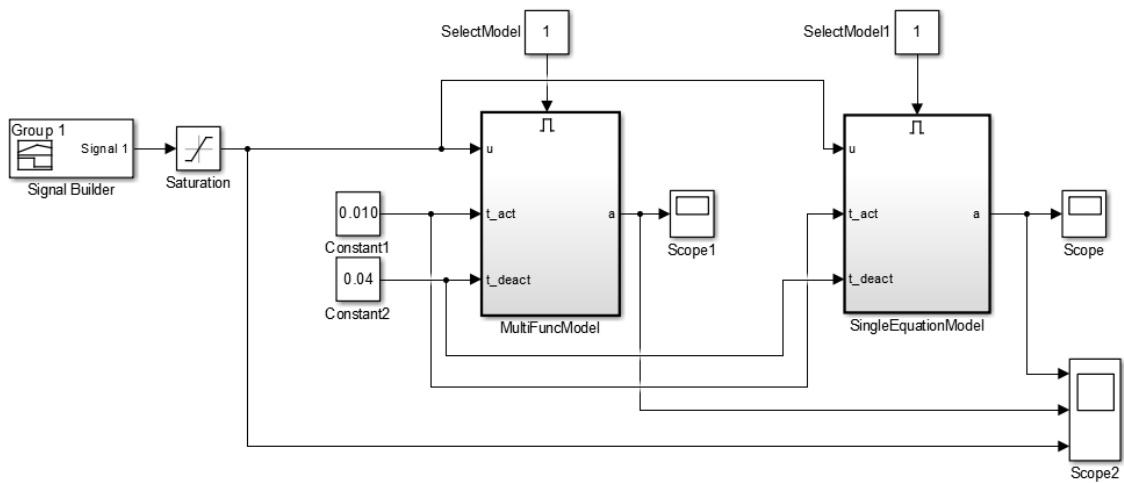Figure 0.1 Simulink simulation block of muscle-tendon dynamic model

Figure 0.2 Simulink block of Activation dynamic simulation

B.2 Part of The designed GUI

Following are some screenshots of GUIs that allow a user to adjust the control system's parameters and view results in real-time. The Simulink models used for the GUIs are almost identical to their fixed-value counterparts.
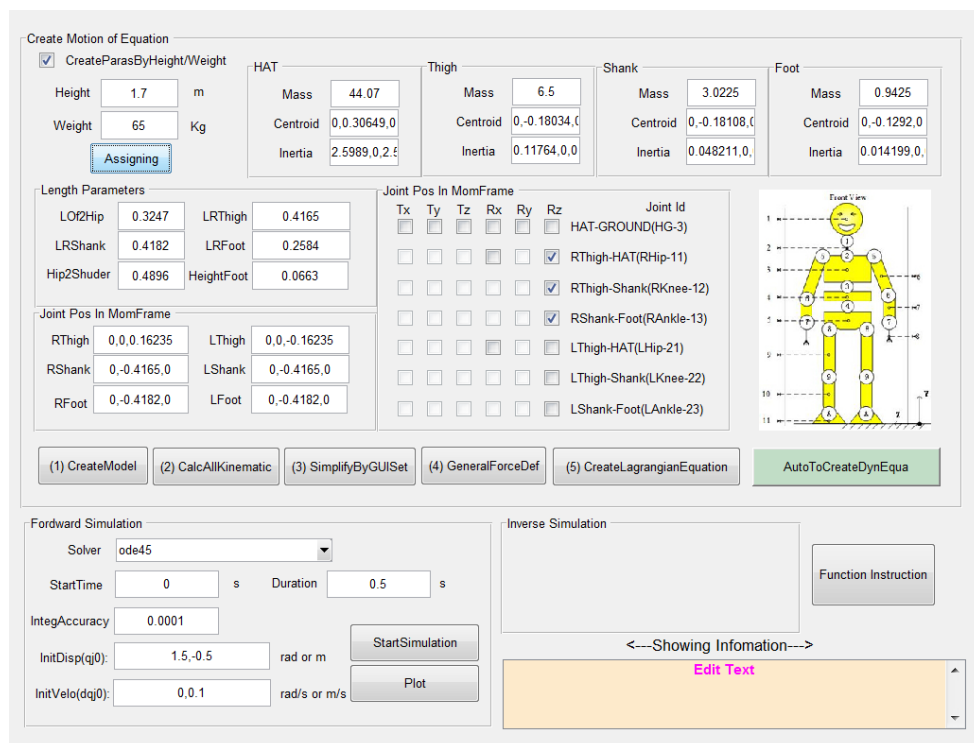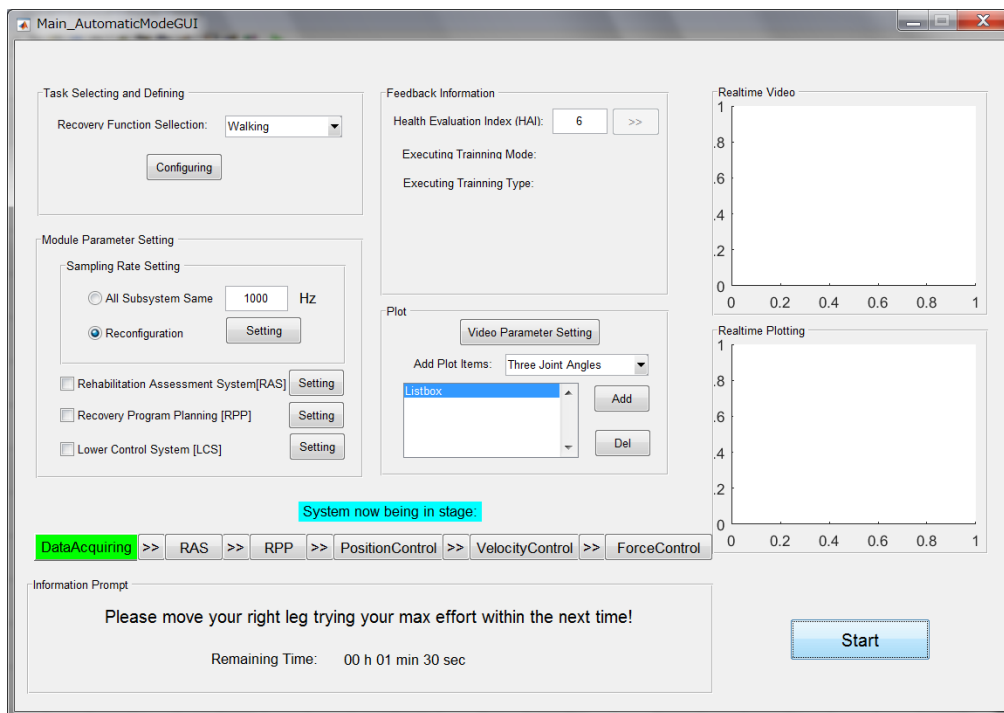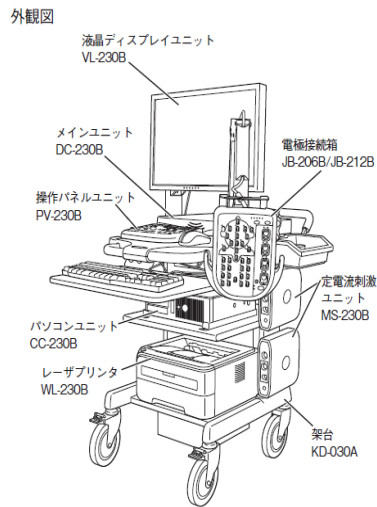

