IZMIR KATIP CELEBI UNIVERSITY

GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES

DEVELOPMENT OF AN EFFECTIVE KINEMATIC SYNTHESIS AND ADMITTANCE CONTROL METHODOLOGY FOR REHABILITATION ROBOTICS

M.Sc. THESIS

Mertcan KOÇAK

Department of Mechanical Engineering

Thesis Advisors: Assoc. Prof. Dr. Erkin GEZGİN

Assist. Prof. Dr. Özgün BAŞER

JULY 2020

IZMIR KATIP CELEBI UNIVERSITY

GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES

DEVELOPMENT OF AN EFFECTIVE KINEMATIC SYNTHESIS AND ADMITTANCE CONTROL METHODOLOGY FOR REHABILITATION ROBOTICS

M.Sc. THESIS

Mertcan KOÇAK

(Y180217001)

0000-0002-3470-989X

Department of Mechanical Engineering

Thesis Advisors: Assoc. Prof. Dr. Erkin GEZGİN

Assist. Prof. Dr. Özgün BAŞER

JULY 2020

İZMİR KATİP CELEBİ ÜNİVERSİTESİ

FEN BİLİMLERİ ENSTİTÜSÜ

ROBOTİK REHABİLİTASYON İÇİN ETKİLİ BİR KİNEMATİK SENTEZ VE ADMİTANS KONTROL YÖNTEMİNİN GELİŞTİRİLMESİ

YÜKSEK LİSANS TEZİ

Mertcan KOÇAK

(Y180217001)
0000-0002-3470-989X

Makina Mühendisliği Ana Bilim Dalı

Tez Danışmanları: Doç. Dr. Erkin GEZGİN

Dr. Öğretim Üyesi Özgün BAŞER

TEMMUZ 2020

Mertcan KOÇAK, a M.Sc. student of **IKCU Graduate School Of Natural And Applied Sciences**, successfully defended the thesis entitled **"DEVELOPMENT OF AN EFFECTIVE KINEMATIC SYNTHESIS AND ADMITTANCE CONTROL METHODOLOGY FOR REHABILITATION ROBOTICS"**, which he prepared after fulfilling the requirements specified in the associated legislations, before the jury whose signatures are below.

Thesis Advisor :

        Assoc. Prof. Dr. Erkin GEZGİN
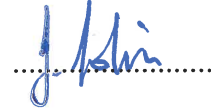   İzmir Kâtip Çelebi University

Thesis Co-Advisor :

        Assist. Prof. Dr. Özgün BAŞER
        İzmir Kâtip Çelebi University

Jury Members :

        Assoc. Prof. Dr. Savaş ŞAHİN
        İzmir Kâtip Çelebi University

        Assist. Prof. Dr. Fatih Cemal CAN
        İzmir Kâtip Çelebi University

        Assist. Prof. Dr. Özgün SELVİ
        Çankaya University

Date of Defense : 29.07.2020

To my wife

FOREWORD

First, I would like to thank my advisor and head of Medical Robotics Laboratory, Assoc. Prof. Dr. Erkin GEZGİN and my co-advisor Assist. Prof. Dr. Özgün BAŞER for their guidance, trust and support in this work. The way they took me is amazing and I am very proud to be part of their academic works.

I also want to express my gratitude to all of the professors and research assistants in the Department of Mechatronics Engineering at İzmir Katip Çelebi University, for their individual kindness and understanding. I am very lucky to have the opportunity to work as research assistant at this department.

The work group in the Laboratory of Medical Robotics is precious and I am much honored to be part of it. I want to thank all of the colleagues in this laboratory, who work and give efforts to make life better place. Their support and advices are very important to shape this thesis as well as my other works in the laboratory.

Despite the physical distances, I always felt my family's support behind me. I want to thank my father Mustafa KOÇAK, my mother Selma KOÇAK and my brother Mehmet KOÇAK for their endless love and encouragement in the whole of my life.

Having many friends that I cannot reveal all the names one by one here shows how lucky I am. They all have effects on me in the social and professional life. I thank all of my friends that gives courage and support me in this work as well as in the social life.

Finally, I would like to thank my wife, Dilara KOÇAK, for her endless patience, support and faith in me. In most pessimistic and depressed times, she gives me the courage and necessary motivation and I appreciate it.

July 2020

Mertcan KOÇAK

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

ABBREVIATIONS

2D	**:**	Two Dimensional
3D	**:**	Three Dimensional
DOF	**:**	Degree of Freedom
OSI	**:**	Open Systems Interconnection
ISO	**:**	International Organization for Standardization
IP	**:**	Internet Protocol
TCP	**:**	Transmission Control Protocol
UDP	**:**	User Datagram Protocol
API	**:**	Application Programming Interface
PCI	**:**	Peripheral Component Interconnect
USB	**:**	Universal Serial Bus
SDK	**:**	Software Development Kit

DEVELOPMENT OF AN EFFECTIVE KINEMATIC SYNTHESIS AND ADMITTANCE CONTROL METHODOLOGY FOR REHABILITATION ROBOTICS

ABSTRACT

Throughout the human life, neurological disorders can occur in body areas due to accidents or other biological factors, which negatively affects human life by destroying nerve commands. In these scenarios, continuous and efficient exercises that focus on injured extremities have great benefits in regaining functional losses by retraining motor movements. Rehabilitation procedures need to be applied quickly in order to create neural plasticity and regain affected motor functions. Although these actions are mostly performed by therapists in current rehabilitation procedures, robotic rehabilitation systems also begin to be included in these treatments with the help of technological developments. In light of this, not only the accuracy of the exercises, but also the chances of recovery of patients have increased, because there are a limited number of therapists compared to the increased rehabilitation needs associated with population growth. Given these facts, this thesis provides an effective methodology that includes the steps to be used for lower degrees of freedom robotic rehabilitation system designs and supports them with applications.

The method used in this thesis starts with the collection of 3D position data from human limbs and then continues with the transfer of this data to the best plane to make it suitable for synthesizing the planar mechanism. As a next step, usage of the reduced 2D data in kinematic synthesis is explained and exemplified by a case study. Utilization of the planar single degree of freedom mechanisms brought up to this stage with admittance control algorithms for rehabilitation procedures has been explained theoretically and case studies have been carried out on two different systems. Control algorithms created by these case studies were combined with a game rehabilitation system and their functionality was tested with three different games that are also developed throughout the thesis.

ROBOTİK REHABİLİTASYON İÇİN ETKİLİ BİR KİNEMATİK SENTEZ VE ADMİTANS KONTROL YÖNTEMİNİN GELİŞTİRİLMESİ

ÖZET

İnsan ömrü boyunca, vücut bölgelerinde kazalar veya diğer biyolojik faktörler nedeniyle nörolojik bozukluklar meydana gelebilir ve bu durum sinir komutlarını yok ederek insan yaşamını olumsuz yönde etkilerler. Bu senaryolar durumunda, yaralı ekstremitelere odaklanan sürekli ve verimli egzersizlerin, motor hareketlerini yeniden eğiterek fonksiyonel kayıpları yeniden kazanma açısından büyük faydaları vardır. Nöral plastisite oluşturulabilmesi ve etkilenen motor fonksiyonlarının yeniden kazanılabilmesi için rehabilitasyon prosedürlerinin hızlıca uygulanması gerekmektedir. Bu eylemler çoğunlukla mevcut rehabilitasyon prosedürlerinde terapistler tarafından yapılsa da, teknolojik gelişmeler yardımıyla robotik rehabilitasyon sistemleri de bu tedavilere hızla dahil olmaya başlar. Bunun ışığında, sadece egzersizlerin doğruluğu değil, aynı zamanda hastaların iyileşme şansı da artmıştır, çünkü nüfus artışı ile ilgili artan rehabilitasyon ihtiyaçlarına kıyasla sınırlı sayıda terapist vardır. Bu gerçekler göz önüne alındığında, bu tez düşük serbestlik dereceli robotik rehabilitasyon sistemlerinin tasarımı için kullanılması gereken basamakları içeren ve bunları uygulamalarla destekleyen etkili bir yöntem sunmaktadır.

Tezde kullanılan yöntem insan uzuvlarından 3 boyutlu pozisyon verisinin toplanması ile başlamakta daha sonra bu verinin düzlemsel mekanizma sentezlemeye uygun hale getirilmesi için en iyi düzleme aktarılması ile devam etmektedir. Bir sonraki aşama olarak indirgenen 2 boyutlu verinin mekanizma sentezinde nasıl kullanılacağı anlatılmış ve bir durum çalışması ile örneklendirilmiştir. Bu aşamaya kadar getirilen düzlemsel tek serbestlik dereceli mekanizmaların, rehabilitasyon amacıyla admitans kontrol algoritmalarıyla nasıl kullanılacağı teorik olarak anlatılmış ve 2 farklı sistem üzerinde durum çalışması gerçekleştirilmiştir. Bu durum çalışmalarıyla oluşturulan kontrol algoritmaları ise oyunlaştırılmış rehabilitasyon sistemiyle birleştirilmiş ve tez kapsamında geliştirilen 3 farklı oyun ile fonksiyonelliği test edilmiştir.

1. INTRODUCTION

After suffering from some diseases or going through traumas, people may lose their neural commands and cannot control their muscles properly. In such cases, in order to increase the probability of recovering, they should start physical rehabilitation as quick as possible. This plays a vital role for recovering the ability of motion, where the problem stems from the brain damage.

Neurological disorders cause damage to motor functions of human, which impacts patient's life in terms of physical, psychological and social disorders. After these kinds of problems, high-level acute care is needed as soon as possible, which is actually the beginning of the treatment. In the long term, physical rehabilitation is needed for patients to regain motor functions [1]. The main purpose of rehabilitation is to keep disability at minimum and reach the functional independency as much as possible. It is important to integrate the patient to daily life again.

Stroke is the most common neurological disorder among the people, which is the second most common cause of death. According to the definition of World Health Organization, stroke is a neurological dysfunction of vascular origin and characterized by the sudden onset of findings relative to the affected brain area [2]. Although researches show the risk factors of the stroke and warn the people [3], it is still the most common factor behind the disabilities among elderly people [4] and it is predicted to rise all over the world [5, 6] as well as Turkey [7-10]. Huge amount of the capability in neurology services and rehabilitation hospitals are reserved for patients with stroke. The purpose of rehabilitation for stroke patients is mainly to support them to learn how to meet their needs successfully.

In studies conducted, it is stated that prognosis is more severe if movement does not start within the first 4 weeks after stroke [11]. However, it is not always possible to achieve successful rehabilitation results due to the low number of physiotherapists and ergotherapists working in the field of stroke rehabilitation compared to the

patient amounts, need for one-to-one treatment and also the requirement of high treatment durations to be allocated to a patient. In addition to these, as the trained health professionals are mostly located in larger centers, it becomes impossible access to treatment resources for many patients.

Considering previous facts, robotic rehabilitation has much more advantages over the classical rehabilitation. It has a higher impact on patients' motivation and by the help of precise measurement devices, evaluation of the treatments can be carried out objectively. Also, this type of rehabilitation supports the process in learning with motor functions, thanks to its ability to execute repetitive procedures. Motor learning is defined as a process that requires skill in mobility and includes permanent changes that occur with experience or practice. Repetitions to be made for the targeted ability should be based on functional goals, a different variety of activities should be used each time, and a process that allows the patient to process information automatically should be followed. In order to learn a skill, a minimum of 400 repetitions per day with 1 hour of treatment and a total of 60 hours of treatment each day is needed. Thus, it is not possible to apply this intensive treatment program to each patient with a physiotherapist [12, 13]. Robot-supported rehabilitation systems reduce the physical burden of physiotherapists and ergotherapists, enabling high-intensity, safe and flexible treatments. One of the most important features of the robotic rehabilitation is that they can assist the completion of the movements as much as they need or show the desired amount of resistance for strengthening exercises [14]. Additionally, it has been shown that healing can be increased by simulating functional movements and skills in virtual environments created with the help of computers [15]. As a summary, robot-assisted treatment increases the motivation of the patient, controls whether exercise is performed effectively or not, reduces the cost of treatments, allows utilization of the same system to be used for different purposes in different patients and provides objective data evaluation.

In light of these, purpose of this thesis is to create a methodology to be followed in designing a low degree of freedom rehabilitation device from scratch. This methodology consists of motion capturing for data extraction, kinematic synthesis for mechanism design, design of interaction control and their implementation via game therapy.

1.1 Robotic Rehabilitation Procedures

The importance of the rehabilitation can be seen in such cases, where the human life is affected strongly. However, although need for rehabilitation therapists is increasing rapidly in the world, the number of proficient therapists is limited. Also, as therapists' work relies on the human body and muscle activities, the overall process is extremely exhausting for the therapists and throughout the treatment period quality of the work is affected by the human-related reasons. As this is the case, nowadays, robotic rehabilitation becomes an important topic.

According to the patients' current recovery stage, the therapy to be applied to the subject changes no matter it is a robotic or classical therapy. The effectiveness, probability and time of the recovery strongly depend on the correct therapy after the incident. For this reason, as in the classical therapy, robotic rehabilitation treatments also need to follow some procedures that stem from the neuro-rehabilitation facts and physiotherapists' experience.

In the literature, there are many works that aim to increase effectiveness of recovering process in both classical rehabilitation and robotic rehabilitation. Suitable rehabilitation working mode for the concerning devices strongly depends on the patients' situation. As mentioned in the work [16], stages of the rehabilitation can be subdivided into three category; preliminary rehabilitation, intermediate rehabilitation and advanced rehabilitation. Depending on their treatment stages, different rehabilitation procedures should be applied to the patients by therapists. These treatments can be generalized as passive and active ones. In passive treatments, patients do not perform the treatment activity, instead, robotic device forces to move patient's concerned area, in order to improve the movement ability and create plasticity in the brain. This treatment methodology, in the early stages of the rehabilitation, plays a vital role in recovering due to the fact that it allows regaining lost motor functions by the help of brain plasticity. In active treatments, patients must provide some level of muscle activities to perform the treatment. Obviously, patients can participate to active treatments, if and only if they regain some amount of strength and ability to send neurologic signals to the affected extremity. Depending on the level of the patient, rehabilitation device can assist, correct or resist the

patients' motions. These sub-modes strongly depend on the stage of the therapy. In the review work of Poli et al. [17], the control strategies are subdivided in to 5 different modes as; passive movement mode, active non-assist mode, active assist mode, resistive mode and bimanual exercise mode. Also, in the control strategies review work for upper limb exoskeletons, the control strategies are divided into three different categories, namely; assistance, correction and resistance [18].

Due to the fact that rehabilitation robots in this thesis are actually one degree of freedom systems, the correction phase is irrelevant because the motion is restricted to the correct motion as the mechanism itself. Also, since the concerned systems work in one part of the body, it is not possible to adapt bimanual exercise mode. In the light of rehabilitation procedures in the literature, in this work, three different modes are applied to the control strategy; passive mode, active assist priority mode and resist mode.

1.1.1 Passive mode

Creating plasticity in patients' brain plays a vital role at the beginning of the rehabilitation process. For this reason, the passive mode is included in which the actuator of the system is driven without any intention from the patient. This is the first part to start the lost muscle activity at the target extremity, which is usually unresponsive. In this mode, the velocity/motion control is activated with predefined characteristics.

1.1.2 Active assist priority mode

Active assist priority mode is the first step of the interaction control in treatment. In this stage, depending on the amount of intention from the patient, he/she is supported by the actuator on the system. The main purpose of the procedure is to amplify the intention in order to achieve the movement. This mode is divided into two different sub strategies; the first one is to assist the patient in terms of the whole movement and the second is to vanish the external mechanical effects of the system itself.

1.1.2.1 Active assist mode

According to the patients' recovery stage, there are some cases where the patient can only intent for the motion but need some assistance from the robot itself in order to execute it, which is a huge part of the treatment. In this sub-mode, rehabilitation system needs to be activated by the patients' small intention. This intention is amplified by the control algorithm to drive the system as if the patients drive themselves. This assist from the robot helps to increase the awareness and concentration of the treatment at the same time leading patients to activate the motion repeatedly. Here, rehabilitation robot let patients to perform any movement by amplifying the muscle activity without suppressing any motor capability [19, 20].

1.1.2.2 Active partial assist mode

In this sub-mode, patients achieve the movement on their own, without any extra help from the robot. However, since the mechanical system has inertia, friction, gravitational effects and non-back drivable actuator, there should be a control algorithm in order to get over those effects.

1.1.3 Active resist mode

At the advanced stage of the treatment, active resist mode comes into the action where the rehabilitation robot resists the patients' motion in order to enhance the muscle strength of the target area. The actuator of the system generate a predefined counter force, which makes the system harder to be driven. Suitable control algorithm is applied to the system that will be discussed in Sub-section 1.4.

1.2 Kinematic Synthesis

Since the ancient times of human life, people have been using some mechanisms and machines in order to achieve some simple or complex tasks. Before humankind can reach the chemical or electrical power, these tasks with mechanisms are powered by the humans or animals. After the usage of different types of power sources, they started to design and use different kinds of mechanisms as well, expectedly. The term mechanism can be roughly defined as the system of bodies to transform input motions and forces to a desired motion and forces with moving components powered by a power source.

The design of this input-output relation in terms of motion and force are one of the major concepts in the history to manufacture different kind of machines that ease the human life. In light of this design procedures have been followed that are divided into two main categories as synthesis and analysis. Synthesis is a design process to design the best type and dimensioned machine or mechanism to perform a given or required task. It is also divided into two sub sections as structural and kinematic synthesis, where the kinematic structure of the mechanism is revealed on the former one while the construction parameters were determined in the latter one if the systems degree of freedom is to be kept lower than the task degrees of freedom. On the other hand analysis part is all about determining the motion characteristics of a mechanism that is already created or existing. Thus the analysis can be done after the synthesis procedures are applied in case of a new design, which is the same as in this thesis in Sub-sections 3.1 and 3.2. It should also be noted that analysis part can be divided into another two sections as kinematic and dynamic analysis where the forces and moments are integrated to the analysis procedures in latter one.

To design any mechanisms, the concept of the structural synthesis is divided into two categories; type synthesis, and number synthesis and dimensional synthesis.

Type synthesis is about choosing the type of the mechanism such as linkage, belt-pulley, gear train or cam mechanisms to achieve the desired motion.

Number synthesis is about determining the number of the links and joints that should be utilized in the system as well as their types. Both type and number synthesis together can be categorized as the conceptual design, which is the beginning of the design procedure and there are many approaches along with this concept in the literature in terms of spatial [21] and variable general constraints [22, 23].

Dimensional synthesis is about determining geometric parameters such as link lengths. This is important part of the synthesis procedure, which is also named as functional dimensioning. This procedure does not concern about the mechanical dimensioning part, which actually deals with the mechanics of materials, such as strength, stress etc. It is obvious that dimensional synthesis can be started only if the conceptual design procedure is revealed. For example, planar five-bar mechanism

can be used in some procedures by considering the conceptual design, which has the type synthesis (linkage) and number synthesis (two degree-of-freedom with 5 links and 5 joints), but the dimensions of the links come after by also utilizing parametrical mechanism analysis.

Although kinematic synthesis is also related with construction parameter dimensioning, it can only be utilized on the conditions where the task mobility is greater than the mechanism mobility. It is for designing lower degree of freedom mechanisms to execute higher degree of freedom tasks. In general, by considering the requirement of the output motion of the mechanisms, kinematic synthesis procedure can be divided into three categories; function generation, path generation and body guidance (a.k.a. motion generation).

In function generation, the motion of the output link is bind with the input motion in terms of a function [24]. In path generation, the motion of the output link is to trace a predefined path [25]. In body guidance or motion generation, the motion of the output is to follow a predefined pose (path and orientation together) at different time steps. For using this kind of method, the problem is reduced to dyad synthesis, which is described in Sub-section 3.1 and in Gezgin's work [26].

The given or required task can be achieved exactly or approximately according to both the mobility of the task and the mechanism itself. If the mobility is sufficient to produce motion task exactly, then it is possible to achieve the motion without error. On the other hand, there is an opportunity to decrease the mobility of the mechanism in order to have a less complex and compact solution for the task motion. In such cases, the task mobility is higher than the mechanism, so it is not possible to obtain an exact solution. Instead, approximate solutions are considered that the designed mechanisms follow the requirement motion with a finite error. In applying this kind of approximation, precision points are taken into the action which is the number of finite number of points where the path or function is exactly fulfilled. The number of the precision points must be equal to the number of unknown construction parameters used in the synthesis [27].

The kinematic synthesis procedure for lower degrees of freedom systems basically relies on the planar position data. In some cases as the body guidance method, orientation data of the output link is needed as well. These data can be formed by a mathematical model or randomly generated depending on the requirement of the final position and orientation path.

The rehabilitation devices can be multi degree of freedom exoskeletons or tip follower type systems. Multi degree of freedom exoskeletons have high amount of movement integrity, and by the help of multiple joints they have high mobility. However, they are complex and hard to control and expensive as expected. Tip follower devices on the other hand only focus on the end-effector movement, depending on the mobility of the system. In the design process, they require end-effector's motion data to be given as a task. They are simple and cheaper.

In this thesis, tip follower type of planar rehabilitation mechanism with one degree of freedom is synthesized by using the body guidance method, where the path and orientation data relies on the real data collected by the motion capture system at different extremities. The captured data is collected at least 3 times and bind into the regression curves to be normalized.

1.3 Motion Capture System

Motion capture is generally the method of recording the movement of the objects, animals or human with different hardware and software. There are different methods of performing systems, such as using optical methods, inertial sensors, magnetic or mechanical systems. Optical methods use mostly the camera-based systems such as the ones manufactured by Vicon, Optitrack or Microsoft (Kinect). While the former two use the reflector markers in recording the motions, the latter uses the infrared based system combining with the special software to detect the human body system. In this thesis, one of the optical methods that provided by Optitrack Motion Capture System (Figure 1.1) is used to record the motion of different extremities in order to design the different rehabilitation robots by using kinematic synthesis methodologies.
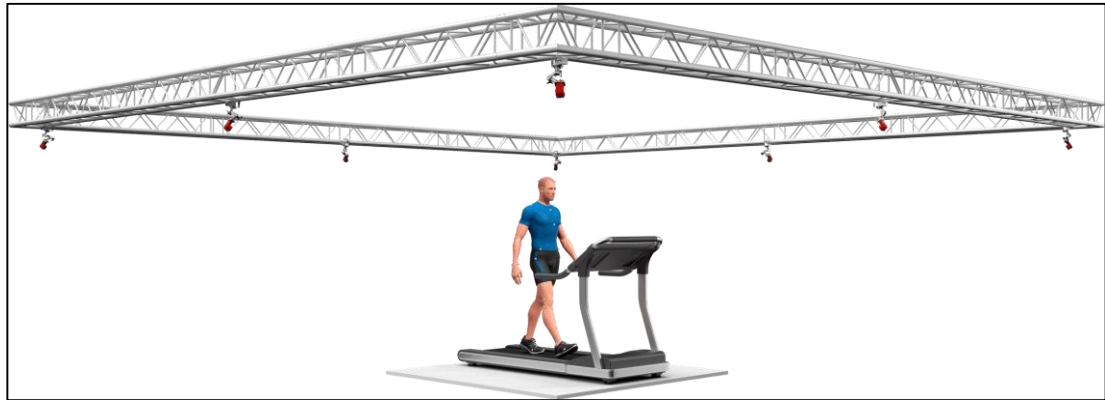
Figure 1.1 Optitrack Motion Capture System [28].

Optitrack motion capture systems provide camera based motion capture solutions in terms of hardware and software [28]. Provided motion capture system is widely utilized in sectors that use the graphical environments such as gaming or film industry. As it is very usable in detecting the motions with its reflector markers, the usage is jumped into the biomechanics researches as well. Optitrack is counted as one of the gold standards in clinics motion systems. According to the technical data of the system, the maximum latency during capturing is 100 milliseconds, with its maximum rate of collecting data 100 frames per second.

The system tracks reflector markers that are placed on objects or extremities in systems workspace and collects position information with respect to time. It is also possible to stream the data in real-time to various platforms, including Unity3D, MATLAB or dotnet based custom software, with its NatNet Software Development Kit (SDK) by using the stream protocols such as UDP, Unicast or Multicast.

Procedural approach begins by constructing a stable working space towards which the cameras are pointed. In order to proceed exact location of the cameras with respect to each other and their orientations should be known. For this reason, calibration process plays a vital role here. As the next step, calibration should be done via special calibration wand, which is included in the system package (Figure 1.2). The calibration wand is a special structure that carries three reflector markers with predefined geometry. In this process, the provided wand is moved in the workspace, especially on most used areas, in order to determine relative positions and orientations of the cameras with respect to each other in target workspace.
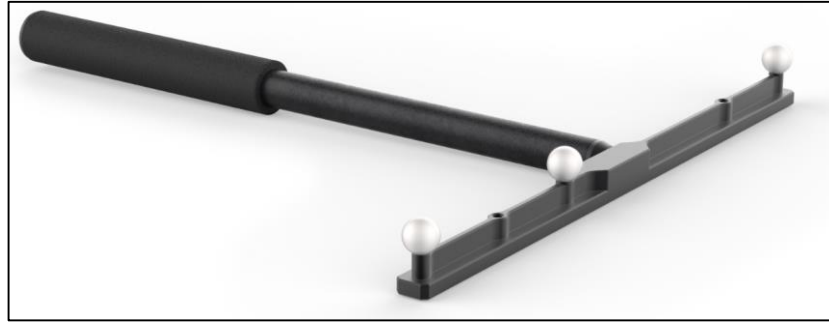
Figure 1.2 Optitrack's calibration wand.

Here, the synchronization between the cameras and the personal computer that executes the software is carried out by using Optihub device (Figure 1.3), which is also responsible to distribute the power to the cameras.

Figure 1.3 Optitrack's Optihub.

The last step of the calibration process is to define the local coordinate system of the motion capture system. This is important when the directions matter in the collected data. In Sub-sections 2.1 and 2.2, Cartesian coordinates play vital role in order to extract the path data onto the predefined planes. This definition is done by using the calibration square, which is also included in the system package (Figure 1.4). Similar to the calibration wand, this device is also a predefined structure that has reflector markers on it. This helps software to determine the Cartesian coordinates and directions, also defines the origin of the coordinate system.

Figure 1.4 Optitrack's calibration square.

One of the most important feature of this system is to be able to create a custom rigid body structure. Rigid bodies are a predefined geometries that have at least three different reflector markers stable with respect to each other. This gives the opportunity to determine the orientation of the whole mini structure besides the position data. The orientation data is very useful for the motions with rotations. Another advantage of utilizing rigid bodies is the fact that if one of the markers on the rigid body becomes out of sight from the cameras' view, with the help of the predefined structure, the position of out of sight marker can be estimated (Figure 1.5).
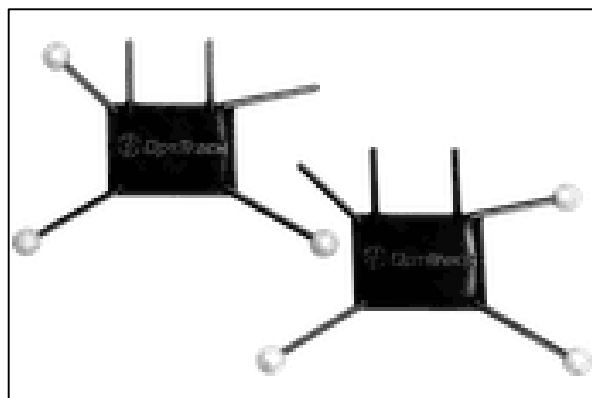
Figure 1.5 Optitrack's special rigid body structure.

In this work, 6 different Flex3 cameras (Figure 1.6a) are used in the collecting data phases as in the Chapter 2. The licensed software, Motive (v1.7.2.), is used to

manage the system in terms of the calibration, configuration and data collecting (Figure 1.6b).
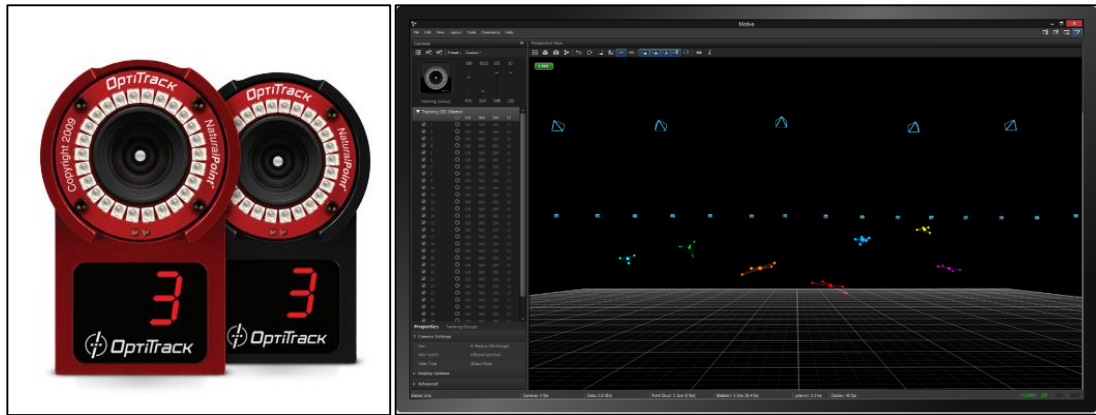
Figure 1.6 a) Optitrack's flex3 cameras **b)** Optitrack's motive software.

Data collecting is the first step of the kinematic synthesis procedure of the tip follower type rehabilitation system. In this thesis, three different data collecting processes are handled for hand rehabilitation system path extraction part, knee orthosis synthesis and one degree of freedom upper extremity mechanism.

1.4 Interaction Control Strategies

Mechanical interaction with objects is very essential for robots that work in collaboration with the environment or humans. This concept is growing fast, since the human-robot interaction becomes the part of the life, which is required in huge amount of robotic applications [29-31]. The health sector, including physical rehabilitation, is one of the area that strongly needs to interact with the robots [32, 33].

In literature and industry, the most of the robotic devices are motion based controlled and there are many approaches to design such controllers for different requirements [34-36]. Such controllers consider the environmental forces and torques as disturbances, which are needed to be overcome to follow the reference motion trajectories. However, if these robotic devices have physical interaction with the environment as well as the humans, pure motion based controllers become clearly dangerous, due to the fact that controller scheme does not consider their possible

output forces and torques to the environment they are in. As their nature, if the motion errors becomes larger, with sufficient amount of control gains, the contact forces can increase dramatically. Thus, it is obvious that if the robotic devices need an interaction with the environment, which is very common in rehabilitation robots, they require additional control approach, named as interaction control that actually must consider the forces as well.

The interaction control strategies are grouped in two categories; direct force control and indirect force control. The direct force control is used for controlling the contact force to a reference value with its force feedback loop. Such controllers require regulation of the contact force to a constant desired value. Indirect force control, however, fulfills the force control via motion control, without certain need of explicit closure of a force feedback loop [37]. In the study of Zeng and Hemami [38], the robot force control methods are examined and their methods are compared.

Rehabilitation robots have direct physical interaction with users, at the same time, they need to follow a predefined or online created motions – position, velocity or acceleration. For this reason, the motion and the force must be regulated or concerned together. As mentioned in Sub-section 1.1, in some stages of the robotic treatment, it is possible to define some force/torque levels to resist the patients' motion, which is directly concerned by the interaction control algorithms. As they work directly in physical contact with human, the forces must be controlled and limited due to the safety reasons also.

1.4.1 Direct force control

Direct force control is a feedback controller type that requires the contact force measurements as the main feedback and controls the contact force to a reference value with the help of the force feedback loop on the system, mostly by using the model-based of system. The main drawback of this type of controller is that the contact force is always needed, thus it is not the best option to be worked in free space. Hybrid position/force control and parallel force/position is counted in this type of controllers. The former type is used by applying either a motion or force/torque based controller in separate axes of the end effectors. In such case, the position

controlled axes are constraint-free from the force, which means that only motion controller is active. Thus it is possible to comment that this type of controller divides task space into two as position controlled and force controlled ones. In Dede's work [39], he clearly stated that this controller takes advantage of both position and force control for each and combined. This leads that there is no dynamic behavior between the applied force and motion of the end-effector. In the block diagram representation of the hybrid position/force control in Figure 1.7, **S** is the compliance selection matrix, which determines the type of the controller in sub space. Here, **S** is a diagonal matrix with the dimension of degree of freedom of the system. The value on the diagonal is selected as 1 or 0 depending on the desired controller type; it is 1 for position control and it is 0 for force control. This **S** matrix can be created with fixed values or the values can be continuously changed to serve a purpose. However, switching may lead to unstable responses during manipulation, which is the main drawback of this type of controller.
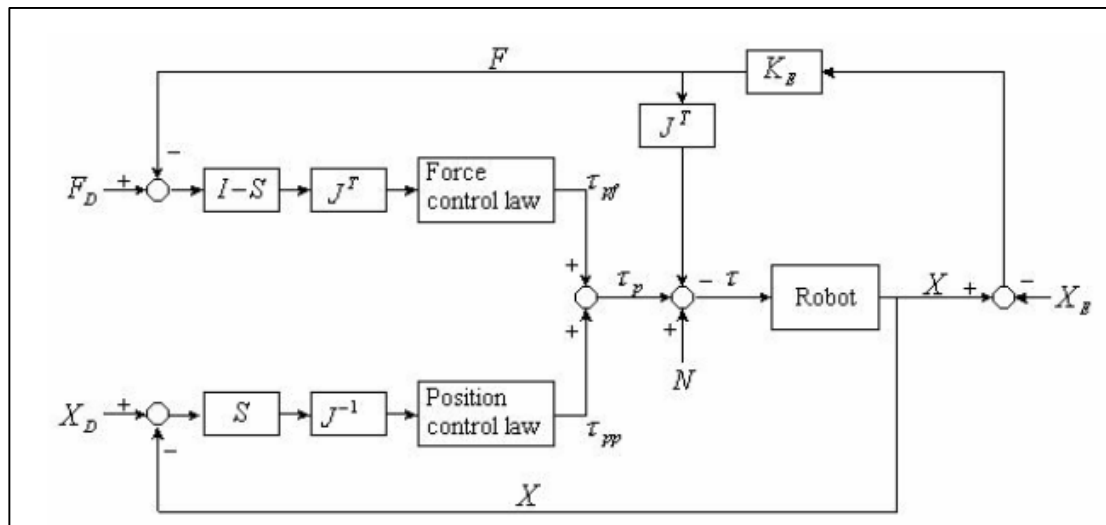
Figure 1.7 Hybrid Position/force Control [39].

Different type of force control law in hybrid force/position controller are used in applications as explicit or implicit.

1.4.2 Indirect force control

Indirect force control methods use the motion and force controllers together. The motion controllers are used as the primarily loop which is fed by the force controlled

external loop. Thus, the force control is achieved indirectly thanks to the primarily motion loop.

Interaction control in terms of the indirect force control is started to be used in literature back in 1984 with the work of Hogan [40] based on impedance control. Until Hogan's work, many works along with the force and motion in control have been conducted because of the critical role of force and motion, however all the interaction control strategies are presented in terms of direct force control, or very similar to indirect force control [41]. Hogan's researches and applications were based on the mechanical impedance, which is the analogy of the electrical impedance that is described as the electrical resistance to electrical current. As mechanical systems, he adopted this concept into the control of the mechanical interaction, the force and the motion [42].

As direct force control applications, before indirect force control concept is introduced, admittance alike force/motion control model application is used by Whitney in 1977 [41]. This work is also referred as damping control [43] or admittance control. Admittance control is defined as the opposite of the impedance control, which accepts the force as the input and with its internal dynamics, which is discussed in the next Chapters, creates the reference motion trajectory for the primarily control loop in the system.

Both impedance and admittance are control methods that describe the manipulator to interact with a variety of environments and they are widely used as interaction control strategy in the literature and among industrial robots. They are counted as the sub category of the indirect force control, which is roughly defined as the dynamic behavior between the applied force/torque and the motion. Also, by contrast to direct force control method such as hybrid position/force controller, the impedance and admittance controls do not have any switching phase that leads to unstable responses.

1.4.2.1 Impedance control

As Hogan mentioned in his work [40], command and control of pure position or force is not enough to control dynamic interaction between systems, therefore the controller must also control the relation between those variables. Starting with the

impedance control, the goal of which is to control the dynamic relation between the force and the motion according to required task by controlling both at the same time. This leads to increase the dynamic performance of the manipulator and safety where the human-robot or environment-robot interaction occurs [44]. Thus, by using the motion feedback, the impedance control actually applies a virtual physical attitude between the reference and actual position of the end effector, if the virtual impedance is arranged as second order dynamics. Therefore, as Hogan mentioned, the manipulator should be designed to regulate the mechanical impedance of the manipulator along with the motion trajectory. In Laplace domain, the mechanical impedance is described as the relation between the velocity and the applied force.

$$Z_l = \frac{F(s)}{v(s)} \tag{1.1}$$

$$Z_r = \frac{\tau(s)}{\omega(s)} \tag{1.2}$$

where the Z_l is the linear mechanical impedance, Z_r is the rotational mechanical impedance, $F(s)$ is the applied force, $\tau(s)$ is the applied torque, $v(s)$ is the linear velocity and $\omega(s)$ is the rotational velocity.

The equations 1.1 and 1.2 can be expressed in terms of position as well, if the motion control is desired to be position based control.

$$sZ_l = \frac{F(s)}{X(s)} \tag{1.3}$$

$$sZ_r = \frac{\tau(s)}{\theta(s)} \tag{1.4}$$

where, $X(s)$ is the linear position, $\theta(s)$ is the angular position.

The general second order dynamic model of the force and motion based systems can be conducted as follows;

$$f = m\ddot{x} + c\dot{x} + kx \tag{1.5}$$

where, f is the external force, m is the mass of the system, c is the linear damping coefficient, k is the spring constant and x is the displacement in the motion direction,

$\dot{x}$ is the velocity in the motion direction and $\ddot{x}$ is the acceleration in the motion direction.

If the equation 1.5 is combined with the relation between position and the force, mechanical virtual impedance can be determined, where the sub letter v denote virtual.

$$sZ_l = \frac{F(s)}{X(s)} = m_v s^2 + c_v s + k_v \qquad (1.6)$$
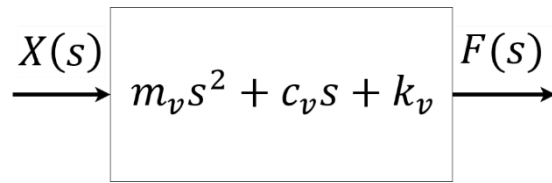
Figure 1.8 Input-output relation of the impedance control.

As seen from the input output relation of the impedance control (Figure 1.8), by using the position feedback, with the second order virtual impedance, physical attitude is created.
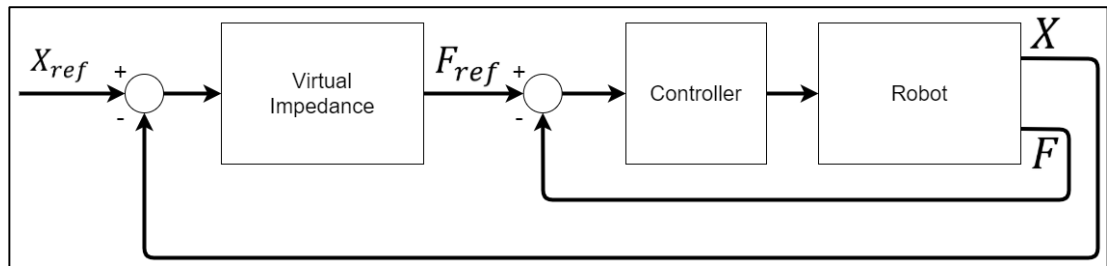
Figure 1.9 Simple block diagram of impedance controlled robot.

In literature and industry, simple and active impedance control strategies (Figure 1.9) are used very widely among the rehabilitation robots that aims to different extremities of the human body [45-48].

1.4.2.2 Admittance control

As implied in previous Sub-section, impedance control measures the positional deviation of the robot and then controls its actuator forces such that the desired impedance is produced. By contrast, the admittance control tries to satisfy the environmental interaction and respond to forces by recasting the reference velocity

trajectory. It focuses more on the force tracking control. The most important application area, in this work as well, is the fact that such controller is suitable for non-back-drivable systems.

Suppose that there is an end-effector of the manipulator, or simply one degree of freedom mechanism moved by an actuator with a huge transmission ratio, and it should be backdriven by an operator from the end-effector with a handle. In such a case, it is not easy to drive the mechanism with a given effort at the end-effector by the user, therefore, a force based motion controller is needed to drive the system as if the operator himself/herself move as a whole. Due to this large inertia and the non-back drivability, the impedance controller is forced to be out of the scope, therefore, in this work, admittance controller is used.

The mechanical admittance is defined as the ratio of the linear/rotational velocity to applied force/torque.

$$A_l = \frac{v(s)}{F(s)} \tag{1.7}$$

$$A_r = \frac{\omega(s)}{\tau(s)} \tag{1.8}$$

where A_l is the linear mechanical admittance and A_r is the rotational mechanical admittance. In this work in both case studies, all the motions are actually rotational and the intentions are the torques, the equation 1.8 will be used in determining the mechanical admittance.

The general second order dynamic model of the force and motion based system is revealed in equation 1.5. Binding this dynamic to mechanical admittance as in equation 1.7, the following relation is conducted with zero initial conditions.

$$F(s) = mv(s)s + cv(s) + \frac{kv(s)}{s} \tag{1.9}$$

where $F(s)$ is the external force in the Laplace domain, $v(s)$ is the linear velocity in the Laplace domain in the motion direction. Therefore, it is possible to construct the virtual mechanical admittance in the rotational direction as;

$$A_r(s) = \frac{\omega(s)}{\tau(s)} = \frac{s}{J_v s^2 + b_v s + \kappa_v} \qquad (1.10)$$

where J_v is the virtual rotational inertia, b_v is the virtual rotational damping coefficient and κ_v is the virtual rotational spring constant.
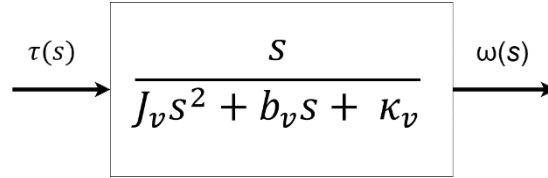
Figure 1.10 Input-output relation of the admittance control.

As seen from the input output relation of the admittance control (Figure 1.10), the relation between the applied torque and the rotational velocity trajectory strongly depends on the virtual effects of the admittance controller.
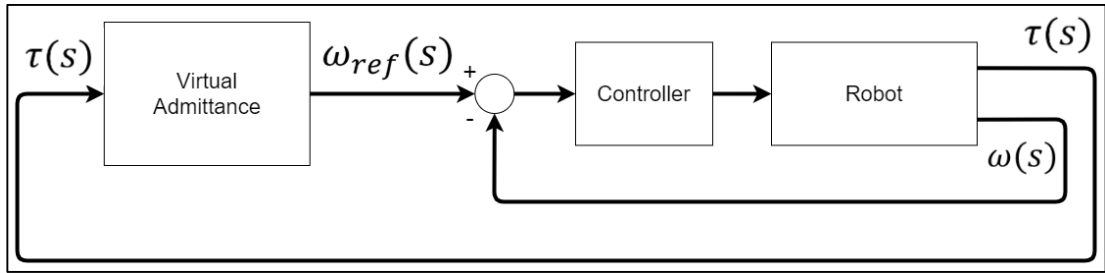
Figure 1.11 Simple block diagram of admittance controlled robot.

Simple block diagram representation of the admittance control scheme in Figure 1.11, due to the force/torque error, the system is affected. With the internal dynamic of the virtual admittance control, the rotational velocity profile is created to activate the inner motion control loop, which is the simple velocity controller in this representation.

The admittance based rehabilitation systems are widely used in the industry and literature [20, 49-52]. This kind of control is very suitable for this work, due to the fact that the force/torque intention is expected from the user while the system has a non-back drivable mechanical configuration. Also, as mentioned in the Sub-section 1.1, rehabilitation procedures will include the resistance motions, which is possible to be adopted by using this kind of controller.

By considering dynamics of general systems, parameters of inertia, damping and spring coefficient affect system behavior in transient and steady state. It is obvious that if admittance parameters are high, a larger effort is needed to move the system because of the virtual dynamic parameters. In such case, fine movements can be easily done. Consequently, if the admittance parameters are low, it is easier to move the robot. At the same time, performed movements with that virtual effects is rougher, so fine movements cannot be done easily.

When rehabilitation procedures are considered, the virtual effects actually fit the distinct modes. For example, if the active resist mode is required, high admittance parameters are expected, naturally. Similarly, if the active partial assist mode is in the action, admittance parameters should be arranged carefully, so that the controller senses the intention and drive the actuator as if there is no external effects like gravity or inertia. It is obvious that the admittance parameters must be very low, on the other hand, they should be over a ratio below which the stability issues comes forward. In this work, a novel method is used in order to achieve the rehabilitation properly in terms of assistance and resistance. Other than the admittance virtual dynamics, offset torque is applied as the desired torque. This offset torque can be positive or negative whether it is desired to assist or resist to motion. This kind of control structure is applied in related works with tuning and game therapy.