Figure 0.3 Simulation Configuring Interface

Figure 0.4 A control GUI window being in automatic working mode

C: sEMG Acquiring

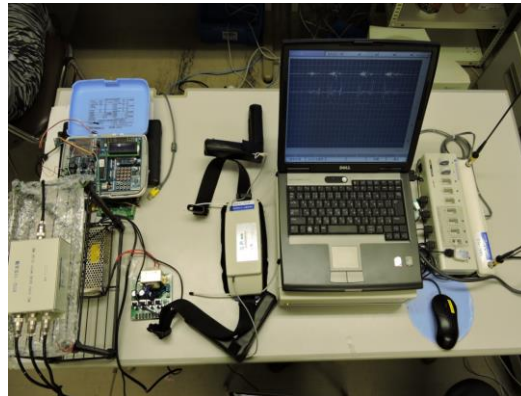### C.1 Devices

(a) Experimental scene

(b) The appearance of sEMG signal acquisition device (MEB-2312 with 12 channels Neuropack X1) that is made by NIHON KOHDEN Inc., Tokyo, Japan

(c) Exporting data by TXT file that saved some setting parameters positioning before digital sEMG data, and please note that the figure shows a part of the txt file

Figure 0.5 The sEMG signal acquiring by using wire device in Nagasaki University Hospital

(a) sEMG Collecting   (b) Experiment Devices (sEMG, force, and angle sampling)
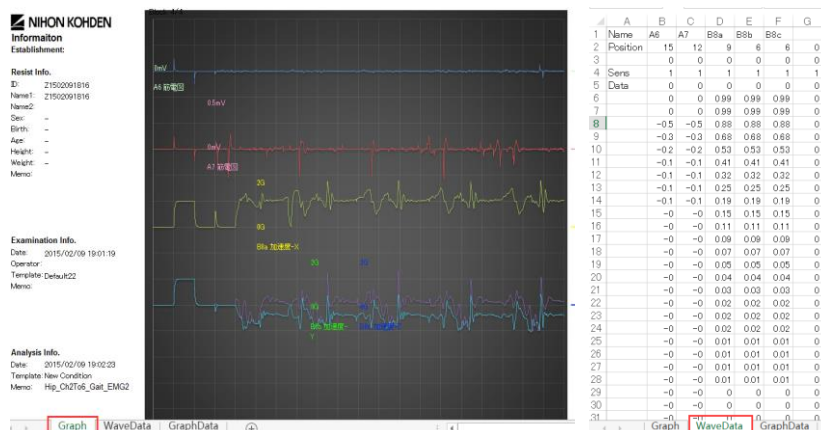
(c) Exporting data format (*.xlsx file)

Figure 0.6 The sEMG signal sampling by using wireless signal acquiring system

C.1 Experimental Result

Table 0.4 Relative muscle activity intensity in Hip-FE exercise

		FromStart2Max	FromMax2Mid	FromMid2Min	FromMin2End
Tensor Fasciae Latae	1/3 before	31.7	17.0	17.6	18.4
	1/3 after	18.8	18.4	19.0	19.3
	All data	27.7	18.1	18.6	19.3
Gluteus Maximus	1/3 before	29.4	17.3	17.7	18.5
	1/3 after	22.8	18.8	19.1	19.4
	All data	27.6	18.5	18.8	19.4
Iliopsoas	1/3 before	20.0	17.2	17.5	18.4
	1/3 after	19.3	18.5	19.0	19.2
	All data	21.1	18.2	18.6	19.3
Gluteus Medius	1/3 before	15.7	16.8	17.4	18.2
	1/3 after	15.0	18.0	18.7	19.0
	All data	16.5	17.8	18.4	19.1
Adductor Brevis	1/3 before	14.3	16.5	17.2	18.1
	1/3 after	13.6	17.7	18.6	18.9
	All data	14.8	17.5	18.3	19.0
Sartorius	1/3 before	13.4	16.3	17.2	18.1
	1/3 after	12.8	17.5	18.6	18.7
	All data	13.8	17.3	18.4	18.9
Adductor Magnus	1/3 before	14.6	16.1	17.2	18.1
	1/3 after	14.0	17.6	18.6	18.9
	All data	15.0	17.3	18.3	19.0
Adductor Longus	1/3 before	16.5	16.6	17.6	18.3
	1/3 after	16.0	18.3	19.0	19.0
	All data	17.0	17.8	18.7	19.4
Biceps Femoris Longhead	1/3 before	17.2	16.7	17.6	18.4
	1/3 after	17.5	18.4	18.9	19.2
	All data	17.6	18.0	18.7	19.2
Rectus Femoris	1/3 before	17.9	17.0	17.7	18.5
	1/3 after	18.4	19.0	19.1	19.3
	All data	18.4	18.6	18.9	19.4
Semitendinosus	1/3 before	17.0	17.4	17.6	18.4
	1/3 after	17.8	18.7	19.0	19.2
	All data	17.6	18.6	18.8	19.6
Semimembranosus	1/3 before	17.4	17.8	17.6	18.4
	1/3 after	18.0	18.7	19.0	19.2
	All data	17.8	18.7	18.8	19.3

D: Publication List

1. Lei Shi, Zhen Liu, The Three-dimensional Biomechanical Model Building of Human Lower Extremity Musculo-Skeletal System, International Journal of Computer Systems Science & Engineering(accepted),2015.
2. Lei Shi, Zhen Liu, Qiang Wang, Yuanning Liu, A New Method of sEMG Segmentation Based on Peak Analysis and Non-equidistant Window, Journal of Multi-Valued Logic and Soft Computing (submitted).
3. Lei Shi, Zhen Liu, Huaibin Wang, Design of a Fuzzy Controller of a Robot-Assisted Recovery System Based on Genetic Algorithms (Submitted).
4. Lei Shi, Zhen Liu, Chao Zhang, A sEMG-driven Musculoskeletal Model to Control Exoskeleton Robot Used in Lower Extremity Rehabilitation. Journal of Information Sciences and Computing Technologies 4(1): pp.265-273, 2015.
5. Shi Lei, Zhen Liu, Design of Soft Human-Robot Interface Based on Neuro-Muscular-Skeletal Model, Proceedings of 12th IEEE International Conference on Dependable, Autonomic and Secure Computing (DASC2014), pp. 536-541, 2014.
6. Shi Lei, Zhen Liu, Qiang Wang, A Novel Method of sEMG Signal Segmentation, Proceedings of 9th IEEE International Conference on Mobile Ad-hoc and Sensor Networks (MSN2013), pp. 515-520, 2013.
7. Shi Lei, Kun Jiang, Qiang Wang, Zhen Liu, A New Method of Quantitative Rehabilitation Rating in Robot-Assisted Recovery System, Proceedings of 5th IEEE International Conference on Intelligent Networks and Intelligent Systems (ICINIS2012), pp. 278-281, 2012.
8. Shi Lei, Qiang Wang, Zhi Jun Sun, Ru Long Xu, Lei Chen, Design of Autonomous Robot Soccer System Based on Sugeno Fuzzy, Applied Mechanics and Materials 195 pp: 1017-1022,2012.
9. Cui Guoguang, Lei Shi, Qiang Wang, Lijun Wu, Haiou Wen, The application research of a fuzzy controller on robot, Proceedings of IEEE International Conference on Computer Science and Service System (CSSS2011), pp. 3658-3661, 2011.
10. Wang Qiang, Yanjun Yang, Lei Shi, A robot path planning system based on a hierarchical fuzzy control, Proceedings of IEEE International Conference on Computer Science and Service System (CSSS2011), pp. 2912-2915, 2011.
11. Lei SHI, and WANG Qiang, Research on Humanoid Robot Soccer System Based on Fuzzy Logic, International Journal of Computer Network and Information Security (IJCNIS) 2(1),38, pp.38-44,2010.

References

[1]. Akdogan, E., M.A. Adli, and M.N. Bennett. A human-machine interface design for direct rehabilitation using a rehabilitation robot. in Proceedings of the 5th international conference on Soft computing as transdisciplinary science and technology. 2008. ACM.

[2]. 平成 26 年度高齢化の状況及び高齢社会対策の実施状況, in 平成 27 年版高齢社会白書（全体版）. 2015, 内閣府: 平成 27 年版高齢社会白書.

[3]. Richardson, R., et al., Impedance control for a pneumatic robot-based around pole-placement, joint space controllers. Control Engineering Practice, 2005. **13**(3): p. 291-303.

[4]. Ballantyne, B.T., et al., Electromyographic activity of selected shoulder muscles in commonly used therapeutic exercises. Physical Therapy, 1993. **73**(10): p. 668-677.

[5]. Durfee, W. and P. Iaizzo, Rehabilitation and muscle testing. Encyclopedia of Medical Devices and Instrumentation.

[6]. Winter, D.A., Biomechanics and motor control of human movement. Fourth ed. 2009, Canada: John Wiley & Sons, Inc, Hoboken, New Jersey.

[7]. Wernig, A., et al., Laufband therapy based on 'rules of spinal locomotion' is effective in spinal cord injured persons. European Journal of Neuroscience, 1995. **7**(4): p. 823-829.

[8]. Jezernik, S., et al., Robotic orthosis lokomat: A rehabilitation and research tool. Neuromodulation: Technology at the neural interface, 2003. **6**(2): p. 108-115.

[9]. Riener, R., Technology of the Robotic Gait Orthosis Lokomat, in Neurorehabilitation Technology. 2012, Springer. p. 221-232.