Here, it is inevitable that the dynamic effects is kept due to the admittance gain when the force intention from the user is cut suddenly. For example, if the current mode is active partial assist in such a way that the admittance parameters are low and the force intention is interrupted, motion will be kept due to the low virtual inertia and friction on the system. This is an unwanted situation as the motion should not be continuing, instead it should stop.

This is the main drawback of the fixed admittance control. In order to get rid of this drawback, virtual parameters of the admittance controller can be altered following some rules. This kind of admittance controller is called variable admittance controller and it is widely used in the literature [53].

As in the works of related literature [54], minimum virtual inertial parameter and virtual rotational damping parameter for two different rehabilitation scenarios are determined experimentally in order to extract the usable area of the stability for the variable parameters. According to this area, the parameters should be chosen in different rehabilitation modes for keeping the system in stable.

In light of given information above, this thesis focuses on creating a methodology for robotic rehabilitation with three topics; synthesis, interaction control and gamification. Tip follower type of rehabilitation device is designed via kinematic synthesis methodologies based on the real motion data of the human body that is collected by the help of motion capture system. One of the explained interaction control strategies is applied to the readily built hand rehabilitation system. This control strategy is combined with the game rehabilitation. The rehabilitation games are designed to be adopted in interaction control strategies. Results are also discussed in detail throughout the thesis.

2. PATH EXTRACTION

Kinematic synthesis procedure requires a task that has to be fulfilled by the resultant mechanism. This task is generally a path or motion to be followed. According to the outcome of the structural synthesis if the mobility of the resultant mechanism is sufficient to achieve the path, then the path can be followed exactly without error. On the other hand, if it is not possible to follow the path exactly due to the nature of the mechanism as the limited degrees of freedom with respect to the task, then approximate solutions are required.

The main aim of this Chapter is to create a methodology for collecting 3D path data for a specified task that mimics a part of the human body with the motion capture system and adapt it to a planar 2D path. The prerequisite for this task is the fact that 3D path must be suitable to be adapted to a planar path, otherwise this procedure may result in huge errors. If the adaptability exists, it will be possible to synthesize a planar mechanism with lower degree of freedom in order to mimic 3D motion by utilizing small errors. Thus, for such specific tasks, body part motions can be reduced to a planar motion, although they seem to be completely spatial. For instance, if a person comb her/his hair, the overall motion of the end-point, hand, can be described in a specific plane where the most of the 3D data is placed on this plane [55]. Also, since the described planar data is actually formed by a point data cloud, a planar regression curve should be extracted in order to synthesize a planar mechanism that tries to follow this regression curve that should be dependent on a single variable (i.e. $y = f(x)$) for a single degree of freedom system.

Path extraction procedure strongly depends on the way to collect the data of the motion. As mentioned in Sub-section 1.2, the working space of the motion capture system has 3D Cartesian coordinates that is created by the help of calibration square which makes it possible to define coordinates in the workspace for user needs. Since coordinates of the workspace can be predefined by using calibration square, motion data can collection be forced to be on one of the Cartesian planes by arranging setup

on the workspace. In such case, path extraction is simple, since the motion is already composed of a planar data and the data that is perpendicular the predefined plane can be ignored due to small deviation. For instance, the setup can be arranged for the motion to occur on xy plane, while small amount of motion occurring on predefined z axis can be ignored. Sub-section 2.1 is dedicated to show an application for this kind of processing.

However, there are some cases where it is not possible to arrange the setup to be strictly on the plane. In such cases, 3D motion data should be processed in order to get a planar motion. Since there is no predefined plane to represent the data, the best fit plane that represents 3D data should be determined mathematically. The motion data on this plane is used to get the regression curve to be treated in synthesizing the mechanism [55]. Sub-sections 2.2 and 2.3 are dedicated to show different applications for this kind of processing.

In the initial sub-sections of this Chapter, path data for finger rehabilitation is extracted via different procedures and methods are compared due to the suitability and simplicity of the concerned human limb for data collecting. In the latter sub-sections of this Chapter, path extraction applications are made for upper and lower extremity motions.

Gezgin et al. [26] worked on a hand rehabilitation system, which basically relies on the real data collected from the motion capture system. In their work, reflector markers are placed on a finger and grasping motion is performed while recording.
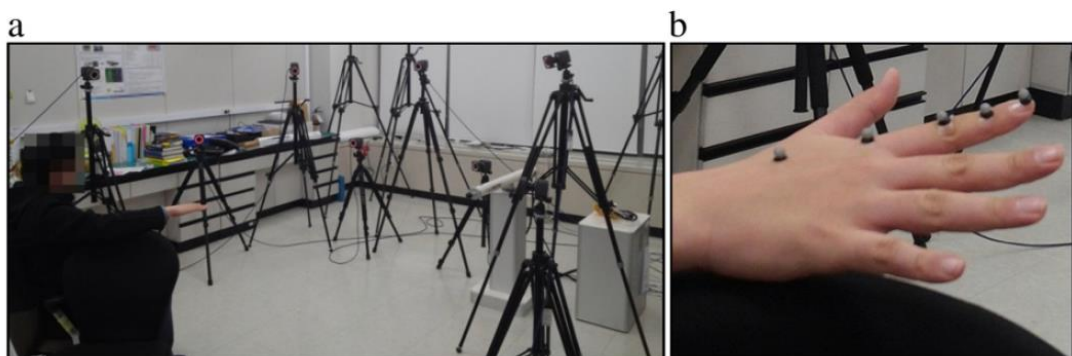
Figure 2.1 Motion capture from the index finger during grasping [26].

The motion is carried out on one of the Cartesian planes, where through other directions the motion is relatively small. As this small off plane data can be ignored, data cloud is created on a plane by only considering 2D motion data as mentioned in Sub-section 2.1. In other words, all data in 3D are projected on one of the predefined Cartesian plane.

In this case, it is clearly seen that motion of the finger is simply strict and as its nature, it is relatively easy to be maintained on a pure plane without pointing any other direction that crosses the workplane. Thus it is not needed to consider other planes.

In order to comprehend the total deviation during the grasping motion, two data collecting procedures are performed with the OptiTrack 3D motion capture system. The first data collecting procedure is completed by trying to execute the motion as straight as much as possible on xy plane without pointing in z direction (Figure 2.2).
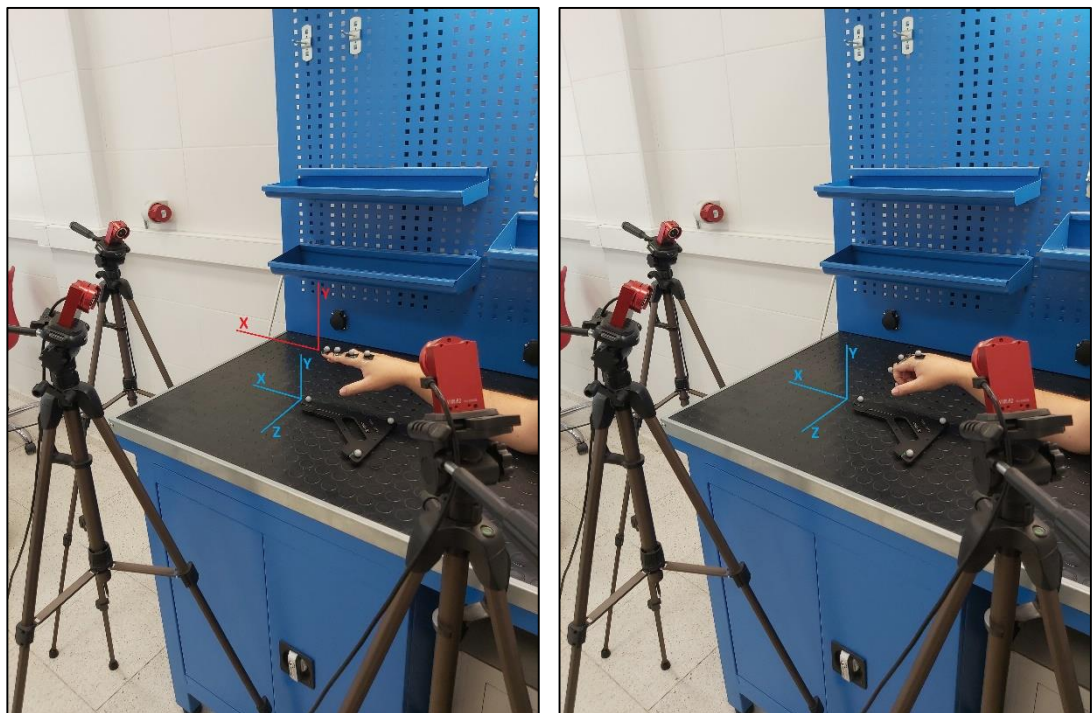
Figure 2.2 Data collection from the finger during grasping motion on XY plane.

Collected four different marker data can be seen in Figure 2.3. Although having effort on the motion to stay on a plane, the data is deviated in z direction by a

maximum of 2 centimeters. All data processing steps are done by using MATLAB program.
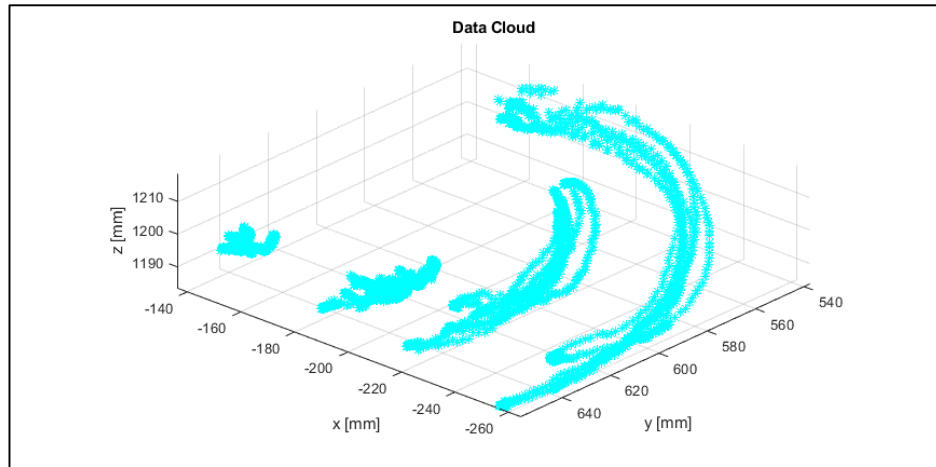
Figure 2.3 3D data cloud (constrained grasping on plane).

The second data collecting procedure is completed without considering any direction and only the grasping movement is focused (Figure 2.4).
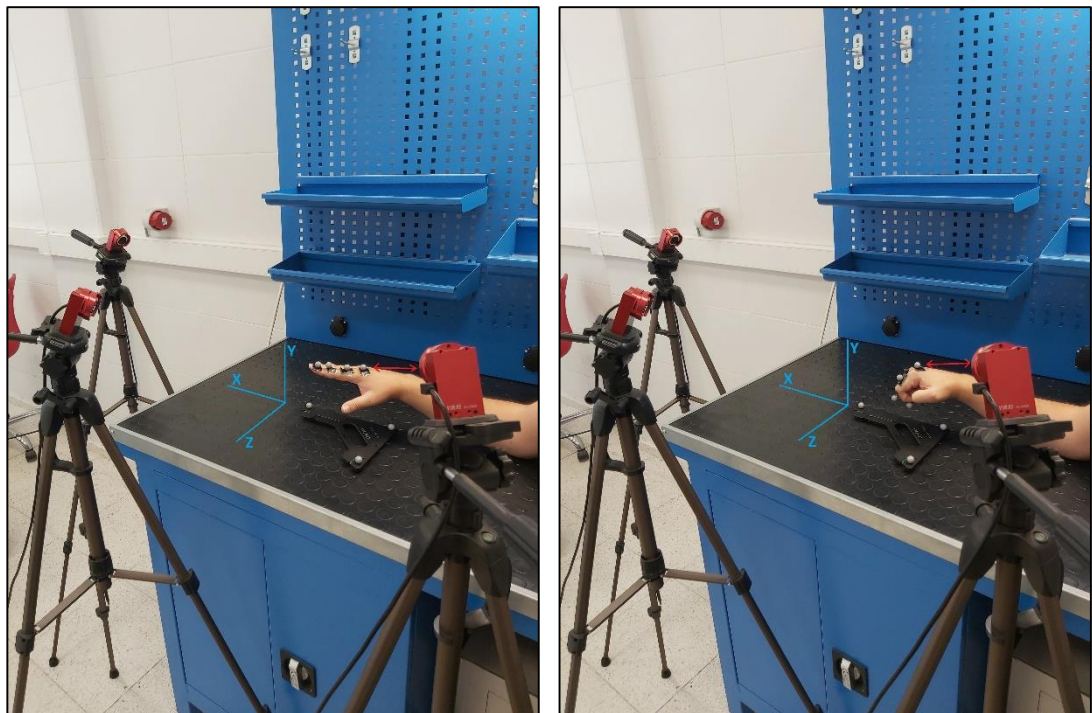
Figure 2.4 Data collection from the finger during grasping motion on a random plane.

As seen in Figure 2.5, four distinct marker data are not constrained to any Cartesian plane. Instead, they are randomly distributed. Here, there is no option as ignoring one of the planes, instead, data must be processed in terms of determining the best fit plane to represent the motion.
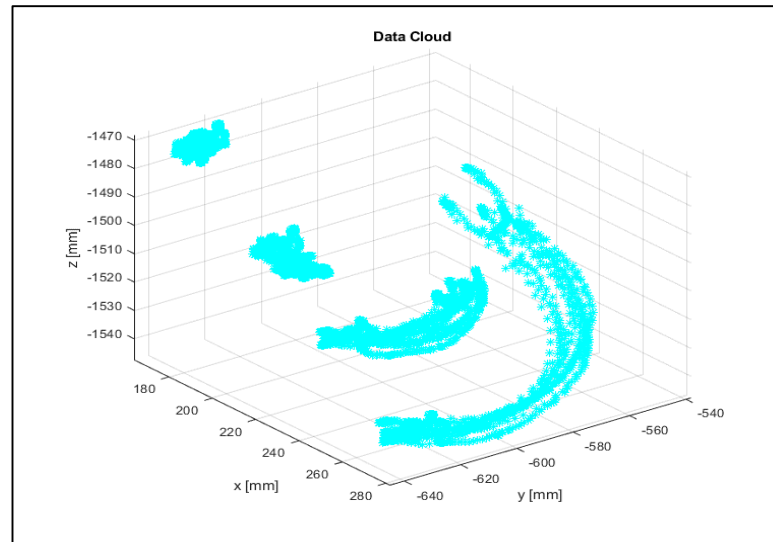
Figure 2.5 3D data cloud (grasping occurs on a random plane).

The procedure steps in Sub-section 2.2 should be applied to the system to determine the regression curve on the best fit plane.

2.1 Constrained Finger Motion Data on Predefined Plane

During performing the motion with reflector markers in the workspace of the motion capture system, the motion is recorded in 3D. There are some cases, especially in this work, where the data should be planar, thus one of the dimensions must be ignored. By arranging the motion capture system setup, the motion data can be intentionally forced to occur in one of the predefined Cartesian planes in a workspace of the motion capture system. Deviation between predefined Cartesian plane and its perpendicular direction may cause some errors. However, depending on the type of the motion, it can be ignored due to the fact that its effect on the overall result is relatively small. Maximum deviation of the trial experimented on this thesis is 2 centimeters and this leads to some difference in the resultant regression curves, where the maximum difference between the processed and not processed regression curves is 7.15 millimeters as seen in Figure 2.14.

This procedure is only valid when the deviation between the predefined plane and perpendicular direction is relatively small during capturing the motion. In such cases, it is easy to process the data for extracting the motion to be used in synthesizing mechanisms based on 2D planar regression curves. When the deviation becomes larger, the regression curve created on the predefined plane becomes unrealistic, thus it leads an error in designed mechanisms.

2.1.1 Utilization of constrained finger motion data

The collected data in Figure 2.3 is suitable to be used as a direct 2D data, where xy Cartesian plane is the predefined plane due to the fact that small deviation in perpendicular axis to this plane is ignored. In such case, it is relatively easy to extract a regression curve on the plane.

The third order polynomial regression curve of the end point data can be formed as;

$$y = -4.42 \cdot 10^{-5}x^3 + 0.01336x^2 - 0.4184x - 121 \qquad (2.17)$$

where x is in between -50.35mm and 90.99mm.

End-point regression curve and non-processed planar data cloud can be seen together in Figure 2.6.
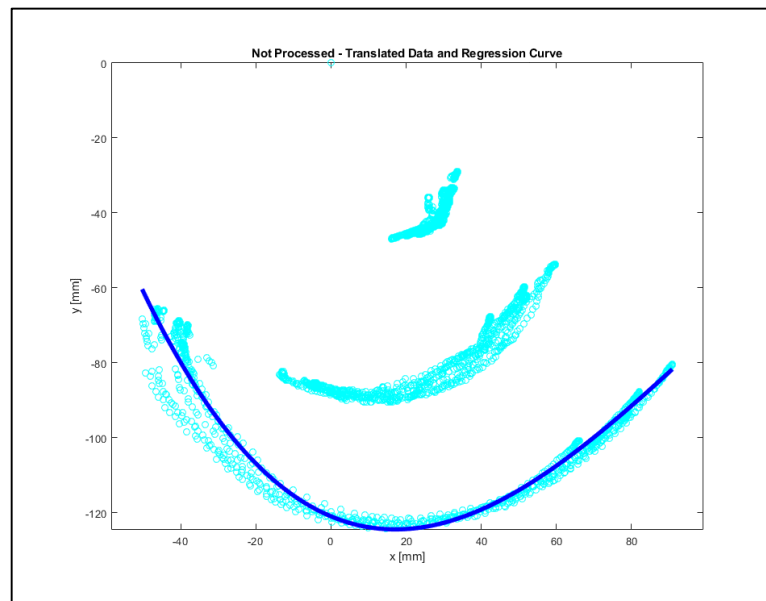
Figure 2.6 Final regression curve of non-processed data.

2.2 Finger Motion Data Extracted from Spatial Workspace

There are some cases where it is almost impossible to collect data efficiently on a predefined plane especially for people who has some disability to do so. In such cases, it is better to collect data on any direction, but repeatedly, without considering the planes, then determine the best fit plane that represents the motion on it with pseudo inverse algorithm. The 3D data should be projected onto a plane by finding the equation of this best fit plane for all points in the workspace.

General plane equation in a Cartesian workspace can be represented as;

$$ax + by + c = z \qquad (2.1)$$

where a, b and c are the plane equation coefficients, x, y and z are the Cartesian axis variables.

Considering the fact that all of the captured points should satisfy this equation in order to be on this plane, following equation set can be written in matrix form as,

$$\underbrace{\begin{bmatrix} x_1 & y_1 & 1 \\ x_2 & y_2 & 1 \\ \vdots & \vdots & \vdots \\ x_n & y_n & 1 \end{bmatrix}}_{A} \underbrace{\begin{bmatrix} a \\ b \\ c \end{bmatrix}}_{x} = \underbrace{\begin{bmatrix} z_1 \\ z_2 \\ \vdots \\ z_n \end{bmatrix}}_{B} \qquad (2.2)$$

where sub number n is the number of data points during performing the motion. For simplicity, equation 2.2 can be described as equation 2.3.

$$Ax = B \qquad (2.3)$$

where A and B are the matrix that contain the data points and x is the array that contains plane coefficients as in the equation 2.2.

Since the system becomes overdefined, in order to determine the plane coefficients a, b and c, left pseudo inverse of the matrix **A** should be used as in the equation 2.4.

$$\begin{bmatrix} a \\ b \\ c \end{bmatrix} = (A^T A)^{-1} A^T B \qquad (2.4)$$

With the determined a, b and c coefficients, the best plane that represents the motion can be found. Afterwards, the data cloud on the workspace is ready to be projected onto the best fit plane so that a planar data cloud can be formed.
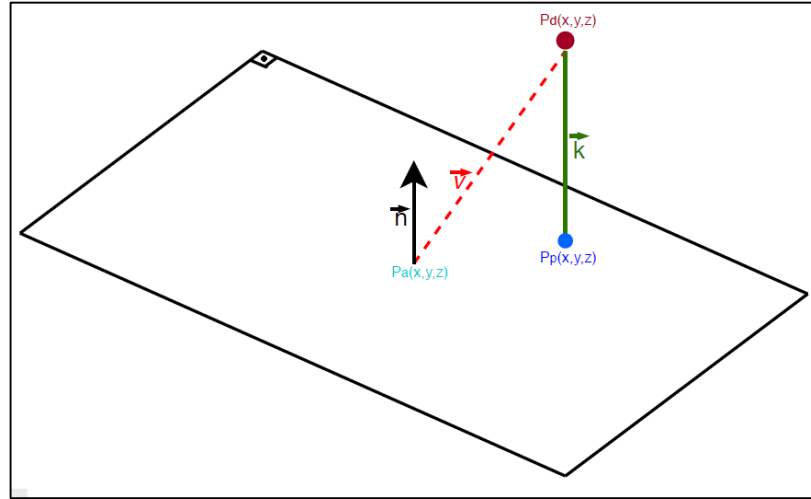
Figure 2.7 Projection of 3D point data on a 2D plane.

In Figure 2.7, P_d represents each data points during motion data capture, which should be projected onto the plane and each should be transformed to the P_p, projected points. Here, $\vec{n}$ vector is the unit normal vector of the plane originated through an arbitrary point, P_a, and the $\vec{v}$ vector is the vector between the arbitrary point on the plane and the concerned data point P_d.

Each of the projected points on the best fit plane can be determined by following vector equations.

$$\vec{v} = P_a(x, y, z) - P_d(x, y, z) \qquad (2.5)$$

$$\left| \vec{k} \right| = \vec{v} \cdot \vec{n} \qquad (2.6)$$

where $\left| \vec{k} \right|$ represents the scalar minimum distance between the points P_d and P_p. At this step, it is possible to find the most distant point from the best fit plane to give the idea about the deviation between the plane and the 3D data cloud.

$$P_p(x, y, z) = P_d(x, y, z) - \left| \vec{k} \right| \vec{n} \qquad (2.7)$$

With this procedure, projected points on the best fit plane for each data points on the cloud can be determined.

After projecting the data cloud onto the best fit plane, as the next step, angle between the best fit plane and the plane to be worked should be determined so that the rotation operation can be done.

The angle between the unit normal vector of the best fit plane and the normal vector of the working plane can be determined by the equation below;

$$\theta = \cos^{-1}\left(\frac{\vec{n}\cdot\vec{h}}{|\vec{n}||\vec{h}|}\right) \tag{2.8}$$

where $\vec{n}$ is the normal vector of the best fit plane and $\vec{h}$ is the normal vector of the predefined working plane.

As the final step before finding the regression curve on the plane, the projected data should be rotated in order to be parallel to predefined working plane. Thus, the axis of rotation has to be orthogonal to $\vec{n}$ and $\vec{h}$, so the axis of the rotation can be introduced as,

$$\vec{u} = \frac{(\vec{n}\ X \vec{h})}{|\vec{n}||\vec{h}|} = (u_1, u_2, u_3)^T \tag{2.9}$$

According to this axis of rotation, rotation matrix can be determined as;

$$R = \begin{bmatrix} \cos\theta + u_1^2(1-\cos\theta) & u_1u_2(1-\cos\theta) - u_3\sin\theta & u_1u_3(1-\cos\theta) + u_2\sin\theta \\ u_1u_2(1-\cos\theta) + u_3\sin\theta & \cos\theta + u_2^2(1-\cos\theta) & u_2u_3(1-\cos\theta) - u_1\sin\theta \\ u_1u_3(1-\cos\theta) - u_2\sin\theta & u_2u_3(1-\cos\theta) + u_1\sin\theta & \cos\theta + u_3^2(1-\cos\theta) \end{bmatrix} \tag{2.10}$$

If the rotation is applied to all of the data points in the cloud via determined matrix, all points becomes parallel to the predefined working plane.

At this point, the data is represented as planar data cloud, and it can be defined as 2D motion path by utilizing a best fit regression curve that depend on a single variable. This regression curve is the main goal of this process, since kinematic synthesis actually relies on the planar motion path.

In order to compensate the deviation between the planes and get a path that is suitable for kinematic synthesis procedures (i.e. $y = f(x)$), the following procedural steps should be applied.

• Determine the best fit plane as if all data is performed on it.

• Project all the data to a best fit plane.

• Find the angle between the best fit plane and Cartesian xy plane (predefined workplane).

• Rotate all the projected data so that they will be parallel to Cartesian xy plane (predefined workplane).

• Rotate all the data around z axis (normal axis of the predefined working plane) in order to have a function as $y = f(x)$.

• Translate all the data at each step that the first marker is placed on the origin point.

Two distinct finger data are collected with a small and huge deviation with xy plane of the Cartesian coordinate system of the motion capture system (Figure 2.3 and Figure 2.5). The effect of the path extraction procedure can be compared by processing the both data. Mentioned steps above are applied in order to have two distinct regression curves in following sub-sections.

2.2.1 Utilization of finger motion data extracted from constrained motion via proposed procedure

Collected finger data as seen in Figure 2.3 represents the motion with small deviation in z axis. Although the deviation can be ignored as in the previous section, the path extraction procedure is still applied to get the regression curve in order to compare all of the results. The procedural steps are applied to this data in the next sub-sections.

2.2.1.1 Determining the best fit plane

By considering the general plane equation in a Cartesian workspace as represented in equation 2.1, all the data is processed with the pseudo inverse algorithm (equations 2.3 and 2.4). Best fit plane equation is determined as below and visualized in Figure 2.8.

$$0.0485x - 0.2681y + 1.3696 = z \tag{2.11}$$
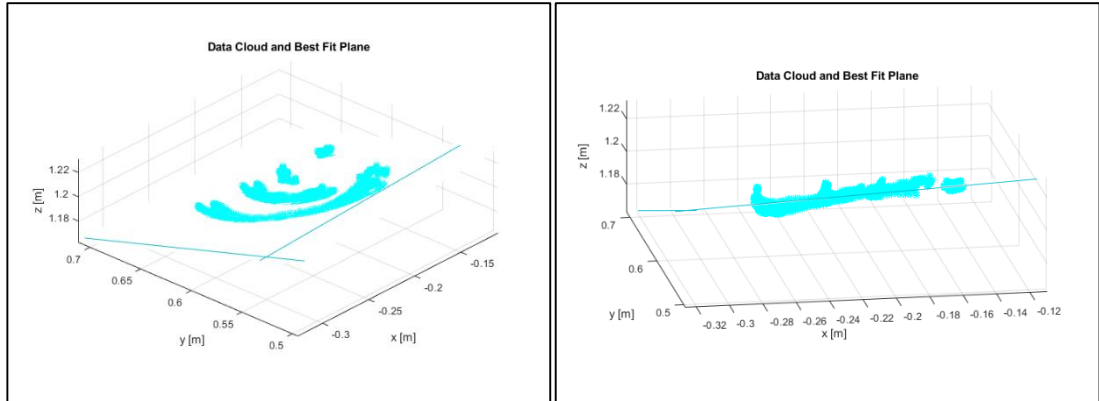
Figure 2.8 Best fit plane from different angle of views.

2.2.1.2 Projecting all of the data onto the best fit plane

Utilizing the equations 2.5 to 2.7, each data point on the cloud is projected onto the best fit plane (Figure 2.9). As mentioned before, it is possible to find the most distant point to the best fit plane. In this data cloud, the most distant point is 8.74 millimeters, which is quite sufficient that the motion is in a planar manner.
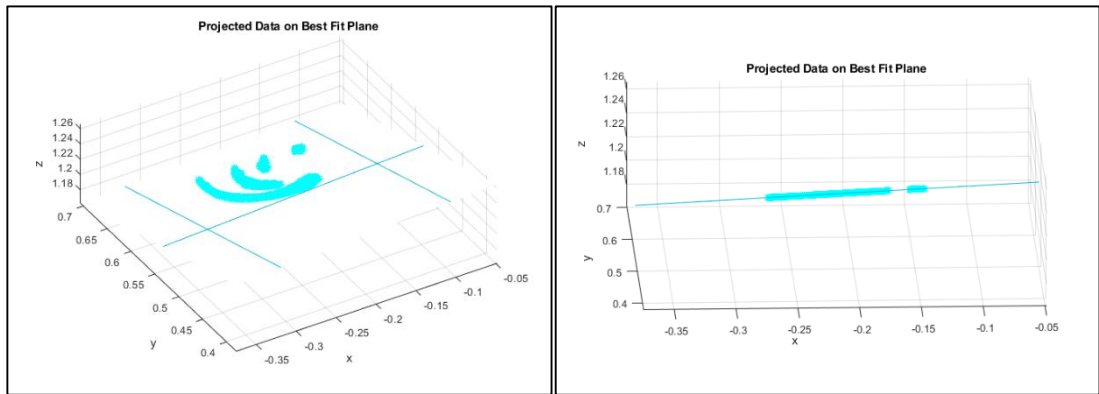
Figure 2.9 Projected points on the best fit plane with different angle of views.

2.2.1.3 Finding the angle between the best fit plane and xy plane

In this work the predefined workplane is chosen as the xy plane, since the motion is tried to be performed on the xy plane without pointing the z axis. Because of this reason, the $\vec{h}$ vector is chosen as;

$$\vec{h} = (0,0,1)^T \tag{2.12}$$

As the best fit plane equation is found, the unit normal of the plane can be revealed as;

$$\vec{n} = (0.0468, -0.2587, -0.9648)^T \tag{2.13}$$

The angle between the best fit plane and the xy plane can be found by using equation 2.8 as $15.24°$.

2.2.1.4 Spatial rotation of the projected data

With the calculations above, the rotation axis and rotation matrix can be found using the equations 2.9 and 2.10;

$$\vec{u} = (-0.2587, -0.0468, 0)^T \tag{2.14}$$

$$R = \begin{bmatrix} 0.9377 & 0.3444 & -0.0468 \\ 0.3444 & -0.9025 & 0.2587 \\ 0.0468 & -0.2587 & -0.9648 \end{bmatrix} \tag{2.15}$$

If the determined rotation matrix is applied to all of the data points in the cloud, all points becomes parallel to xy plane (Figure 2.10).
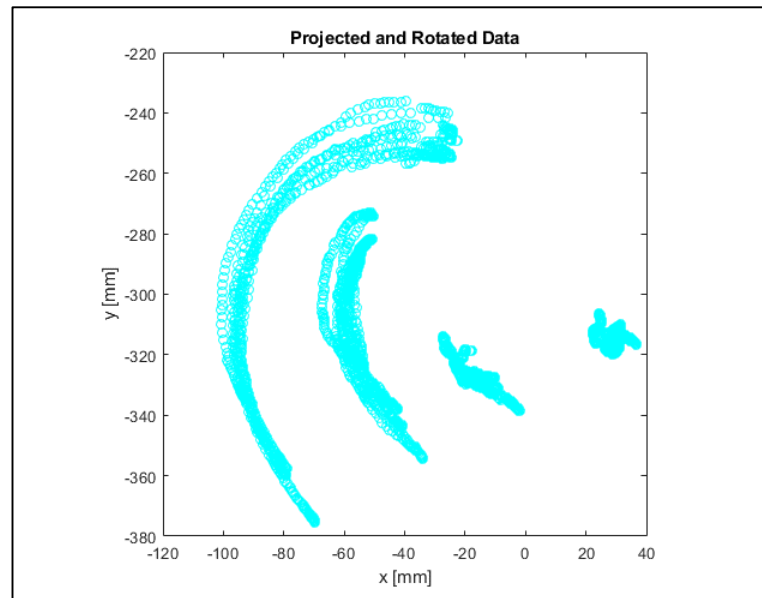
Figure 2.10 Projected and rotated data on the best fit plane.

2.2.1.5 Rotation of the data around z-axis

As seen from the Figure 2.10, it is not possible to create a regression curve with the processed data due to the fact that there is no single independent variable. In order to have a single independent variable, these data should be rotated around z axis. The rotation axis of z should be applied to the rotation matrix in equation 2.10.

2.2.1.6 Translation of the data to a reference origin point

As seen in Figure 2.11, due to the movement at hand during data collection, the data has no fixed point.
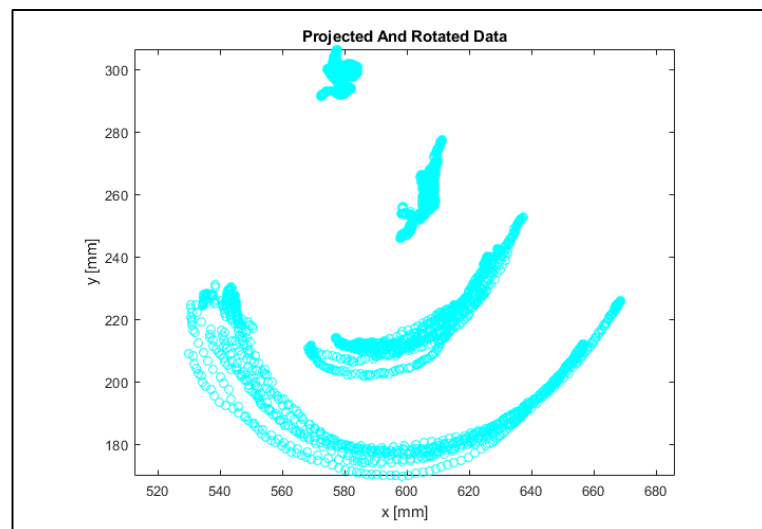
Figure 2.11 Projected data that are rotated around z-axis.

The grasping motion on the finger should be relative to the marker on the hand. For this reason, each data should be translated relative to the hand (Figure 2.12). This fixed point should also be fixed in the planar coordinates in order to be compared with all tries.
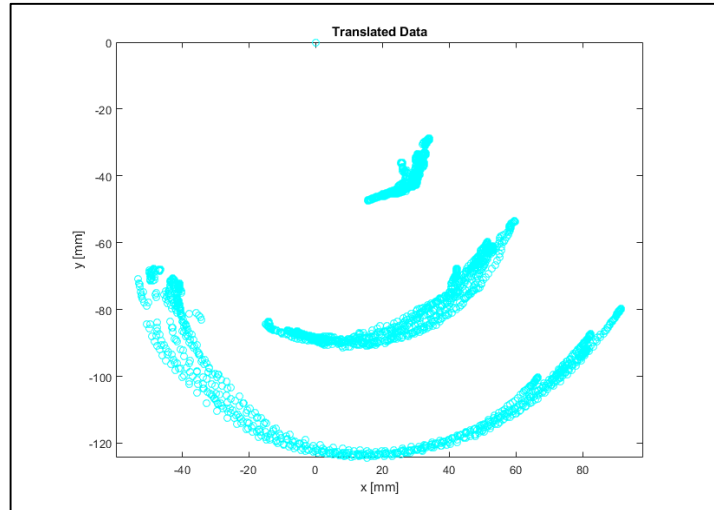
Figure 2.12 Translated data that is referenced to origin.

The end-point of the data cloud is considered for synthesizing procedure, so the regression curve is determined according to that data cloud, which becomes a third order polynomial as extracted from the MATLAB curve fitting tool by means of the millimeters.

$$y = -4 \cdot 10^{-5}x^3 + 0.01245x^2 - 0.357x - 122.3 \tag{2.16}$$

where x is in between -53.3mm and 91.58mm.

The end-point regression curve and the processed planar data cloud can be seen together in Figure 2.13.
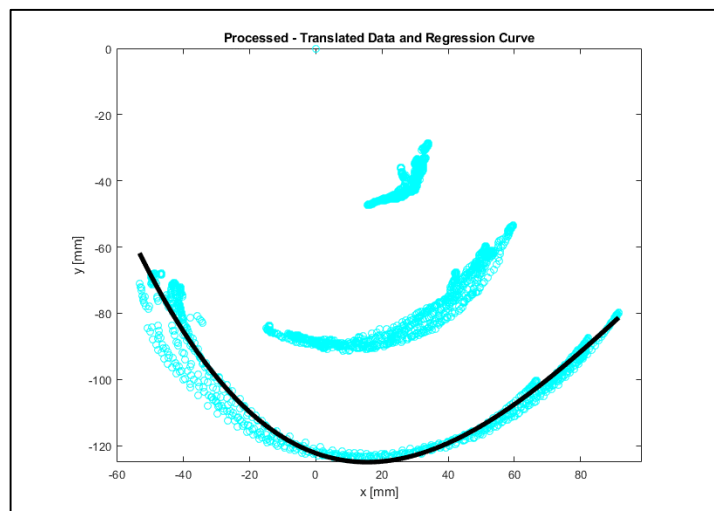
Figure 2.13 Final regression curve of processed data.

The same collected data as in the previous section is used without considering the motion in z direction because of the fact that the motion capturing is done on predefined xy plane. However, although having an effort to form a motion on a predefined plane, there still exists some deviation between planes. As the results are compared, this deviation actually does not affect the final regression curves effectively, while both of them stayed on the workspace of the motion data as seen in Figure 2.14.
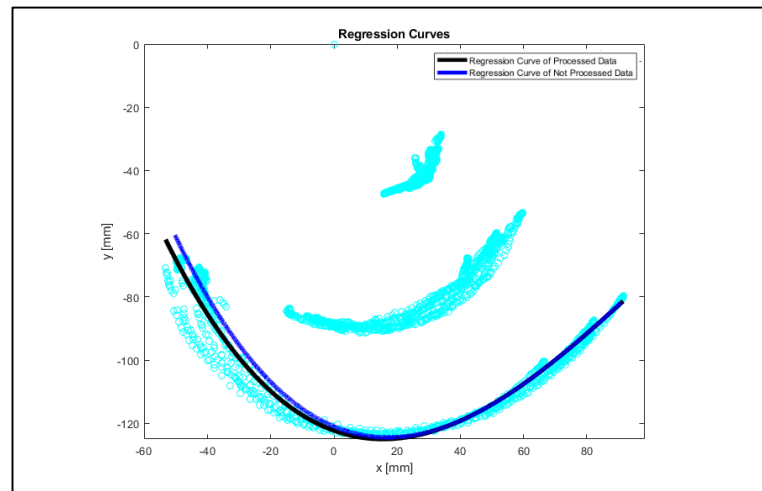
Figure 2.14 Comparison of regression curves with the same data collecting process.

2.2.2 Utilization of finger motion data extracted from spatial motion

As in Figure 2.5, despite the deviation, the data processing procedure is still applied in order to compare ignoring the deviation and not. In order to compensate that deviation, the same procedure steps are applied to the motion data on random plane as well.

2.2.2.1 Determining the best fit plane

The same general plane equation in a Cartesian workspace is processed with pseudo inverse algorithm as in the equations 2.3 and 2.4, the best fit plane (Figure 2.15) is determined as;

$$-0.5833x - 0.2125y - 1.5077 = z \tag{2.18}$$
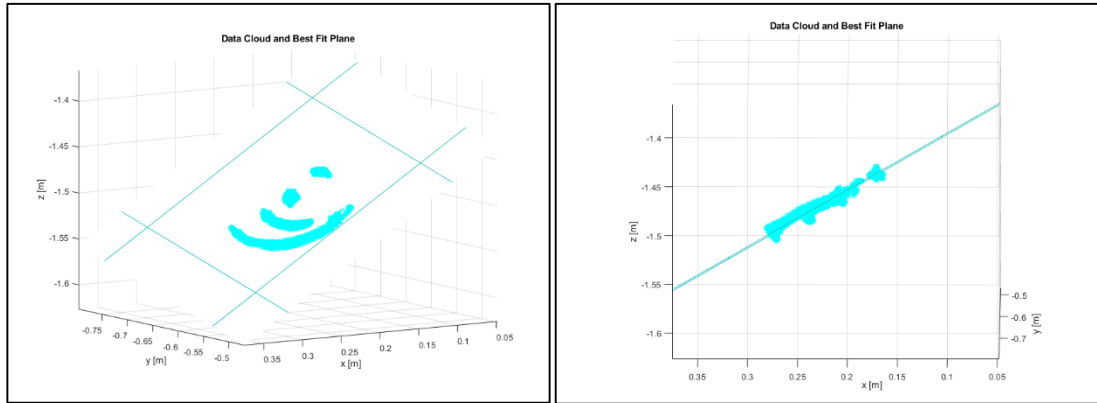
Figure 2.15 Best fit plane with different angle of views.

2.2.2.2 Projection of the data onto the best fit plane

After projecting all the data onto the best fit plane (Figure 2.16), the most distant point from the plane is determined as 9.84 millimeters, which is convenient to use the plane the motion workspace.
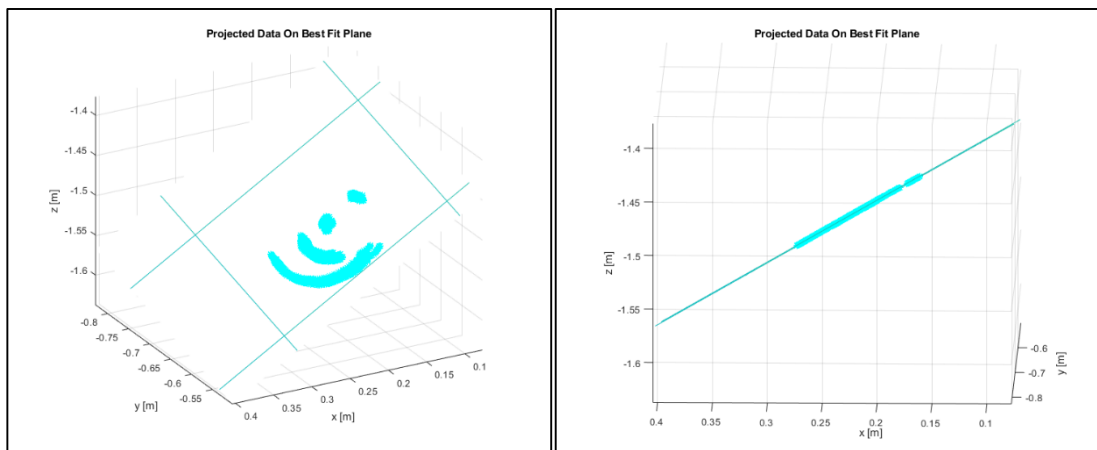
Figure 2.16 Projected points on the best fit plane with different angle of views.

2.2.2.3 Finding the angle between the best fit plane and xy plane

In this data processing part, the predefined working plane is chosen as xy plane again, which is suitable to be used. The $\vec{h}$ vector is chosen again the same as in the sub-section 2.2.1.3;

$$\vec{h} = (0,0,1)^T \tag{2.19}$$

Unit normal of the plane is determined as;

$$\vec{n} = (-0.5833, -0.2125, 1.5077)^T \qquad (2.20)$$

The angle between the best fit plane and the xy plane can be found with the equation 2.8 as 31.83°. As seen from the numeric value of the angle, it is quite inclined with the workplane.

2.2.2.4 Spatial rotation of the projected data

Rotation axis and the rotation matrix is found as;

$$\vec{u} = (-0.1806, 0.4955, 0)^T \qquad (2.21)$$

$$R = \begin{bmatrix} -0.6328 & -0.5950 & 0.4955 \\ -0.5950 & 0.7832 & 0.1806 \\ -0.4955 & -0.1806 & -0.8496 \end{bmatrix} \qquad (2.22)$$

When the rotation matrix is multiplied by all of the data points on the projected plane, the markers distribution can be seen in Figure 2.17.
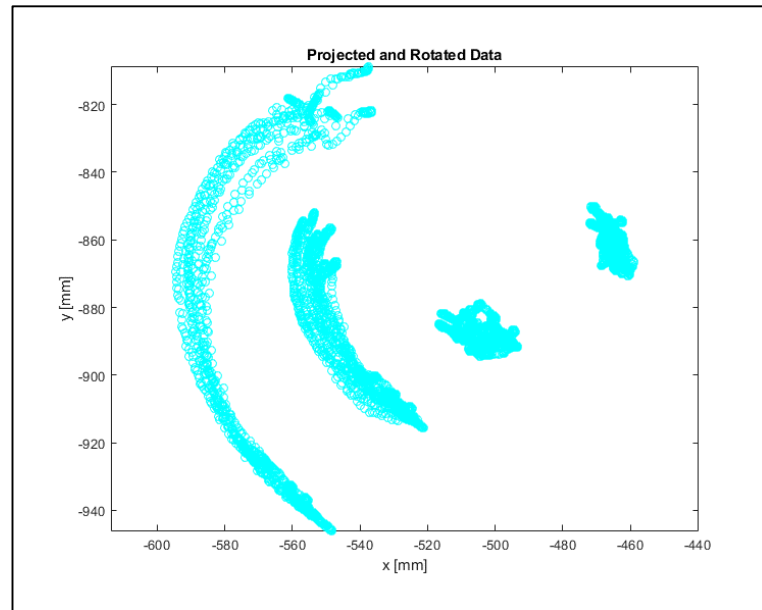
Figure 2.17 Projected and rotated data on the best fit plane.

2.2.2.5 Rotation of the data around z-axis

In consideration of creating a regression curve with single independent variable, the data is rotated around z axis.