[10]. Husemann, B., et al., Effects of locomotion training with assistance of a robot-driven gait orthosis in hemiparetic patients after stroke a randomized controlled pilot study. Stroke, 2007. **38**(2): p. 349-354.

[11]. Westlake, K.P. and C. Patten, Pilot study of Lokomat versus manual-assisted treadmill training for locomotor recovery post-stroke. Journal of neuroengineering and rehabilitation, 2009. **6**: p. 18.

[12]. Rosen, J., et al., A myosignal-based powered exoskeleton system. Systems, Man and Cybernetics, Part A: Systems and Humans, IEEE Transactions on, 2001. **31**(3): p. 210-222.

[13]. Sylos-Labini, F., et al., EMG patterns during assisted walking in the exoskeleton. Frontiers in human neuroscience, 2014. **8**.

[14]. Pons, J.L., Wearable robots: biomechatronic exoskeletons. Vol. 70. 2008, Wiley Online Library.

[15]. Dick, G.J. and E.A. Edwards, Human bipedal locomotion device. 1991, Google Patents.

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

[17]. Makinson, J.B., D.P. Bodine, and B.R. Fick, Machine Augmentation of Human Strength and Endurance Hardiman i Prototype Project. 1969, DTIC Document.

[18]. NR, W.U.N., RESEARCH AND DEVELOPMENT PROTOTYPE FOR MACH INE AUGMENTATION OF HUMAN STRENGrH AND ENDURANCE. Contract, 1971. **14**: p. 66-C0051.

[19]. Fleischer, C. and G. Hommel, A Human--Exoskeleton Interface Utilizing Electromyography. Robotics, IEEE Transactions on, 2008. **24**(4): p. 872-882.

[20]. Ferris, D.P., et al., An ankle-foot orthosis powered by artificial pneumatic muscles. Journal of Applied

Biomechanics, 2005. **21**(2): p. 189.

[21]. Misuraca, J.J. and C. Mavroidis. Lower limb human muscle enhancer. in Proc. 2001 ASME Int. Mech. Eng. Conf. Expo.(IMECE). 2001.

[22]. Bradley, D., et al., NeXOS–The design, development and evaluation of a rehabilitation system for the lower limbs. Mechatronics, 2009. **19**(2): p. 247-257.

[23]. Moughamir, S., et al., Control law implementation for Multi-Iso: a training machine for lower limbs. 2001, DTIC Document.

[24]. Pratt, J.E., et al. The RoboKnee: an exoskeleton for enhancing strength and endurance during walking. in Robotics and Automation, 2004. Proceedings. ICRA'04. 2004 IEEE International Conference on. 2004. IEEE.

[25]. Ferris, D.P., G.S. Sawicki, and M.A. Daley, A physiologist's perspective on robotic exoskeletons for human locomotion. International Journal of Humanoid Robotics, 2007. **4**(03): p. 507-528.

[26]. Sawicki, G.S., K.E. Gordon, and D.P. Ferris. Powered lower limb orthoses: applications in motor adaptation and rehabilitation. in Rehabilitation Robotics, 2005. ICORR 2005. 9th International Conference on. 2005. IEEE.

[27]. Akdoğan, E. and M.A. Adli, The design and control of a therapeutic exercise robot for lower limb rehabilitation: Physiotherabot. Mechatronics, 2011. **21**(3): p. 509-522.

[28]. Monaco, V., et al., Design and evaluation of neurobike: a neurorehabilitative platform for bedridden post-stroke patients. Neural Systems and Rehabilitation Engineering, IEEE Transactions on, 2012. **20**(6): p. 845-852.

[29]. LokoHelp. [cited 2015 12.4]; Available from: http://www.woodway.de/medicaltreadmills/lokohelp.html.

[30]. Dzahir, M.A.M. and S.-i. Yamamoto, Recent trends in lower-limb robotic rehabilitation orthosis: Control scheme and strategy for pneumatic muscle actuated gait trainers. Robotics, 2014. **3**(2): p. 120-148.

[31]. Schmidt, H., et al., Gait rehabilitation machines based on programmable footplates. Journal of neuroengineering and rehabilitation, 2007. **4**(1): p. 2.

[32]. Bernhardt, M., et al. Hybrid force-position control yields cooperative behaviour of the rehabilitation robot LOKOMAT. in Rehabilitation Robotics, 2005. ICORR 2005. 9th International Conference on. 2005. IEEE.

[33]. Banala, S.K., et al., Robot assisted gait training with active leg exoskeleton (ALEX). Neural Systems and Rehabilitation Engineering, IEEE Transactions on, 2009. **17**(1): p. 2-8.

[34]. Veneman, J.F., et al., Design and evaluation of the LOPES exoskeleton robot for interactive gait rehabilitation. Neural Systems and Rehabilitation Engineering, IEEE Transactions on, 2007. **15**(3): p. 379-386.

[35]. van Asseldonk, E.H. and H. van der Kooij, Robot-Aided Gait Training with LOPES, in Neurorehabilitation Technology. 2012, Springer. p. 379-396.

[36]. Dzahir, M. and S.-i. Yamamoto, Recent Trends in Lower-Limb Robotic Rehabilitation Orthosis: Control Scheme and Strategy for Pneumatic Muscle Actuated Gait Trainers. Robotics, 2014. **3**(2): p. 120-148.

[37]. Sankai, Y., HAL: Hybrid assistive limb based on cybernics, in Robotics Research. 2011, Springer. p. 25-34.

[38]. Chu, A., H. Kazerooni, and A. Zoss. On the biomimetic design of the berkeley lower extremity exoskeleton (BLEEX). in Robotics and Automation, 2005. ICRA 2005. Proceedings of the 2005 IEEE International Conference on. 2005. IEEE.