2.2.2.6 Translation of the data to a reference origin point

The translation is applied to the origin point (Figure 2.18).
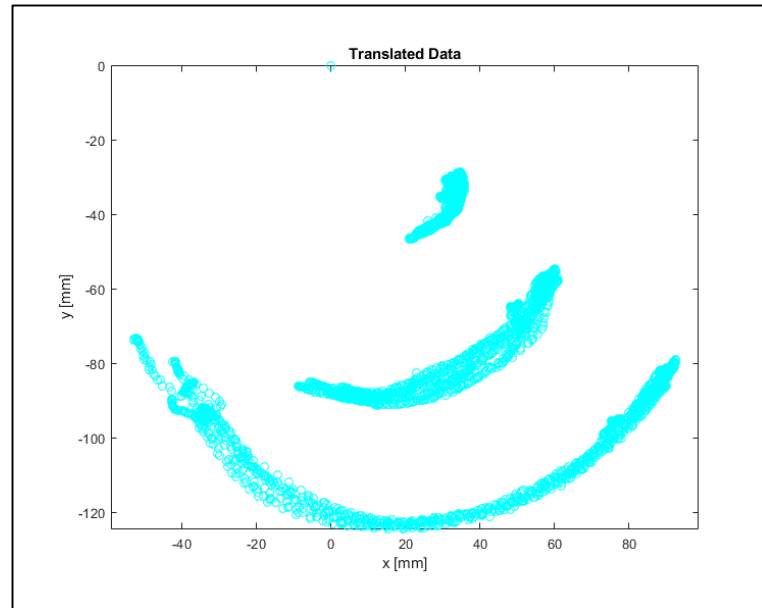
Figure 2.18 Translated data that is referenced to origin.

The regression curve of end-point becomes a third order polynomial as extracted from the MATLAB curve fitting tool by means of the millimeters.

$$y = -7.5 \cdot 10^{-6}x^3 + 0.009619x^2 - 0.3981x - 119.7 \qquad (2.23)$$

where x is in between -52mm and 93mm.

The end-point regression curve and the processed planar data cloud can be seen together in Figure 2.19.
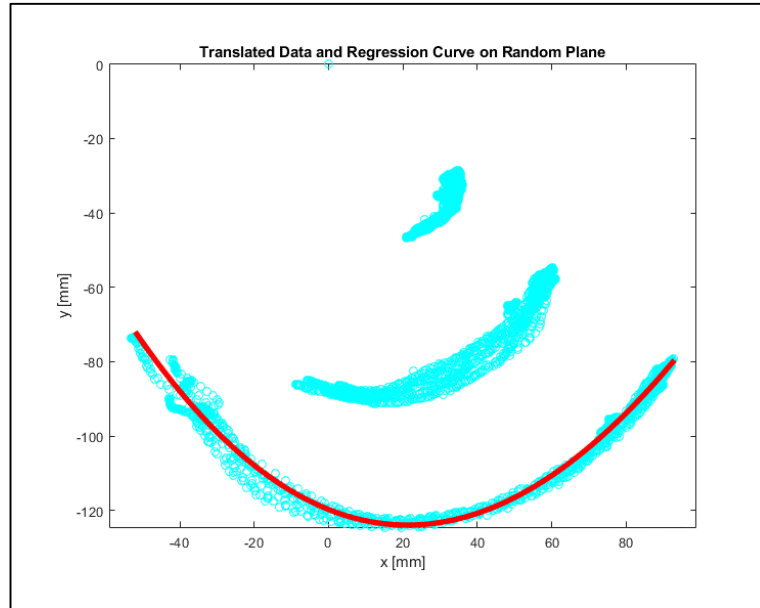
Figure 2.19 Final regression curve of processed data.

As composing, three different procedures to two different data collection processes is applied. In order to have a comparison, all the results can be seen in the Figure 2.20.
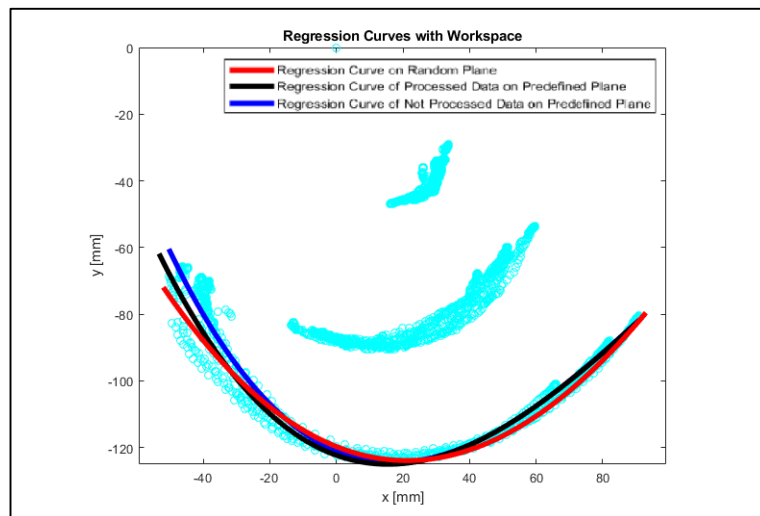
Figure 2.20 Comparison of regression curves with the two different data collecting process.

The error graphs between the not processed in predefined plane (blue curve in Figure 2.20) and processed in random plane (red curve in Figure 2.20) can be found in Figure 2.21.
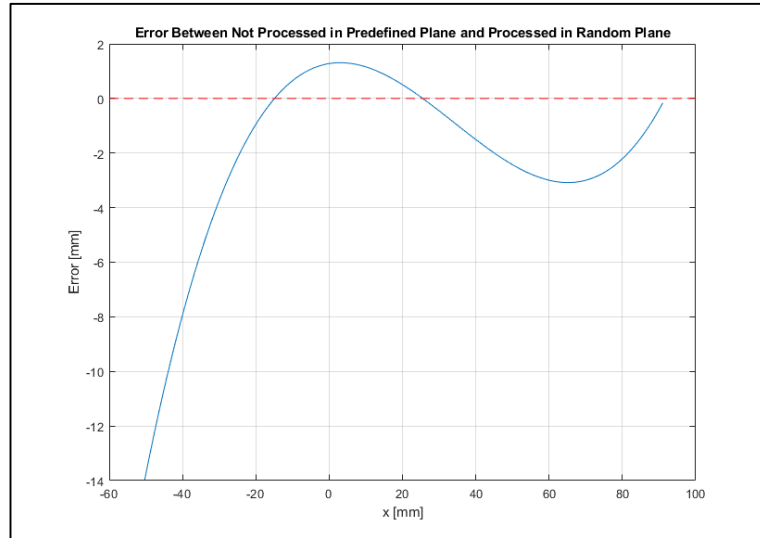
Figure 2.21 Error between non-processed in predefined plane and processed in random plane.

In order to have a planar finger following path, two distinct data is collected from the finger of a subject in grasping motion. One of the motion is restricted to be maintained on a predefined Cartesian plane of the motion capture system (Figure 2.3), and the other one is randomly collected without considering any Cartesian plane (Figure 2.5). The reason behind the idea of collecting two distinct data is to see the effect of the path extraction methodology as described in the sub-section 2.2. Also, it is obvious that hand grasping motion is relatively easy for subjects to maintain the motion on any predefined plane.

Initially, the first data is used directly without processing by ignoring the deviation on the perpendicular direction of the predefined plane. Then, the first data is used again, but this time it is processed with the proposed path extraction methodology. Finally, the same methodology is applied to the second data.

As a result, for this process, since it is easy to maintain the motion on the predefined plane, it is possible to vanish the huge data processing to a simple manner by ignoring the small deviation on the perpendicular direction of the predefined plane. However, the drawback is that the motion capturing environment should be prepared to be as strict as possible to keep the motion on a predefined plane.

On the other hand, if the motion or the subject is not convenient to be maintained on any predefined plane, with this path extraction methodology, it is shown that after data collecting procedure, it is possible to post process the data in order to extract a 2D regression curve depending on an independent variable. It is very useful to be used with paths in random environments, eventually, the data collection procedure is performed, and the path extraction methodology is applied as discussed in previous sub-sections.

2.3 Extraction of Hand Motion Data during Hair Combing

The efficiency of a neurological disorder treatment is strongly related with the type of the activity. For this reason, in classical or robotic rehabilitation, the activities are generally chosen from the daily life in order to increase the awareness of the treatment and bind the activity to human social life. Therefore, another path extraction methodology is applied to the chosen activity of daily life, combing the hair, where the shoulder and elbow joints are used during the movement. This upper extremity parts are quite complex, thus it is not very easy to get recovery after the disorder begins. As the complexity increases, the effort and the difficulty increases as well. The robotic rehabilitation devices that aim to recover this body areas are generally very huge, complex and expensive.

Although combing hair motion seems to be very complex due to the fact that it involves the upper extremity, actually the motion of the end-effector, which is the hand of the subject, is quite straight and can be considered and reduced to a planar motion with some amount of deviation. The main aim of this section is to use the 3D hand position data that are collected from a middle-aged healthy subject during combing the hair motion for applying the path extraction methodology in order to reveal a path that is suitable for kinematic synthesis procedure so that one degree of freedom mechanism can be synthesized.

As mentioned in previous sub-sections, there are some cases where it is almost impossible to collect data efficiently on a predefined Cartesian plane of the motion capture system. In such cases, it is better to collect data on any direction, and then find the best fit plane in order to reduce 3D path motion onto 2D plane. Hand motion

during combing the hair is one of this cases, where it is not possible to keep the motion in any Cartesian plane of the motion capture system. With the procedure steps on sub-section 2.2, it is not obligatory to be close enough to perpendicular any axis of the frame. In this work, combing hair motion is mimicked repeatedly, while the optical marker is placed on the subject's right hand during the motion capture with cameras as seen in Figure 2.22.

Figure 2.22 Optitrack 3D motion capture system with the subject [55].

Combing hair motion is performed repeatedly eight times in the workspace of the motion capture system. Since it is a 3D motion, the positions of the marker in the workspace is a volume and can be seen in Figure 2.23.
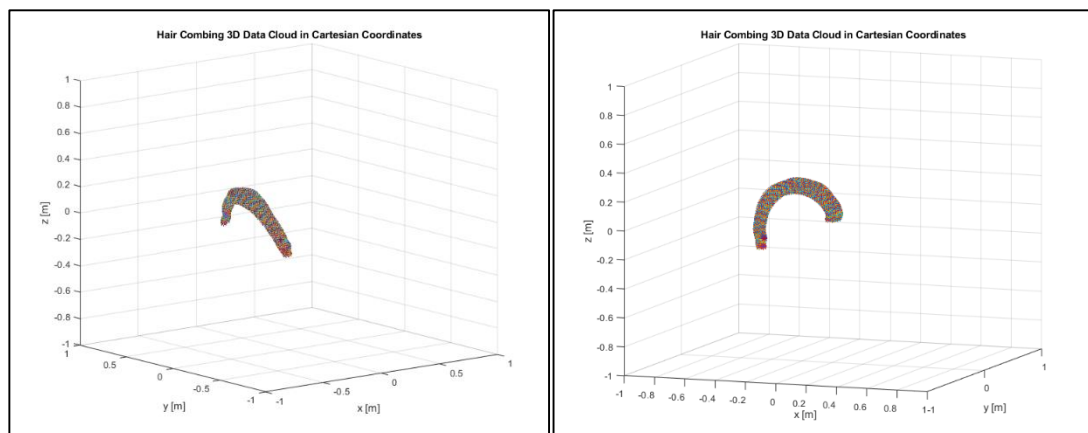
Figure 2.23 Hair combing 3D data cloud in Cartesian Coordinates.

43

As following the path extraction methodology procedures, the presented 3D data should be represented by the best fit plane, which can be seen in Figure 2.24.
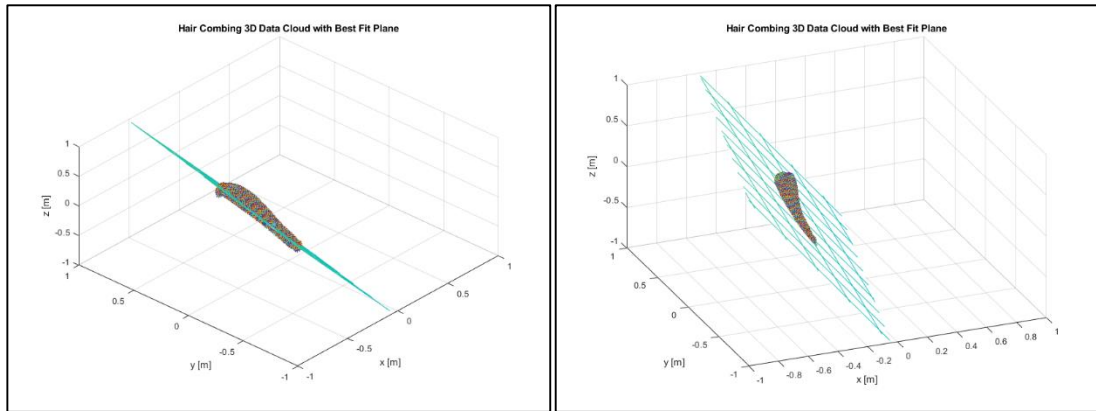
Figure 2.24 Hair combing 3D data cloud with best fit plane.

The 3D data should be projected onto the best fit plane in order to be represented on it. The projected data can be seen in Figure 2.25.
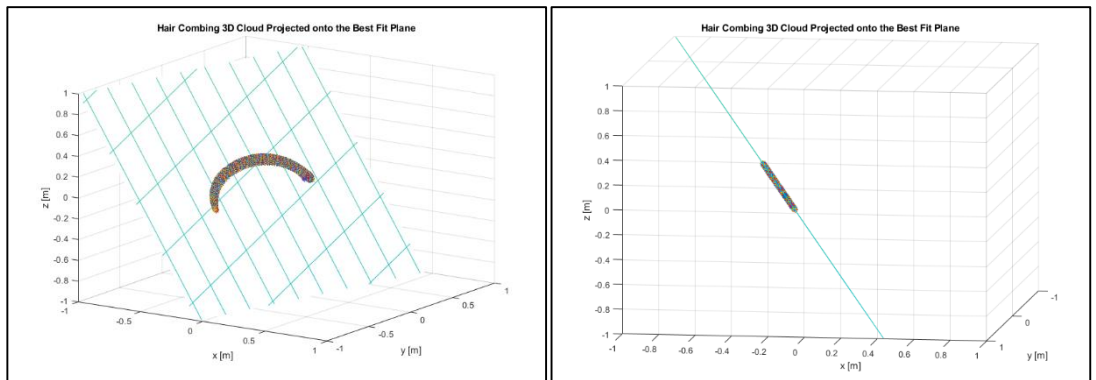
Figure 2.25 Hair combing 3D cloud projected onto the best fit plane.

Since the best fit plane is tilted, the projected data should be rotated in order to be parallel with the xy plane. The data cloud on the new plane is represented by a regression curve and can be seen together as in the Figure 2.26.
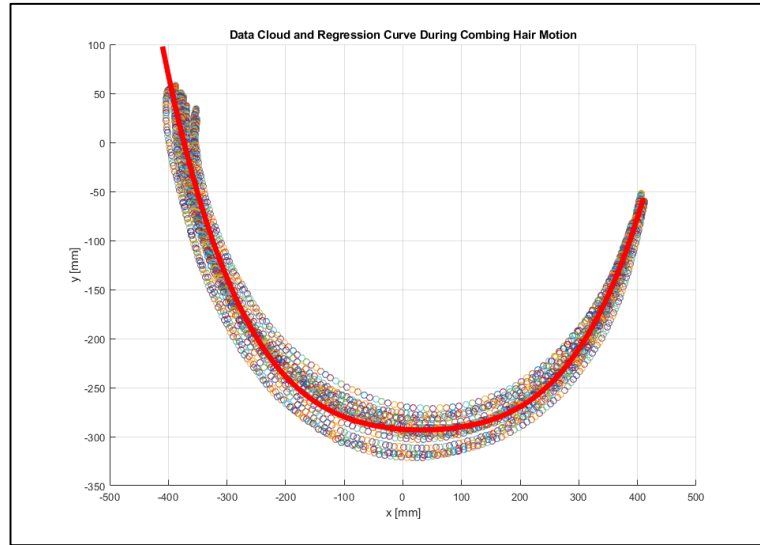
Figure 2.26 Data cloud and regression curve during combing hair motion.

With the results above, it can be concluded that for planar mechanisms to be utilized, during collecting motion data from motion capture system, the motion direction and plane can be chosen arbitrarily, especially where the cases are not possible to be integrated on a plane due to the movement type and the person's disability. In the hair combing case, although the motion is actually a complex due to the fact that it involves the shoulder and elbow together, the repeatedly motion is actually follows a planar path, which can be achieved by a one degree of freedom mechanism with a finite amount of error. This procedure is useful when the motion is possible to be reduced to a planar motion.

2.4 Extraction of Knee Orthosis Data

In daily living, walking has a great importance for people. The quality in walking strongly depends on the motion of the center rotational point of knee joint, which has actually a very complex motion with the help of the muscles and liquids between bones. The motion of this point can show different attitudes between individuals. In this section of this thesis, motion capture system is used to collect real walking motion data from a healthy person in order to extract the position path of the knee centroid, ankle as well as the orientation path of the lower leg during walking. In the next Chapter, this unique knee centroid locomotion data will be used in synthesizing

the four-bar mechanism kinematically with minimum error in order to follow the path and orientation data.

2.4.1 Extraction of walking data via with motion capture system

In this work, Optitrack motion capture system is used with 6 cameras and a working space that allows at least two normal steps for an adult male. In Figure 2.27, red cylinders on the walls corresponds to OptiTrack cameras additionally adults walking orientation with respect to cameras visualize.
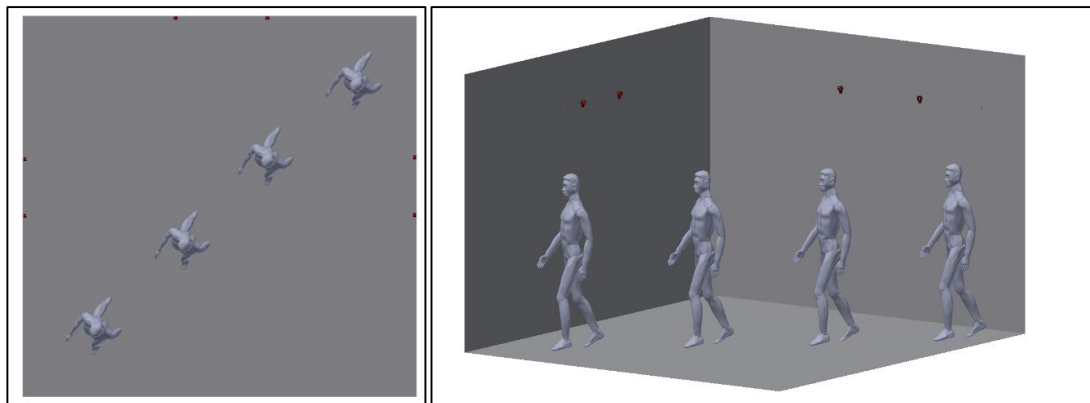
Figure 2.27 Motion capture system working space.

As mentioned in sub-section 1.3, the motion capture system has a unique licensed computer program, Motive, that is used to manage the system in terms of calibration, configuration and data collection. The motion is captured by using special reflector markers to be placed specific points on the subject.

2.4.1.1 Placing the markers

In order to extract a planar motion path of the lower extremity, the reflector markers are placed on the subject. Location of the markers plays a vital role for getting best motion data during walking and visualize the change in knee centroid. Since the locomotion of the subject occurs, a stable fixed point according to ankle should be determined. The hip point is chosen as the stable reference point, where eventually all the data is dependent on that point. The first single reflector marker is located on the hip joint. Since the upper and lower legs will be modeled as the vectors, at least two markers are needed along the vector directions for each rigid body. For this reason, these two markers are placed where both creates a vector that is

approximately parallel to legs orientation itself. Optitrack's special tools for the rigid body are used in creating the vectors as placed in Figure 2.28. Also, one more marker is added to the tool, in order to create a rigid body as plane. The reason behind creating the rigid body is that the motion plane of the upper and lower legs can be determined. Without this information, it is not possible to reduce the motion to sagittal plane of the subject. Another reason is that when one of the reflector markers on the rigid body tool is disappeared from the view of the cameras, the positions of the other two can be used to estimate the location of the concerned marker, thanks to their predefined structure. In such systems, using rigid bodies results more precise output of the systems rather than using single marker. The ankle point is also considered to be followed due to the fact that the end point of the polycentric four bar mechanism will be tracked as well. For this reason, one marker is placed on the ankle point.
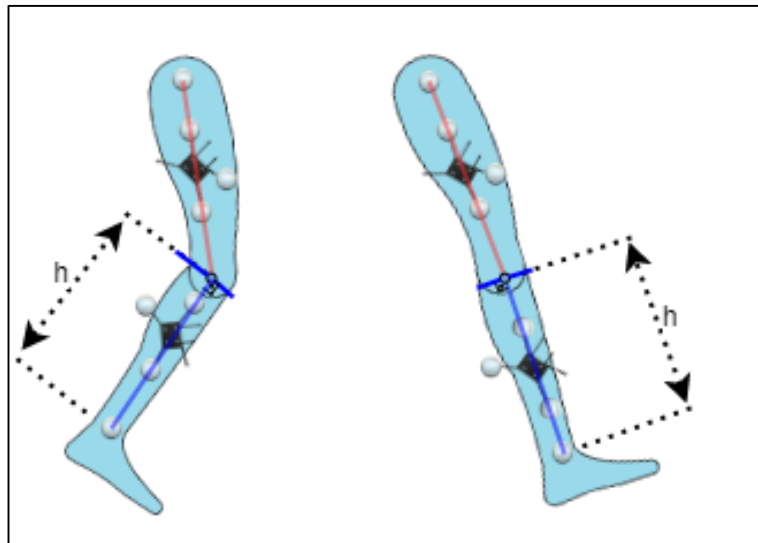
Figure 2.28 Placed markers during walking.

As seen from the Figure 2.28, with planar data of the upper and lower leg markers, the upper and lower leg vectors are created.

2.4.2 Extracting data to sagittal plane

On the local coordinate system of the Optitrack, path extraction methodology is performed to the 3D position data of the reflector markers on the upper and lower leg. Here, the best fit planes for each leg are determined, then they are rotated in

order to be parallel with the sagittal plane of the subject. The path data are projected onto the sagittal plane for creating 2D vectors for upper and lower leg of the subject during walking. Due to the translation of the subject as well as the markers, the marker data on the sagittal plane translates along with the walking direction. However, the synthesizing methodology needs to have a regression curve without translation by considering one fixed reference point on the plane. For this reason, the marker which is placed on hip joint is considered as the fixed origin point of the sagittal point, the other markers and vectors are translated according to this origin point. Since the knee joint is being synthesized, the upper part of the knee should be considered stable as well. Due to this fact, the upper leg is considered to be constant as perpendicular to the ground and all the markers and vectors are rotated according to the upper leg vector. As seen from Figure 2.29, the hip point is located stable on the origin, and the upper leg is considered as a single vector (colored as cyan) for multiple frames. The lower leg vector (colored as red) and the ankle joint (colored as blue) change their positions and orientations during the movement.
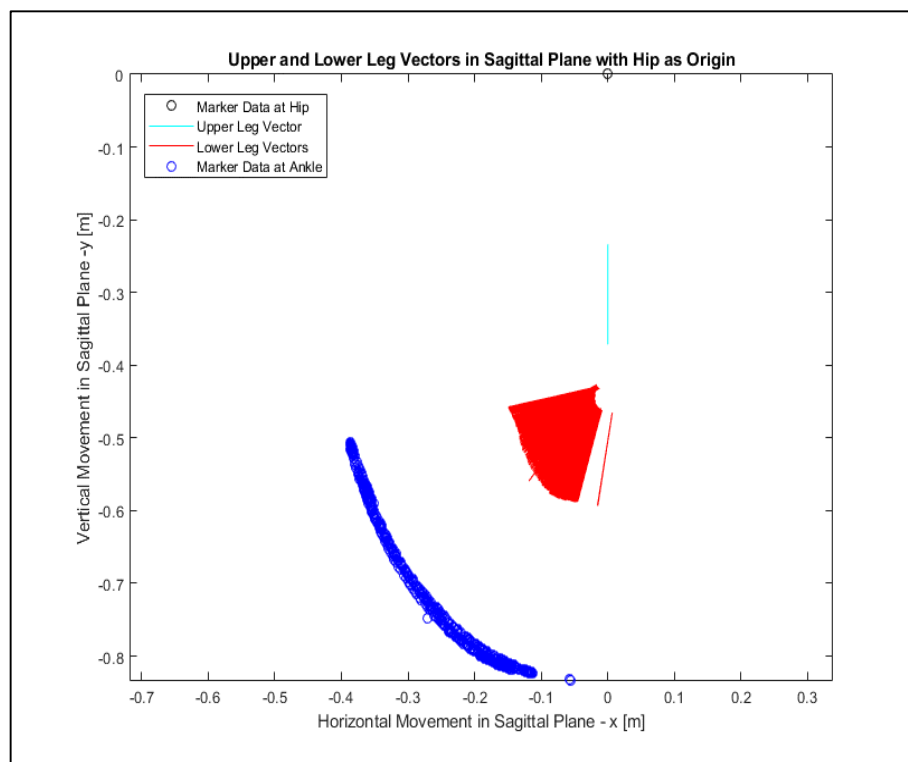
Figure 2.29 Rotated and translated vectors and points on sagittal plane during walking.

2.4.3 Locomotion of the knee centroid during walking

During walking, knee centroid actually makes a complex polycentric motion (Figure 2.30). This polycentric motion can be mimicked with a mechanism that rotates from a point which is altered continuously, like a four-bar mechanism. If the polycentric mechanism is synthesized by considering knee centroid locomotion, then the outcome becomes more natural. Thus it is vital to determine the location of the knee centroid over time by utilizing upper leg and hip location data. Acquiring the necessary data, it will be possible to deal with the syntheses methodology that is focused on the knee rotation centroid.
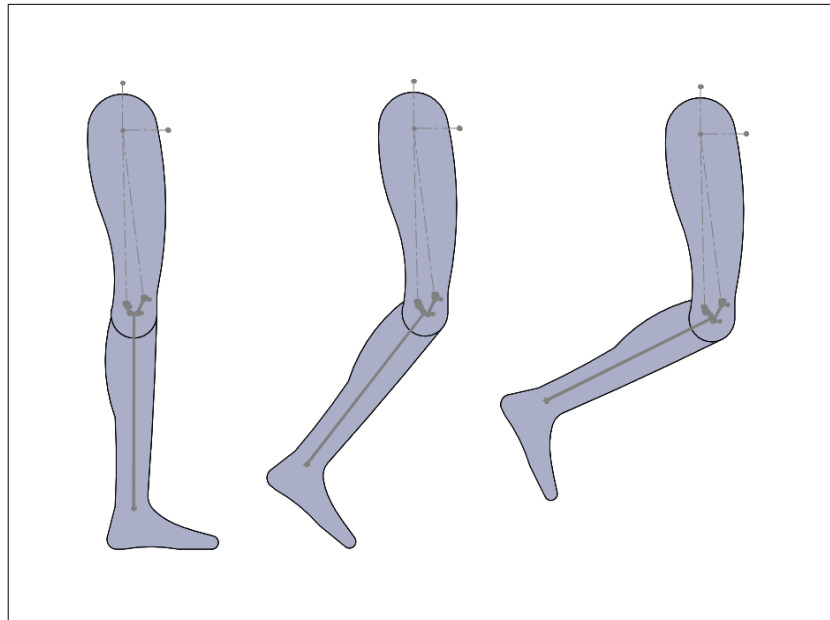
Figure 2.30 Knee flexion with virtual mechanism.

Collecting knee centroid position data during walking gait cycle is challenging due to the fact that the movements on outer skin and muscles are not the same as the movement of the skeletal rigid bodies. For this reason, more complex solutions are tried in literature such as MR following [56].

In this work, a novel method is used to estimate the location of the knee centroid point. Here, upper and lower legs are considered as vectors. The polycentric motion of the knee center is constantly changing due to the fact that the lower leg bone slips on knee joint when the knee flexion angle increases. It is possible to assume that the distance between knee centroid point and the ankle, which can be considered as the

height of the lower leg, stays constant (h in Figure 2.28). According to ankle trajectory, the knee centroid point also moves on sagittal plane, in other words, it does not stays on a single point. This is the reason that the flexion of the knee joint is polycentric. The knee centroid point in terms of horizontal and vertical motion in sagittal plane can be determined at every instant as below;

$$P_{x(knee\ centroid)} = P_{x(ankle)} + h\ cos\ (\psi - \pi) \tag{2.24}$$

$$P_{y(knee\ centroid)} = P_{y(ankle)} + h\ sin\ (\psi - \pi) \tag{2.25}$$

where $P_{x(ankle)}$ and $P_{y(ankle)}$ are the positions of the ankle in sagittal plane, h is the height of the lower leg, ψ is the orientation angle of the knee joint, which depends on the flexion angle of the knee.

2.4.4 Ankle trajectory and orientation angle

The height of the lower leg (h) can be determined when the subject remains standing, when the slip on knee does not occur. Locomotion of the knee centroid on the sagittal plane is problematic by means of extracting a regression curve with independent and dependent values as seen in Figure 2.31 (i.e. $y = f(x)$).
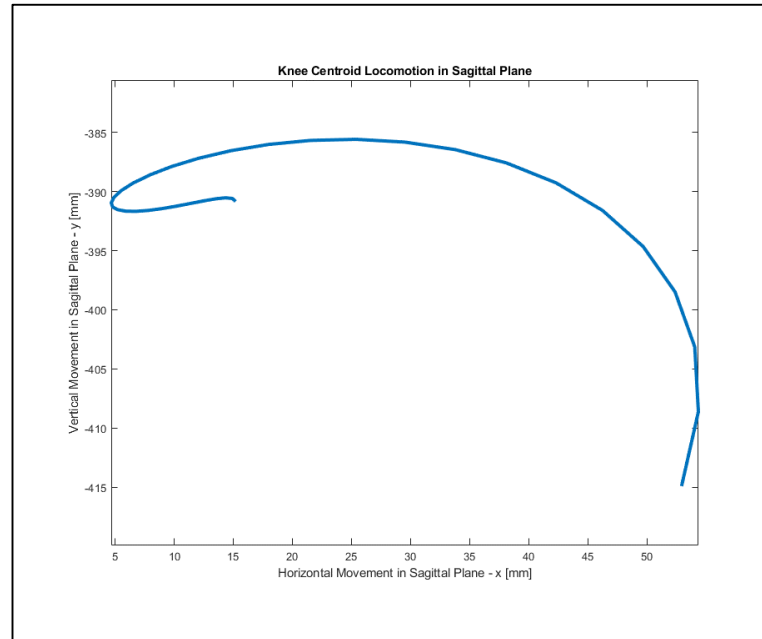
Figure 2.31 Knee centroid locomotion on sagittal plane.

Also, the relation between the centroid data and the orientation of the lower leg with respect to the ground is not easy to find if only this data is considered. For this reason, the regression curve is created by using the locomotion of ankle trajectory on sagittal plane, then knee rotation centroid and orientation angle is determined according to it (Figure 2.32).
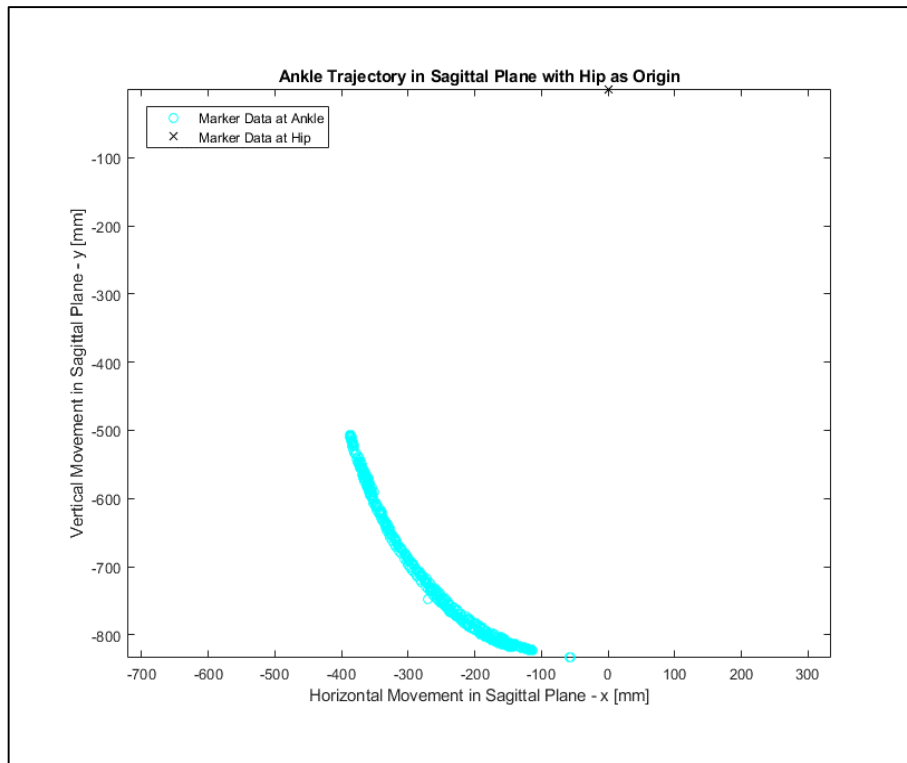
Figure 2.32 Ankle trajectory on sagittal plane with respect to hip joint.

The data is distributed as a curve which can be modeled as a function of independent variable of horizontal movement in sagittal plane and dependent variable of vertical movement in the same plane (Figure 2.33). The third order polynomial regression curve of varying positions is determined in terms of meters as;

$$y = -8.461x^3 - 1.86x^2 - 0.3244x - 0.849 \qquad (2.26)$$
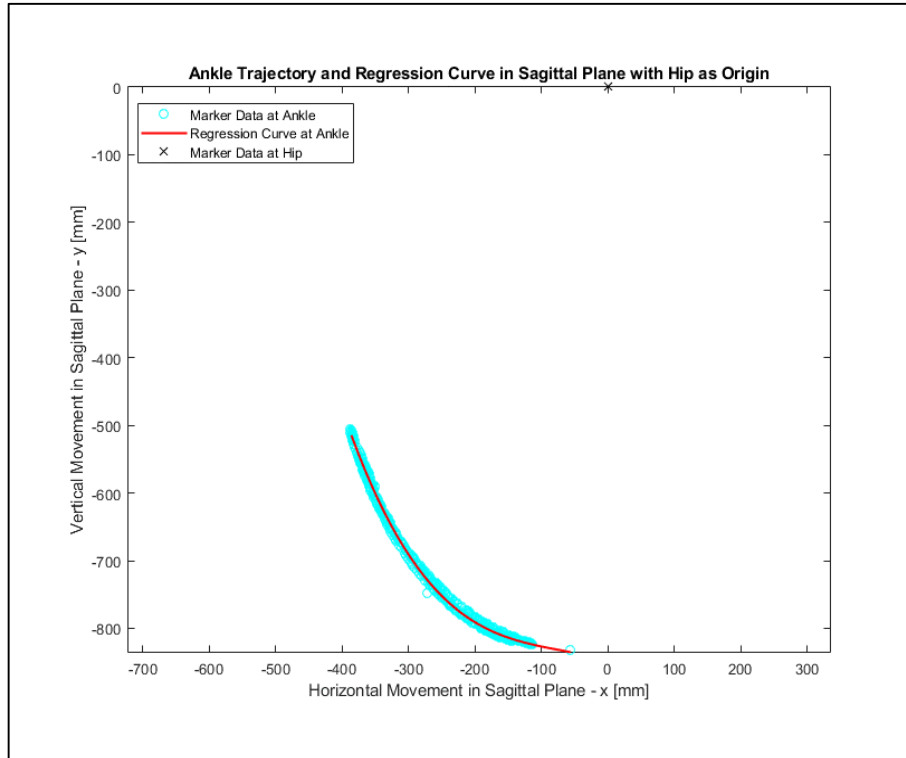
Figure 2.33 Regression curve of ankle trajectory on sagittal plane.

During walking, knee creates flexion and extension angle which is generalized in human gait cycles as in the Figure 2.30. The leg orthosis that will be used in rehabilitation procedures has to mimic the motion of the knee centroid in order to give a natural walking outcome, therefore the resultant polycentric mechanism should follow the path as well as the flexion-extension angle, which is renamed as the orientation angle of the mechanism. This orientation angle must also be considered in synthesis procedure, thus it is also extracted depending on the corresponding position path. The independent variable of precision points in synthesis procedure is chosen from the horizontal points in sagittal plane (x variable) and according to this independent value, the dependent value is determined from the regression curves of varying positions and the orientation angle. For this reason, another regression curve for the orientation angle is needed (Figure 2.34). It is determined in terms of radians as;

$$\emptyset = -24.15x^3 + 7.86x^2 - 2.88x + 4.69 \tag{2.27}$$
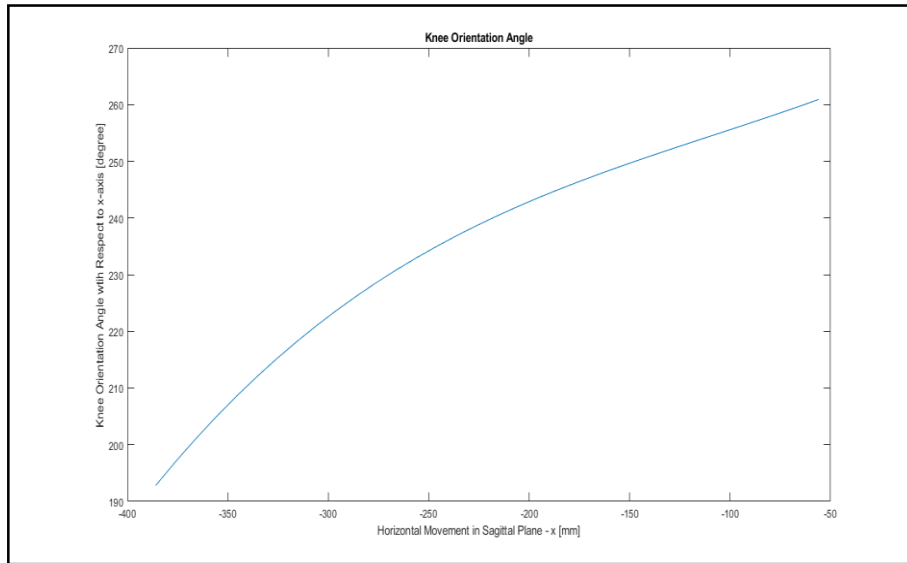
Figure 2.34 Regression curve of knee orientation angle.

The task of the patient-specific knee orthosis mechanism is to follow the path and the orientation that are extracted with the path extraction methodology in this Sub-section. These regression functions of extracted position and orientation trajectory are the main functions to be used in synthesizing mechanism.

In the next Chapter, kinematic synthesis procedure is explained for a single degree of freedom polycentric mechanism and by using these extracted regression curves of the position and orientation, a knee orthosis mechanism is synthesized.

3. KINEMATIC SYNTHESIS OF A KNEE ORTHOSIS DESIGN

For the people who have suffered from lower-limb motor functions disorder, lower-limb robotic devices are being developed in order to execute rehabilitation therapy. These devices are mostly wearable devices coupled with or without treadmills [57-59]. If knee movement is considered only, same kinematic structure as the prosthesis can also be examined in understanding and designing the structure [60-62]. In such devices, systems are treated as a whole, with its all connections, polycentric knee joint and etc.

In literature and commercial usage, wide range of structural types can be observed. In this study, trajectory following with minimum error and force affectivity are the main concerns. It is important that trying to mimic the motion of the related area, revolute pair is insufficient due to the locomotion characteristic of the knee centroid during walking. Instantaneous knee center makes the system polycentric [63]. In each sequence of walking gait cycle, knee center and ankle trajectory follow special paths due to the nature of human walking. Because of the polycentric motion of knee joint, it can be modeled with a four-bar mechanism as the simplest way as seen from Figure 3.1.

Although extracted knee centroid planar path along with proper tibia orientation can be followed exactly with three degrees of freedom planar mechanism, in this work, this path is tried to be followed approximately with a single degree of freedom mechanism. Thus, degree of freedom of given task will be greater than the degree of freedom of designed mechanism. Therefore, it is needed to apply a kinematic synthesis methodology.

54
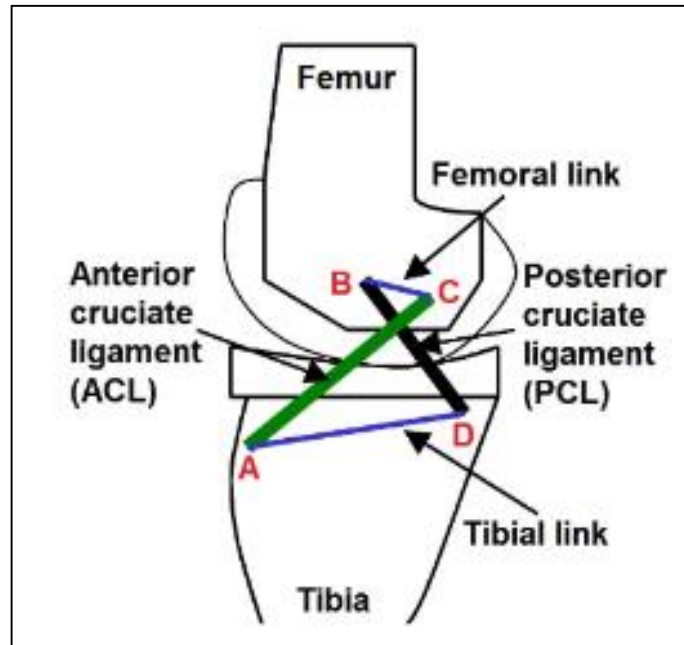
Figure 3.1 Symbolic four-bar mechanism depicted on knee joint [57].

In this Chapter, kinematics of knee is focused and a mechanism is synthesized with body guidance kinematic synthesis procedure by using planar path and orientation data extracted from motion capture system as in sub-section 2.4. Utilizing the proposed of methodology, from motion capture to kinematic synthesis, patient-specific low degree of freedom mechanisms can be designed and used in rehabilitation by means of orthosis or prosthesis.

3.1 Kinematic Synthesis Procedure

During walking, orientation angle of tibia towards the ankle is important as well as its position. Therefore, this orientation angle should be involved in the design procedure. Any unexpected orientation of tibia would lead to harmful damage to the user and system itself. As this is the case, synthesis procedure must contain the orientation of the end-effector with position itself. In light of this requirement, body guidance kinematic synthesis procedure must be applied to the knee joint synthesis. In this method, four-bar mechanism design problem is reduced to dyad synthesis (Figure 3.2).
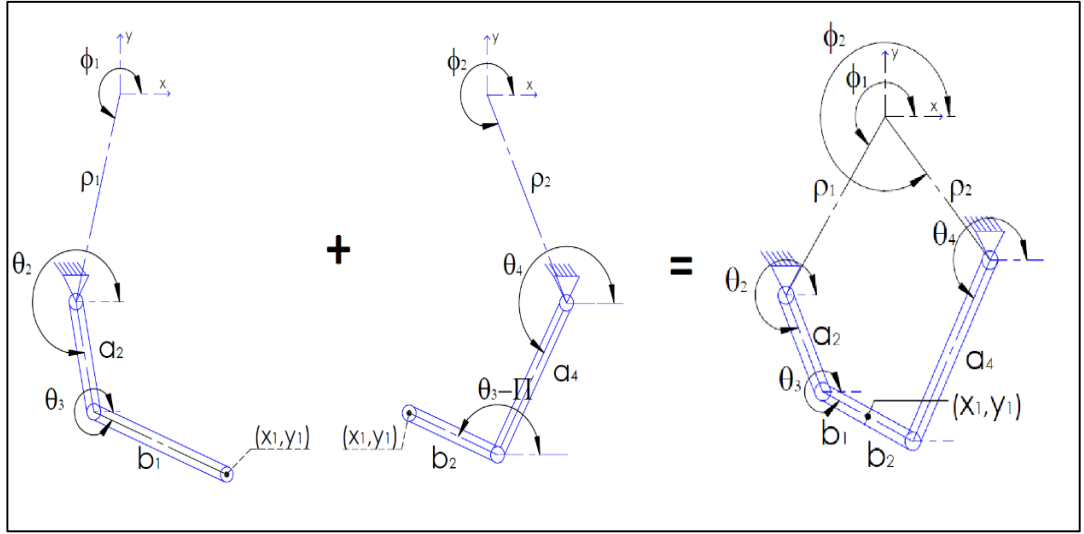
Figure 3.2 Division of four-bar mechanism into two serial arms.

Close loop equations of the serial arm-1 can be written as

$$\rho_1 cos\phi_1 \ + \ a_2 cos\theta_2 \ + \ b_1 cos\theta_3 = X \tag{3.1}$$

$$\rho_1 sin\phi_1 \ + \ a_2 sin\theta_2 \ + \ b_1 sin\theta_3 = Y \tag{3.2}$$

$$a_2 cos\theta_2 \ = X - \ \rho_1 cos\phi_1 \ - \ b_1 cos\theta_3 \tag{3.3}$$

$$a_2 sin\theta_2 \ = Y - \ \rho_1 sin\phi_1 \ - \ b_1 sin\theta_3 \tag{3.4}$$

Taking squares of the equation 3.3 and equation 3.4 and adding them together will lead to an objective function

$$\frac{X^2 + Y^2 + \rho_1^2 + b_1^2 - a_2^2}{2} - X\rho_1 cos\phi_1 - Xb_1 cos\theta_3 + \rho_1 b_1 cos\phi_1 \ cos\theta_3 - Y\rho_1 sin\phi_1 -$$

$$Yb_1 sin\theta_3 + \rho_1 b_1 sin\phi_1 \ sin\theta_3 = 0 \tag{3.5}$$

If ankle trajectory is examined, in orthosis frame, the end point (ankle) is far away from the hip point, which is the origin of frame. After kinematic synthesis procedure, it is possible that the ground point might be located unnaturally very far from the ordinary place of the knee. For this reason, ϕ_1 angle is fixed for the system as the ground location should be placed towards that angle. Thus, number of construction parameters becomes ρ_1, a_2 and b_1 for serial arm-1.

Since X, Y and θ_3 (dependent on the orientation angle) are known variables and fixed ϕ_1 is given to the system equations, equation 3.5 can be rearranged with construction parameters and known parameters.

$$\left(\frac{\rho_1^2 + b_1^2 - a_1^2}{2b_1}\right)1 + \left(\frac{1}{b_1}\right)\left(\frac{X^2 + Y^2}{2}\right) + (\rho_1)(cos(\phi_1 - \theta_3)) + \left(\frac{-\rho_1}{b_1}\right)(Xcos\phi_1 +$$
$$Ysin\phi_1) - [Xcos\theta_3 + Ysin\theta_3] = 0 \qquad (3.6)$$

If equation 3.6 will be written in polynomial form;

$$P_0f_0 + P_1f_1 + P_2f_2 + P_3f_3 - F = 0 \qquad (3.7)$$

where

$$P_0 = \frac{\rho_1^2 + b_1^2 - a_1^2}{2b_1} \qquad (3.8)$$

$$P_1 = \frac{1}{b_1} \qquad (3.9)$$

$$P_2 = \rho_1 \qquad (3.10)$$

$$P_3 = \frac{-\rho_1}{b_1} \qquad (3.11)$$

$$f_0 = 1 \qquad (3.12)$$

$$f_1 = \frac{X^2 + Y^2}{2} \qquad (3.13)$$

$$f_2 = cos(\phi_1 - \theta_3) \qquad (3.14)$$

$$f_3 = Xcos\phi_1 + Ysin\phi_1 \qquad (3.15)$$

$$F = Xcos\theta_3 + Ysin\theta_3 \qquad (3.16)$$

Although equation 3.7 seems to have 4 unknowns, it can be seen that P_3 is dependent on P_1 and P_2. Thus before advancing further, equation 3.7 should be linearized. Let's assume dependent parameters as nonlinear parameters of equation as,

$$P_3 = -P_1P_2 = \lambda \qquad (3.17)$$

and all of the remaining construction parameters are linearly dependent on this nonlinear parameter as,

$$P_i = l_i + m_i \lambda \quad (i = 0,1,2) \tag{3.18}$$

If equation 3.18 is inserted into equation 3.7, polynomial form of the objective function is transformed into equation 3.19, where 3 precision points are needed to solve the problem.

$$(l_0 + m_0 \lambda)f_0 + (l_1 + m_1 \lambda)f_1 + (l_2 + m_2 \lambda)f_2 + \lambda f_3 - F = 0 \tag{3.19}$$

After linear and related nonlinear parts are separated from equation 3.19, two sets of equations can be constructed as,

$$l_0 f_0 + l_1 f_1 + l_2 f_2 = F \tag{3.20}$$

$$m_0 f_0 + m_1 f_1 + m_2 f_2 = -f_3 \tag{3.21}$$

Rearranging equations into matrix form will be represented as

$$\begin{bmatrix} f_0^1 & f_1^1 & f_2^1 \\ f_0^2 & f_1^2 & f_2^2 \\ f_0^3 & f_1^3 & f_2^3 \end{bmatrix} \begin{bmatrix} l_0 \\ l_1 \\ l_2 \end{bmatrix} = \begin{bmatrix} F_1 \\ F_2 \\ F_3 \end{bmatrix} \tag{3.22}$$

$$\begin{bmatrix} f_0^1 & f_1^1 & f_2^1 \\ f_0^2 & f_1^2 & f_2^2 \\ f_0^3 & f_1^3 & f_2^3 \end{bmatrix} \begin{bmatrix} m_0 \\ m_1 \\ m_2 \end{bmatrix} = \begin{bmatrix} -f_3^1 \\ -f_3^2 \\ -f_3^3 \end{bmatrix} \tag{3.23}$$

In order to find constants of the polynomial function (equation 3.7), equation 3.24 should be used as the only unknowns left are nonlinear parameters,

$$\lambda = (l_1 + m_1 \lambda)(l_2 + m_2 \lambda) \tag{3.24}$$

Since equation 3.24 is a second order parabolic function, two different λ values will be found, one of which or both of them can be used to construction parameters.

$$b_1 = \frac{1}{P_1} \tag{3.25}$$

$$\rho_1 = P_2 \tag{3.26}$$

$$a_2 = \sqrt{\rho_1^2 + b_1^2 - 2b_1 P_0}$$ (3.27)

For serial arm-2, the same procedure with the same precision points should be applied in order to find other part of the four-bar mechanism. However, the ground location direction (ϕ_2) must be altered so that the ground point can be located in different place in a desired area.

3.2 Kinematic Analysis Procedure

In the beginning of the body guidance kinematic synthesis method, input angle of four-bar mechanism, θ_2 (or θ_4 for other side of mechanism), is vanished in order to use orientation angle in the algebra of kinematic synthesis procedure as in equation 3.5. As a result, construction parameters of four-bar mechanism are determined by applying the procedure as in previous section. However, there is no clue about input angle itself in the algebra. Since the output angle of four-bar mechanisms coupler can be limited with selected precision points, it can be concluded that within these limits, mechanism acts as a rocker-rocker type, where it is strongly possible that it is a rocker-rocker type kinematically as well. Thus, input range within that mechanism is driven, should be calculated in order to determine position and orientation errors as well as to analyze overall mechanism. Since it is a parallel kinematic chain, it has close loop kinematic constraints as seen in Figure 3.3.
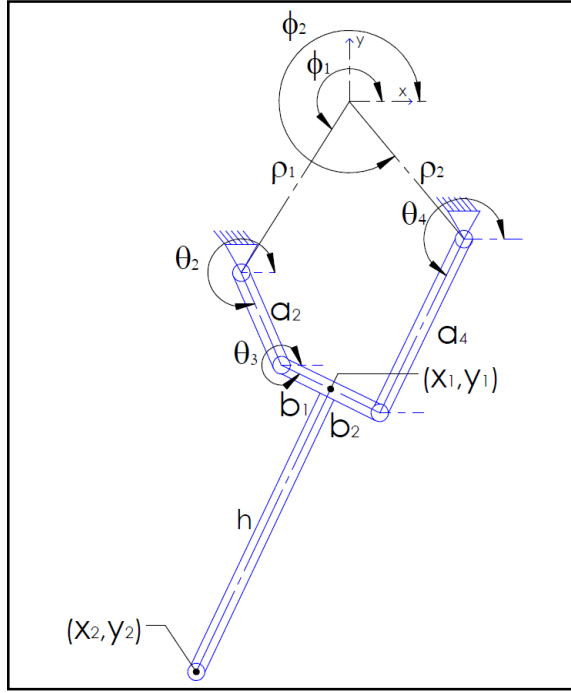
Figure 3.3 Four-bar mechanism formed by synthesized two serial arms.

In order to determine the input limits and range of the mechanism, close loop equations should be used. From serial arm-1, using loop equations 3.1 and 2, θ_2 limits at end points can be found as,

$$\theta_2(@limit1) = atan2(Y_{@limit1} - \rho_1 sin\phi_1 - b_1 sin\theta_{3@limit1}, X_{@limit1} -$$
$$\rho_1 cos\phi_1 - b_1 cos\theta_{3@limit1}) \tag{3.28}$$

$$\theta_2(@limit2) = atan2(Y_{@limit2} - \rho_1 sin\phi_1 - b_1 sin\theta_{3@limit2}, X_{@limit2} -$$
$$\rho_1 cos\phi_1 - b_1 cos\theta_{3@limit2}) \tag{3.29}$$

In equations 3.28 and 3.29, θ_4 and θ_3 angles are unknowns with given θ_2, which is input angle of four-bar mechanism. In order to find instantaneous θ_3 and θ_4 angles, kinematic analysis of mechanism should be done.

Kinematic analysis of combined mechanism stems from loop closure equations (equations 3.1 and 3.2).

After following analytical procedures, following equations are revealed.

$$K_1 cos\theta_3 + K_2 sin\theta_3 = K_3 \tag{3.30}$$

60

where,

$$K_1 = \rho_1(b_1 + b_2)cos\phi_1 + a_2(b_1 + b_2)cos\theta_2 - \rho_2(b_1 + b_2)cos\phi_2 \quad (3.31)$$

$$K_2 = \rho_1(b_1 + b_2)sin\phi_1 + a_2(b_1 + b_2)sin\theta_2 - \rho_2(b_1 + b_2)sin\phi_2 \quad (3.32)$$

$$K_3 = a_2\rho_2cos(\theta_2 - \phi_2) - \rho_1a_2cos(\phi_1 - \theta_2) + \rho_1\rho_2cos(\phi_1 - \phi_2) -$$
$$\left(\frac{\rho_1^2+a_2^2+(b_1+b_2)^2+\rho_2^2-a_4^2}{2}\right) \quad (3.33)$$

Let;

$$K_1 = Dcos\varphi \quad (3.34)$$

$$K_2 = Dsin\varphi \quad (3.35)$$

Then,

$$cos\theta_3cos\varphi + sin\theta_3sin\varphi = \frac{K_3}{D} \quad (3.36)$$

$$\varphi = atan2\ (K_2, K_1) \quad (3.37)$$

$$D = \sqrt{K_1^2 + K_2^2} \quad (3.38)$$

$$cos(\theta_3 - \varphi) = \frac{K_3}{D} \quad (3.39)$$

$$\theta_3 = (\frac{K_3}{D}) + \varphi \quad (3.40)$$

Or

$$\theta_3 = -(\frac{K_3}{D}) + \varphi \quad (3.41)$$

$$cos\theta_4 = \frac{(\rho_1cos\phi_1 + a_2cos\theta_2 + (b_1+b_2)cos\theta_3 - \rho_2cos\phi_2\ )}{a_4} = E \quad (3.42)$$

$$sin\theta_4 = \frac{(\rho_1sin\phi_1 + a_2sin\theta_2 + (b_1+b_2)sin\theta_3 - \rho_2sin\phi_2\ )}{a_4} = F \quad (3.43)$$

From equation 3.42 and equation 3.43, θ_4 can be evaluated as below;

$$\theta_4 = atan2(F, E) \tag{3.44}$$

Velocity analysis of the synthesized system should also be revealed in order to utilize virtual work methodologies later. Velocity analysis procedure starts from the time derivative of close loop equations (equation 3.1 and equation 3.2).

$$-a_4\omega_4 sin\theta_4 = -a_2\omega_2 sin\theta_2 - (b_1 + b_2)\omega_3 sin\theta_3 \tag{3.45}$$

$$a_4\omega_4 cos\theta_4 = a_2\omega_2 cos\theta_2 + (b_1 + b_2)\omega_3 cos\theta_3 \tag{3.46}$$

$$[-a_2 sin\theta_2 \ a_2 cos\theta_2 \ ]\{\omega_2\} = [(b_1 + b_2)sin\theta_3 \ -a_4 sin\theta_4 \ -(b_1 + b_2)cos\theta_3 \ a_4 cos\theta_4 \ ][\omega_3 \ \omega_4 \ ] \tag{3.47}$$

$$\frac{\omega_2}{\omega_3} = \frac{(b_1 + b_2)sin(\theta_3 - \theta_4)}{a_2 sin(\theta_4 - \theta_2)} \tag{3.48}$$

With the revealed kinematic synthesis and analysis equations, numerical case study is convenient to be applied by using path and orientation targets that are worked in sub-section 2.4. Since acquired parameters from kinematic equations have multiple options, optimization takes an important part in the final decision process.

3.3 Optimization

During kinematic synthesis procedure, there are infinite number of possibilities to define predefined parameters (ϕ_1 and ϕ_2). Accordingly, results and effectiveness of the mechanism would also change dramatically. As mentioned in kinematic synthesis part, direction of the ground point (ϕ_1) of mechanism to be placed on a logical place is so important where it should not eventually affect the real life usage and the aesthetics of final mechanism. For these reasons, an algorithm is created and multiple synthesis algorithm is applied to the system equations. With this algorithm, several mechanisms are synthesized in order to achieve goals of needed mechanism as in the flowchart in Figure 3.4.
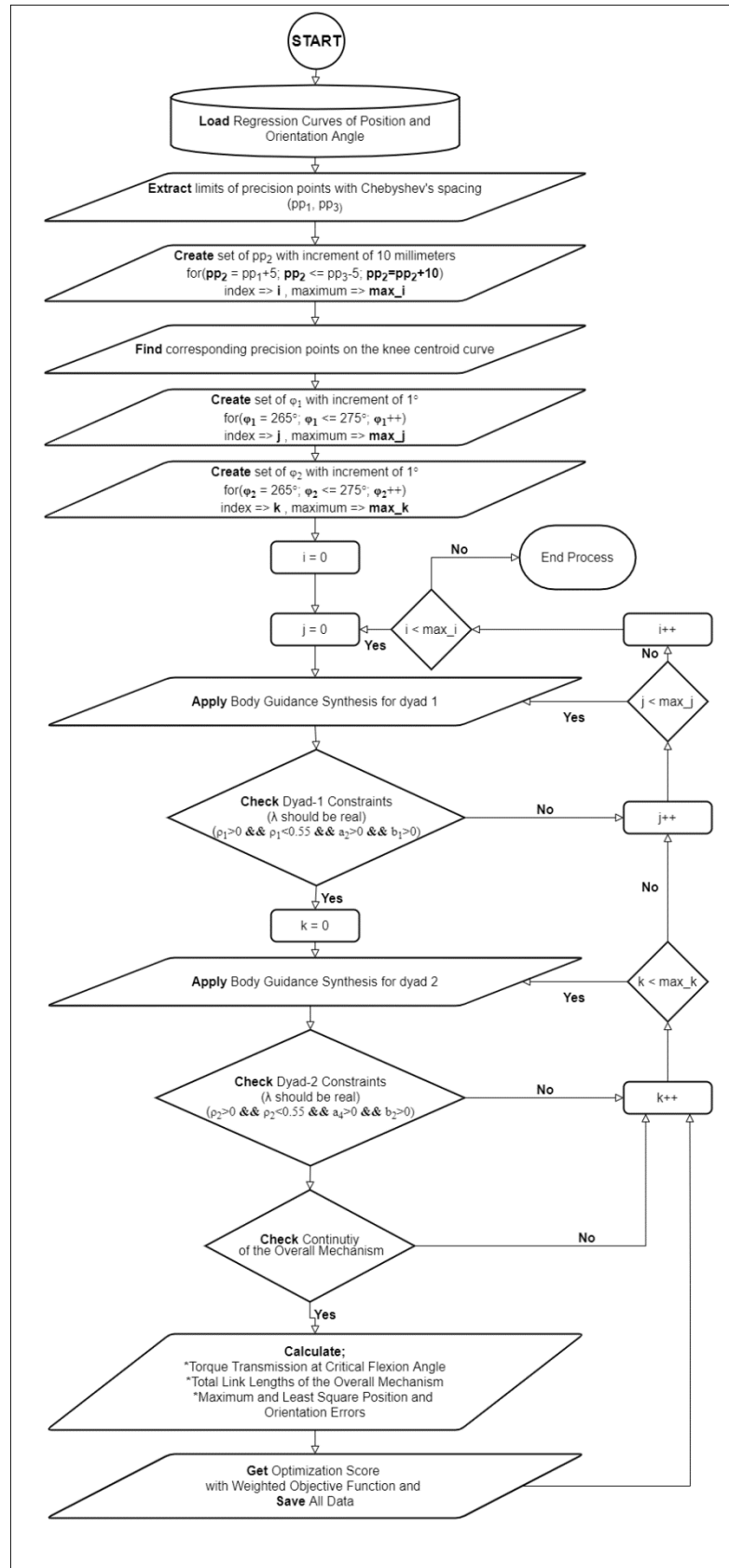
Figure 3.4 Flowchart of multiple synthesis algorithm.

From the origin (hip location) to ground point of the mechanism, virtual lines are created with changing angles, which point out the placing area for the mechanism as in Figure 3.5 and Figure 3.6. For each virtual line, kinematic synthesis procedure is applied as in the flowchart.
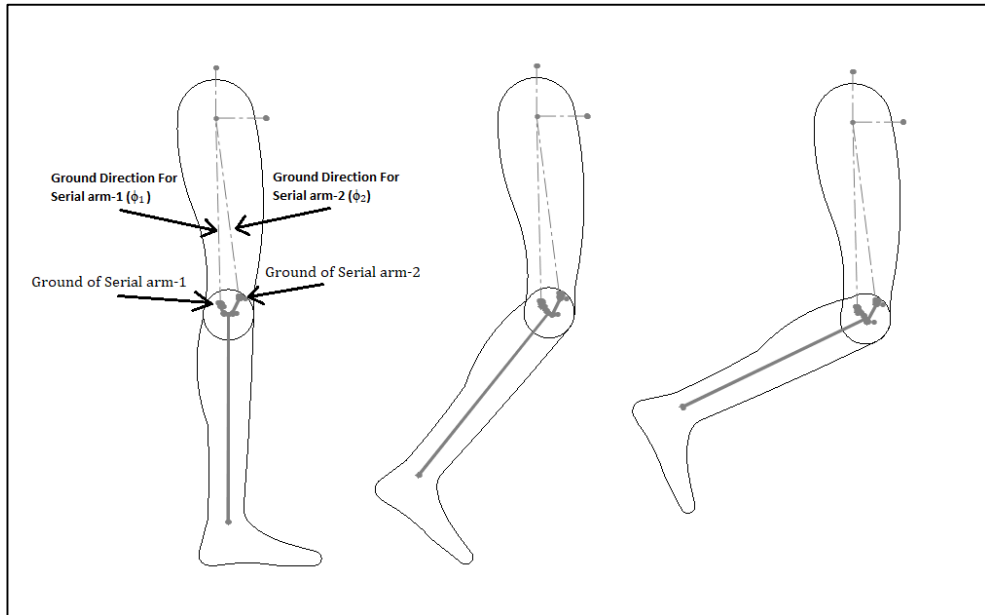
Figure 3.5 Knee flexion via synthesized mechanism.
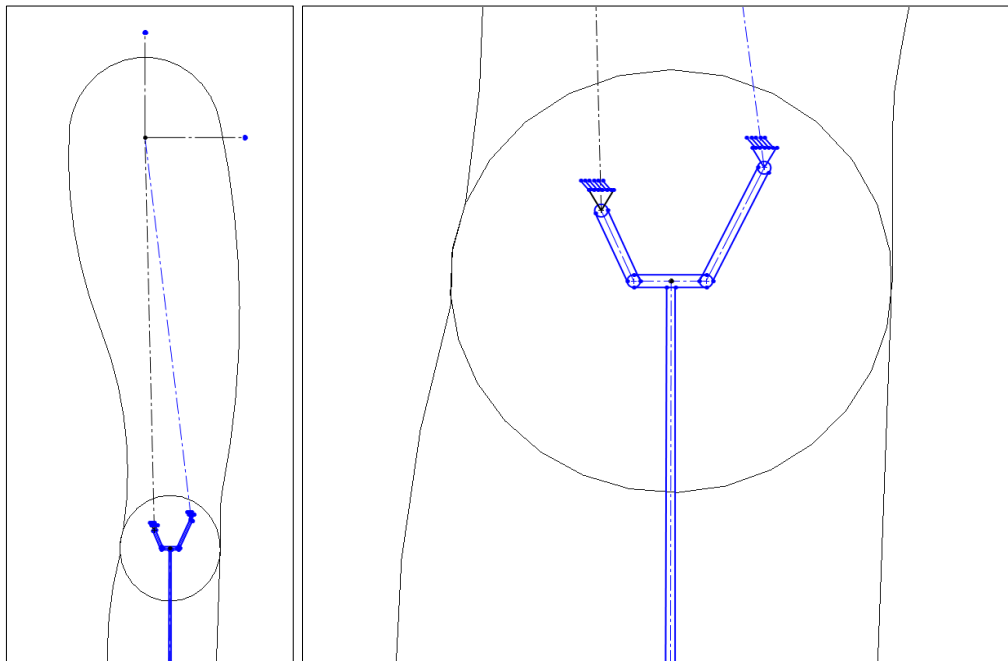
Figure 3.6 Kinematic representation of the mechanism to be placed on knee joint.

It should be pointed out that selection of precision points is so important, which affects the amount of error between desired function and actual generated function formed by the synthesized mechanism. Various known methods can be utilized in selection procedure such as Chebyshev's spacing, equal spacing and random spacing. In this work, with Chebyshev's spacing method, two end limits of precision points are determined. Third precision point is passed through from one limit to another with a predefined 10 mm length in order to see results of all. At each set of precision points, kinematic synthesis procedure is applied, where if output of the procedure is successful, data is stored to be processed in optimization part. According to distinct successful results found by using different precision points, each resultant mechanism has different position and orientation errors. Along with these errors, distinctness of mechanisms is also considered due to the fact that different functionality, torque delivery and link lengths, are also important for effectiveness and compactness of the knee orthosis.

In Figure 3.7, there is an example of precision point distribution obtained by using Chebyshev's spacing method. Precision points are picked from ankle trajectory path and corresponding points on knee centroid curve is applied to the kinematic synthesis methodology.
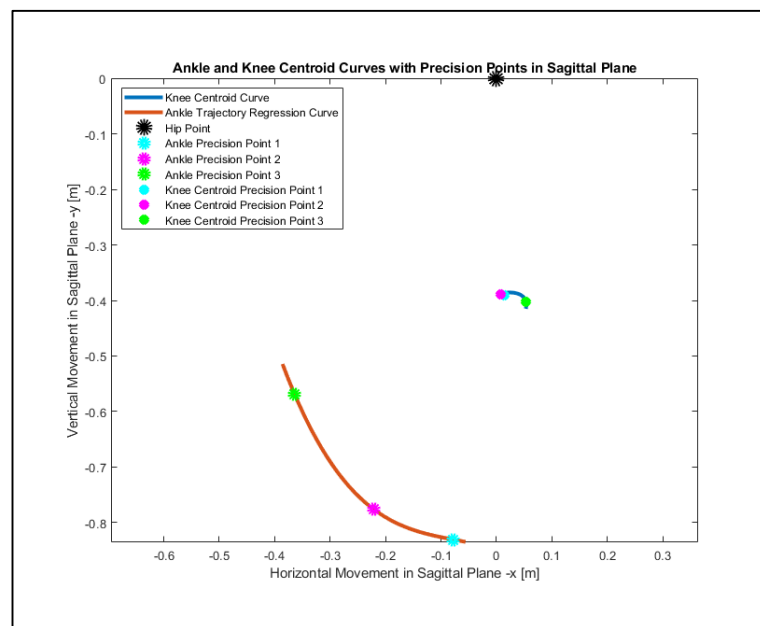
Figure 3.7 Ankle and knee centroid curves with precision points on sagittal plane.

As in literature [64], needed knee moment for walking and corresponding flexion angle can be seen in Figure 3.8. Here, it is possible to see that there are two different peak points required where torques are extreme. These two peak points occur at 14% and 53% of gait cycle, respectively. Corresponding knee flexion angles can be found from Figure 3.8b as 18.96° and 18.68°. In optimization, delivering input torque to output is strongly considered. Thanks to the situation that flexion angles at the moment peak points are close to each other, highest torque transfer should be occurred at around 19° flexion angle of knee joint.
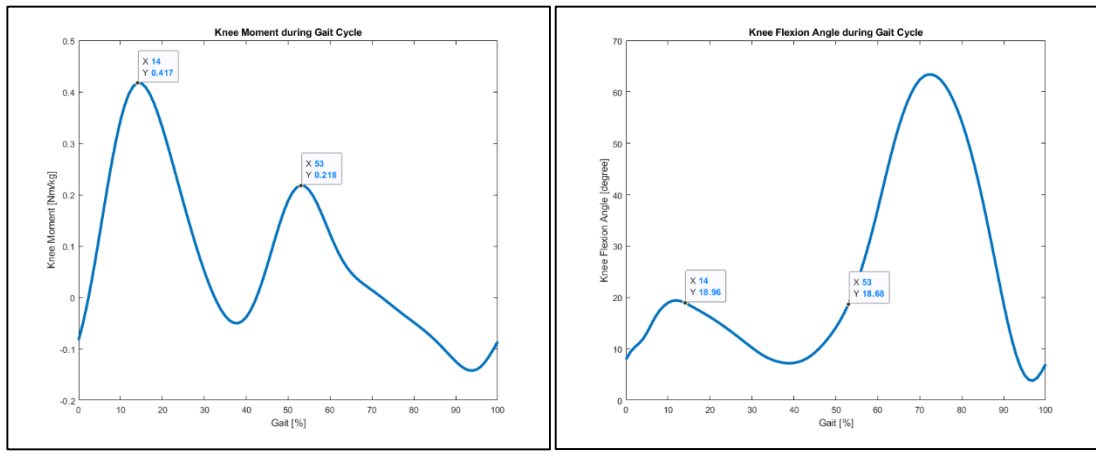
Figure 3.8 a) Knee moment during gait cycle **b)**Knee flexion angle during gait cycle.

Virtual work method is applied for torque transfer with assumption of neglecting mass and inertial effects of links, which are quite small relative to the required torque for walking.

$$\tau_2 w_2 = \tau_3 w_3 \tag{3.49}$$

$$\frac{\tau_3}{\tau_2} = \frac{w_2}{w_3} \tag{3.50}$$

One of the main purposes of optimization part is to maximize the ratio between output torque and input torque when flexion angle is around 19°, which is basically determined by considering analytic solution in equation 3.48.

By considering mechanism as knee orthosis with its initial structure as seen in Figure 3.5 and Figure 3.6, ground orientation angle should be in between 265° and 275° with respect to the horizontal direction (x). When the angle is in predefined range,

location of the ground point of mechanism is placed logically in terms of functionality as well as the aesthetics. Multiple synthesis algorithm contains this procedure in its logic as seen from the flowchart in Figure 3.4.

All of the procedures explained in previous sections are applied to the collected data via motion capture system. There are multiple solutions with multiple input sets into the system in terms of precision points and ground orientation angles (ϕ_1 and ϕ_2). There is no guarantee that all synthesized mechanisms can be continuously work within its input angle limits (θ_2 or θ_4), since mechanisms are designed by using body guidance method. In this method, input angle is eliminated at the beginning of the kinematic equations, due to the fact that it is actually considered as two different serial arms. For the given precision points, united serial arms can achieve required poses, on the other hand it is possible that one of the poses is achieved when the parallel mechanism changes its assembly mode, and thus, there would be non-logical mechanisms within the obtained results.

For instance, mechanism tabulated in Table 3.1 actually fulfills the task at its three precision points in terms of positions and orientation.

Table 3.1 Precision points of non-continuous mechanism.

Precision Point	PP – X Coordinate [mm]	PP – Y Coordinate [mm]	θ_2	θ_3	θ_4
1	-78.0	-831.0	92.9°	-11.8°	231.1°
2	-148.0	-814.3	98.4°	-20.1°	220.5°
3	-363.8	-569.8	61.3°	291.9°	-13.7°

As seen from Figure 3.9, although mechanism achieves the goal in terms of position and orientation at these different precision points, it changes its assembly mode, which leads to discontinuity in given input angles. However, overall system should be continuous due to the fact that walking gait cannot tolerate being out of workspace or any mode change.
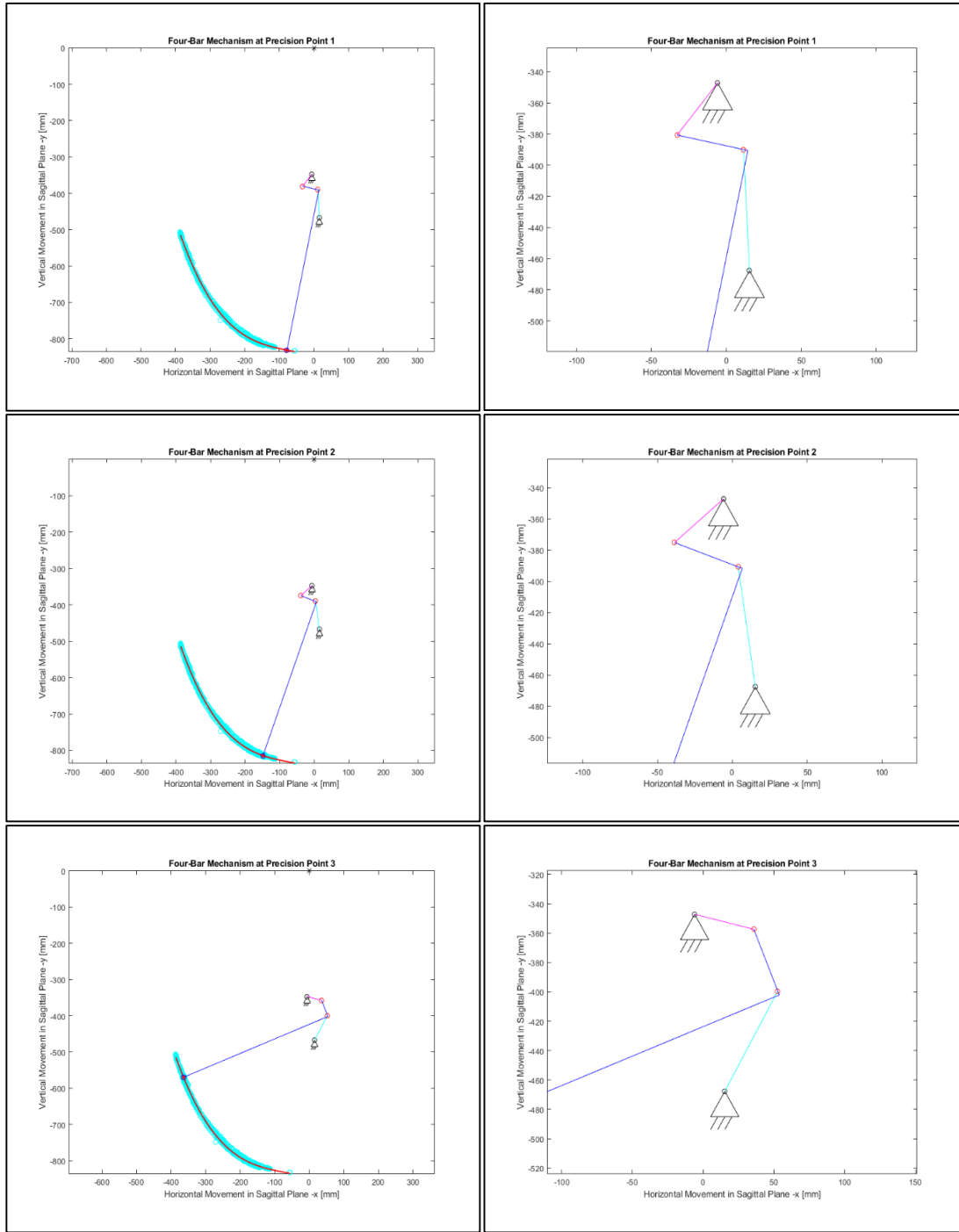
Figure 3.9 Four-bar mechanism **a)** precision point-1 at first mode **b)** precision point-2 at first mode **c)** precision point-3 at second mode.

As a result, continuity of input angles of parallel mechanisms should be checked, whether they can move in one phase within limits of input as determined in equations 3.28 and 3.29. Continuity is controlled for every instant of parallel four bar

mechanism, by using the direct task of mechanism as mentioned in flowchart (Figure 3.4).

When non-continuous mechanisms are removed from data set of results, remained sets are used in applying maximum torque delivering at 19° of flexion angle. Top 10 torque delivering mechanisms that are driven from serial arm-1 (with θ_2 input) and driven from serial arm-2 (with θ_4 input) can be seen in Table 3.2.

Table 3.2 Top 10 torque delivering mechanisms in data set.

From θ_2	Torque Ratio	From θ_4	Torque Ratio
1	1.6072	1	0.6153
2	1.6056	2	0.2852
3	1.6039	3	0.2849
4	1.6032	4	0.2845
5	1.6021	5	0.2508
6	1.6018	6	0.2501
7	1.6002	7	0.2496
8	1.5999	8	0.2491
9	1.5984	9	0.2486
10	1.5975	10	0.2482

From the table, it is clearly seen that torque delivering ratio at required instant is high and close to each other when input comes from serial arm-1, therefore results of serial arm-2 are ignored.

Due to the lack of distinct difference among resultant mechanisms in terms of torque delivering ratio, more elements are examined for each mechanism for additional requirements as; total link length, maximum position and orientation error, least square error for position and orientation. Since there are multiple elements that affect efficiency of resultant mechanism, weighted objective method [65] is applied in order to get the most efficient result.

Scores of each resultant mechanism in terms of required elements are normalized by using equation 3.51. Thus, it is obvious that scores are actually objective and depend on real numerical values, since they are revealed from characteristics of mechanism.

$$S = \frac{X - X_{min}}{X_{max} - X_{min}} \tag{3.51}$$

where, S is the score, X is the value of considered element for each resultant mechanism, X_{min} and X_{max} are minimum and maximum values of considered element among all of the dataset of mechanisms.

In this decision making process, minimum torque delivering ratio is defined as a constraint that strongly affects scoring. Thus, torque delivering score of mechanism with defined minimum value is actually zero. In this case, it is numerically defined as 1.50.

Weights of the elements are subjectively evaluated according to the requirements. Each element is provided relative weights to the equation as torque delivering ratio 0.25, maximum position error 0.1, maximum orientation error 0.1, least square position error 0.1, least square orientation error 0.1 and total link length (compactness) 0.35. Although torque delivering ratio is important for our case and mentioned above, its relative weight is the second highest due to the fact that this parameter was already affected from the normalization part as a constraint. Its effect is zero when it is smaller or equal to 1.5. Highest weight is reserved for total link lengths for two reasons. One of the reasons is about the requirement for compactness. Since orthosis will be connected to the knee joint from outside, compactness is one of the most important ally in functionality as well as aesthetically. Second reason is about increasing reliability of analysis. In determining torque delivering ratio for resultant mechanisms, virtual work method is applied, in which masses and inertias of the links are not considered. By considering needed torques for walking, this approach is actually quite applicable, however, when link lengths are increased, reliability of this assumption decreases accordingly. For these two reasons, total link lengths have a huge impact on decision making process. Other errors have small effects on the decision making, since errors have relatively smaller effect on the topic.

Scores (S) for each requirement elements are determined and these scores are multiplied by relative weights (W) to find values (V) for each mechanism. Those values are summed up to get final score (FS). Eventually, the highest final score is considered to be the most efficient mechanism for our case. Results of selected mechanisms with weighted objective method can be seen in Table 3.3.

Table 3.3 Evaluation of the mechanisms using weighted objective method.

	Torque Ratio		Max Error (Position) [mm]		Least Square Error (Position) [mm]		Max Error (Orientation Angle)		Least Square Error (Orientation Angle)		Total Link Lengths [mm]		FS
W	0.25		0.1		0.1		0.1		0.1		0.35		
	S	V	S	V	S	V	S	V	S	V	S	V	
1	1.00 (1.61)	.25	0.49 (13.6)	.049	0.65 (105.4)	.065	0.75 (2.30°)	.075	0.73 (18.0°)	.073	0.25 (264.3)	.09	0.597
2	0.99 (1.61)	.25	0.49 (13.6)	.049	0.65 (105.1)	.065	0.75 (2.29°)	.075	0.74 (18.0°)	.074	0.26 (258.4)	.09	0.601
3	0.97 (1.60)	.24	0.49 (13.5)	.049	0.65 (104.8)	.065	0.76 (2.29°)	.076	0.74 (18.0°)	.074	0.29 (252.6)	.10	0.606
8	0.93 (1.60)	.23	0.50 (13.5)	.050	0.66 (104.0)	.066	0.76 (2.28°)	.076	0.75 (17.8°)	.075	0.33 (241.0)	.12	0.616
10	0.91 (1.60)	.23	0.51 (13.4)	.051	0.67 (103.5)	.067	0.76 (2.28°)	.076	0.75 (17.8°)	.075	0.35 (235.2)	.12	0.620
15	0.87 (1.59)	.22	0.48 (13.7)	.048	0.63 (106.4)	.063	0.77 (2.26°)	.077	0.75 (17.8°)	.075	0.19 (280.5)	.07	0.548
20	0.82 (1.59)	.21	0.53 (13.2)	.053	0.70 (101.2)	.070	0.75 (2.30°)	.075	0.75 (17.9°)	.075	0.50 (204.1)	.18	0.654
25	0.77 (1.58)	.19	0.54 (13.1)	.054	0.71 (100.7)	.071	0.77 (2.25°)	.077	0.79 (17.4°)	.079	0.46 (211.6)	.16	0.636
30	0.72 (1.58)	.18	0.51 (13.4)	.051	0.69 (102.1)	.069	0.72 (2.37°)	.072	0.70 (18.5°)	.070	0.38 (227.7)	.13	0.577
35	0.69 (1.57)	.17	0.56 (13.0)	.056	0.74 (98.8)	.074	0.76 (2.28°)	.076	0.78 (17.5°)	.078	0.61 (186.1)	.21	0.669
40	0.66 (1.57)	.17	0.54 (13.2)	.054	0.70 (101.5)	.070	0.79 (2.21°)	.079	0.81 (17.2°)	.081	0.36 (233.7)	.13	0.575
45	0.65 (1.57)	.16	0.64 (12.3)	.064	0.66 (104.4)	.066	0.86 (2.08°)	.086	0.72 (18.1°)	.072	0.21 (276.8)	.07	0.522
49	0.63 (1.57)	.16	0.66 (12.2)	.066	0.68 (102.9)	.068	0.84 (2.10°)	.084	0.70 (18.4°)	.070	0.35 (236.9)	.12	0.568
50	0.62 (1.57)	.16	0.57 (12.9)	.057	0.76 (97.7)	.076	0.77 (2.27°)	.077	0.79 (17.4°)	.079	0.65 (179.8)	.23	0.673
51	0.62 (1.57)	.16	0.47 (13.7)	.047	0.62 (107.2)	.062	0.78 (2.23°)	.078	0.77 (17.6°)	.077	0.14 (300.2)	.05	0.470
55	0.61 (1.57)	.15	0.53 (13.2)	.053	0.72 (100.4)	.072	0.73 (2.38°)	.073	0.72 (18.2°)	.072	0.57 (192.8)	.20	0.621

As seen from Table 3.3, by using weighted objective method with defined weights and determined numerical scores, 50th mechanism becomes the most efficient mechanism (total score 0.673 out of 1). As a result, this mechanism is chosen.

3.4 Results

Optimum mechanism has following precision points as in Table 3.4, where position and orientation errors are zero on sagittal plane.

Table 3.4 Precision points of chosen mechanism.

Precision Point 1 (x,y) [mm]	Precision Point 2 (x,y) [mm]	Precision Point 3 (x,y) [mm]
(-78.0,-831.0)	(-308.0,-678.3)	(-363.8,-569.8)

Construction and given parameters can also be found in Table 3.5.

Table 3.5 Construction parameters of chosen mechanism.

	a [mm]	b [mm]	ρ [mm]	φ [degree]	h [mm]
Serial Arm-1	47.5	51.7	394.3	91.3°	450
Serial Arm-2	56.4	24.2	444.7	91.3°	450

Within limits of input angle of four-bar mechanism, θ_2, it is possible to determine θ_3 and θ_4 angles at all instants so that position and orientation errors can be calculated. In this work, MATLAB based simulation interface is prepared by including all the data cloud acquired from the motion capture system, as well as the regression curve and mechanism itself. With determined construction parameters, simulation is done as seen in Figure 3.10. Axis placing to human body is visualized in Figure 3.11.
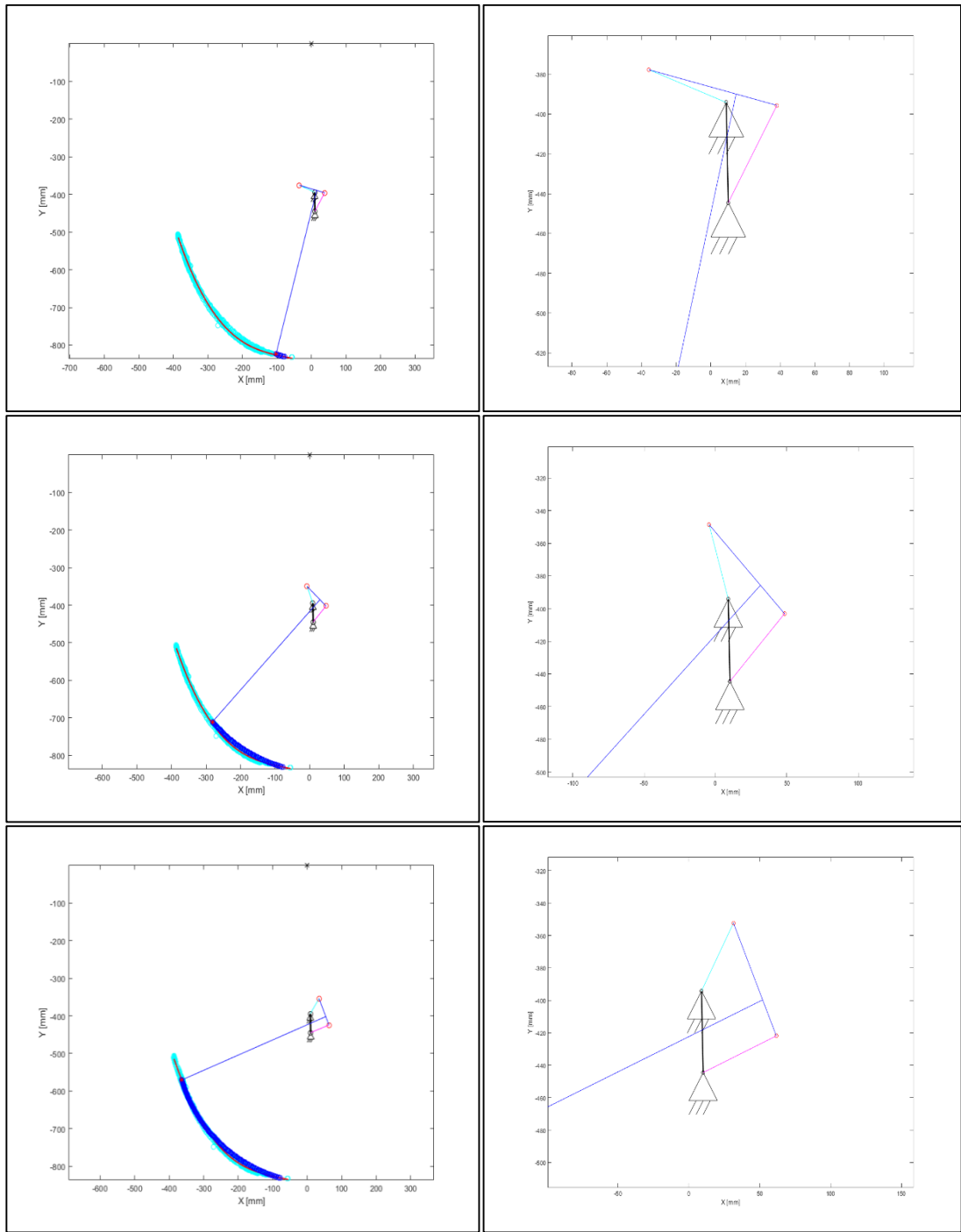
Figure 3.10 Chosen four-bar mechanism **a)** at precision point-1 **b)** at precision point-2 **c)** at precision point-3.
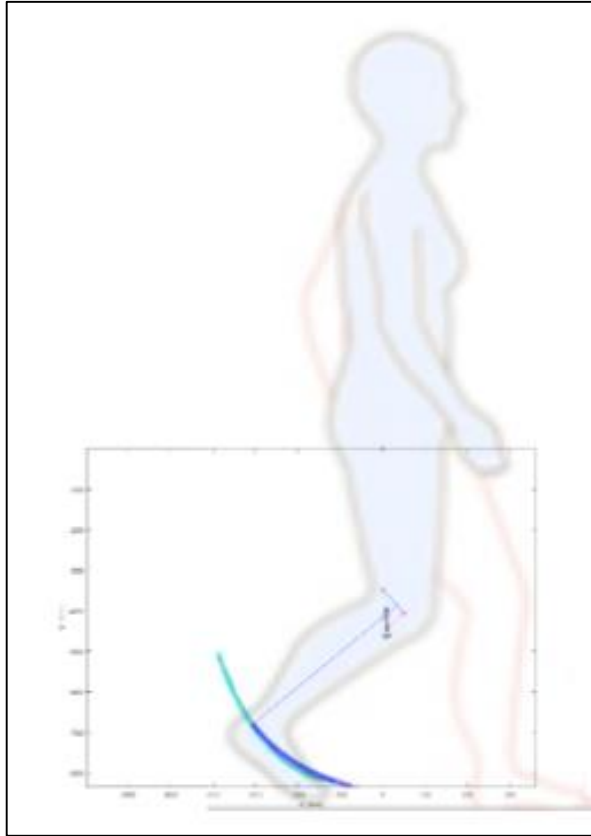
Figure 3.11 Mechanism location on human body. Background layer is modified from [66].