[39]. Kong, K. and D. Jeon, Design and control of an exoskeleton for the elderly and patients. Mechatronics, IEEE/ASME Transactions on, 2006. **11**(4): p. 428-432.

[40]. Suzuki, K., et al., Intention-Based Walking Support for Paraplegia Patients with Robot Suit HAL. 2010: INTECH Open Access Publisher.

[41]. Farris, R.J., H.A. Quintero, and M. Goldfarb, Preliminary evaluation of a powered lower limb orthosis to aid walking in paraplegic individuals. Neural Systems and Rehabilitation Engineering, IEEE Transactions on, 2011. **19**(6): p. 652-659.

[42]. REX Bionics. 2013   [cited 2015 11.25]; Available from: http://www.rexbionics.com/.

[43]. Wang, L., et al. Actively controlled lateral gait assistance in a lower limb exoskeleton. in Intelligent Robots and Systems (IROS), 2013 IEEE/RSJ International Conference on. 2013. IEEE.

[44]. Ekso™. 2014; Available from: http://intl.eksobionics.com/.

[45]. Métrailler, P., et al. Improvement of rehabilitation possibilities with the MotionMaker TM. in Biomedical Robotics and Biomechatronics, 2006. BioRob 2006. The First IEEE/RAS-EMBS International Conference on. 2006. IEEE.

[46]. Fleischer, C., Controlling exoskeletons with EMG signals and a biomechanical body model,. 2007, Scuola Superiore Sant'Anna.

[47]. Kawamoto, H., S. Kanbe, and Y. Sankai. Power assist method for HAL-3 estimating operator's intention based on motion information. in Robot and human interactive communication, 2003. proceedings. ROMAN 2003. The 12th IEEE international workshop on. 2003. IEEE.

[48]. Kawamoto, H. and Y. Sankai, Power assist system HAL-3 for gait disorder person. Computers helping people with special needs, 2002: p. 19-29.

[49]. Mukherjee, G., Design and development of an assistive exoskeleton for independent sit-stand transitions among the elderly. 2014, University of Cincinnati.

[50]. Kazerooni, H. and M.-G. Her, The dynamics and control of a haptic interface device. Robotics and Automation, IEEE Transactions on, 1994. **10**(4): p. 453-464.

[51]. Kazerooni, H., The human power amplifier technology at the University of California, Berkeley. Robotics and autonomous systems, 1996. **19**(2): p. 179-187.

[52]. de Souza Ramos, J.L.A. and M.A. Meggiolaro. Use of surface electromyography to control an active upper limb exoskeleton actuated by pneumatic artificial muscles and optimized with genetic algorithms. in 22nd International Congress of Mechanical Engineering. Ribeirao Preto, Brazil. 2013.

[53]. Lee, J.-W. and G.-K. Lee, Gait angle prediction for lower limb orthotics and prostheses using an EMG signal and neural networks. International Journal of Control, Automation, and Systems, 2005. **3**(2): p. 152-158.

[54]. Cavanagh, P. and P. Komi, Electromechanical delay in human skeletal muscle under concentric and eccentric contractions. European journal of applied physiology and occupational physiology, 1979. **42**(3): p. 159-163.

[55]. Zhou, S., et al., Electromechanical delay in isometric muscle contractions evoked by voluntary, reflex and electrical stimulation. European journal of applied physiology and occupational physiology,

1995. **70**(2): p. 138-145.

[56]. He, H. and K. Kiguchi. A study on emg-based control of exoskeleton robots for human lower-limb motion assist. in Information Technology Applications in Biomedicine, 2007. ITAB 2007. 6th International Special Topic Conference on. 2007. IEEE.

[57]. Akdoğan, E. and Z. Şişman, A muscular activation controlled rehabilitation robot system, in Knowledge-Based and Intelligent Information and Engineering Systems. 2011, Springer. p. 271-279.

[58]. Schultz, A.E. and T.A. Kuiken, Neural interfaces for control of upper limb prostheses: the state of the art and future possibilities. PM R, 2011. **3**(1): p. 55-67.

[59]. Park, J. and O. Khatib, A haptic teleoperation approach based on contact force control. The International Journal of Robotics Research, 2006. **25**(5-6): p. 575-591.

[60]. Ferris, D.P. and C.L. Lewis, Robotic lower limb exoskeletons using proportional myoelectric control. Conf Proc IEEE Eng Med Biol Soc, 2009. **2009**: p. 2119-24.

[61]. Lawrence, J.H. and C. De Luca, Myoelectric signal versus force relationship in different human muscles. Journal of Applied Physiology, 1983. **54**(6): p. 1653-1659.

[62]. Dietz, V., Spinal cord pattern generators for locomotion. Clinical Neurophysiology, 2003. **114**(8): p. 1379-1389.

[63]. Dietz, V., R. Müller, and G. Colombo, Locomotor activity in spinal man: significance of afferent input from joint and load receptors. Brain, 2002. **125**(12): p. 2626-2634.

[64]. Wolpaw, J.R., D.J. Braitman, and R.F. Seegal, Adaptive plasticity in primate spinal stretch reflex: initial development. Journal of neurophysiology, 1983. **50**(6): p. 1296-1311.

[65]. Sterr A, F.S., Motor-improvement following intensive training in low-functioning chronic hemiparesis. Neurology,2003,6: p. 842–844.

[66]. 陈景藻主编，康复医学[M]. 2001.7: 北京：高等教育出版社.

[67]. EDGERTON, V., et al., Use-dependent plasticity in spinal stepping and standing. Advances in neurology, 1997. **72**: p. 233-247.

[68]. Erol, D. and N. Sarkar, Intelligent control for robotic rehabilitation after stroke. Journal of Intelligent and Robotic Systems, 2007. **50**(4): p. 341-360.

[69]. Lotze, M., et al., Motor learning elicited by voluntary drive. Brain, 2003. **126**(4): p. 866-872.

[70]. Schouenborg, J., Learning in sensorimotor circuits. Current opinion in neurobiology, 2004. **14**(6): p. 693-697.

[71]. Kaelin-Lang, A., L. Sawaki, and L.G. Cohen, Role of voluntary drive in encoding an elementary motor memory. Journal of neurophysiology, 2004.

[72]. Bolliger, M., et al., Standardized voluntary force measurement in a lower extremity rehabilitation robot. J Neuroeng Rehabil, 2008. **5**: p. 23.

[73]. Peshkin, M., et al. KineAssist: A robotic overground gait and balance training device. in Rehabilitation Robotics, 2005. ICORR 2005. 9th International Conference on. IEEE.

[74]. Kirtley, D.C. CGA Normative Gait Database. 2005; Available from: http://www.clinicalgaitanalysis.com/data/.

[75]. Cherelle, P., et al. The amp-foot 2.0: Mimicking intact ankle behavior with a powered transtibial prosthesis. in Biomedical Robotics and Biomechatronics (BioRob), 2012 4th IEEE RAS & EMBS International Conference on. 2012. IEEE.

[76]. DeWoody, Y., The role of musculoskeletal dynamics and neuromuscular control in stress development in bone,. 1998.5, Graduate Faculty of Texas Tech University.

[77]. Delp, S.L. PARAMETERS FOR A MODEL OF THE LOWER LIMB. 2015; Available from: http://isbweb.org/data/delp/delp_jnt.

[78]. Wang, S., W. Van Dijk, and H. van der Kooij. Spring uses in exoskeleton actuation design. in Rehabilitation Robotics (ICORR), 2011 IEEE International Conference on. 2011. IEEE.

[79]. Wang, S., C. Meijneke, and H. van der Kooij. Modeling, design, and optimization of Mindwalker series elastic joint. in Rehabilitation Robotics (ICORR), 2013 IEEE International Conference on. 2013. IEEE.

[80]. Kazerooni, H., et al., Exoskeleton and method for controlling a swing leg of the exoskeleton. 2014, Google Patents.

[81]. Wikipedia. Model-based design. 2015; Available from: https://en.wikipedia.org/wiki/Model-based_design.

[82]. Prasad, G., et al. Using motor imagery based brain-computer interface for post-stroke rehabilitation. in Neural Engineering, 2009. NER'09. 4th International IEEE/EMBS Conference on. 2009. IEEE.

[83]. Knudson, D., Fundamentals of biomechanics. 2007: Springer Science & Business Media.

[84]. Wolkotte, P., et al., Modelling human locomotion. Aalborg University,Institute of Electronic Systems,Department of Health Science and Technology, 2003.

[85]. O'Brien, A.J., FLIHI: fuzzy logic implemented Hill-based muscle model. 2006: ProQuest.

[86]. Farina, D., A. Crosetti, and R. Merletti, A model for the generation of synthetic intramuscular EMG signals to test decomposition algorithms. Biomedical Engineering, IEEE Transactions on, 2001. **48**(1): p. 66-77.

[87]. Hermens, H., et al., The median frequency of the surface EMG power spectrum in relation to motor unit firing and action potential properties. Journal of Electromyography and Kinesiology, 1992. **2**(1): p. 15-25.

[88]. Duchêne, J. and J.-Y. Hogrel, A model of EMG generation. Biomedical Engineering, IEEE Transactions on, 2000. **47**(2): p. 192-201.

[89]. Hernandez Arieta, A., et al., Development of a multi-DOF electromyography prosthetic system using the adaptive joint mechanism. Applied Bionics and Biomechanics, 2006. **3**(2): p. 101-111.

[90]. Merletti, R. and P.A. Parker, Electromyography: physiology, engineering, and non-invasive applications. Vol. 11. 2004: John Wiley & Sons.

[91]. Criswell, E., Cram's introduction to surface electromyography. 2010: Jones & Bartlett Publishers.

[92]. Hof, A., et al., Speed dependence of averaged EMG profiles in walking. Gait & posture, 2002. **16**(1): p. 78-86.

[93]. Stansfield, B.W., et al., Normalized speed, not age, characterizes ground reaction force patterns in 5- to 12-year-old children walking at self-selected speeds. Journal of Pediatric Orthopaedics, 2001. **21**(3): p. 395-402.

[94]. Shi, L., Z. Liu, and Q. Wang. A Novel Method of sEMG Signal Segmentation. in Mobile Ad-hoc and Sensor Networks (MSN), 2013 IEEE Ninth International Conference on. 2013. IEEE.

[95]. Clancy, E., E.L. Morin, and R. Merletti, Sampling, noise-reduction and amplitude estimation issues in surface electromyography. Journal of Electromyography and Kinesiology, 2002. **12**(1): p. 1-16.

[96]. Reaz, M., M. Hussain, and F. Mohd-Yasin, Techniques of EMG signal analysis: detection, processing,

classification and applications. Biological procedures online, 2006. **8**(1): p. 11-35.

[97]. Armstrong, D. and T. Drew, Forelimb electromyographic responses to motor cortex stimulation during locomotion in the cat. The Journal of physiology, 1985. **367**(1): p. 327-351.

[98]. Wikipedia. Feature (pattern recognition). The Free Encyclopedia 2015; Available from: https://en.wikipedia.org/wiki/Feature_(machine_learning).

[99]. Rouffet, D.M. and C.A. Hautier, EMG normalization to study muscle activation in cycling. Journal of Electromyography and Kinesiology, 2008. **18**(5): p. 866-878.