Since system is constructed by using body guidance methodology, there are two different error parameters, which are position and orientation angles. Trajectory in terms of position and orientation angle of the final mechanism with regression curves and total errors in horizontal movement can be found in Figure 3.12.
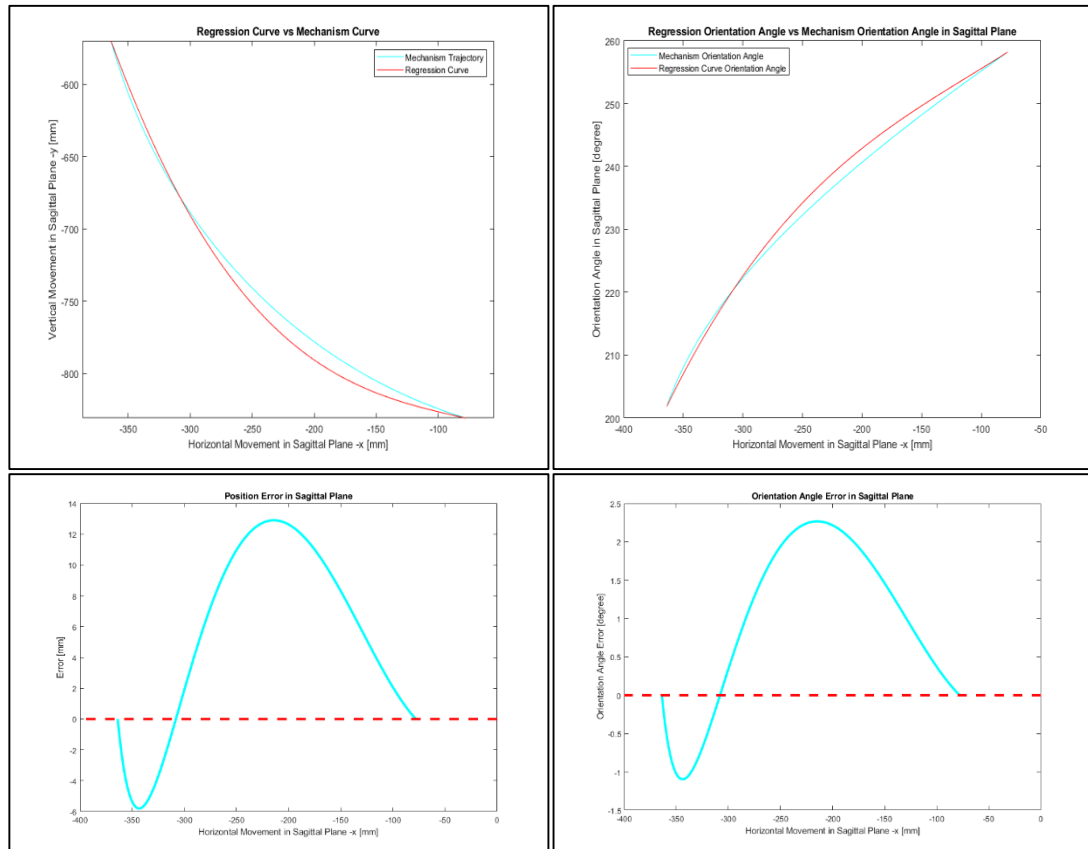
Figure 3.12 a) Regression trajectory curve vs mechanism trajectory curve **b)**Regression orientation angle vs mechanism orientation angle **c)**Position error **d)**Orientation error.

In conclusion, single degree of freedom four-bar mechanism is synthesized theoretically for following desired path with desired orientation angle. Here, it can be seen that ankle joint, end effector of the system, follows regression curve which is generated in previous sections, as close as possible with maximum error of 12.9 mm and 2.3°. On the other hand, it is also clear that end effector motion always stays in data cloud that is projected onto the motion plane (Figure 3.13), and thus it can be concluded that this mechanism is very convenient to be used as knee prosthesis or orthosis.
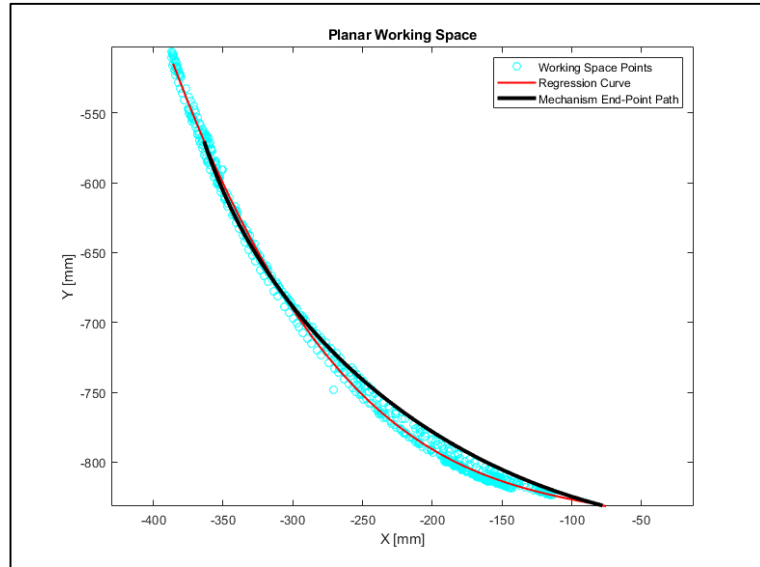
Figure 3.13 Planar working space with regression curve and mechanism output.

Following steps of the methodology, planar motion task is created from motion capture and path extraction procedures. Afterwards, single degree of freedom planar mechanism is synthesized to achieve given task. Next step is to create a control algorithm to adopt any concerned mechanism to rehabilitation purposes.

4. REHABILITATION CASE STUDIES WITH ADMITTANCE CONTROL

One of the main purposes of this thesis is to build a lower degree of freedom overall system that eventually works for physical rehabilitation in any considered part of human body. In order to achieve this goal, procedure starts with motion capturing from human extremities and goes on with planar path extraction that to be used in synthesizing a planar mechanism. Kinematic synthesis plays an important role to design a mechanism that aims to achieve body part movement in the right way due to the fact that it is based on real path and orientation information. Finally, in the light of the rules of rehabilitation procedures, system needs to be controlled with an actuator.

In this thesis, admittance control strategies in rehabilitation are combined with game rehabilitation in virtual environment by using readily built electromechanical system that is based on the same pre-procedures of this work such as motion capture, path extraction and kinematic synthesis [26]. In Gezgin's work, a hand rehabilitation setup (Figure 4.1) is designed and manufactured. The system was created to be able to mimic natural hand grasping motion for the post stroke patients. It is a single degree of freedom six-bar mechanism that is driven from a crank link. The repetitive continuous grasping motion is achieved with full rotations of the crank link in one direction, where there is no need to change the direction of the actuator to fulfill the motion. This electromechanical system is ready to be controlled for rehabilitation purposes with its torque sensor, electric motor, kinematic structure and all electrical components such as motor driver and power supply.

Besides its mechanical parts, system has a FUTEK TRS300 20 Nm shaft to shaft rotary torque sensor which is located between output shaft of gearbox of electric motor and flexible coupling which is the main connection component to rest of the system. This sensor is used as a feedback device for measuring user intention torque

as well as actuator torque. Also, the system has an electric motor as main actuator, which is Maxon brushless DC motor (250 W, 5000 rpm, 331 mNm). A gearbox with reduction ratio of 43 is connected to output of electric motor in order to increase delivered torque and reduce rotational velocity to requested levels.
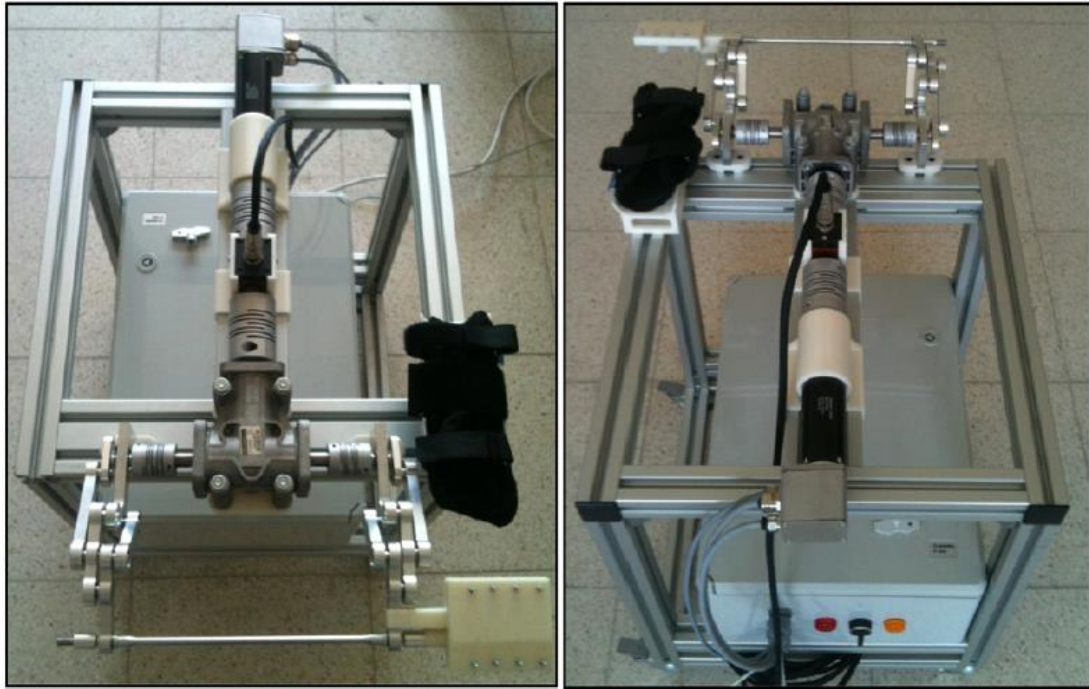
Figure 4.1 Hand Rehabilitation System [26].

Overall system is ready to be used as hand rehabilitation device, when a proper controller is applied. However, since admittance controller adds virtual dynamics to the system with mathematic explained in Chapter 1, it should be tuned in order to be used effectively. Besides of the complexity of admittance control, six-bar mechanism on hand rehabilitation system adds some additional complexity to the system as well. For this reason, six-bar mechanism is deconstructed, instead of it, simple pendulum (like a one dimensional joystick) is used with the same electrical and electronic components as the preliminary work. This simple setup can also serve as a rehabilitation device that aims to wrist rotation and forearm pronation-supination.

From now on, two different system setups that aim to rehabilitate wrist rotation, forearm pronation-supination and hand grasping (finger extension) are used one after another. Considering different rehabilitation modes and control strategies in literature as mentioned in Sub-sections 1.1 and 1.4, respectively, two different case studies

with admittance control are studied for rehabilitation purposes. Admittance control strategy involves both free motion systems, where there is no reference fixed point on that motion is dependent, and fixed-to-reference point motion systems, where motion is dependent on a reference position proportionally, like a virtual spring. Therefore, mathematic behind this virtual admittance is generalized with fixed-to-reference point system in Sub-section 1.4. Thus, if free motion system is required to be used, related virtual spring parameter must be omitted. Variable admittance control strategies with different approaches according to rehabilitation modes are applied with pointing the purpose of each. These case studies are both single degree of freedom mechanisms that focus on finger, wrist and forearm rehabilitation.

4.1 Setting-Up the System

Used system in this thesis is actually already studied by Gezgin et al. [26] and built as an electromechanical system. By considering the system as whole, some modifications are made on the pre-built system in order to apply wrist and forearm rehabilitation case study as well. Also, since the system has an electric motor and a torque sensor, overall system should be prepared to be adopted into both case studies, which are wrist and hand rehabilitation.

4.1.1 Torque sensor calibration

In this work, FUTEK TRS300 20 Nm shaft to shaft rotary torque sensor is used as main feedback device in control loop along with an amplifier module FUTEK CSG110, which is capable of giving analog output between -10/+10 V with resolution of 0.5 V/Nm (Figure 4.2).

Figure 4.2 a) FUTEK TRS300 20 Nm Torque Sensor b) FUTEK CSG110 Amplifier.

This analog signal was measured by Humusoft MF624 PCI card, which was installed onto a PC. All of acquired sensor data were digitalized and observed with Desktop Real Time Toolbox of Simulink. Prior to the implementation, torque sensor was calibrated to be used properly.

During dedicated calibration procedure, one of two shafts of rotary torque sensor was fixed. For remaining shaft, a single link with a 14 cm length was designed and manufactured with a rapid prototyper in order to generate a known torque value at sensors rotation axis by hanging an object with a known mass to link tip (Figure 4.3).
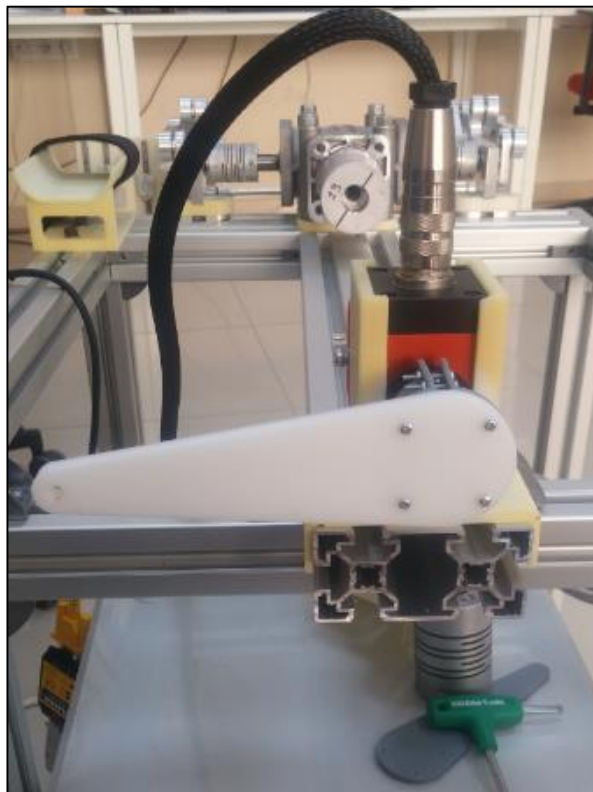
Figure 4.3 Torque sensor calibration setup.

At horizontal position of linkage (Figure 4.3), an object with a mass of 2 kg was hanged to tip of link and torque value of 2.35 Nm was created on the system. Since it is a linear torque sensor with 0.5 V/Nm resolution, corresponding voltage was measured via torque sensor and it should have been approximately 1.18 V. At stable state, by turning the span knob on amplifier module, measured value was adjusted to match calculated voltage value. After arrangements, object and link were removed from the system. At this point, expected value that was measured from the sensor

should have been 0 V, thus, zero knob on amplifier was adjusted until 0 V was read from output of the sensor. At the end of the procedure, calibration of the torque sensor was verified with different objects.

4.1.2 Modification of maxon motor drive from Simulink

As mentioned in Gezgin's work [26], Maxon brushless DC motor with built-in gearbox, encoder and Hall Effect sensor is used as the actuator system. Driver system of the motor is from the same brand, which is specific hardware EPOS2 (Figure 4.4). EPOS2 driver system allows users to communicate with personal computers via serial communication.

Figure 4.4 EPOS2 Maxon Motor Driver.

Utilizing EPOS2 communication command libraries from Eugenio [67], the system is driven real time with Simulink, thus all control algorithms are simply applied in Simulink environment.

To date, commanding Maxon Motors from MATLAB and Simulink via EPOS2 driver system is developed from Eugenio at file exchange center of MATLAB. This is very useful for developers that desire to use MATLAB as a main platform of commanding. Developed files are based on libraries that are published from Maxon,Inc. However, according to this shared files for Simulink, main s-function of Maxon Motor gives output of encoder position alone, not including any other sensor parameters such as motor current or motor velocity. Also, by using related s-function, it is not possible to switch drive mode of the system in real time, which are

position control mode, velocity control mode or current control mode. However, according to this work's requirements, at the same control structure, velocity control is required to be switched to position control or vice versa in real time. For these two reasons, this s-function is modified to fulfill this needs.

As the main structure of admittance control has two different control loops, inner one is a classical motion controller, where it is mainly rotational velocity. In this work, inner velocity controller is driven with auto tuned readily built drive mode of Maxon controller block. Thus, in this work, inner loop control parameters were not intervened, instead, they are left as Maxon Motor's auto tuning parameters. It is only focused on admittance control parameters in terms of rehabilitation procedures, however, eventually the performances of Maxon Motor velocity drive mode are presented.
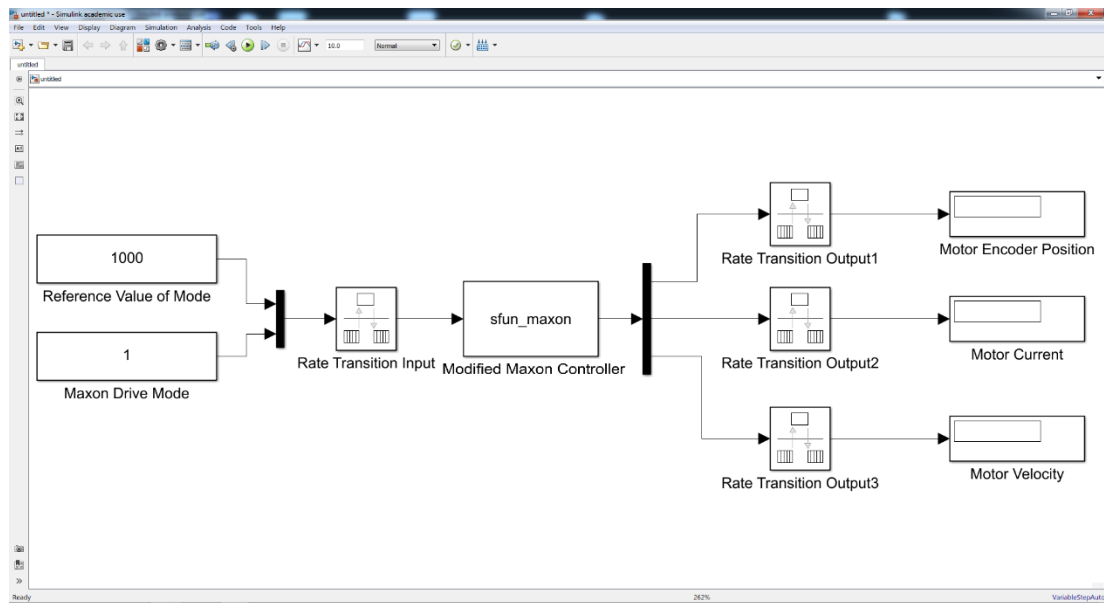
Figure 4.5 Modified s-function of maxon drive model from Simulink.

With small modifications in terms of hardware and software with calibration of torque sensor that are placed in the system, case studies are studied. Modified Maxon controller block, for which Eugenio's work [67] selected as base, can be seen in Figure 4.5, where it eventually has additional one input and additional two outputs. Maxon controller's s-function requires totally two inputs, one of which is reference value of drive mode parameter, and second is drive mode itself. In this case, if value

of drive mode is 0, then current controller becomes drive mode with reference current in terms of milliamps. If value of drive mode is 1, then drive mode becomes velocity and reference velocity should be given to the system as first input in terms of revolution per minute. Finally, if drive mode is chosen as 2, then drive mode becomes position based, where encoder pulse number is required for reference value. Modified system gives totally three different outputs; encoder position, motor current and motor velocity in order to be used in control structure or just observing results.

Since the system has single shaft to shaft torque sensor, which is located between the output shaft of electric motor reduction and the handle itself, force or torque intention of user cannot be measured directly. For this reason, once user applies some torque, the torque sensor is affected by user intention as well as electric motor and system dynamics. However, user intention is required to visualize and analyze effects of user intention on movement. Thus, torque intention is estimated by using following equations as in pre-work [68].

The output torque of electric motors can be estimated, if motor current is measurable. In such cases, the output torque is proportional with the motor current, where proportion depends on a fixed motor-related value known as torque constant.

$$\tau_{motor} = i_{motor} k_t \qquad (4.1)$$

where, τ_{motor} is output torque of electric motor, i_{motor} is electric current of motor that is measured by the EPOS2 drive system and k_t is torque constant of specific electric motor that is supplied from the Maxon,Inc.

Due to the location of shaft to shaft torque sensor, its output measurements depend on torques on both shafts. Input shaft is affected by electric motor and output shaft is affected by user intention and system dynamics. Its measurement can be generalized with including the directions as;

$$\tau_{measured} = \tau_{intention} - \tau_{motor} - \tau_{system} \qquad (4.2)$$

where, $\tau_{measured}$ is torque value that is measured with the torque sensor on the system and τ_{system} is torque from system dynamics depending on inertia, friction

83

and gravity of the system that is extracted in equation 4.11. Here, user torque intention is the main parameter to be estimated in real time, thus, it is determined as in equation 4.3.

$$\tau_{intention} = \tau_{measured} + \tau_{motor} + \tau_{system} \qquad (4.3)$$

Estimated user torque intention value is used in the following sections in order to compare approximate torque intentions between case studies. Effects of the different virtual admittance gains and offset torques in terms of assistive and resistive will be obseved numerically and according to these values, parameters will be chosen for game rehabilitation. Also, estimated user intention torque value is used as independent parameter of variable admittance control in free move structures, which will be discussed in following sub-sections. According to changing on user torque intention, virtual gains of variable admittance control (virtual damping coefficient and virtual rotational inertia) change as well (Figure 4.9 and 4.29).

4.2 Single DOF Wrist and Forearm Rehabilitation Case Study

This simple mechanism aims to work on the movement of wrist rotation and forearm pronation-supination (Figure 4.6). It can be counted as preliminary work of this thesis in terms of applying control algorithms in real environment. With simplicity of this system, complication of control algorithms and other related problems are left as only things to be resolved, thus focusing to algorithms is made possible.

Wrist and forearm rehabilitation mechanism actually is a simple pendulum whose main shaft is connected to shaft to shaft torque. As actuator, Maxon motor drive system is used and needed couplings for transmitting motion and torque are connected.
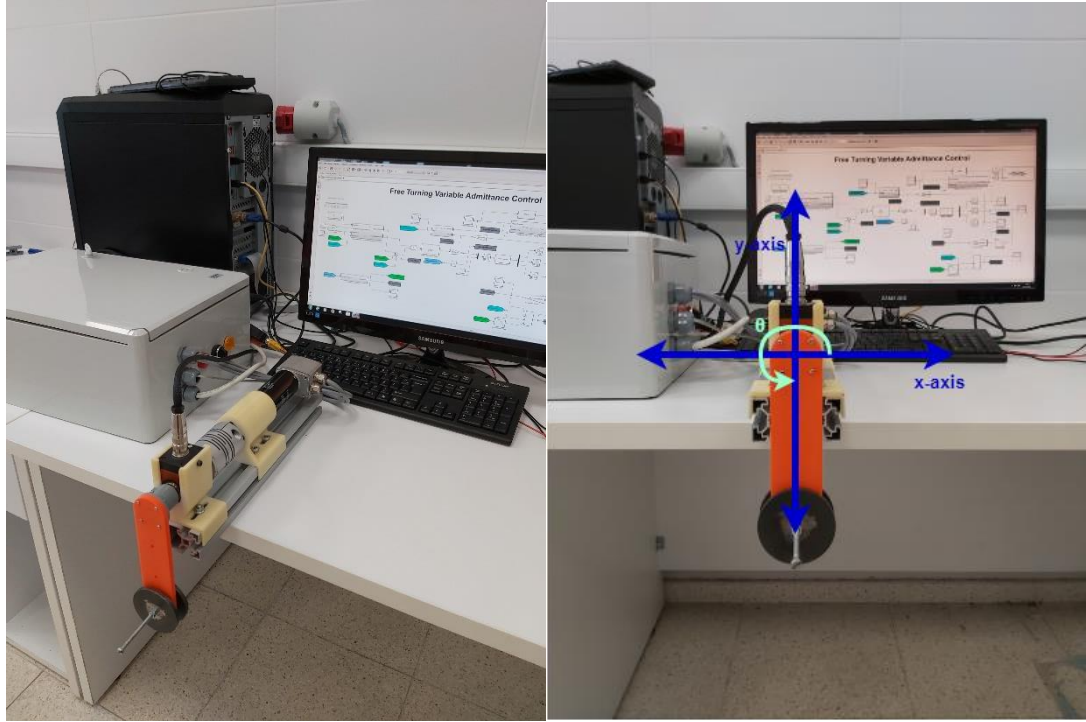
Figure 4.6 a) Wrist and forearm rehabilitation system b) System coordinate system with angle.

In order to determine real time value of system torque, equation of motion of the system is required. Thanks to simplicity of overall system, dynamic equations can be conducted with energy equations.

$$T = \frac{1}{2}mv_{cog}^2 + \frac{1}{2}J\omega^2 \tag{4.4}$$

where, T is total kinetic energy, m is mass of the system, v_{cog} is velocity of center of gravity of the system, J is rotational inertia and ω is rotational velocity of link.

$$v_{cog}^2 = l_{cog}^2\omega^2 \tag{4.5}$$

where l_{cog} is distance of center of gravity from axis of rotation.

$$V = mgl_{cog}\sin\theta \tag{4.6}$$

where V is total potential energy, θ is rotation angle with respect to defined x-axis.

Apart from energy equations, the system undergoes a certain amount of friction, which cannot be omitted. In Lagrange equations, Rayleigh parameter, R, should be

conducted in order to be involved into system equations. Here, friction coefficient is treated as constant rotational damping coefficient, c, which is assumed to affect the system proportional to rotational velocity.

$$R = \frac{1}{2}c\omega^2 \tag{4.7}$$

With Lagrange equations, equation of motion of pendulum can be determined as;

$$L = T - V + R \tag{4.8}$$

$$L = \frac{1}{2}ml_{cog}^2\omega^2 + \frac{1}{2}J\omega^2 - mgl\sin\theta + \frac{1}{2}c\omega^2 \tag{4.9}$$

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \omega}\right) - \frac{\partial L}{\partial \theta} + \frac{\partial R}{\partial \omega} = \tau_{system}(\theta, \omega, \alpha) \tag{4.10}$$

where, α is angular acceleration of the system.

And finally, equation of motion can be determined as;

$$\tau_{system}(\theta, \omega, \alpha) = \left(ml_{cog}^2 + J\right)\alpha + c\omega + mgl_{cog}\cos\theta \tag{4.11}$$

Since overall rehabilitation system is controlled with admittance control method, it must have at least two different control loops. Outer loop is related with indirect force control, which mainly focuses on the torque on end effector. As mentioned in previous sections, admittance control is used as indirect force control type, due to non-back drivability of the overall system. According to virtual dynamics of admittance control, it creates a desired rotational velocity trajectory to be followed by inner velocity loop (Figure 4.7).
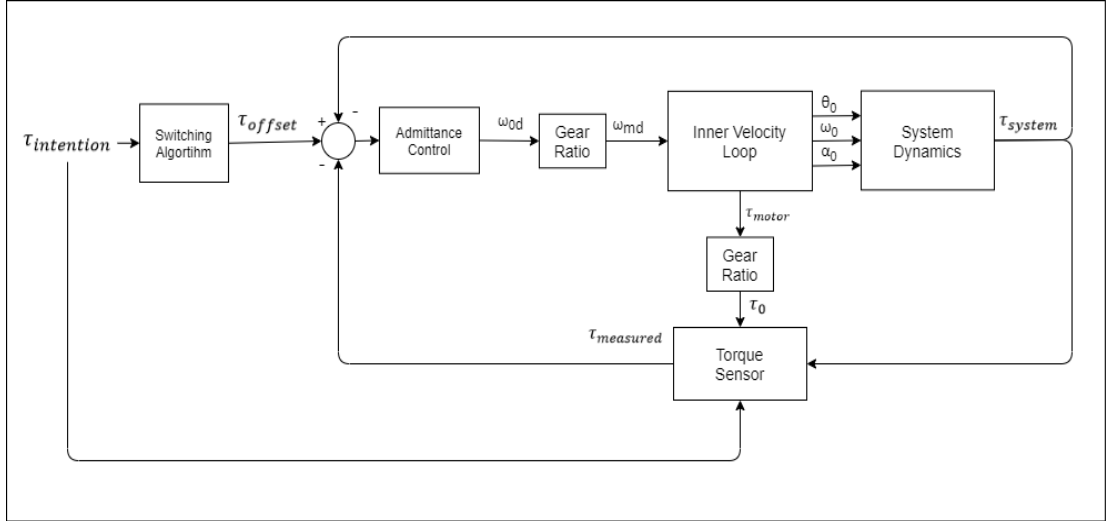
Figure 4.7 Control structure of the system.

In Figure 4.7, τ_{offset} represents offset torque on the system, $\tau_{measured}$ is measured torque by the sensor. Admittance control block accepts torque error and determines desired rotational velocity of application shaft, ω_{0d}. Corresponding motor velocity is represented by ω_{md}. According to this reference value, inner velocity control is executed and dynamic motion creates system torque, represented by τ_{system}.

Offset torque (τ_{offset}) is a predefined positive or negative value that is related with rehabilitation mode (assistive or resistive). It works together or against user that increase or decrease effort by changing torque error that is applied to admittance control block. Together with virtual dynamics of admittance control, it strongly affects the tendency of rehabilitation procedure. However, when no intention comes from user ($\tau_{intention} = 0$), offset torque should be zero, since it must not affect the torque error. For this reason, switching algorithm is driven where offset torque is created depending on existence of user torque intention.

4.2.1 Experimental environment and results

Single degree of freedom simple pendulum mechanism is constructed by adding a handle with end-weight to hold the system as seen in Figure 4.6. Two different case studies are studied with the system in terms of using admittance control and based on both of the case studies, distinct rehabilitation games are constructed. As an initial step, free move scenario is applied, in which there is no virtual spring parameter. The

system does not require any reference point so that motion is completely velocity based freely. Second case is a motion with virtual spring related. Here, the system has a reference point which is the starting position. Apart from virtual inertia and damping coefficients, effect of virtual spring is also included, as if there is a torsional spring that is connected at the end of motor shaft so that it tries to rotate back the system to reference point. Both cases are studied and admittance gains are applied as fixed and variable.

4.2.1.1 Free move structure

In this case study, motion does not have a reference point, instead, motion is activated freely in environment with the effect of virtual inertia and damping coefficient, which are affected by acceleration and velocity of the system, respectively. General control structure of applied Simulink system can be seen in Figure 4.8. Here, different blocks are used to construct the control structure and some of the variables are saved to be examined later.
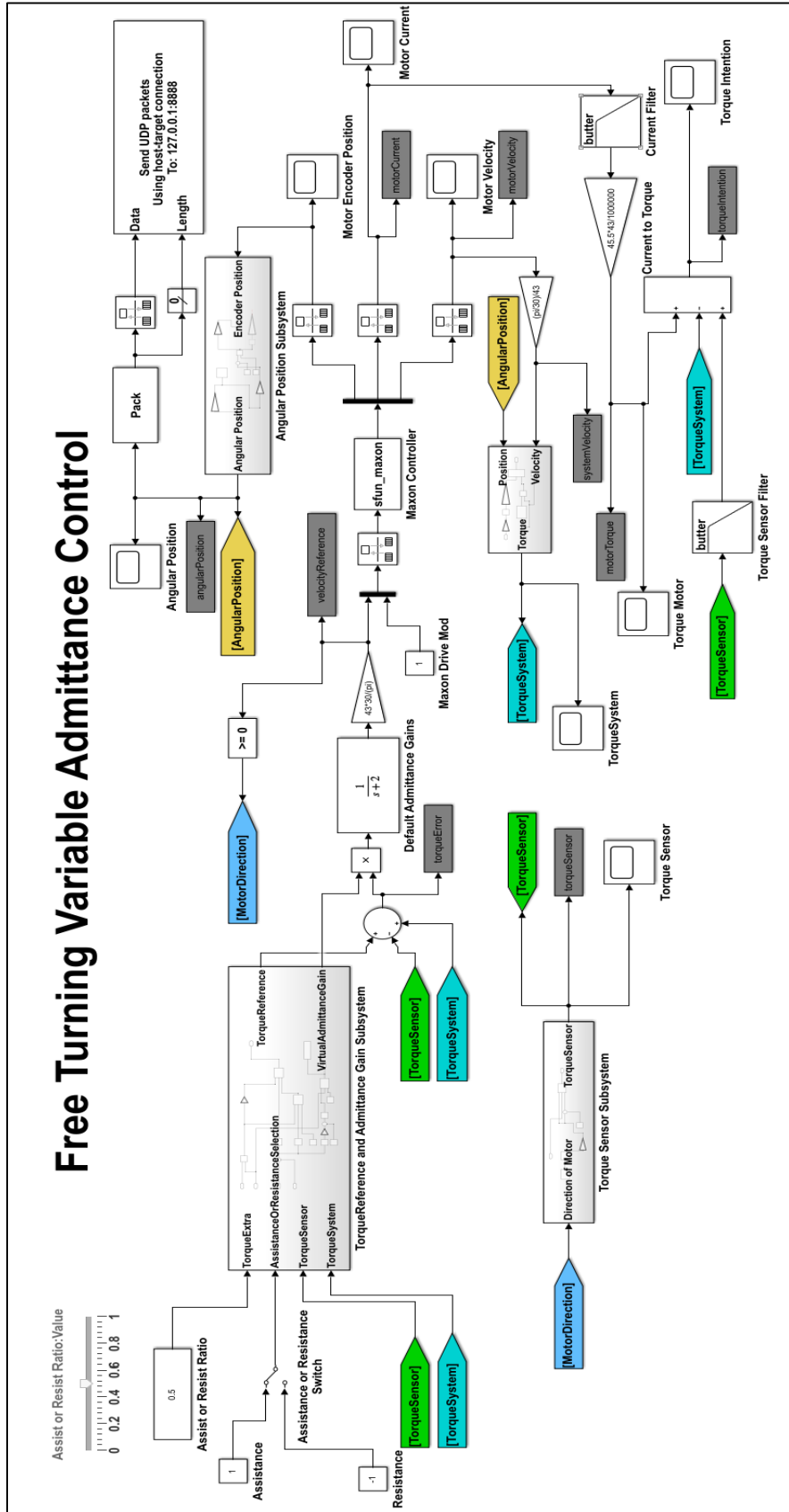
Figure 4.8 Free turning admittance control model in Simulink.

As mentioned in Sub-section 1.4, using variable admittance gains has some advantages over fixed gains. Thus, first variable approach is developed by considering the system as in action. Virtual gains of free move structure (virtual inertia and virtual damping coefficient) are bind to estimated torque intention of user. According to torque intention, virtual gains together are adjusted in order to achieve the requirements during motion. These requirements are strongly related with rehabilitation procedures in terms of assistance and resistance. Also, system performances are considered in choosing characteristic of virtual gains such as overshoot and settling time of the system. Virtual gains are linearly dependent on torque intention of user with minimum and maximum saturation points. Tuned characteristic of gains can be seen in Figure 4.9.
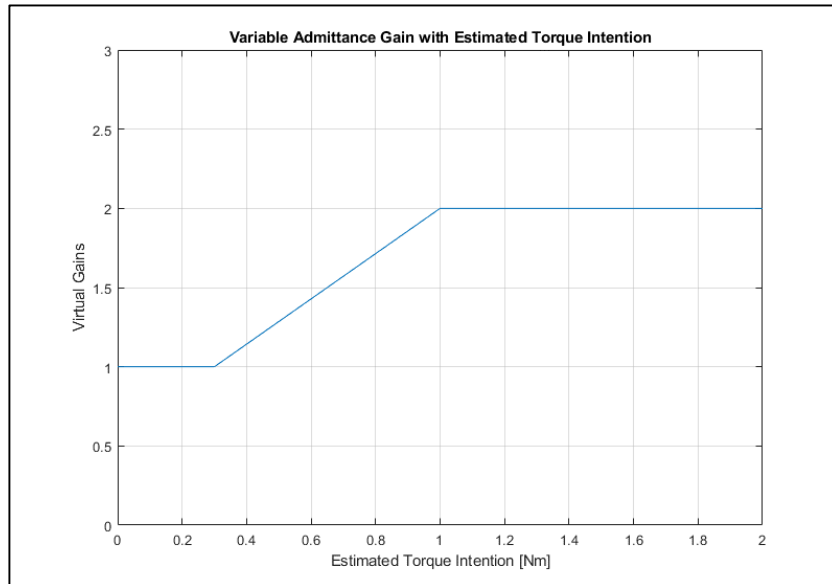
Figure 4.9 Variable virtual gain characteristic with respect to estimated torque intention.

As seen in Figure 4.9, when torque intention becomes higher, virtual admittance gains on the system becomes higher as well until the highest saturation point. This characteristic has actually negative effect on virtual parameters due to the inverse proportion. Therefore, if more intention is occurred, much easier to perform movement. This is important especially when assistance to user is needed, while stability is kept as well. When intention from user is cut, virtual gains become higher, so stopping effect on the system as well, especially due to additive amount of

damping coefficient, increase braking torque. This leads to stop motion immediately without overshoot, which is very important for system performance.

In this work, a novel method is applied to the system in terms of applying assistance and resistance apart from changing parameters of virtual admittance gains. Normally, when intention comes from user, the system tries to keep torque measurement from the torque sensor as zero. However, at the same time, this torque error is main input of admittance of the system, which has dynamic effects in it and outs reference velocity to inner velocity control loop. Admittance of the system should have at least a finite amount of dynamic effects due to stability reasons of the system. However, requirement of this system is to increase or decrease the amount of intention torque by overcoming system inertial effects such as inertia and friction. For this reason, in this work, offset torque value is altered according to amount of desired assistance or resistance. With this method, it is possible for a virtual constant torque to help or resist user in defeating admittance dynamic as well.

In experimental environment, one full counter-clockwise rotation motion from 270° (-90°) from x-axis is applied to the system with estimated amount of torque intention by changing the amount of assistance and resistance level and virtual variable admittance gains.
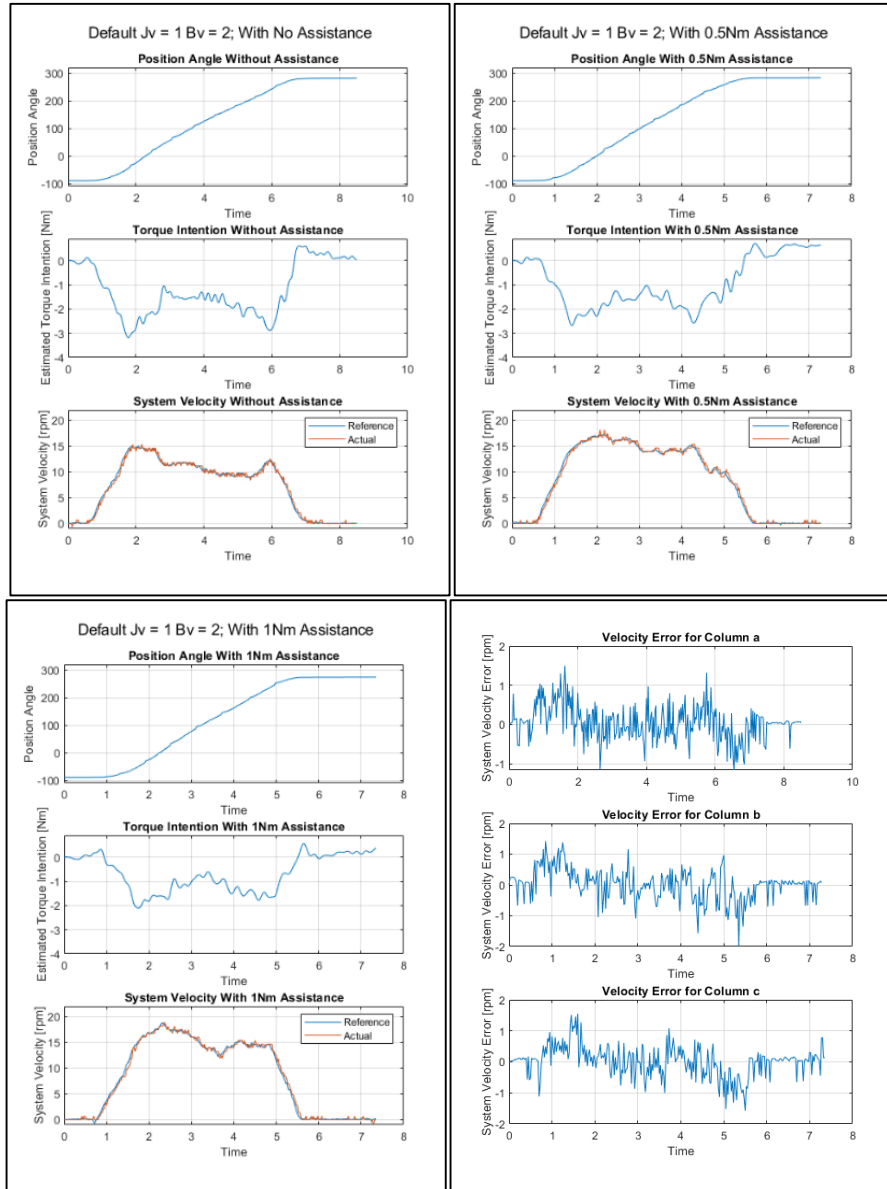
Figure 4.10 Free move full turn with J_v=1, B_v=2 and assistance.

First three columns of Figure 4.10 represent three different amount of offset torque assistance in full turn by using default virtual inertia of 1 kgm^2/rad and virtual damping coefficient of 2 Nms/rad. In first column, user gives approximately 2.5 Nm torque in full turn without extra offset torque assistance. Despite this amount of torque input, the system reaches to maximum velocity of 15 rpm in very short time interval so complete full turn takes about 7 seconds. In second column, user gives less effort, approximately 2 Nm in full turn. However, additionally 0.5Nm offset torque assistance is provided by control structure of the system, which results in faster rotating, about 15 rpm constantly. One full turn is completed in about 5.5

seconds. In third column, user intention is the smallest, below 2 Nm continuously. With high extra offset torque assistance from control system, 1 Nm, motion is the fastest one to be completed, with less effort. According to results, extra offset torque assistance gives opportunity to be used in active assistance mode in rehabilitation with different amount of assistance. Final column is the velocity error of the handle for three different cases in terms of Maxon Motor controller auto-tuned parameters. Since there is no sudden changes in reference values, the performance of the controller is sufficient, where instantaneous errors are not more than 1 rpm.
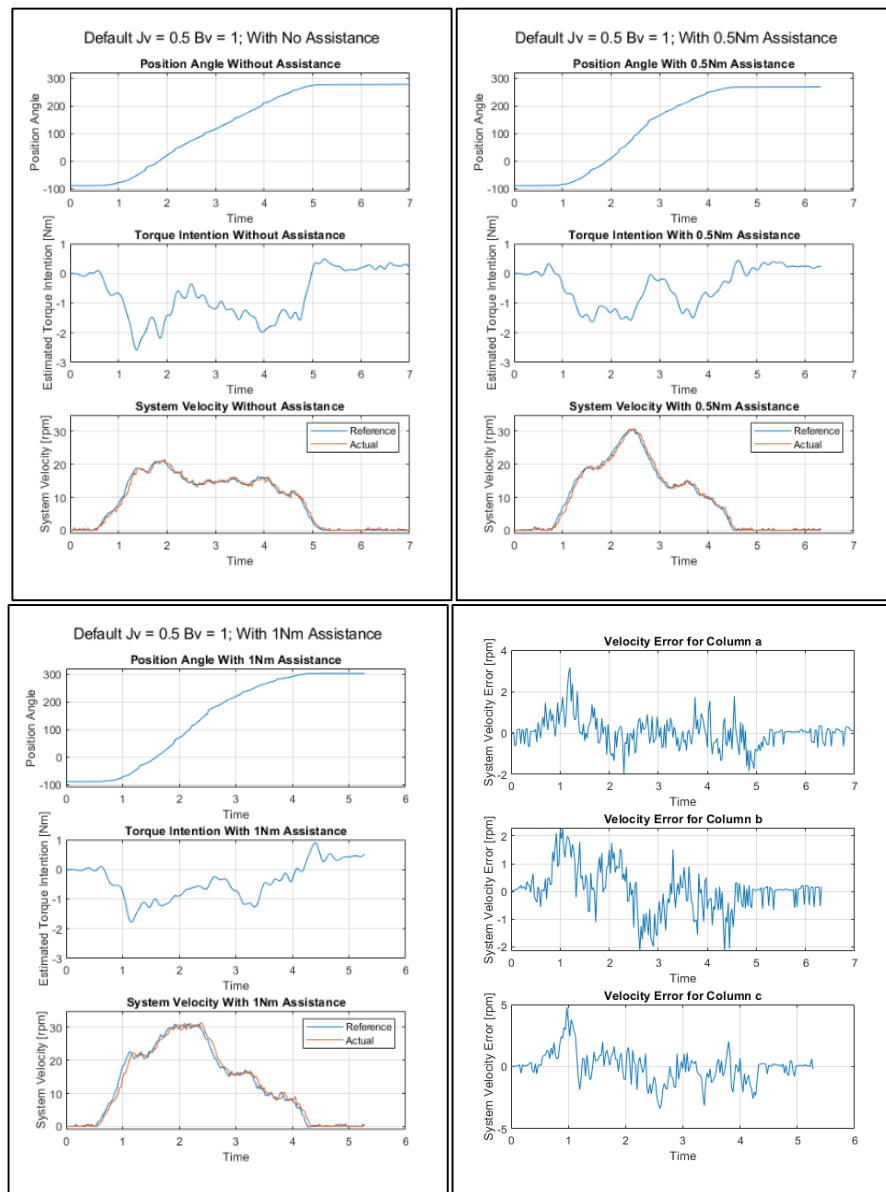
Figure 4.11 Free move full turn with J_v=0.5, B_v=1 and assistance.

In Figure 4.11, default virtual admittance coefficients are scaled down to 0.5 kgm^2/rad and 1 Nms/rad with keeping procedure as the same. As overall, it is clear that extra offset torque assistance makes the system to be achieved tiny amount of effort. In fact, as seen from third column, although intention is cut, motion is kept flowing due to fact that the virtual damping coefficient is very small. In forth column, it is possible to see that peak velocity errors are bigger due to sudden changes in reference value, which is output of admittance control. Despite this amount of peak errors, the system reduces errors in a tiny amount of time interval.
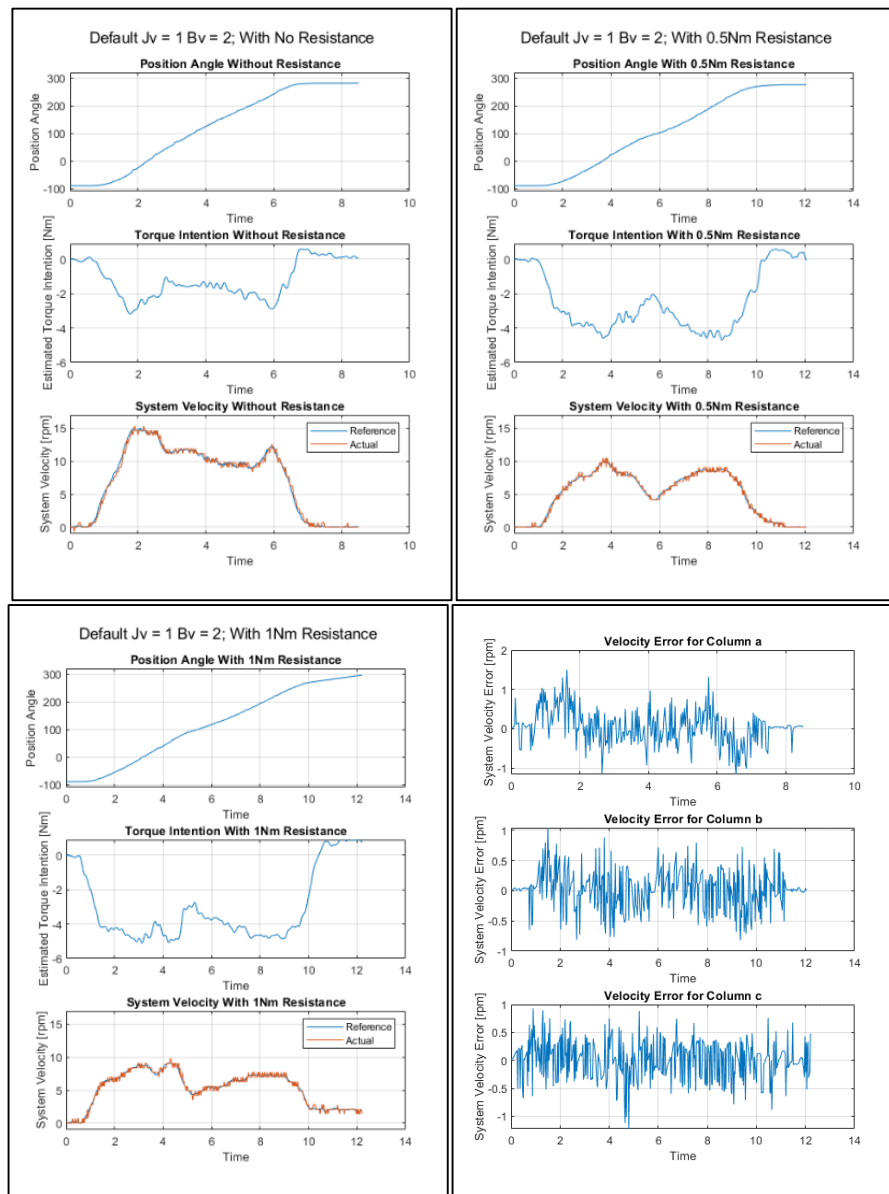
Figure 4.12 Free move full turn with J$_v$=1, B$_v$=2 and resistance.

In contrast to assistance applications, in Figure 4.12 user tries to complete a full turn with extra offset torque resistance and with default virtual coefficients; 1 kgm^2/rad and 2 Nms/rad. Results on first column stay for comparison. In second column, although user torque intention becomes higher, maximum velocity cannot exceed 10 rpm, so one full turn is completed within 10 seconds. In third column, user effort becomes relatively higher, resultant velocity becomes lower due to extra 1 Nm resistance. Here, it is clear that extra offset torque resistance affects user, which is very suitable for active resist mode. As in the first case study, peak errors are not high, below 1 rpm, due to the lack of sudden changes in reference value.
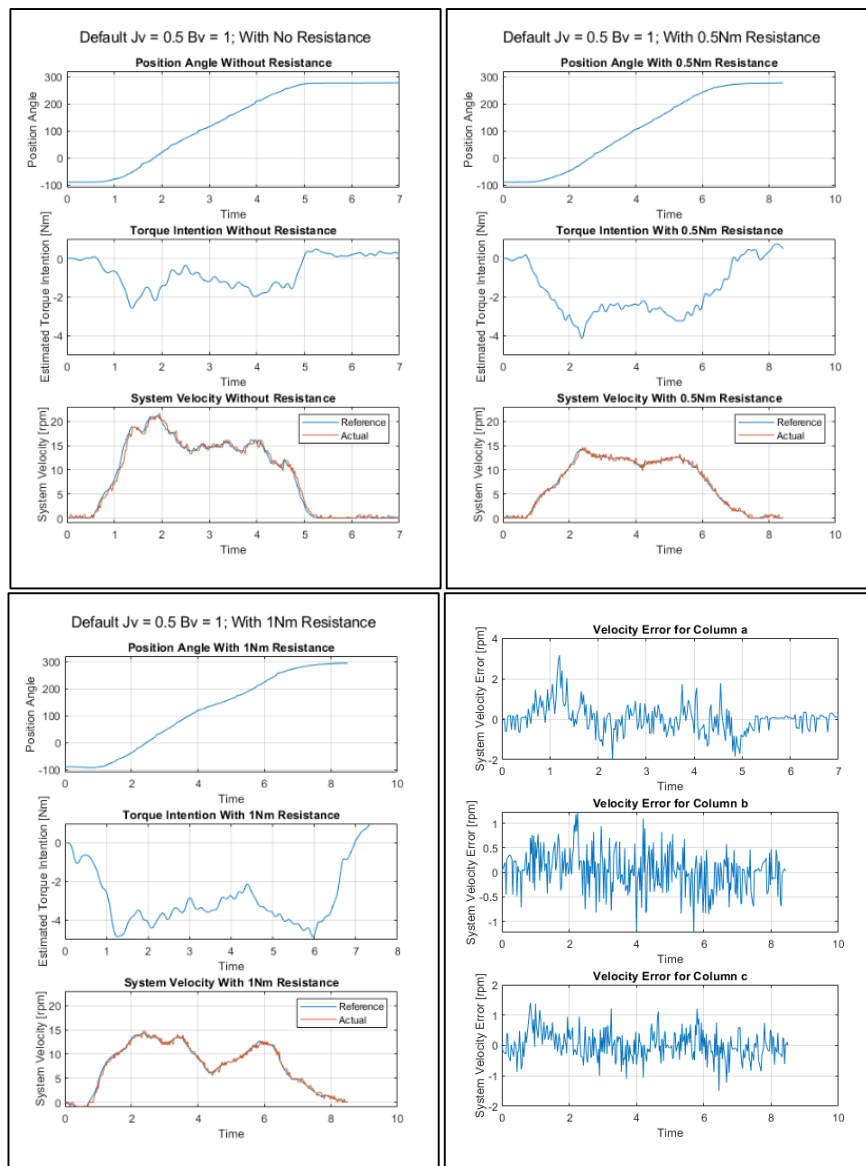
Figure 4.13 Free move full turn with J_v=0.5, B_v=1 and resistance.

As in assistance case, in Figure 4.13 default virtual admittance coefficients are scaled down to 0.5 kgm^2/rad and 1 Nms/rad. Again, first column is kept to comparison to extra resistance cases. In second and third column, more effort is needed to achieve one full turn due to extra offset torque resistance to the system. However, effect of virtual admittance gains is still important, so combination of extra offset torque resistance and virtual admittance gains variation can be applied for active resistance mode in rehabilitation procedures. In the last column, the velocity error values can be seen for each case study. Due to the tendency of overdamped characteristic of the system, peak errors are small.

Until now, all assistance and resistive procedures are applied with variable admittance control related with the rules at Sub-section 1.1. Effect of variable admittance gains is examined by comparing fixed admittance on the system.
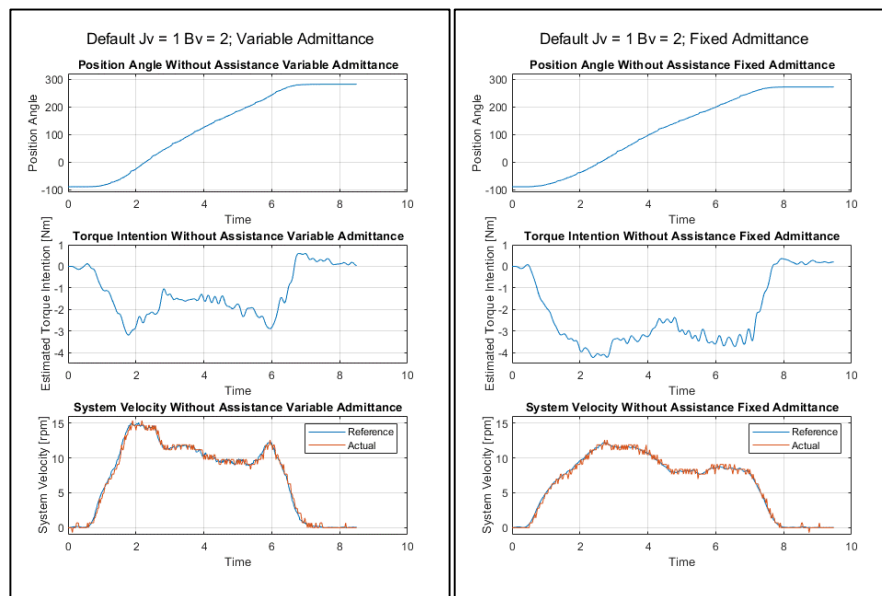
Figure 4.14 Free move full turn with variable and fixed admittance (J_v=1, B_v=2).

In Figure 4.14, procedure is similar to above, where user gives an effort to complete one full rotation. Estimated torque, resultant velocity and position is plotted in two columns that represent variable admittance gains with default inertia and damping coefficients and fixed admittance gains, respectively. The effect of virtual dynamics is clearly seen in second column where acceleration at starting and deceleration at stopping are small relative to first column. Also, during approximate constant

96

velocity area, more effort is needed in fixed admittance case, which is directly related with virtual coefficient.
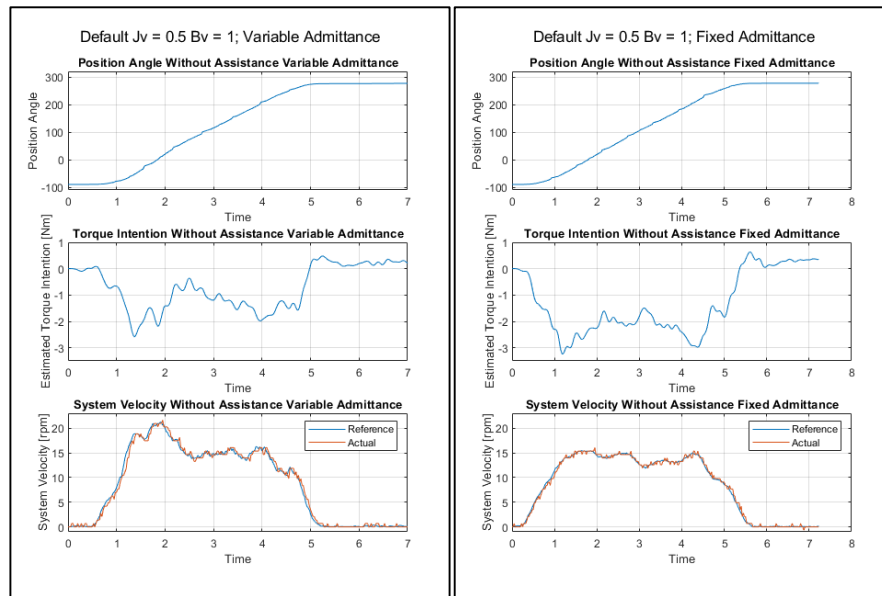
Figure 4.15 Free move full turn with variable and fixed admittance (J_v=0.5, B_v=1).

In Figure 4.15, default and fixed admittance coefficients are scaled down to 0.5 kgm^2/rad and 1 Nms/rad. Estimated torque intention becomes also smaller because of smaller dynamic parameters. As in previous case, in second column, fixed admittance control, acceleration and deceleration are high due to unchanged parameter values. However, the system is rotated more easily.
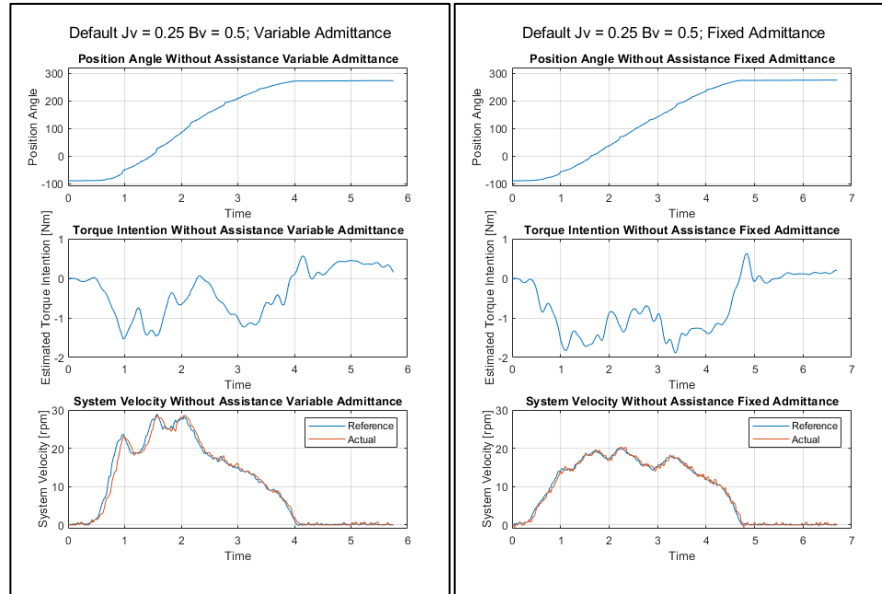
Figure 4.16 Free move full turn with variable and fixed admittance (J_v=0.25, B_v=0.5).

In Figure 4.16, virtual dynamic system parameters are altered to very low values; 0.25 kgm²/rad and 0.5 Nms/rad. In variable admittance case, once intention torque is increased, low admittance parameters are changed to smaller value and the system becomes very easy to drive, where there is almost no effort at all. Still, stability is kept for case. This attitude shows itself similar way in fixed admittance as well. Usage is still easy and there is no stability or overshoot problem. Results show both cases are suitable to be used in active assist mode, where variable admittance is more assistive than fixed admittance case.

Overall, system performance is better in variable admittance case. For fixed admittance gains, it is not always suitable for assistance applications where dynamic effects play huge role in every moment of movement.

4.2.1.2 Referenced rotation structure with virtual spring

In this case study, the motion has a reference point which has the effect of virtual spring. It means that there is an additive torque effect to the system, which is created by control structure according to spring coefficient of virtual admittance gains. Constructed model in Simulink can be seen in Figure 4.17.
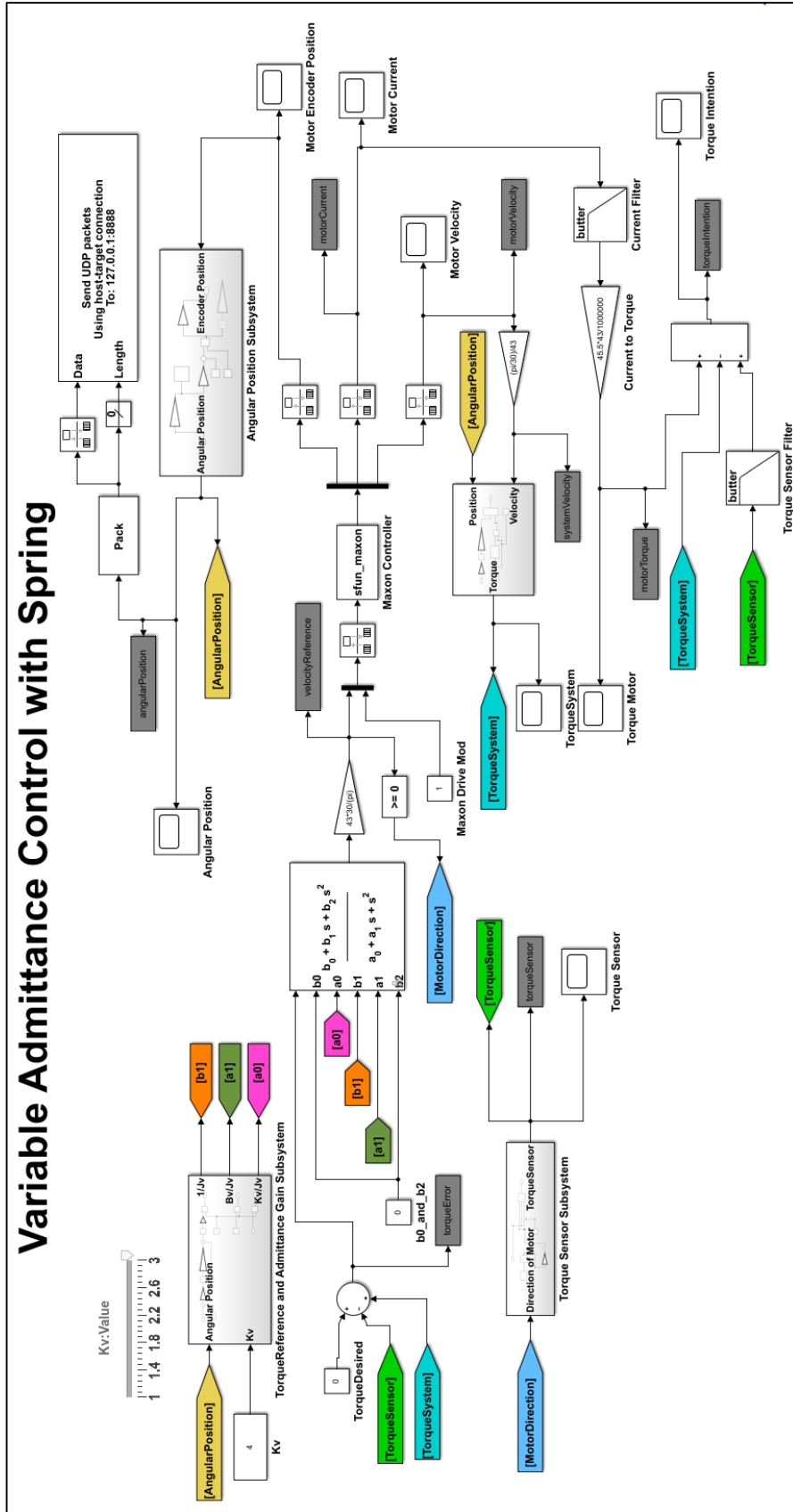
Figure 4.17 Spring move admittance control model in Simulink.

Admittance control structure is created with variable admittance gains, which has some advantages over fixed gains. In this work, virtual rotational inertia and virtual spring coefficient are kept constant, while damping coefficient is bind to position angle of the system. The reason behind this application is that when the system becomes distant from reference point by means of rotation, effect of spring coefficient should be dominant over damping coefficient and rotational inertia. Characteristic of variable admittance can be seen in Figure 4.18.
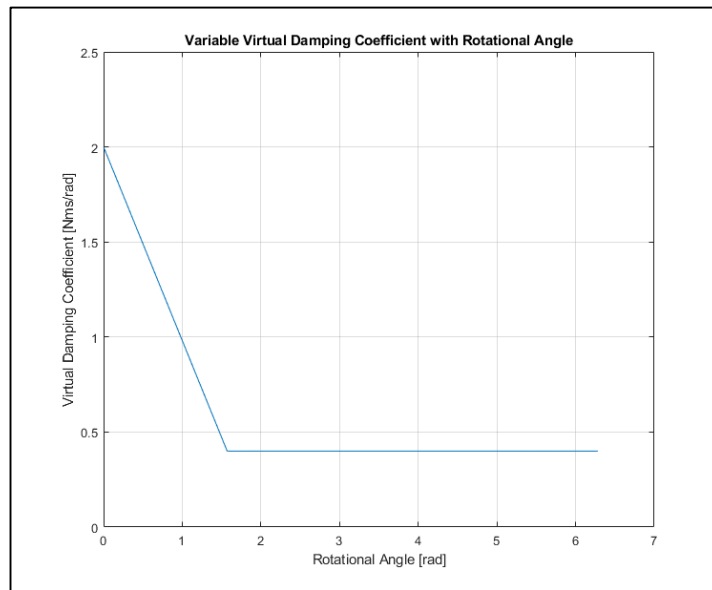
Figure 4.18 Variable virtual damping characteristic with respect to rotational angle.

As seen from Figure 4.18, when handle position is rotated away from reference point, $\theta = 0$, virtual damping coefficient becomes lower. As a result, effect of spring force becomes dominant.
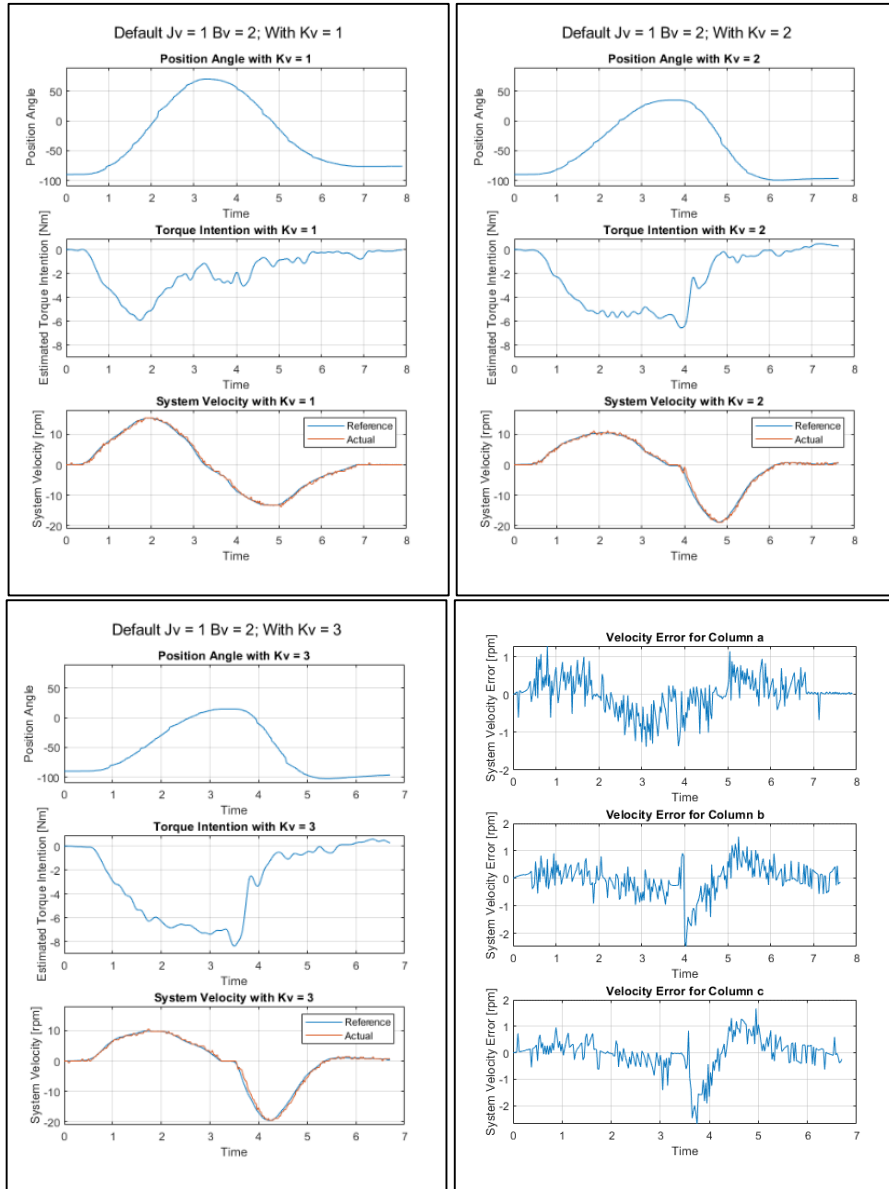
Figure 4.19 Spring move with J_v=1, B_v=2, K_v=1, K_v=2 and K_v=3.

In Figure 4.19, three different case studies are applied with the same default virtual inertia of 1 kgm^2/rad and virtual damping coefficient of 2 Nms/rad with increasing spring coefficients, 1 N/rad, 2 N/rad and 3 N/rad, respectively. In first three columns, approximately same amount of torque intention are applied to the system. In first column, motion is reached to approximately 170°, due to small amount of virtual spring coefficient. With the same effort, in second and third columns, motions are reached to 120° and 100°, respectively. Once torque intention cut suddenly, negative torque is applied to the system from the control structure. In third column, system

overshoots a bit due to high spring coefficient. As seen in last column, velocity errors are sufficient for overall controller performance.
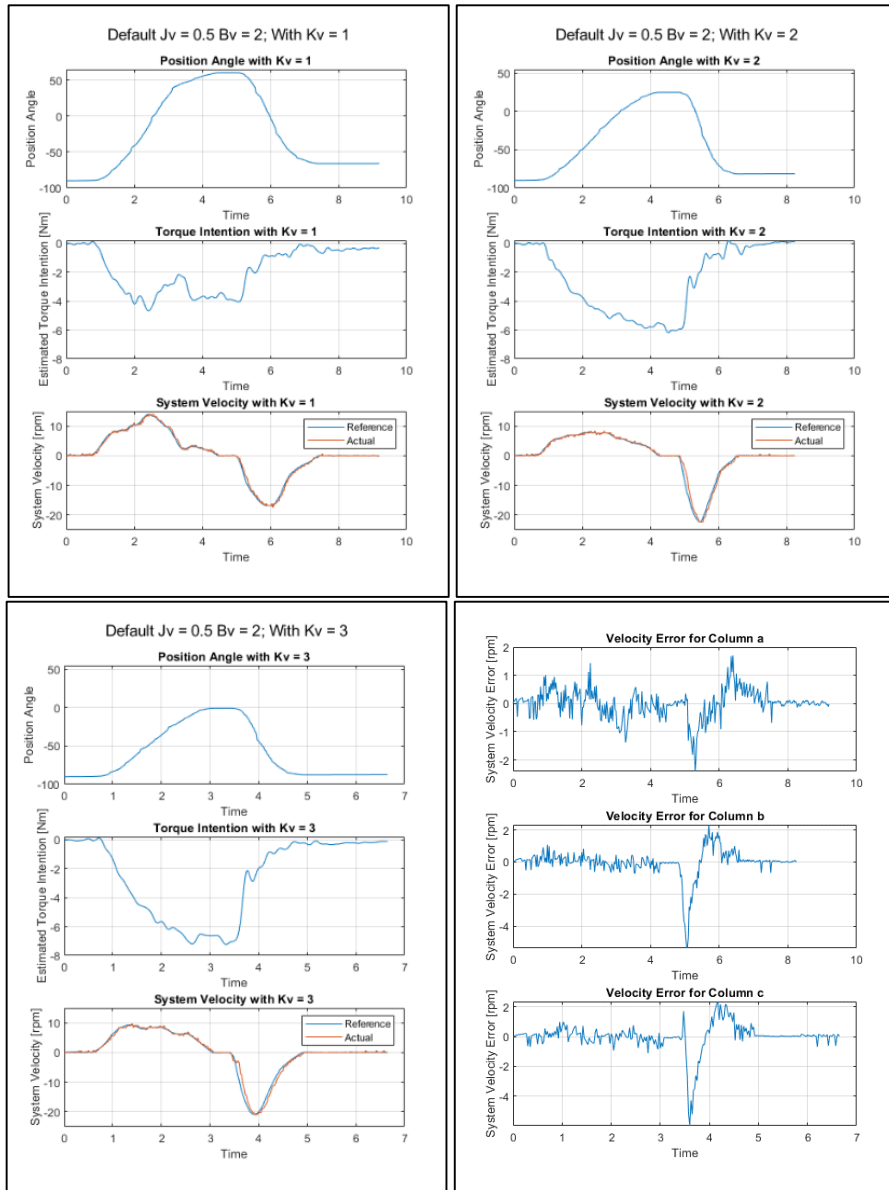
Figure 4.20 Spring move with J_v=0.5, B_v=2, K_v=1, K_v=2 and K_v=3.

In Figure 4.20, virtual inertia coefficient is shrank to 0.5 kgm^2/rad and the same procedure is applied. In this case, acceleration and deceleration becomes higher due to change on virtual inertia. In third column, small overshoot is also vanished and system performance becomes better. Sudden changes in reference value due to the spring effect, causes velocity errors to make peaks, eventually with small error overshoots, the overall errors are vanished.
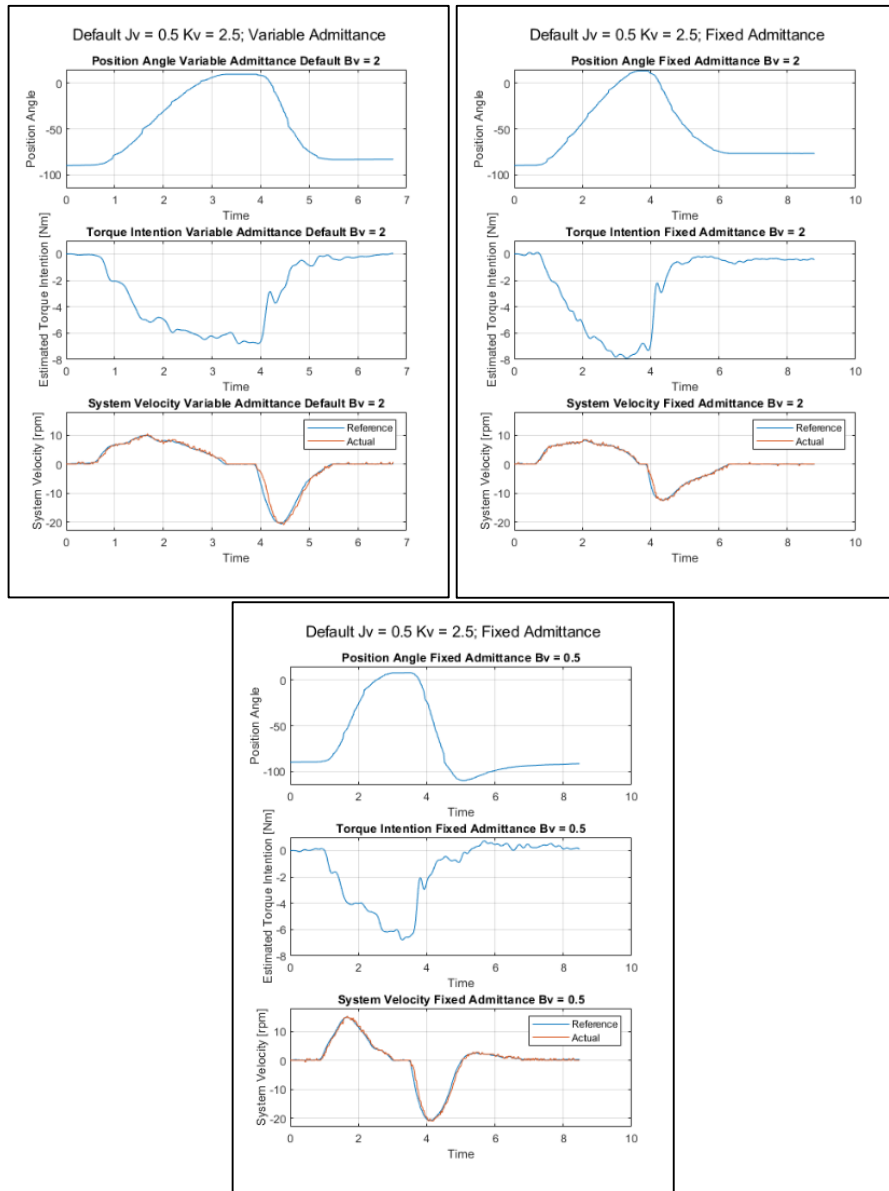
Figure 4.21 Spring move with variable and fixed admittance (J_v=0.5, K_v=2.5).

In Figure 4.21, variable and fixed admittance control structures are compared with two different fixed admittance coefficients, one of which is 2 Nms/rad, other is 0.5 Nms/rad. In second column, once torque is applied to the system, more effort is needed due to the fact that high damping coefficient. However, once torque intention is cut suddenly, due to fixed admittance gains, motion acts as over-damped, so system performance is bad. In third column, fixed damping coefficient is changed to 0.5 Nms/rad, which is relatively small, motion becomes under-damped so overshoot is seen in action. As a result, variable admittance control has positive effects over fixed admittance control in the referenced rotation structure with virtual spring.

4.3 Hand Rehabilitation Case Study

In Gezgin's work [26], hand rehabilitation setup is designed and manufactured in order to focus on hand rehabilitation process (Figure 4.22). In this thesis, this work is moved to one step further, where interaction force control is applied with help of the torque sensor and Maxon motor with needed mechanical parts. Interaction control is constructed as in previous case study, with variable admittance control structure, which assists and resists motion according to rehabilitation stage.
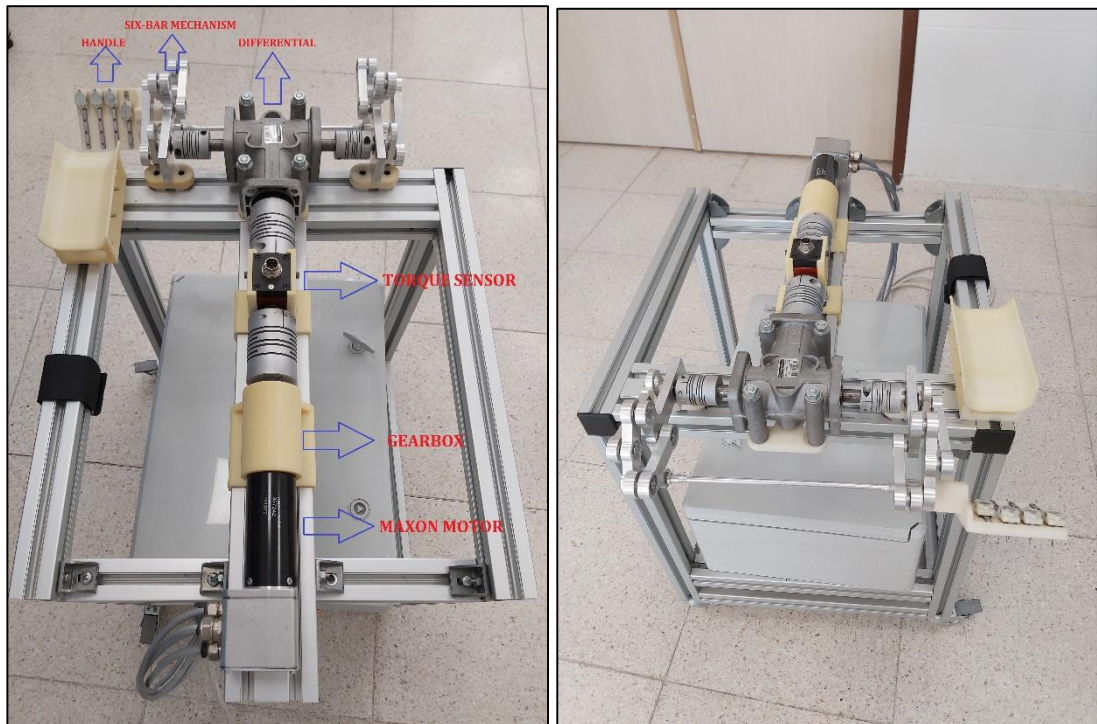
Figure 4.22 Hand Rehabilitation Mechanism.

In this case study, complexity of control strategy is increased due to the fact that six bar mechanism itself results in nonlinear attitude to the system. Accordingly, motion is affected dynamically in terms of velocity and acceleration. In rehabilitation application, output link that is affected by fingers is considered as main location for control part where it can be mentioned as handle. It is obvious that there is a motion ratio between output link and input link, which changes dynamically. Therefore, this motion ratio should be taken into control algorithm due to the fact that the actuator and the torque sensor is located in input part. This ratio can be determined

analytically with the kinematic motion analysis. Kinematic indication of six-bar mechanism on hand rehabilitation system can be seen in Figure 4.23.
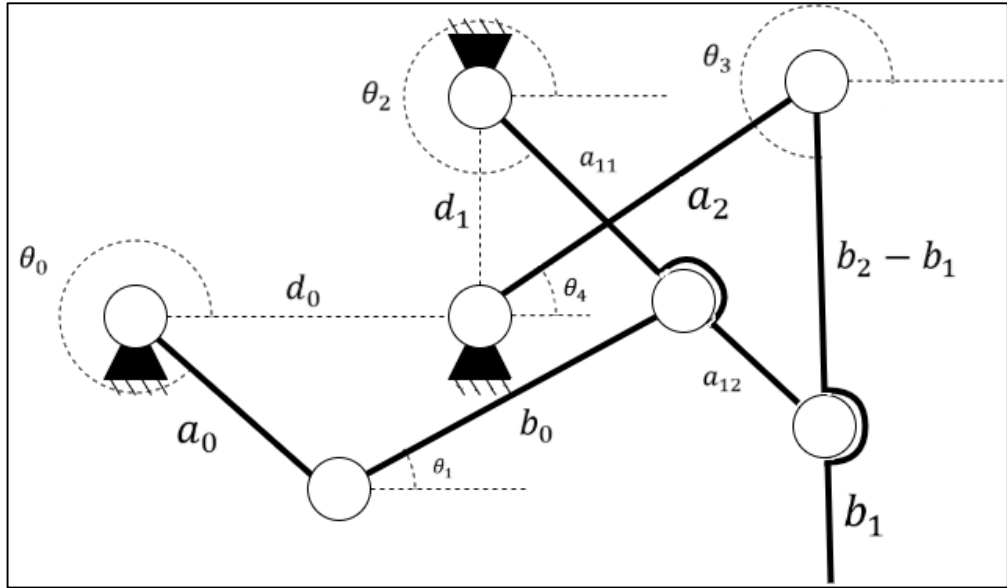
Figure 4.23 Six-bar kinematic structure of hand rehabilitation system.

The motion analysis starts with loop closure equations of six bar mechanism. There are two distinct loop closure equations, which are both considered independently.

$$a_0 e^{i\theta_0} + b_0 e^{i\theta_1} = d_0 e^{i0} + d_1 e^{i\pi/2} + a_{11} e^{i\theta_2} \tag{4.12}$$

$$d_1 e^{i\pi/2} + a_1 e^{i\theta_2} = a_2 e^{i\theta_4} + (b_2 - b_1) e^{i\theta_3} \tag{4.13}$$

The equation 4.12 and 4.13 represents loop 1 and loop 2 in exponential form, respectively.

$$a_0 \cos\theta_0 + b_0 \cos\theta_1 = d_0 + a_{11}\cos\theta_2 \tag{4.14}$$

$$a_0 \sin\theta_0 + b_0 \sin\theta_1 = d_1 + a_{11}\sin\theta_2 \tag{4.15}$$

Equations are x and y components of loop 1. Next, θ_1 is vanished in order to determine θ_2.

$$\left(\frac{d_0^2 + d_1^2 + a_{11}^2 + a_0^2 - b_0^2}{2}\right) + d_0 a_{11}\cos\theta_2 - d_0 a_0\cos\theta_0 - a_0 a_{11}\cos\theta_0\cos\theta_2 +$$
$$d_1 a_{11}\sin\theta_2 - d_1 a_0\sin\theta_0 - a_0 a_{11}\sin\theta_0\sin\theta_2 = 0 \tag{4.16}$$

$$[d_0 a_{11} - a_0 a_{11} \cos \theta_0] \cos \theta_2 + [d_1 a_{11} - a_0 a_{11} \sin \theta_0] \sin \theta_2 =$$
$$-\left(\frac{d_0^2 + d_1^2 + a_{11}^2 + a_0^2 - b_0^2}{2}\right) + d_0 a_0 \cos \theta_0 + d_1 a_0 \sin \theta_0 \tag{4.17}$$

In order to make calculation simpler, let;

$$K_1 = d_0 a_{11} - a_0 a_{11} \cos \theta_0 \tag{4.18}$$

$$K_2 = d_1 a_{11} - a_0 a_{11} \sin \theta_0 \tag{4.19}$$

$$K_3 = -\left(\frac{d_0^2 + d_1^2 + a_{11}^2 + a_0^2 - b_0^2}{2}\right) + d_0 a_0 \cos \theta_0 + d_1 a_0 \sin \theta_0 \tag{4.20}$$

Then the system is reduced to equation 4.21.

$$K_1 \cos \theta_2 + K_2 \sin \theta_2 = K_3 \tag{4.21}$$

Since K_1 and K_2 are dependent to each other, dependency can be shown as in equation 4.22 and 4.23.

$$K_1 = D \cos \varphi \tag{4.22}$$

$$K_2 = D \sin \varphi \tag{4.23}$$

where, D and φ are arbitrary dependent variables. These variables can be determined with equations below.

$$\varphi = atan2(K_2, K_1) \tag{4.24}$$

$$D = \sqrt{K_1^2 + K_2^2} \tag{4.25}$$

$$\cos \varphi \cos \theta_2 + \sin \varphi \sin \theta_2 = \frac{K_3}{D} \tag{4.26}$$

Then, two different solution for θ_2 is presented as;

$$\theta_2 = \pm acos\left(\frac{K_3}{D}\right) + \varphi \tag{4.27}$$

From loop equations with known θ_2, θ_1 can be determined as well.

$$\theta_1 = atan2(d_1 + a_{11} \sin \theta_2 - a_0 \sin \theta_0 , d_0 + a_{11} \cos \theta_2 - a_0 \cos \theta_0) \quad (4.28)$$

By considering the same procedure applied above for loop 1, unknown variables of loop 2 can be determined as well based on loop equation 4.13.

$$\theta_3 = \pm acos \left({K_6}/{E} \right) + \beta \qquad (4.29)$$

where,

$$\beta = atan2(K_5, K_4) \qquad (4.30)$$

$$K_4 = a_1(b_2 - b_1) \cos \theta_2 \qquad (4.31)$$

$$K_5 = d_1(b_2 - b_1) + a_1(b_2 - b_1) \sin \theta_2 \qquad (4.32)$$

$$E = \sqrt{K_4^2 + K_5^2} \qquad (4.33)$$

$$K_6 = \left(\frac{d_1^2 + a_1^2 + (b_2 - b_1)^2 - a_2^2}{2} \right) + d_1 a_1 \sin \theta_2 \qquad (4.34)$$

With known θ_3, it is possible to determine θ_4 from closure loop equation.

$$\theta_4 = atan2(d_1 + a_1 \sin \theta_2 - (b_2 - b_1) \sin \theta_3 , a_1 \cos \theta_2 - (b_2 - b_1) \cos \theta_3)(4.35)$$

Position angles of each joint of six-bar mechanism with given input angle of θ_0 are determined with two different loop closure equations above. Transmission ratio in terms of velocity should be found as well in order to take ratio into the consideration in admittance control algorithm. For this reason, velocity analysis is done for two distinct loops considered above. Analysis starts with time derivatives of position loop equations.

$$-b_0\omega_1 \sin \theta_1 = -a_{11}\omega_2 \sin \theta_2 + a_0\omega_0 \sin \theta_0 \qquad (4.36)$$

$$b_0\omega_1 \cos \theta_1 = a_{11}\omega_2 \cos \theta_2 - a_0\omega_0 \cos \theta_0 \qquad (4.37)$$

Equation system can be taken into the matrix form as below.

$$\begin{bmatrix} a_0 \sin \theta_0 \\ -a_0 \cos \theta_0 \end{bmatrix} \{\omega_0\} = \begin{bmatrix} -b_0 \sin \theta_1 & a_{11} \sin \theta_2 \\ b_0 \cos \theta_1 & -a_{11} \cos \theta_2 \end{bmatrix} \begin{bmatrix} \omega_1 \\ \omega_2 \end{bmatrix} \qquad (4.38)$$

For loop 1, rotational velocities can be determined analytically as below.

$$\omega_1 = \frac{a_0 \sin(\theta_2 - \theta_0)}{b_0 \sin(\theta_1 - \theta_2)} \omega_0 \tag{4.39}$$

$$\omega_2 = \frac{a_0 \sin(\theta_1 - \theta_0)}{a_{11} \sin(\theta_1 - \theta_2)} \omega_0 \tag{4.40}$$

The same procedure must be applied to loop 2 in order to determine unknown rotational velocities.

$$-a_1 \omega_2 \sin \theta_2 = -a_2 \omega_4 \sin \theta_4 - (b_2 - b_1) \omega_3 \sin \theta_3 \tag{4.41}$$

$$a_1 \omega_2 \cos \theta_2 = a_2 \omega_4 \cos \theta_4 + (b_2 - b_1) \omega_3 \cos \theta_3 \tag{4.42}$$

$$\begin{bmatrix} -a_1 \sin \theta_2 \\ a_1 \cos \theta_2 \end{bmatrix} \{\omega_2\} = \begin{bmatrix} -(b_2 - b_1) \sin \theta_3 & -a_2 \sin \theta_4 \\ (b_2 - b_1) \cos \theta_3 & a_2 \cos \theta_4 \end{bmatrix} \begin{bmatrix} \omega_3 \\ \omega_4 \end{bmatrix} \tag{4.43}$$

For loop 2, rotational velocities can be determined analytically as below.

$$\omega_3 = \frac{a_1 \sin(\theta_4 - \theta_2)}{(b_2 - b_1) \sin(\theta_4 - \theta_3)} \omega_2 \tag{4.44}$$

$$\omega_4 = \frac{a_1 \sin(\theta_2 - \theta_3)}{a_2 \sin(\theta_4 - \theta_3)} \omega_2 \tag{4.45}$$

Compared to applied forces and torques from user and the electric motor on the system, link lengths and inertias of six-bar mechanism is small, thus in torque transmission, only velocity ratios should be considered as in virtual work method.

$$\tau_3 \omega_3 = \tau_0 \omega_0 \tag{4.46}$$

$$\tau_3 = \frac{\tau_0 \omega_0}{\omega_3} \tag{4.47}$$

As seen from equation 4.47, in order to transform torque value from the torque sensor to applied point, proportional rotational velocity ratio of shafts is needed, which are actually found analytically above.

$$\frac{\omega_0}{\omega_3} = \frac{a_{11}(b_2 - b_1) \sin(\theta_1 - \theta_2) \sin(\theta_4 - \theta_3)}{a_0 a_1 \sin(\theta_1 - \theta_0) \sin(\theta_4 - \theta_2)} \tag{4.48}$$

In this case study with six-bar mechanism of the hand rehabilitation setup, control signal as torque measurement is transformed on application handle shaft with virtual work method. Thus, result of admittance control is in level of considered handle shaft as well, which is desired rotational velocity on application handle shaft. However, due to the transmission between the electric motor and the output shaft, desired rotational velocity of the electric motor must be determined in real time in order to create desired effect on application handle shaft, whose index in motion analysis is 3 (ω_3).