[100]. Ha, S.-m., et al., A reliability of electromyographic normalization methods for the infraspinatus muscle in healthy subjects. Journal of human kinetics, 2013. **36**(1): p. 69-76.

[101]. Tanawongsuwan, R. and A. Bobick. Gait recognition from time-normalized joint-angle trajectories in the walking plane. in Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on. 2001. IEEE.

[102]. Kukulka, C.G. and H.P. Clamann, Comparison of the recruitment and discharge properties of motor units in human brachial biceps and adductor pollicis during isometric contractions. Brain research, 1981. **219**(1): p. 45-55.

[103]. M. Lichman, T. Theodoridis, Editor. 2013, UCI Machine Learning Repository: http://archive.ics.uci.edu/ml/citation_policy.html.

[104]. Bass, E.J., Editorial IEEE Transactions on Human–Machine Systems: Year in Review for 2013. Human-Machine Systems, IEEE Transactions on, 2014. **44**(1): p. 1-12.

[105]. Theodoridis, T., et al. Mechanical feature attributes for modeling and pattern classification of physical activities. in Information and Automation, 2009. ICIA'09. International Conference on. 2009. IEEE.

[106]. Di Fabio, R.P., Reliability of computerized surface electromyography for determining the onset of muscle activity. Physical Therapy, 1987. **67**(1): p. 43-48.

[107]. Suykens, J.A. and J. Vandewalle, Least squares support vector machine classifiers. Neural processing letters, 1999. **9**(3): p. 293-300.

[108]. Boser, B.E., I.M. Guyon, and V.N. Vapnik. A training algorithm for optimal margin classifiers. in Proceedings of the fifth annual workshop on Computational learning theory. 1992. ACM.

[109]. van der Kooij, H., B. Koopman, and F.C. van der Helm, Human motion control. Reader for Delft University course wb2407 and Twente University course, 2008. **115047**: p. 211-220.

[110]. Haeufle, D., et al., Hill-type muscle model with serial damping and eccentric force–velocity relation. Journal of biomechanics, 2014. **47**(6): p. 1531-1536.

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

[112]. Huxley, H. and J. Hanson, Changes in the cross-striations of muscle during contraction and stretch and their structural interpretation. Nature, 1954(173): p. 973-6.

[113]. Thelen, D.G., Adjustment of Muscle Mechanics Model Parameters to Simulate Dynamic Contractions in Older Adults. Journal of Biomechanical Engineering, 2003. **125**(1): p. 70.

[114]. Shi, L. and Z. Liu. Design of Soft Human-Robot Interface Based on Neuro-Muscular-Skeletal Model. in Dependable, Autonomic and Secure Computing (DASC), 2014 IEEE 12th International Conference on. 2014. IEEE.

[115]. Winters, J.M., How detailed should muscle models be to understand multi-joint movement

coordination? Human Movement Science, 1995. **14**(4): p. 401-442.

[116]. Thelen, D.G., F.C. Anderson, and S.L. Delp, Generating dynamic simulations of movement using computed muscle control. Journal of biomechanics, 2003. **36**(3): p. 321-328.

[117]. Winters, J.M., An improved muscle-reflex actuator for use in large-scale neuromusculoskeletal models. Annals of biomedical engineering, 1995. **23**(4): p. 359-374.

[118]. Lei, S., Z. Liu, and C. Zhang, A sEMG-driven Musculoskeletal Model to Control Exoskeleton Robot Used in Lower Extremity Rehabilitation. Journal of Information Sciences and Computing Technologies, 2015. **4**(1): p. 265-273.

[119]. Yamaguchi, G.T., Dynamic modeling of musculoskeletal motion: a vectorized approach for biomechanical analysis in three dimensions. 2005: Springer Science & Business Media.

[120]. Meador, C.D., The dynamics and control of a planar articulating segmental model. 2000, Texas Tech University.

[121]. Zajac, F.E., Muscle and tendon: properties, models, scaling, and application to biomechanics and motor control. Critical reviews in biomedical engineering, 1989. **17**(4): p. 359-411.

[122]. Millard, M., et al., Flexing computational muscle: modeling and simulation of musculotendon dynamics. Journal of biomechanical engineering, 2013. **135**(2): p. 021005.

[123]. Rosen, J., M.B. Fuchs, and M. Arcan, Performances of Hill-type and neural network muscle models—Toward a myosignal-based exoskeleton. Computers and Biomedical Research, 1999. **32**(5): p. 415-439.

[124]. Winters, J.M., Hill-based muscle models: a systems engineering perspective, in Multiple muscle systems. 1990, Springer. p. 69-93.

[125]. Gordon, A., A.F. Huxley, and F. Julian, The variation in isometric tension with sarcomere length in vertebrate muscle fibres. The Journal of physiology, 1966. **184**(1): p. 170-192.

[126]. McMahon, T.A., Mechanics of locomotion. The International Journal of Robotics Research, 1984. **3**(2): p. 4-28.

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

[128]. Delp, S.L. and J.P. Loan, A graphics-based software system to develop and analyze models of musculoskeletal structures. Computers in biology and medicine, 1995. **25**(1): p. 21-34.

[129]. Delp, S.L., et al., An interactive graphics-based model of the lower extremity to study orthopaedic surgical procedures. Biomedical Engineering, IEEE Transactions on, 1990. **37**(8): p. 757-767.

[130]. Winters, J.M., Terminology and foundations of movement science, in Biomechanics and Neural Control of Posture and Movement. 2000, Springer. p. 3-35.

[131]. Yamaguchi, G., et al., A survey of human musculotendon actuator parameters. Multiple muscle systems: Biomechanics and movement organization, 1990: p. 717-773.