Similar to simplified mechanism, overall hand rehabilitation system is controlled with admittance control method as well, where there are two different loops. Outer loop considers indirect force control type (admittance control) and inner loop considers motion control as referenced with output of admittance control (Figure 4.24).
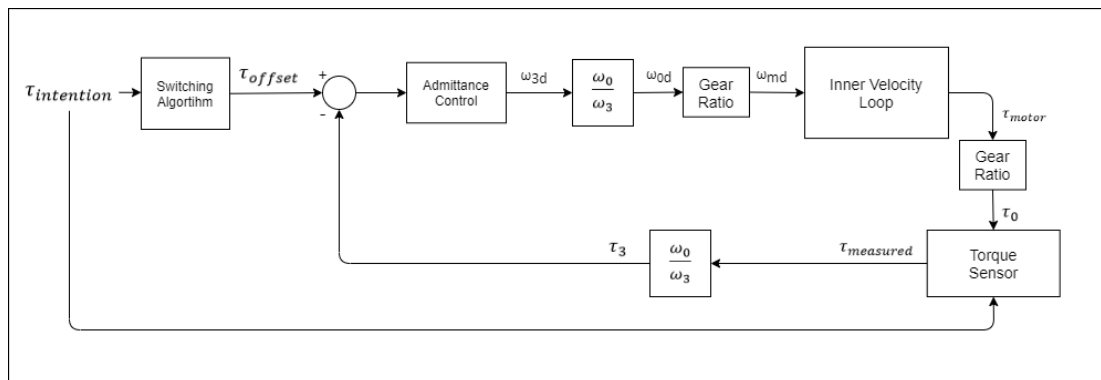
Figure 4.24 Control structure of the system.

In Figure 4.24, τ_{offset} represents offset torque on the system, $\tau_{measured}$ is torque measured with the torque sensor, τ_3 is determined torque on application handle shaft. Admittance control block accepts torque error and determines desired rotational velocity, in this case it is desired velocity on application handle shaft that is represented by ω_{3d}. As discussed above, velocity ratio with gear ratio is for determining desired motor velocity, ω_{md}. According to this reference value, inner velocity control is executed.

With the same purpose of the simplified mechanism, offset torque (τ_{offset}) is defined according to type of rehabilitation mode; assistive or resistive. The same switching method is applied to this structure as well due to the need in not affecting torque error when user intention torque is zero ($\tau_{intention} = 0$).

4.3.1 Experimental environment and results

Single degree of freedom six-bar mechanism for hand rehabilitation purposes is constructed with mechanism with mechanical components, electric motor with needed electronic components and the torque sensor as seen in Figure 4.22. In order to tune admittance control for hand rehabilitation system, two different case studies are applied as in the previous simplified mechanism part, wrist and forearm rehabilitation setup. Case studies are based on free move and virtual spring related. In both cases, variable and fixed admittance control structures are applied and compared. Main target is to focus on only finger extension, where user tries to extend his/her finger from reference starting point. Starting point is one end of the mechanism can reach. According to rehabilitation type, it is required to extend fingers as much as possible with assistance or resistance.

Actuator system has an embedded encoder in order to have position feedback in usage. However, this encoder is an incremental type of encoder as commonly. It means that once the system is powered up, encoder starts to count from zero position. In order to get rid of this kind of problem, the system should have a reference point, on which motion as well as angles should be depended. In this system, it is chosen as position where crank angle is 180° ($\theta_0 = 180°$) with respect to horizontal line (x-axis) due to the fact that it is convenient to take crank to this position manually.
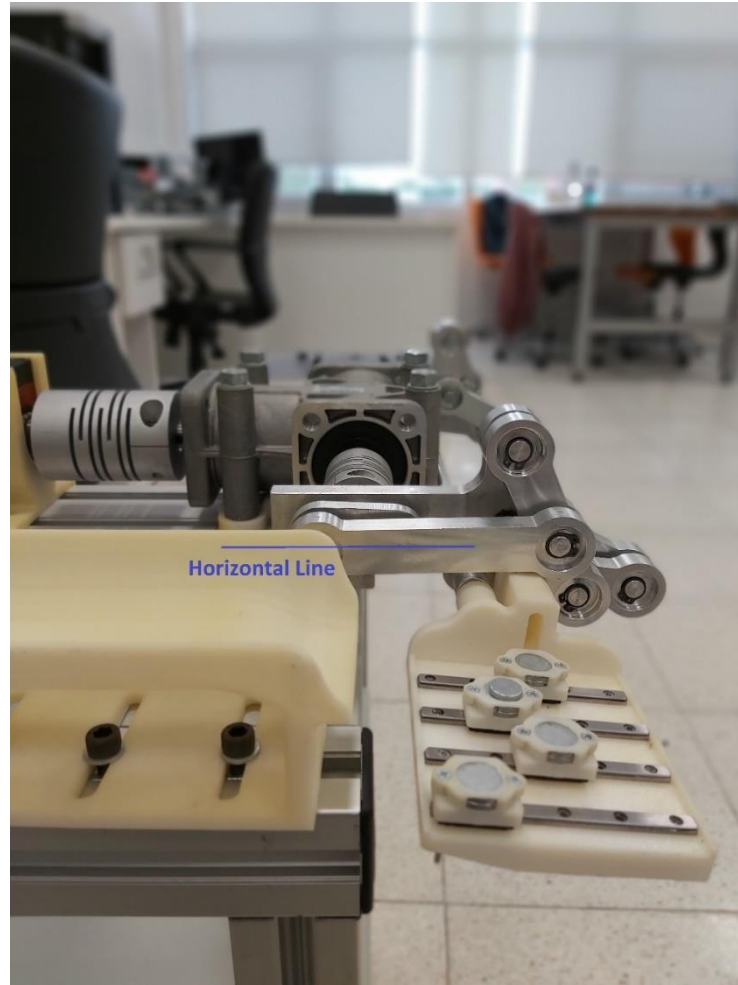
Figure 4.25 Six-bar mechanism with 180° crank angle (θ_0).

As seen from Figure 4.25, the system should be powered on, when the crank is in position as it is, which is known as the crank angle. However, due to the closeness to singularity point of six-bar mechanism, it should not be exact position starting position of the handle. In order to get rid of singularity point, there should be some safe zone, where torque transfer ratio is in reasonable levels. When crank angle is 140° with respect to horizontal axis, torque ratio becomes 0.9586, which is quite reasonable and handle orientation with respect to horizontal line becomes 194° (Figure 4.26), due to the fact that the crank and the handle positions are dependent to each other. Thus, it is very suitable angle to be starting position.
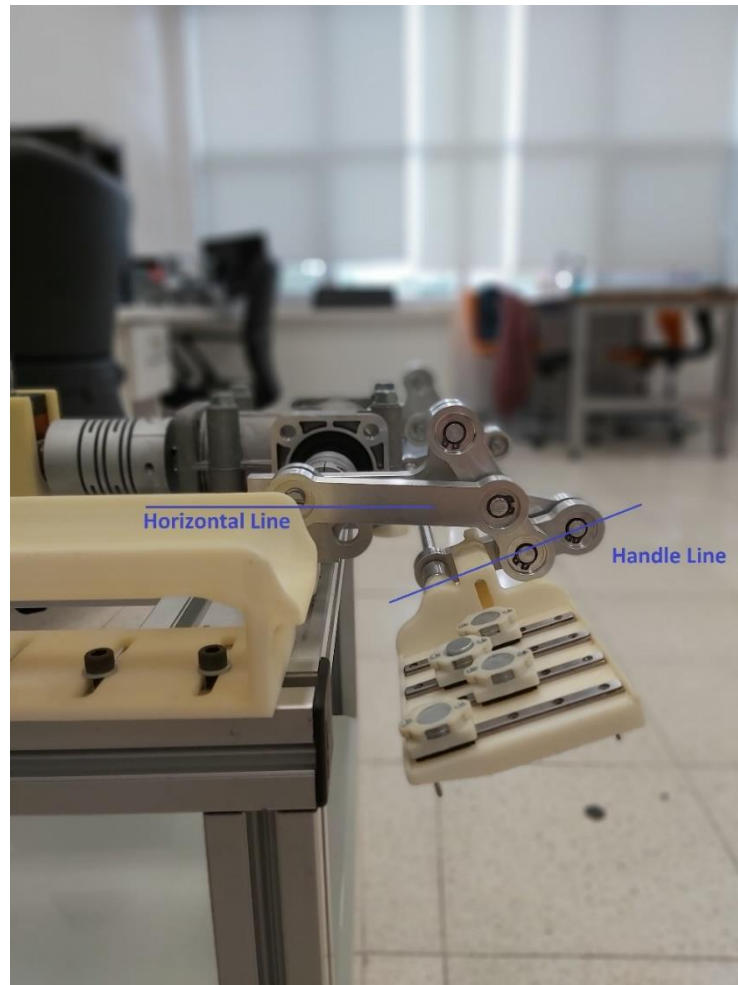
Figure 4.26 Six-bar mechanism with 140° crank angle (θ_0).

For positioning the crank to exact starting location, position controlled Simulink model is constructed as in Figure 4.27, where drive mode of Maxon motor and required position of encoder is given to s-function of Maxon Controller block.
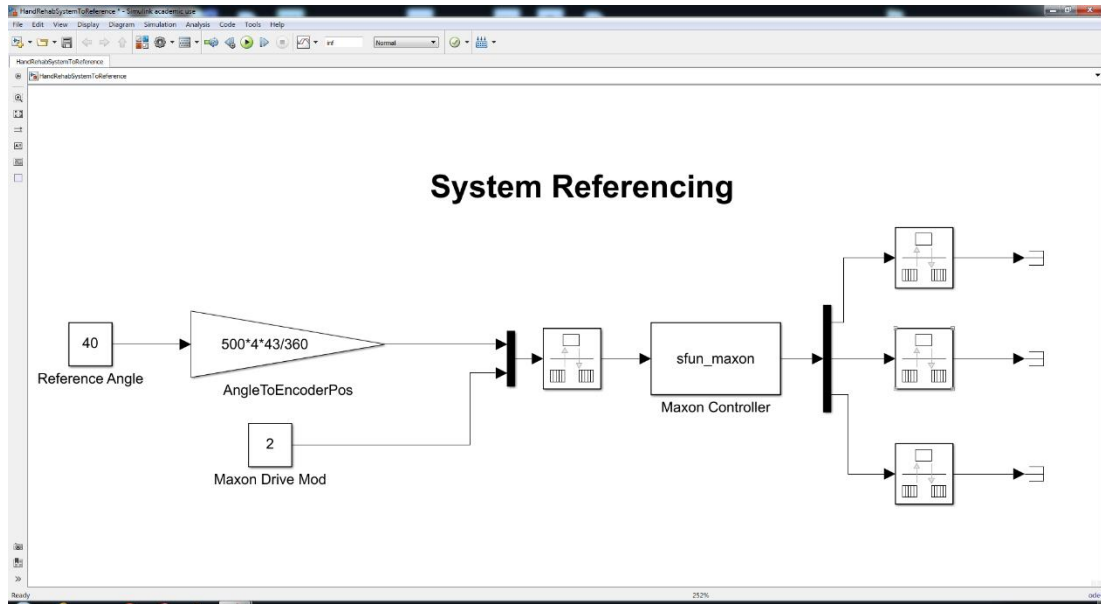
Figure 4.27 Position control model in Simulink.

After driving motor via Simulink position control model, the crank is positioned so that the handle itself as needed. It can be concluded that motion is arranged to be starting point from now on.

4.3.1.1 Free move structure

In this case study, movement is carried out freely, without any reference point. According to control type, user tries to fulfil movement under assistance or resistance. Apart from additional offset assistance or resistance, the system is also affected by virtual inertia and damping coefficients. Control system is constructed in Simulink environment and different tuning parameters are applied (Figure 4.28).

Figure 4.28 Free turning admittance control model in Simulink.

As mentioned and applied in the wrist and forearm rehabilitation setup (simplified mechanism) case study, variable admittance gain has considerable advantages, thus the same structure is constructed in this hand rehabilitation case study as well. In this free move structure admittance gains are bind to torque intention that is affected by user, however, it must be noted that there is torque transmission ratio between the application handle shaft and the motor shaft. Tuned virtual gain characteristics are determined as in Figure 4.29.
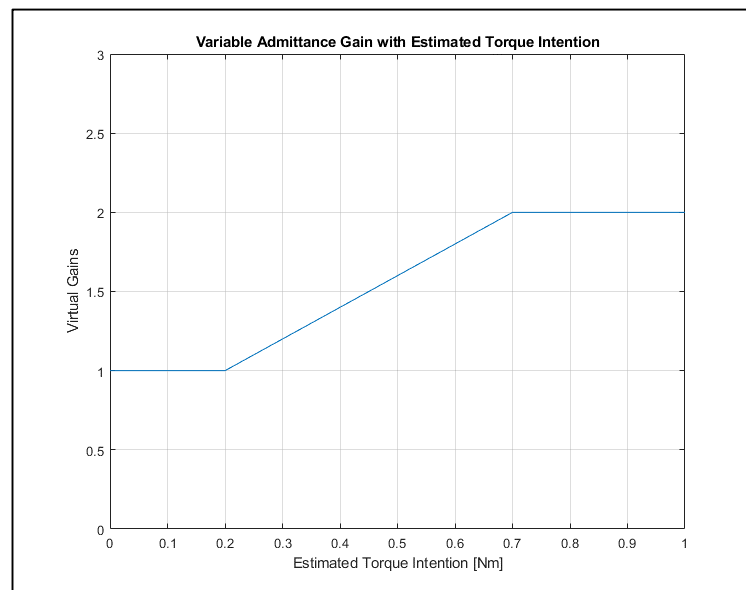
Figure 4.29 Variable virtual gain characteristic with respect to estimated torque intention.

Since virtual gain is relatively small at stable undisturbed position, which actually means that higher virtual inertia and damping coefficient, this variable gain characteristic helps to keep stability when there is small or no user intention. When user tries to drive the system, needed effort becomes smaller due to decreasing in virtual inertia and damping coefficients. As in the previous cases, the system performance is affected positively as well.

As in the previous cases, novel method that is based on extra assistance or resistance torque that gets into the system as offset torque is applied in this case as well. With presence of these extra effects, user intention is directly affected in terms of rehabilitation procedures as well as admittance dynamics as well. For instance, in

such cases where user requires to be fully supported, admittance dynamics should also be got over.

By applying this kind of control structure, the system is driven by healthy user in order to be tuned so that parameter stage can be predicted. From starting point that is determined as crank angle $\theta_0 = 140°$, this healthy user tries to extend his/her fingers under different offset amounts of assistance, resistance level and the virtual variable admittance gains.
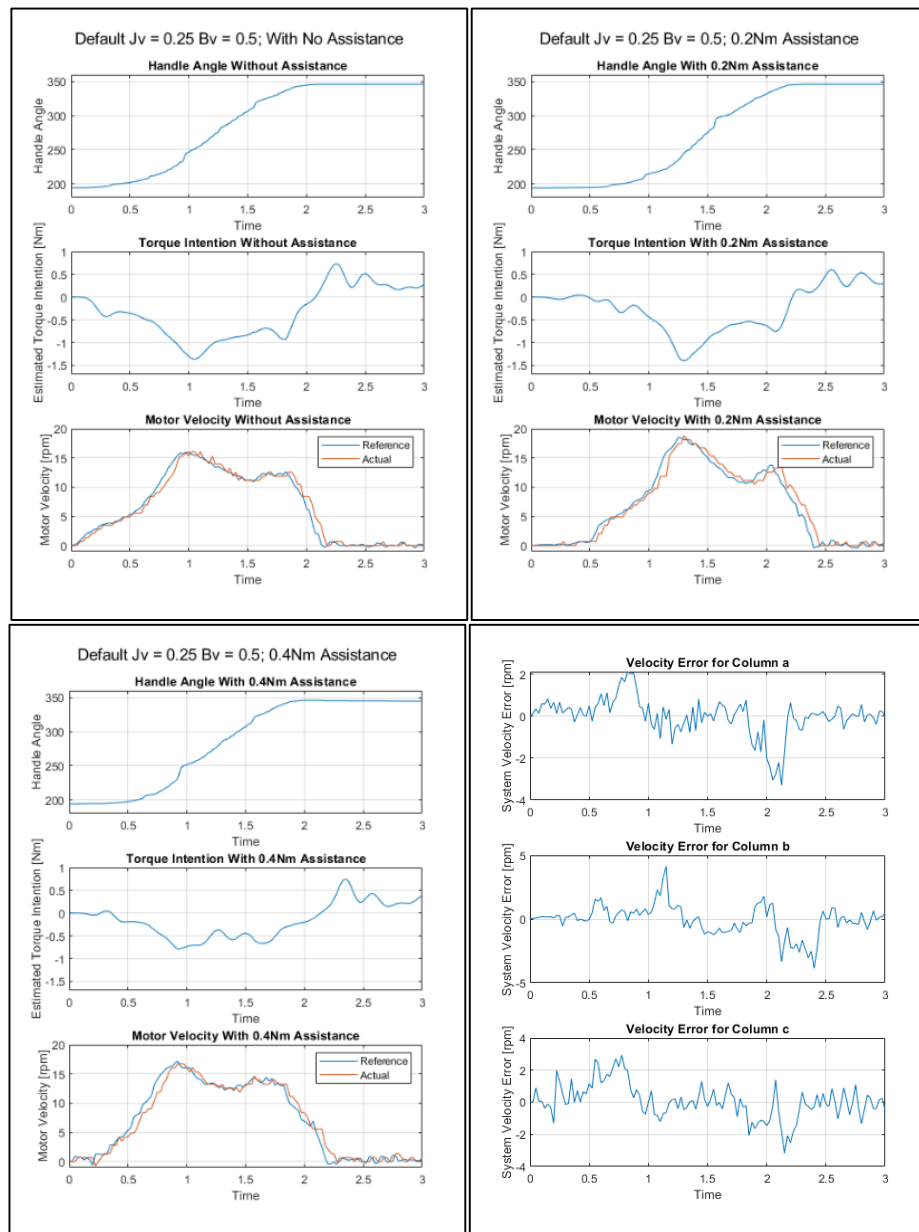
Figure 4.30 Free move finger extension with $J_v=0.25$, $B_v=0.5$ and assistance.

Three different columns of Figure 4.30 represents three different amount of offset torque assistance at handle; 0 Nm, 0.2 Nm and 0.4 Nm with default virtual inertia of 0.25 kgm^2/rad and virtual damping coefficient of 0.5 Nms/rad. In first column, there is no offset assistance in control model, thus torque intention is higher than other two columns as expected. Motor velocity cannot reach to 20 rpm, with a peak intention of 1 Nm. With user's finger extension, the handle is moved around 150°. In second column, with lower effort from user, motor velocity reaches just below 20 rpm, thanks to offset assistance control structure. In third column, the lowest effort is presented, where it does not reach to 1 Nm, still, it is the fastest experiment due to 0.4 Nm offset assistance. In the last column, performance of Maxon Motor auto-tuned controller in terms of velocity can be seen for each case. As seen in reference velocity values for each cases, due to the non-linearity of six-bar structure, the value has sudden changes continuously. This fluctuations lead in some errors in motor shaft, where its effect can be seen in the figure.
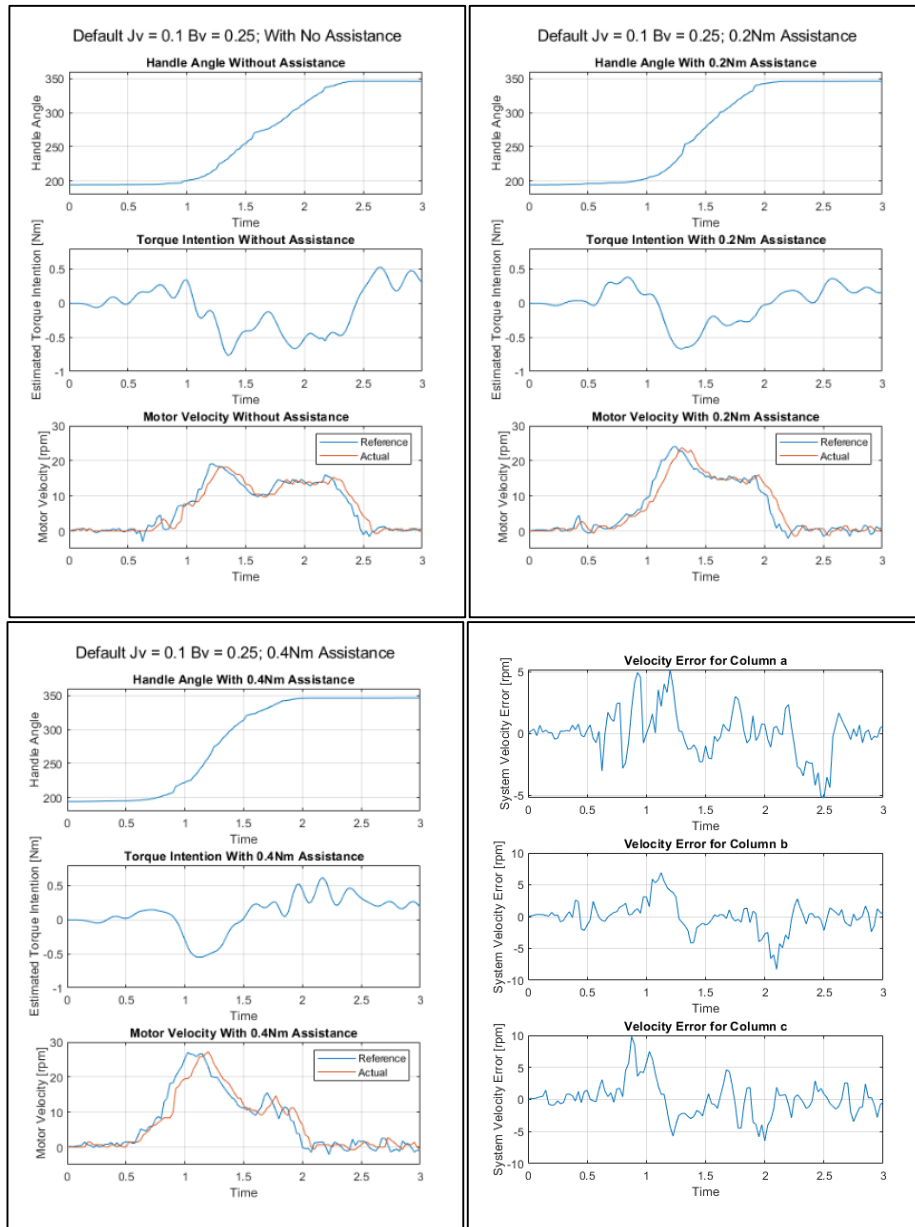
Figure 4.31 Free move finger extension with J_v=0.1, B_v=0.25 and assistance.

In Figure 4.31, the same extra assistance torques are applied by scaling virtual admittance coefficients down to 0.1 kgm²/rad and 0.25 Nms/rad. Since dynamic effects of virtual gains are scaled down, effort is reduced as expected. When offset assistance torque is 0.4 Nm as seen in third column, velocity passes over 20 rpm although intention torque cannot reach 0.6 Nm. Around this level of parameters are suitable to be used in active-assist mode of rehabilitation. The sudden changes and fluctuation in velocity reference results in some velocity errors, which can be seen in third column.
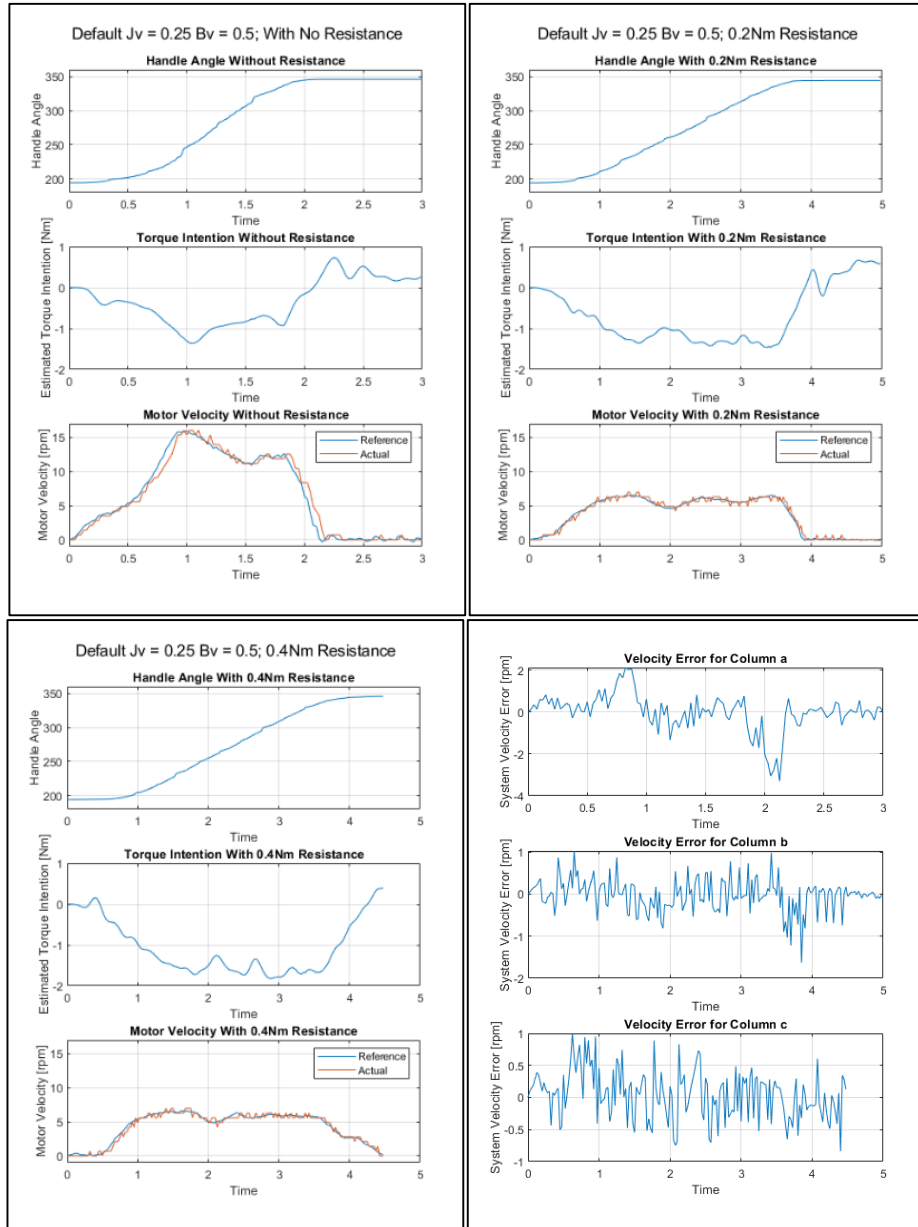
Figure 4.32 Free move finger extension with J_v=0.25, B_v=0.5 and resistance.

By aiming resistive type of rehabilitations, offset resistance is applied to the handle in control structure as 0.2 Nm and 0.4 Nm and plotted some outputs in Figure 4.32. Here, first column is kept in order to compare, which has no offset assistance or resistance. In experiment, default virtual inertia is 0.25 kgm^2/rad and virtual damping coefficient is 0.5 Nms/rad. It is possible to see that although torque intention increases with offset resistance, velocity levels decreases. In this case, since the fluctuation in terms of velocity is smaller, peak errors are smaller as well.
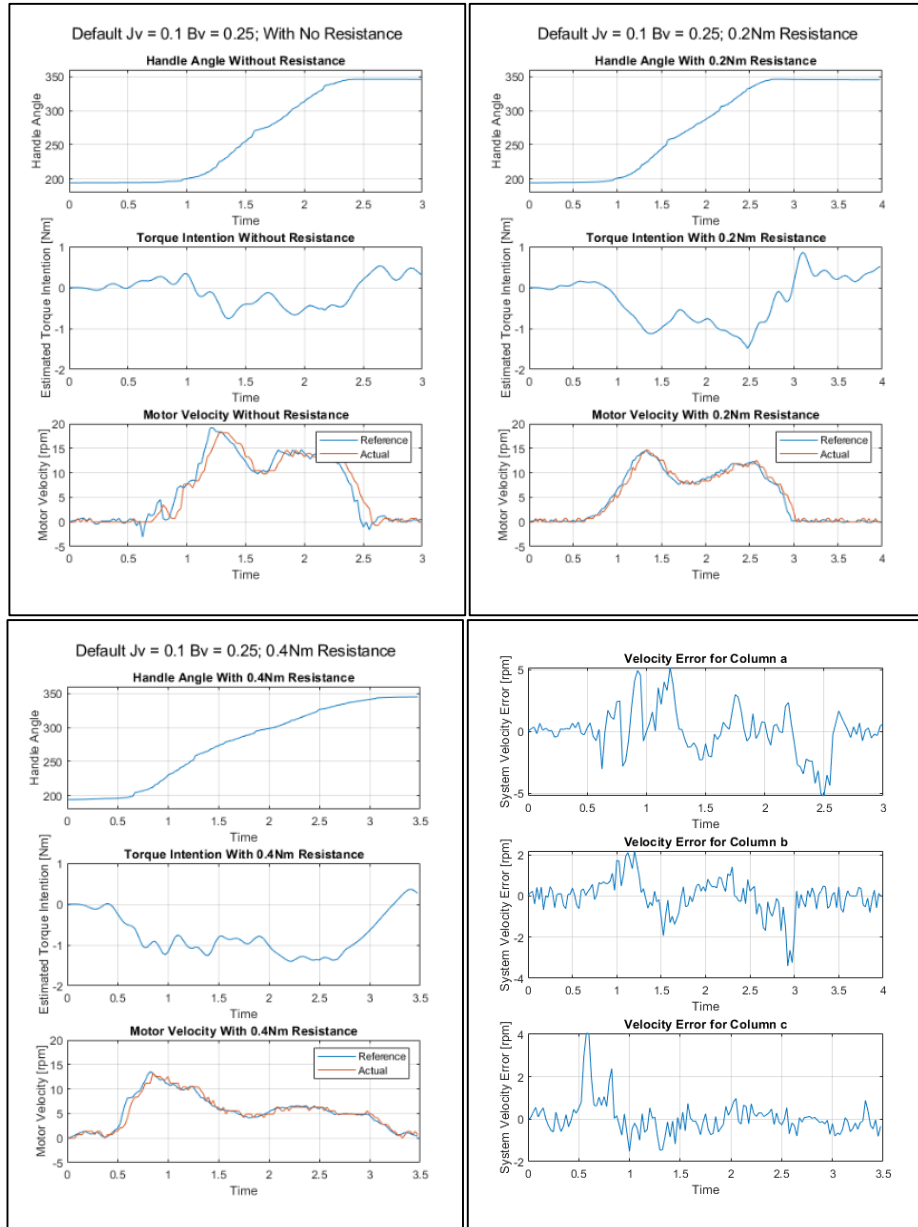
Figure 4.33 Free move finger extension with J_v=0.1, B_v=0.25 and resistance.

As in assistance case, default virtual admittance coefficients are scaled down to 0.1 kgm^2/rad and 0.25 Nms/rad with offset 0.2 Nm and 0.4 Nm resistances on the application handle shaft, while first column is kept to comparison (Figure 4.33). Here, it can be seen that level of user torque intention is not too high the due to low virtual admittance coefficients, but still this level of parameters seems suitable to be used in early stages of active resist rehabilitation.

Conducted experiments above are carried out by using variable admittance gain characteristics as given in Figure 4.29. In order to compare variable admittance gain and fixed admittance gain, one more experiment is conducted with this system as well.
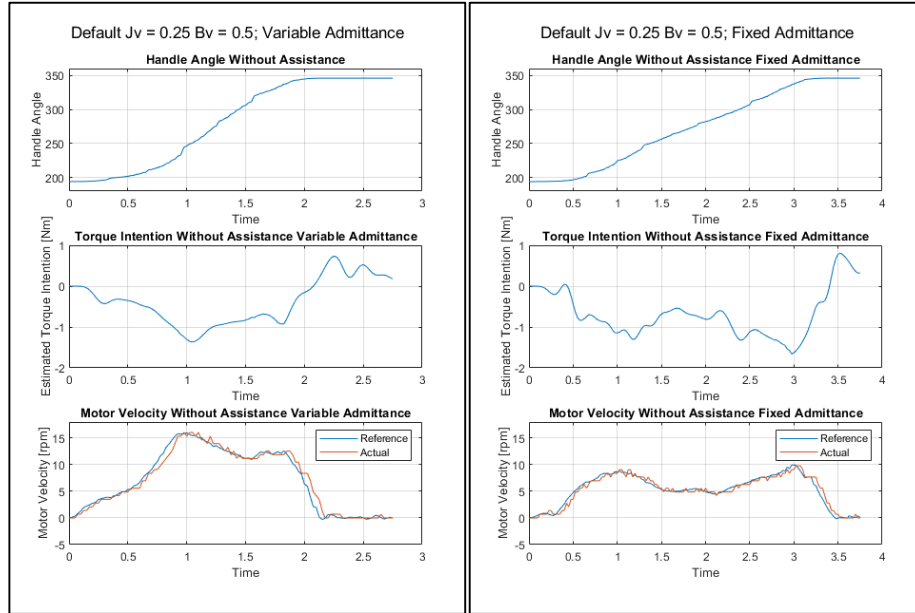
Figure 4.34 Free move finger extension with variable and fixed admittance (J_v=0.25, B_v=0.5).

In second column of Figure 4.34, virtual admittance gains do not change no matter how changes user intention. By comparing two columns, fixed admittance is more resistive in usage although there is no offset resistance in the system.

4.3.1.2 Referenced rotation structure with virtual spring

In this case study, apart from virtual effects of inertia and damping coefficients, additional virtual spring coefficient affects handle motion. According to handle position angle with respect to starting point, torque is created by the electric motor with considering torque delivering ratio as determined in equation 4.47. Here, it is important to note that kinematic relation of six-bar mechanism affects delivered torque as well as position between links.

Simulink model with control algorithm and needed blocks in order to fulfill other requirements is constructed as seen in Figure 4.35.
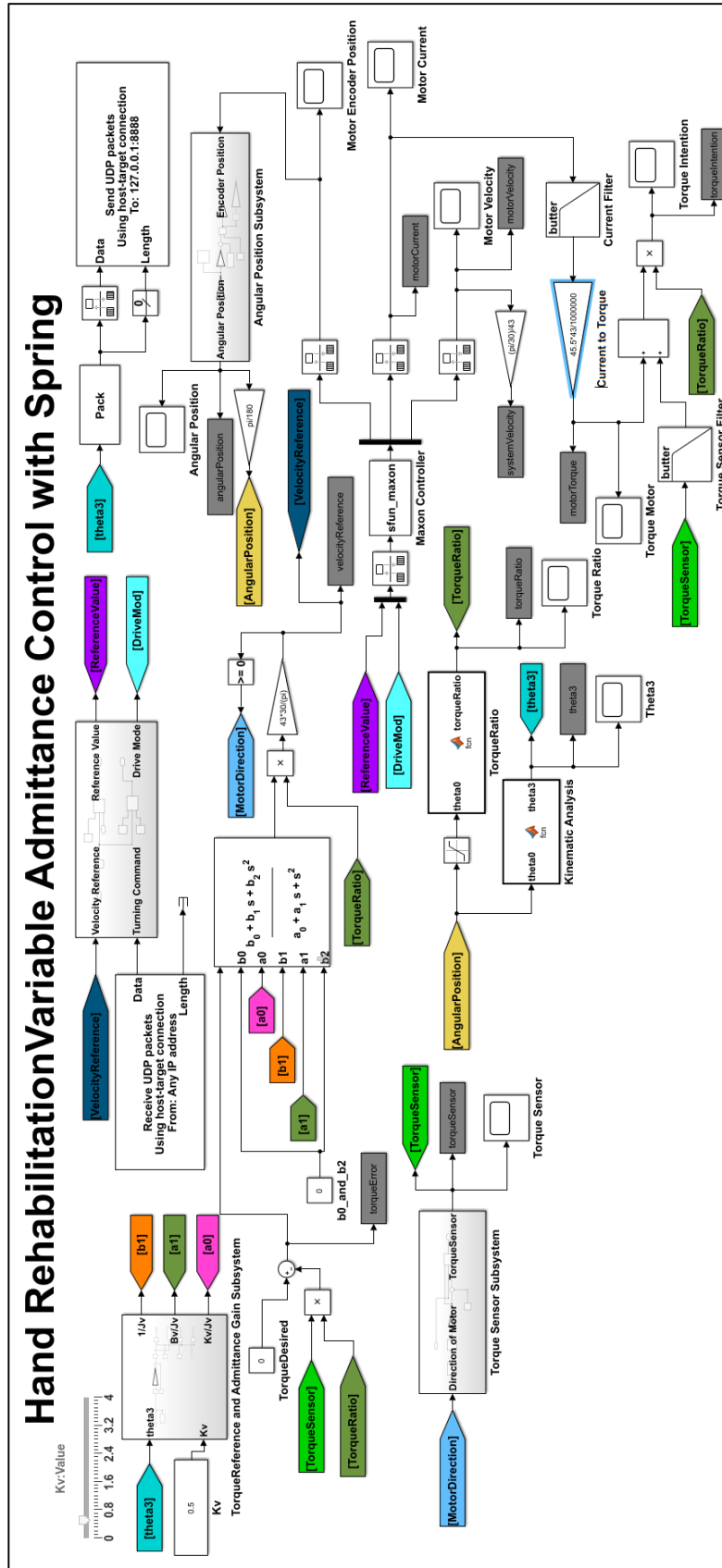
Figure 4.35 Spring move admittance control model in Simulink.

Before applying admittance control model with virtual spring, orientation of six-bar mechanism should be initialized as starting position, which is critical due to the fact that it is zero reference position of virtual spring. For this reason, Simulink position control model should be driven as visualized in Figure 4.27.

Due to the fact that there are some advantages to use variable admittance gains, structure is applied in this system as well. Here, a relation between virtual damping coefficient and application handle position angle is created while other admittance gains are kept constant at their default values.
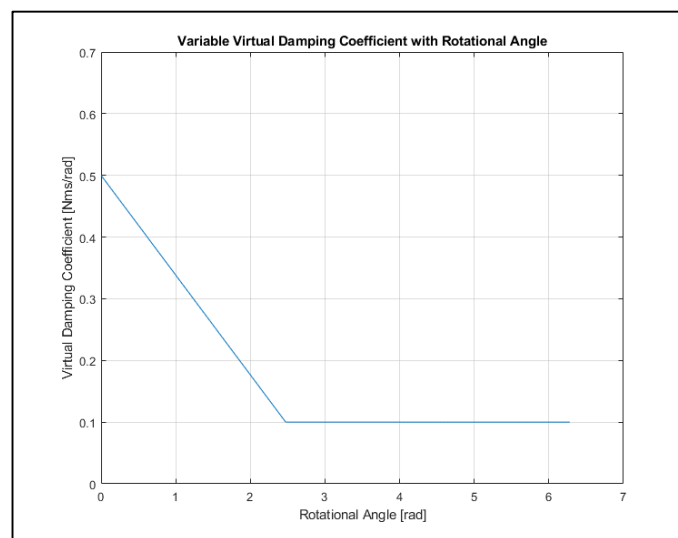
Figure 4.36 Variable virtual damping characteristic with respect to rotational angle.

As seen in Figure 4.36, it is possible to comment that effect of virtual spring becomes dominant when the handle is pushed away from starting point.

In the next steps, with different values of parameters, a healthy person tries to extend fingers against additional virtual spring torque from determined starting point of the handle.
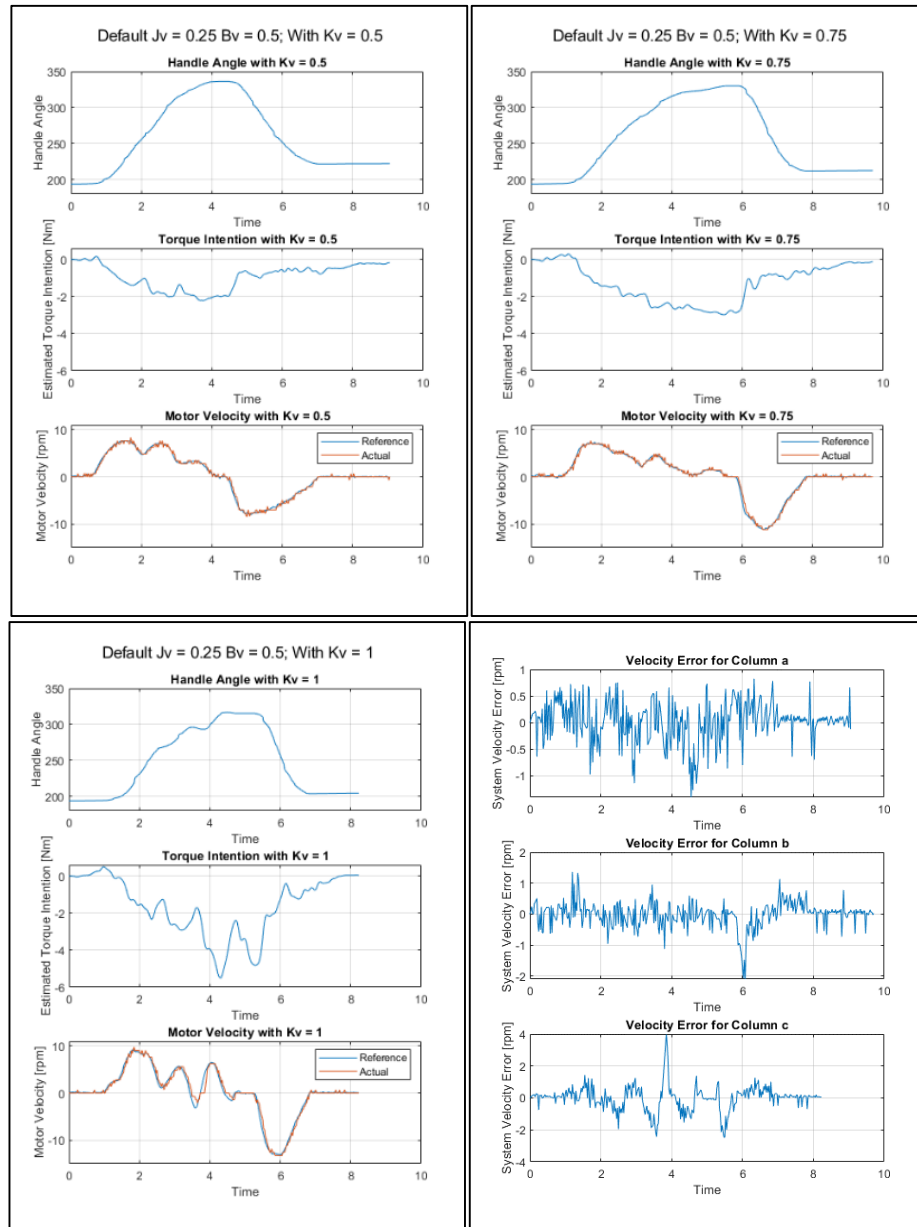
Figure 4.37 Spring move with J_v=0.25, B_v=0.5, K_v=0.5, K_v=0.75 and K_v=1.