[132]. Visser, J., et al., Length and moment arm of human leg muscles as a function of knee and hip-joint angles. European journal of applied physiology and occupational physiology, 1990. **61**(5-6): p. 453-460.

[133]. Hermens, H.J., et al., European recommendations for surface electromyography. Roessingh Research and Development, 1999. **8**(2).

[134]. Hedrick, T.L., Software techniques for two-and three-dimensional kinematic measurements of

biological and biomimetic systems. Bioinspiration & biomimetics, 2008. **3**(3): p. 034001.

[135]. Barre, A. and S. Armand, Biomechanical ToolKit: Open-source framework to visualize and process biomechanical data. Computer methods and programs in biomedicine, 2014. **114**(1): p. 80-87.

[136]. De Luca, C.J., The use of surface electromyography in biomechanics. Journal of applied biomechanics, 1997. **13**: p. 135-163.

[137]. Weir, J.P., Quantifying test-retest reliability using the intraclass correlation coefficient and the SEM. The Journal of Strength & Conditioning Research, 2005. **19**(1): p. 231-240.

[138]. Laananen, D.H., A.O. Bolukbasi, and J.W. Coltman, Computer Simulation of an Aircraft Seat and Occupant in a Crash Environment. Volume 1. Technical Report. 1983, DTIC Document.

[139]. Silva, M., J. Ambrosio, and M. Pereira, Biomechanical model with joint resistance for impact simulation. Multibody System Dynamics, 1997. **1**(1): p. 65-84.

[140]. Young, J.W., et al., Anthropometric and mass distribution characteristics of the adult female. 1983.

[141]. Rustin, C., Physiological modelling and dynamic simulation of human walking. Faculty Polytechnique de Mons, Mons University, Mons, 2010.

[142]. DeWoody, Y., C. Martin, and L. Schovanec, A forward dynamic model of gait with application to stress analysis of bone. Mathematical and computer modelling, 2001. **33**(1): p. 121-143.

[143]. Martin, C.F. and L. Schovanec, The control and mechanics of human movement systems, in Dynamical Systems, Control, Coding, Computer Vision. 1999, Springer. p. 173-202.

[144]. Cheung, J.T.-M. and M. Zhang. Finite element modeling of the human foot and footwear. in ABAQUS Users' Conference. 2006.

[145]. Spong, M.W., S. Hutchinson, and M. Vidyasagar, Robot modeling and control. Vol. 3. 2006: Wiley New York.

[146]. Westervelt, E.R., et al., Feedback control of dynamic bipedal robot locomotion. Vol. 28. 2007: CRC press.

[147]. Peterson, D.R. and R.S. Adrezin, BIODYNAMICS: A LAGRANGIAN APPROACH. 2004.

[148]. Antsaklis, P.J. and X.D. Koutsoukos, Hybrid systems: Review and recent progress. Software-Enabled Control: Information Technology for Dynamical Systems, 2003: p. 273-298.

[149]. Albu-Schäffer, A., C. Ott, and G. Hirzinger, A unified passivity-based control framework for position, torque and impedance control of flexible joint robots. The International Journal of Robotics Research, 2007. **26**(1): p. 23-39.

[150]. Mallapragada, V., D. Erol, and N. Sarkar. A new method of force control for unknown environments. in Intelligent Robots and Systems, 2006 IEEE/RSJ International Conference on. 2006. IEEE.

[151]. Branicky, M.S., Multiple Lyapunov functions and other analysis tools for switched and hybrid systems. Automatic Control, IEEE Transactions on, 1998. **43**(4): p. 475-482.

[152]. Michel, A.N. and B. Hu, Towards a stability theory of general hybrid dynamical systems. Automatica, 1999. **35**(3): p. 371-384.

[153]. Shi, L., et al. A New Method of Quantitative Rehabilitation Rating in Robot-Assisted Recovery System. in Intelligent Networks and Intelligent Systems (ICINIS), 2012 Fifth International Conference on. 2012. IEEE.

[154]. Van den Bogert, A.J., Exotendons for assistance of human locomotion. Biomedical engineering online, 2003. **2**(17): p. 1-8.

[155]. Oberg, T., A. Karsznia, and K. Oberg, Joint angle parameters in gait: reference data for normal

subjects, 10-79 years of age. Journal of rehabilitation Research and Development, 1994. **31**(3): p. 199-213.

[156]. Lemmon, M.D., K.X. He, and I. Markovsky, Supervisory hybrid systems. Control Systems, IEEE, 1999. **19**(4): p. 42-55.

[157]. Quintero, H.A., R.J. Farris, and M. Goldfarb, A method for the autonomous control of lower limb exoskeletons for persons with paraplegia. Journal of medical devices, 2012. **6**(4): p. 041003.

[158]. Kandel, E.R., J.H. Schwartz, and T.M. Jessell, Principles of neural science. Vol. 4. 2000: McGraw-Hill New York.

[159]. Nigg, B.M. and W. Herzog, Biomechanics of the musculo-skeletal system. 2007: Wiley.

[160]. Brand, R.A., D.R. Pedersen, and J.A. Friederich, The sensitivity of muscle force predictions to changes in physiologic cross-sectional area. Journal of biomechanics, 1986. **19**(8): p. 589-596.

[161]. Friederich, J.A. and R.A. Brand, Muscle fiber architecture in the human lower limb. Journal of biomechanics, 1990. **23**(1): p. 91-95.

[162]. Arnold, E.M., et al., A model of the lower limb for analysis of human movement. Ann Biomed Eng, 2010. **38**(2): p. 269-79.

[163]. Wickiewicz, T.L., et al., Muscle architecture of the human lower limb. Clinical orthopaedics and related research, 1983. **179**: p. 275-283.