In Figure 4.37, plots of important parameters can be seen when three different case studies with three different virtual spring coefficients are done while virtual inertia and virtual damping coefficients are kept constant as 0.25 kgm^2/rad and 0.5 Nms/rad, respectively. Here, virtual spring coefficient affects user torque intention dramatically. User torque intention passes over 4 Nm when spring coefficient is 1 N/rad while differing almost 140° of position, which is close to previous cases. However, when the system is released, spring torque is not enough to beat virtual and

real dynamics even in third column. This is because virtual damping coefficient is relatively high close to starting position.
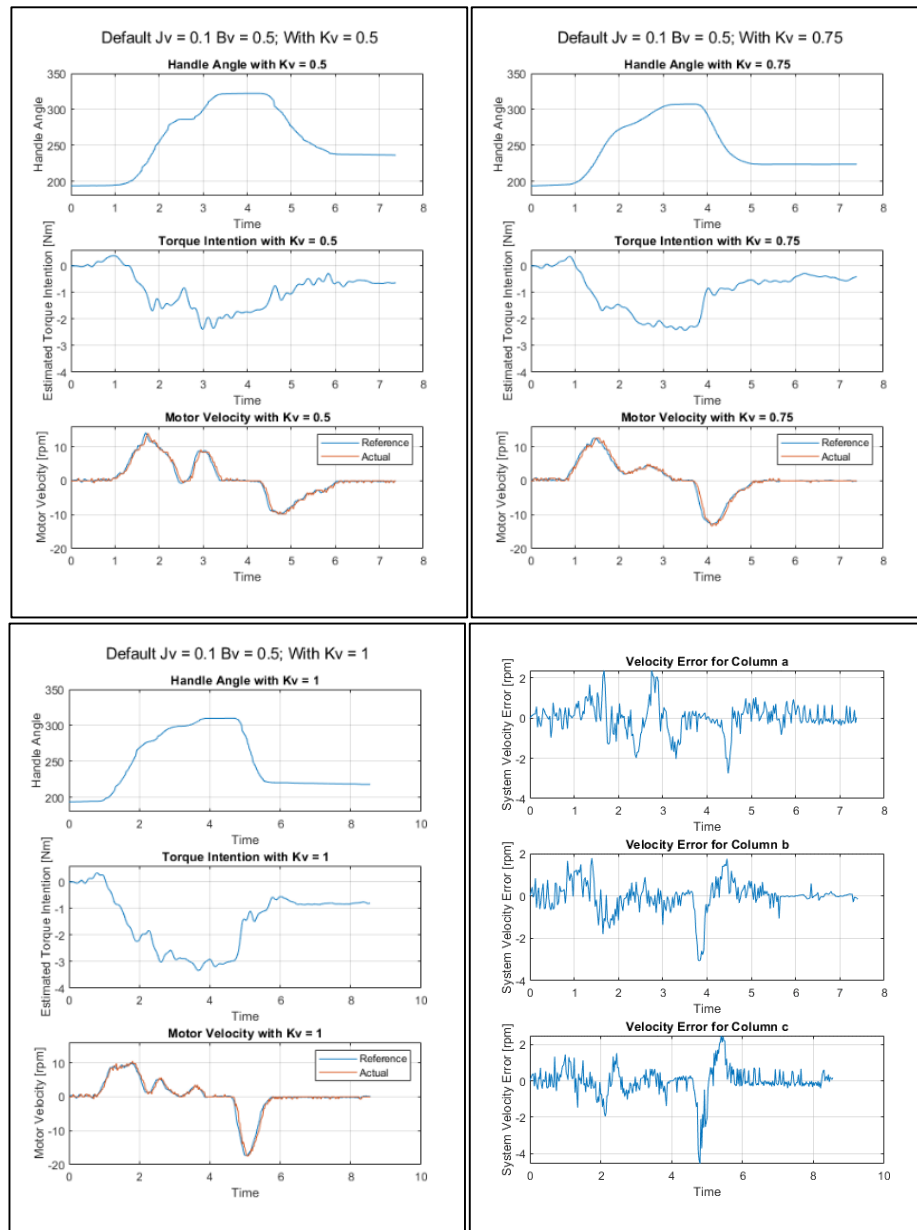
Figure 4.38 Spring move with J_v=0.1, B_v=0.5, K_v=0.5, K_v=0.75 and K_v=1.

With Figure 4.38, it is possible to comment on effect of virtual inertia by shrinking down to 0.1 kgm^2/rad. Similar to the previous part, user torque intention gets high with increasing amount of virtual spring coefficient. In this case, when the handle is released, it does not reach to starting position, instead, it stops further away from starting position compared to previous one due to the fact that virtual inertia is

smaller. In order to increase effectiveness of usage, damping coefficient on the handle should be decreased.
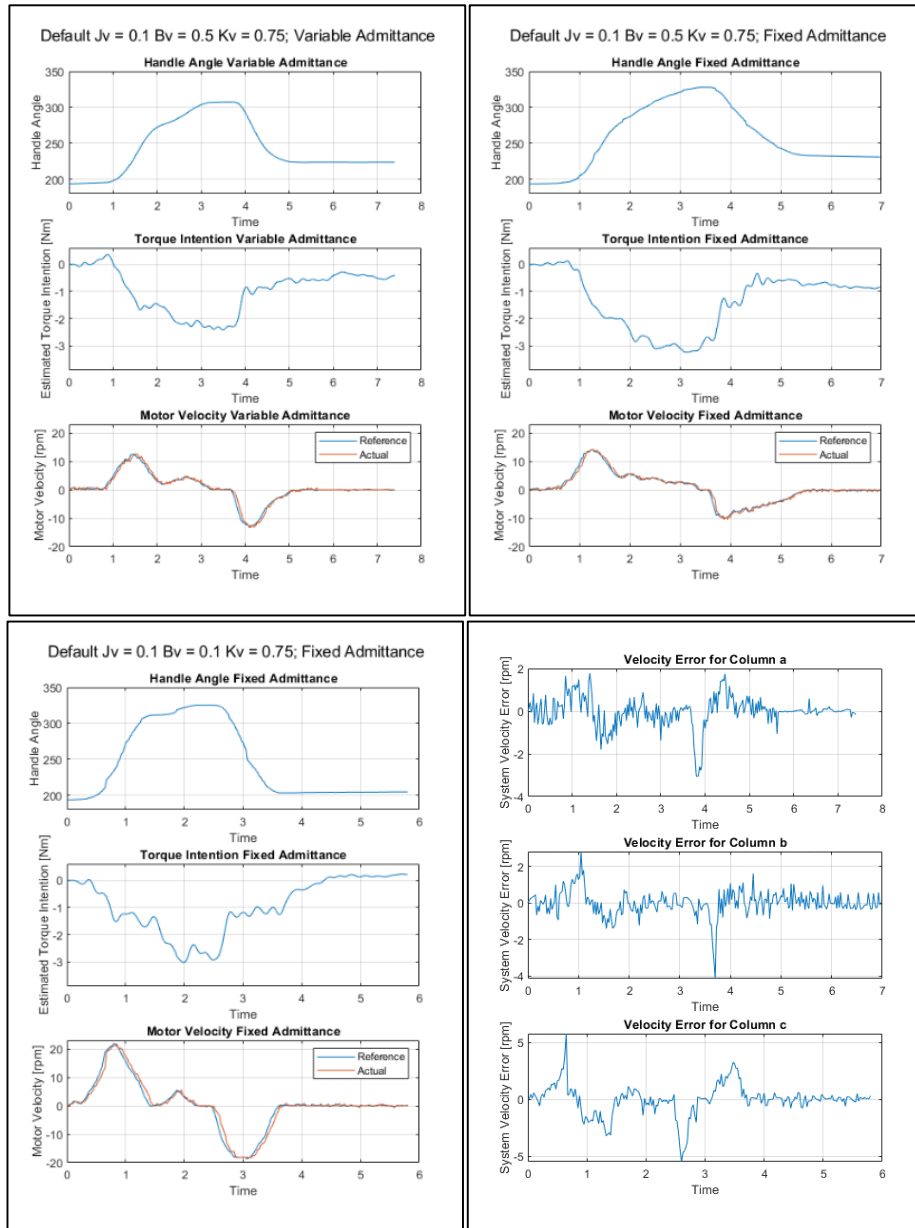
Figure 4.39 Spring move with variable and fixed admittance (J_v=0.1, K_v=0.75).

In Figure 4.39, as in the previous case studies, comparison between variable and fixed admittance gain control structures is done by choosing one variable damping coefficient and two different fixed damping coefficients as seen in columns sequentially. By comparing first and second columns, user torque intention in second is higher due to the fact that constant damping coefficient. This makes the system

very resistive. Also, when the handle is released, due to high amount of virtual friction, the system does not respond quickly, instead, it responds with an over-damped characteristic. In third column, it is possible to see that although fixed virtual damping coefficient is very small, torque intention is not as low as in first column. Also, it is very close to unstable behavior. For this reason, it can be concluded that even if fixed virtual damping coefficient is small, it is not very useful to be used in active assistance mode due to the fact that variable coefficient gives better results without getting closer to unstable behavior.

This study shows that variable type of admittance controller in both systems is applicable as rehabilitation robot with treatment modes. Next step is to increase functionality of the systems with game therapy. The outcomes of this Chapter are used in the next Chapter in order to define levels of games.

5. GAME REHABILITATION AND VIRTUAL ENVIRONMENT

Since ultimate aim is to use designed systems in physical rehabilitation, cognitive effects should also be included to the rehabilitation procedures due to their importance during recovery. Thus a methodology, game rehabilitation, should be proposed considering the integrity of virtual environment and admittance control structure together with electromechanical system itself.

In this Chapter, importance of game rehabilitation is introduced and methods to make games that communicate with the Simulink environment are discussed. Three different games are designed and created for different functionality. At the end of the study developed, games are played by a middle-aged healthy person, whose data are collected and compared in terms of different designed game levels to reveal the functionality of the proposal.

5.1 Game Rehabilitation

As mentioned in Chapter 1, after some incidents that affect neurological system, people should start physical therapy as quick as possible to increase probability of recovering. In such cases, some parameters strongly affect outcome of therapy, such as the time between trauma and starting therapy, number of repeatable movements of concerned extremity and passion to recover with high motivation. In literature, there are many works that cognitive effects have huge impact on recovering [15, 69, 70]. However, classical rehabilitation is unfortunately a relatively boring process that it is really possible for patient to lose motivation over some time. As robotic rehabilitation is increased in therapy sessions, works on increasing cognitive effects on rehabilitation is increased as well. Game based rehabilitations are started to be developed so that participants can increase their motivation for recovery process without just focusing on treatments, instead, focusing on games that are actually prescription of the illness. It is also important that this kind of rehabilitation can

provide the patient to experience the sense of achievement, even if he/she cannot achieve task in real world, by characterizing virtual environment and robotic system. It is concluded that use of virtual reality and video games for rehabilitation increase motivation of the patients to perform the therapy tasks [69]. Works in the literature that are focused on game-rehabilitation with virtual environment show that such systems can be used effectively in order to improve motor skill rehabilitation.

In the work [70], Mirelman and his team aim to determine effects of using a virtual environment in lower extremity based rehabilitation system. An experiment is conducted including therapy groups with and without virtual game environment. Eventually, the results show that using rehabilitation robots coupled with virtual environment improve patients' motor functions.

In this work, in order to increase cognitive effects on recovering, game-rehabilitation is applied via creating a virtual environment on Unity3D and communicating this tool with rehabilitation control model.

5.2 Virtual Environment: Unity 3D

Unity is a powerful and popular 3D game engine developed by Unity Technologies, which is first released in 2005. It is currently offered as freeware for small income companies and personal use, however, there exist still an option for paid licensing which has more features and it is obligatory to be used in companies whose annually income is above prescribed level. Due to these facts, it is extremely suitable for independent game developers and small companies. One of the most impressive specifications of this game-engine is its property being a cross-platform, to date, it supports 27 different platforms with one project. With its robust ecosystem, as well as usage in developing games, it is also possible to use this programming tool in real time or offline virtual simulations that includes physical environments. As a matter of fact, it is already used in some engineering and architecture purposes, in automotive sector as well as filming industry [71]. Moreover, Unity combines visual physics engine and coding parts together in a simplest way. With limited amount of visual effect knowledge and programing experience, it is possible to create a game or simulation environment. The programing language can be visual C# or java. The

architecture of those object-oriented languages makes programing clean and methodical.

In this work, Unity-3D is the main tool for creating virtual environments and simulating the designs to be used in rehabilitation procedures. Simulation is taken place in real-time, which is made possible with bidirectional communication between the Unity-3D and the rehabilitation control model in Simulink. Since the author's experience in programing supports the object-oriented programing with visual C#, this programing language is used in scripting the simulation environment.

5.3 Socket Communication

Besides its primary works, Matlab Simulink model should also deal with communication with front-end system, which is user interface that shows virtual environment to user for increasing cognitive effects of recovery process as well as contributing to the system efficiency. Therefore, Simulink model and user interface should share some information bidirectional in real-time. Here, "real-time" is an important aspect, since interface should react as fast as possible to keep user attention at high level. For these reasons, communication type between two programs that are executed in the same machine, should be chosen and constructed.

Any two computer programs that are executed in the same machine or remotely connected, can communicate each other, if only they have the necessary hardware as well as the software. To date, there are many hardware and software protocols to deal with communication processes. Roughly, it is possible to comment that while hardware work on electrical level, low-level software deal with their embedded structure to create electrical signals. In light of this, two machines or programs can communicate, if two software know how to interpret low-level electrical signals in bitwise. Since it is one of the most important needs in any sector that deals with the processors with software, some global protocols are created to meet the needs.

In this work, one of requirements is that communication should be real-time under load of calculation process of Simulink model. Also, since Simulink model and interface works on the same computer, any additional hardware is not requested.

Finally necessary infrastructure for communication protocol should be supported by both Simulink and Unity-3D programs.

By considering these facts, socket communication is chosen as the most suitable for the system. Socket is an object that allows a program to accept incoming connections and send/receive data. It supports client-server computing and thanks to its general Application Program Interface (API), it has capability to work as client and server on the same machine [72]. Application programming interface for internet protocol (IP) networking is socket API. According to Open Systems Interconnection (OSI) model, which is developed by International Organization for Standardization (ISO) for interoperating of diverse communication systems with standard communication protocols, socket API relates to transport layer (OSI layer 4). This API supports various protocols such as Transmission Control Protocol (TCP) [73] and User Datagram Protocol (UDP) [74] as in Figure 5.1.
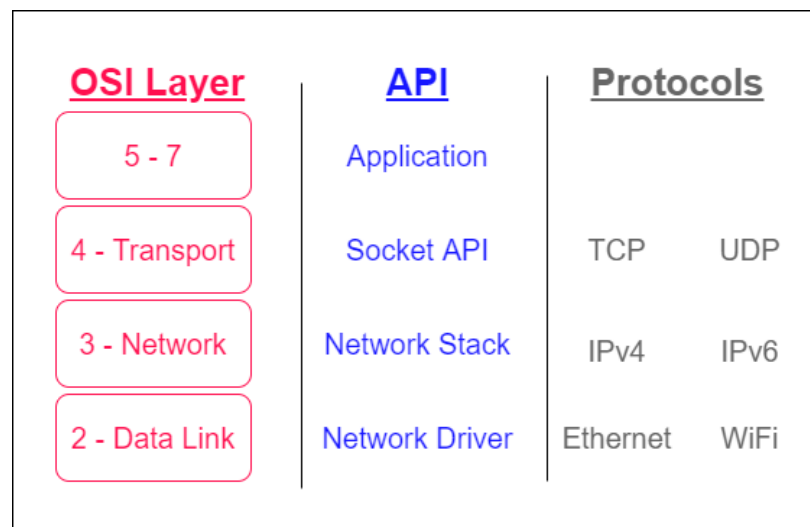
Figure 5.1 OSI Layers.

Both UDP and TCP run on the top of the Internet Protocol (IP). For two distinct machines to communicate each other with socket communication, both software should have information including network location (IP address) and TCP/UDP port. IP addresses are the unique addresses in a network to be used in communicating for devices, locally or globally. When the transport layer with socket API calls an IP address, it actually aims to communicate another machine over the internet. If

internet is connected, data packets are sent via transport protocol in socket API (TCP/IP or UDP/IP) to the specified location on global network [75]. Besides, this data transfer protocol, as over internet with transport layer, can also be used in local networks without internet connection. In this case, connection should be done on the same machine, where localhost comes in forward. With the same procedure as in internet case, if IP address is given as address of localhost, then with help of socket API based software, program tries to contact to the very same machine. Special IP address for localhost is defined as 127.0.0.1. This procedure creates a process called loopback, which creates a virtual interface through operating system. Using loopback interface bypasses any local network interface hardware.

For example, in a closed local network via router, this data transport protocol can be used by addressing IP addresses that are supplied by router. With the same logic, if internal communication is requested in the same machine, same procedure can be applied using localhost. In this work, mentioned communication protocol is used for transferring data packets between two different programs, MATLAB Simulink and Unity3D. As mentioned before, MATLAB Simulink is responsible to construct control model, drive motors and read sensors. At the same time, the system should have a human interface in order to create a virtual environment to steer user for cognitive concerns that are very important for recovering in neuro-rehabilitation. This end interface made with Unity also supplies to have a purpose for rehabilitation process.

5.3.1 TCP/IP and UDP/IP

Socket API has two popular socket types in transport layer, TCP and UDP. Both have advantages and disadvantages over each other.

TCP/IP socket, is also called stream socket, provides reliable two-way communication. This protocol is designed to send and receive data packets to other side without errors and in sequence. Before transmitting data, two peers should establish a connection, which is also called handshake process. Also, data packet is a heavy-weight and includes bidirectional communication for checksums. This, of course, slows down communication process; therefore TCP is not a good option for

real-time applications, which is very critical for this work. This type of communication is used in webservers, mail servers etc., where communication latency does not so important.

UDP/IP socket, also called datagram socket, cannot guarantee data transfer and there is no overhead for opening a connection, so it is an unreliable datagram oriented and connectionless protocol. Besides these drawbacks, UDP protocol is super-fast compared to TCP and very suitable for broadcasting in real-time. This type of communication is commonly used in video streaming, networked games, voice over IP and applications that cannot tolerate latency but tolerate some data losses.

As a result, since real-time concerns are important, UDP/IP type of connection is chosen to be applied in this work. Data reliability is removed with precautions in software. Final software and hardware architecture can be seen in Figure 5.2.
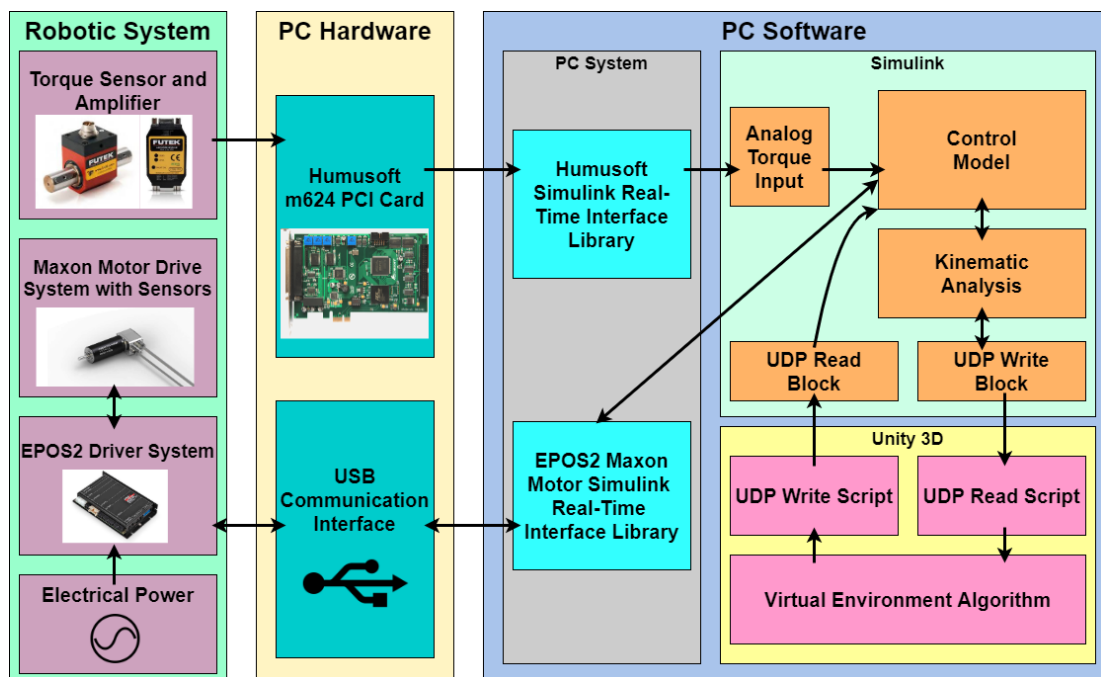
Figure 5.2 Hardware-Software Architecture.

In Simulink model, real-time UDP library blocks are used. In Unity software, library for sockets that are developed by Microsoft (System.Net.Sockets) is used with class of UdpClient.

5.4 Setting-Up Visuals

Throughout the thesis two different games are designed to be used in rehabilitation procedures for the single degree of freedom wrist and forearm rehabilitation system. In these games, visuals are set as they are in real-life in order to increase awareness. For this reason, exactly same components including actuator and mechanical parts are transferred to Unity environment. However, for hand rehabilitation system that consists of six-bar mechanism, simple 2D game is designed without considering physical appearance of the system in virtual environment.

Autodesk Inventor 2019 3D computer aided design program is used for creating mechanism parts and they are assembled together. Here, assembled mechanism is saved as STL (stereolithography) format to be converted to OBJ (object) format, which is accepted by Unity software (Figure 5.3).
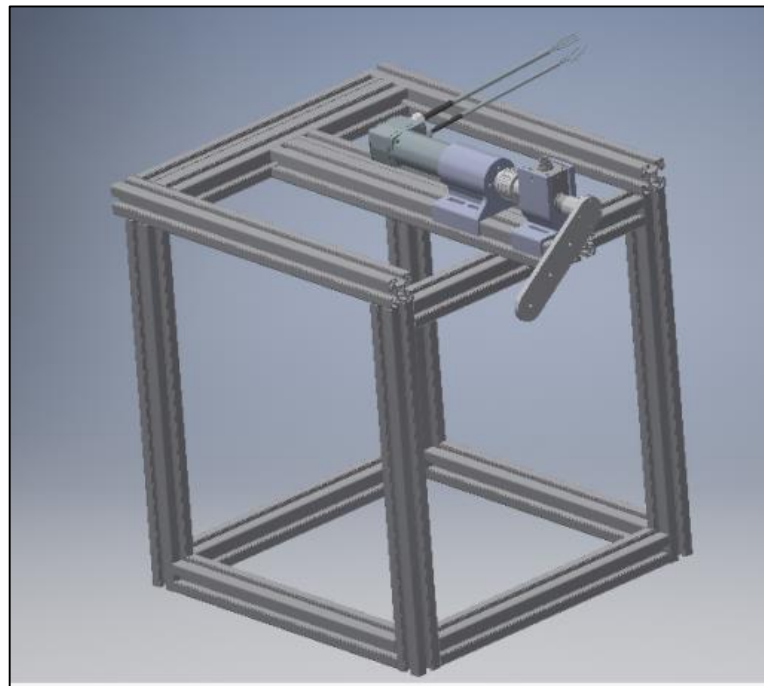
Figure 5.3 System model in Autodesk Inventor 2019.
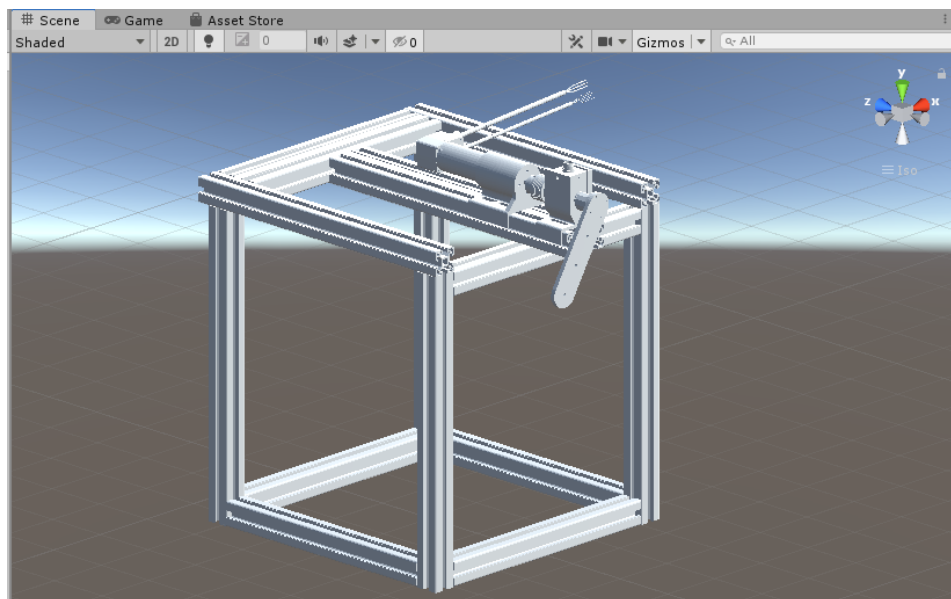
Figure 5.4 System model that is imported to Unity 3D.

Mechanism as a whole is transferred into Unity 3D as seen in Figure 5.4. With small modifications and additions, two different games are built as discussed in next sections.

5.5 Unity Hierarchy and Scripting

Unity is a program that combines graphical environment directly to coding section. For this reason, it is required to build a hierarchy between graphical components and scripts. In Unity, graphical components can be bind to scripts. These graphical components such as object of the hand rehabilitation system can be driven from these bind scripts to be moved. In this work, all graphical components are basically dependent on position information that comes from Simulink environment. According to encoder position, position angle is determined in Simulink and this information is broadcasted real-time over UDP to Unity. UDP reading script is based on library that is developed by Microsoft (System.Net.Sockets). UDP client is created with this library in order to receive data with specific port over UDP. This data is attended to public variable in order to be shared with other scripts of game objects to be used in visualizing. This public data that is updated in Simulink is used in scripts of graphical components such as handle, spring, rope or bucket, which will be discussed and visualized in the next section.

5.6 Single DOF Wrist and Forearm Rehabilitation Game Scenario and Its Application

As mentioned in sub-section 5.1, game therapy in physical rehabilitation has positive effects on recovering due to cognitive effects in brain. Desired motions to be repeated can be implemented into games by aiming entertainment and raising patients' mood. In this section, admittance control is combined with game environment by considering rehabilitation modes. All concerned types of admittance control are used in different levels of games that are designed for using in rehabilitation.

In this thesis, primary game concepts are based on functionality. Visual effects inside the game and other design concepts such as graphical user interface, flow on the game is not primarily concerned and instead their developments are left for future works.

Related single degree of freedom wrist and forearm rehabilitation setup can be driven with free move and reference position based depending on the strategies discussed above. Here, two different basic games are designed with respect to concerning the rehabilitation procedure modes as discussed in sub-section 1.1.

5.6.1 Free move game scenario

First game is based on free move of the wrist rehabilitation setup. Motion is strongly focused on wrist rotation by the help of muscles on single or several fingers. The main concept behind the game is the fact that user tries to rotate a swatter in order to catch a plane that flies on a constrained motion, which is the same rotational trajectory that is traced by the swatter itself (Figure 5.5). Once the plane is caught with the scatter, dedicated level is counted as achieved (Figure 5.6). However, if the plane crashes to the scatter from behind, which means that user stays slow, level will be failed (Figure 5.7). The scatter is created by the Inventor 2019 and the plane is extracted from Unity's Asset Store.
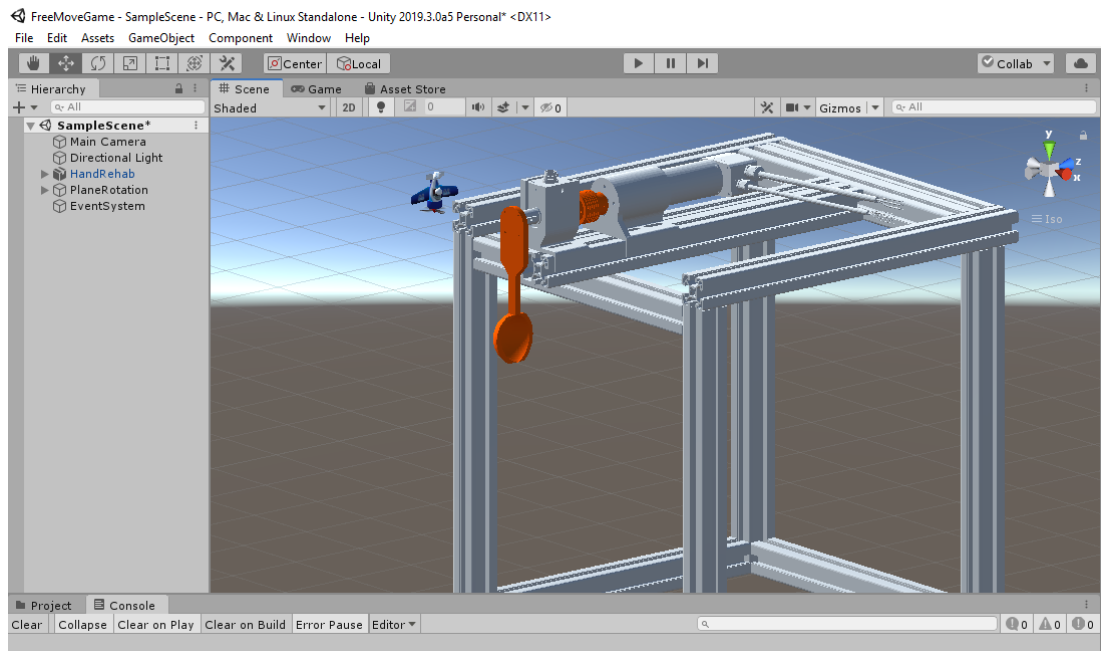
Figure 5.5 Free move game scene in Unity 3D.
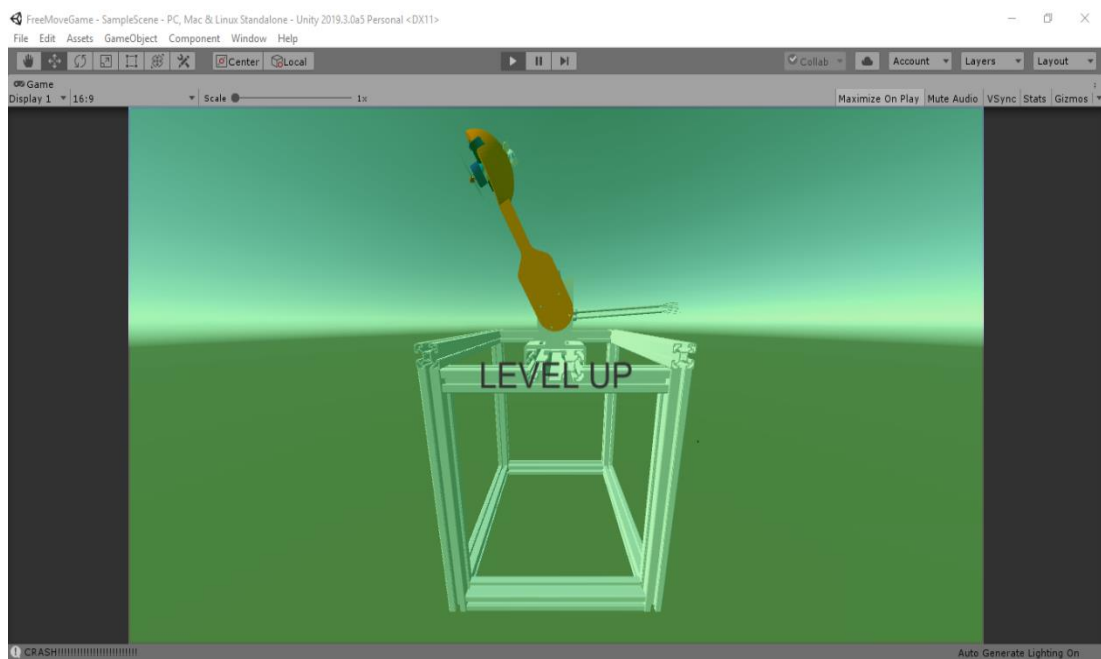
Figure 5.6 Free move game level achieved scene in Unity 3D.
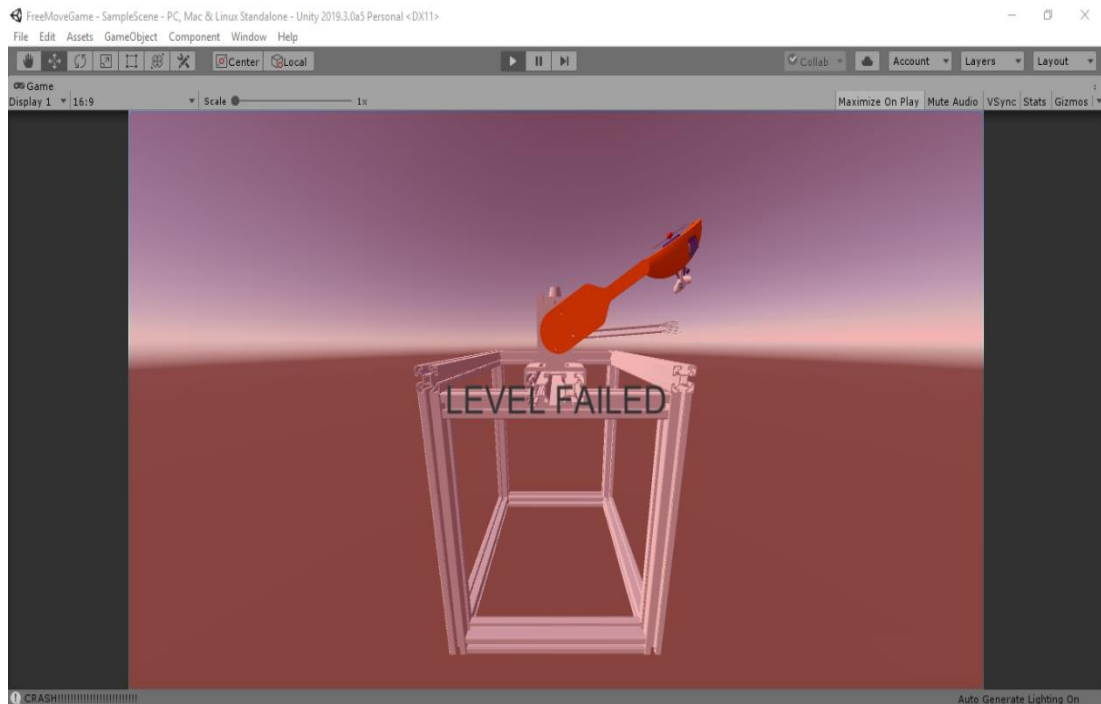
Figure 5.7 Free move game level failed scene in Unity 3D.

Levels of the game are constructed by considering rehabilitation modes. First, the system tries to create plasticity on the brain, which is related with passive mode of rehabilitation procedures. In this procedure, only velocity control is used as main control structure because no interaction control is required of this type of rehabilitation mode. Here, two different game levels are dedicated to passive mode by changing velocity reference. In next levels of the game, active assist mode with different levels of extra assistance is applied. In these levels, velocity of the plane is also considered as level of difficulty because catching action is the main purpose. Active partial assist mode is dedicated to next levels where there is no extra assistance apart from achieving inertial, frictional and gravitational effects on the system. In latest stages of rehabilitation progress, in order to increase muscle activities, resistance comes into action, which is discussed in rehabilitation procedures as active resist mode. In this mode, extra resistances are applied according to different level of the game. Again, velocity of the plane is also considered as level of difficulty, at the end of game rehabilitation, user should catch the plane although there is considerably high resistance in front and the plane flies relatively fast. Game in action can be seen in Figure 5.8.
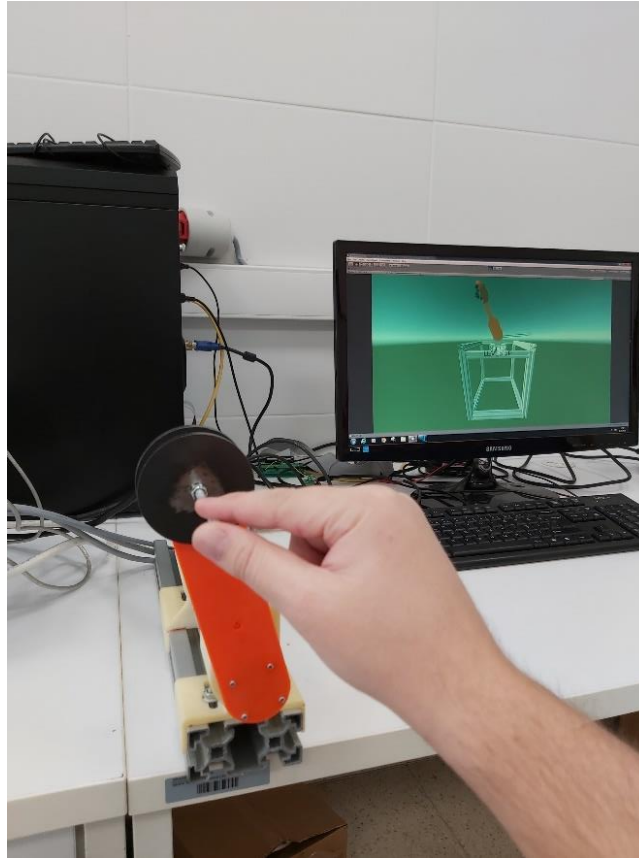
Figure 5.8 Free move game in action.

According to the needs of rehabilitation procedures, game with 20 levels are designed and tabulated below to show utilized parameters. In levels with interaction control, velocity of the plane plays also an important role for achieving levels. With effects of interaction control, no matter how user rotates the scatter, plane rotates itself with velocity that is related with the ratio on 5th column of Table 5.1. However, once user rotates the scatter very slowly or stops, plane rotates with minimum velocity which is specified in 6th column.

Table 5.1 Free move game difficulty level parameters.

Level	Control Mode	Controller References	Admittance Control Default Parameters	Plane Default Velocity	Plane Minimum Velocity
L1	Passive Mode (Velocity Control)	$V_{ref} = 15$ rpm	$J_v = 0, B_v = 0$	$V_{plane} = 7.5$ rpm	$V_{min} = 7.5$ rpm
L2	Passive Mode (Velocity Control)	$V_{ref} = 10$ rpm	$J_v = 0, B_v = 0$	$V_{plane} = 5$ rpm	$V_{min} = 5$ rpm
L3	Active Assist Mode	$T_{assist} = 1$ Nm	$J_v = 0.5, B_v = 1$	$V_{plane} = 0.5V_{system}$	$V_{min} = 4$ rpm
L4	Active Assist Mode	$T_{assist} = 1$ Nm	$J_v = 0.5, B_v = 1$	$V_{plane} = 0.8V_{system}$	$V_{min} = 4$ rpm
L5	Active Assist Mode	$T_{assist} = 1$ Nm	$J_v = 1, B_v = 2$	$V_{plane} = 0.5V_{system}$	$V_{min} = 4$ rpm
L6	Active Assist Mode	$T_{assist} = 1$ Nm	$J_v = 1, B_v = 2$	$V_{plane} = 0.8V_{system}$	$V_{min} = 4$ rpm
L7	Active Assist Mode	$T_{assist} = 0.5$ Nm	$J_v = 0.5, B_v = 1$	$V_{plane} = 0.5V_{system}$	$V_{min} = 4$ rpm
L8	Active Assist Mode	$T_{assist} = 0.5$ Nm	$J_v = 0.5, B_v = 1$	$V_{plane} = 0.8V_{system}$	$V_{min} = 4$ rpm
L9	Active Assist Mode	$T_{assist} = 0.5$ Nm	$J_v = 1, B_v = 2$	$V_{plane} = 0.5V_{system}$	$V_{min} = 4$ rpm
L10	Active Assist Mode	$T_{assist} = 0.5$ Nm	$J_v = 1, B_v = 2$	$V_{plane} = 0.8V_{system}$	$V_{min} = 4$ rpm
L11	Active Partial Assist Mode	$T_{assist} = 0$ Nm	$J_v = 0.5, B_v = 1$	$V_{plane} = 0.5V_{system}$	$V_{min} = 4$ rpm
L12	Active Partial Assist Mode	$T_{assist} = 0$ Nm	$J_v = 0.5, B_v = 1$	$V_{plane} = 0.8V_{system}$	$V_{min} = 4$ rpm
L13	Active Partial Assist Mode	$T_{assist} = 0$ Nm	$J_v = 1, B_v = 2$	$V_{plane} = 0.5V_{system}$	$V_{min} = 4$ rpm
L14	Active Partial Assist Mode	$T_{assist} = 0$ Nm	$J_v = 1, B_v = 2$	$V_{plane} = 0.8V_{system}$	$V_{min} = 4$ rpm
L15	Active Resist Mode	$T_{resist} = 0.5$ Nm	$J_v = 0.5, B_v = 1$	$V_{plane} = 0.5V_{system}$	$V_{min} = 4$ rpm
L16	Active Resist Mode	$T_{resist} = 0.5$ Nm	$J_v = 0.5, B_v = 1$	$V_{plane} = 0.8V_{system}$	$V_{min} = 4$ rpm
L17	Active Resist Mode	$T_{resist} = 0.5$ Nm	$J_v = 1, B_v = 2$	$V_{plane} = 0.5V_{system}$	$V_{min} = 4$ rpm
L18	Active Resist Mode	$T_{resist} = 1$ Nm	$J_v = 0.5, B_v = 1$	$V_{plane} = 0.5V_{system}$	$V_{min} = 4$ rpm
L19	Active Resist Mode	$T_{resist} = 1$ Nm	$J_v = 0.5, B_v = 1$	$V_{plane} = 0.8V_{system}$	$V_{min} = 4$ rpm
L20	Active Resist Mode	$T_{resist} = 1$ Nm	$J_v = 1, B_v = 2$	$V_{plane} = 0.5V_{system}$	$V_{min} = 4$ rpm

5.6.2 Free move game results

First game is based on free move structure. All difficulty levels are applied in the system during the play of a healthy person in order to verify proof of concept. Results are tabulated in Table 5.2.

Table 5.2 Free move game results for each level.

Level	Average Velocity	Average Torque	Rotation Degree	Estimated Energy	Level Time
L1	14.83 rpm	0 Nm	980°	0 Nm rad	11.00 s
L2	9.98 rpm	0 Nm	1620°	0 Nm rad	27.10 s
L3	21.94 rpm	0.99 Nm	775°	121.83 Nm rad	5.90 s
L4	26.83 rpm	0.95 Nm	2366°	357.85 Nm rad	14.70 s
L5	16.36 rpm	1.49 Nm	906°	214.59 Nm rad	9.20 s
L6	19.94 rpm	1.91 Nm	4381°	1329.50 Nm rad	36.60 s
L7	22.37 rpm	1.30 Nm	765°	158.90 Nm rad	5.70 s
L8	30.04 rpm	1.64 Nm	2000°	522.40 Nm rad	11.10 s
L9	16.74 rpm	2.14 Nm	892°	304.08 Nm rad	8.90 s
L10	21.96 rpm	2.95 Nm	3333°	1563.30 Nm rad	25.30 s
L11	21.82 rpm	1.82 Nm	777°	224.66 Nm rad	5.90 s
L12	26.30 rpm	2.11 Nm	2381°	801.13 Nm rad	15.10 s
L13	14.79 rpm	2.58 Nm	993°	408.03 Nm rad	11.20 s
L14	20.58 rpm	3.56 Nm	3828°	2167.50 Nm rad	31.00 s
L15	15.20 rpm	3.40 Nm	965°	522.99 Nm rad	10.60 s
L16	20.59 rpm	4.42 Nm	3876°	2723.70 Nm rad	31.40 s
L17	13.12 rpm	5.37 Nm	1094°	934.33 Nm rad	13.90 s
L18	16.33 rpm	4.41 Nm	903°	633.92 Nm rad	9.20 s
L19	21.12 rpm	5.42 Nm	3698°	3189.90 Nm rad	29.20 s
L20	13.96 rpm	6.43 Nm	1048°	1073.00 Nm rad	12.50 s

As seen from the results of different levels, user increases his/her effort to the system due to the changes in control algorithm. According to the results, hardest levels are the ones in that velocity ratio of the plane is high due to the fact that it is harder to catch the plane with the scatter and naturally it takes time to be caught. This control structure that is combined with rehabilitation modes is suitable for game rehabilitation system, which has the ability to drain effort with entertainment environment.

5.6.3 Reference motion with spring game scenario

Second game is based on reference motion with virtual spring of the forearm rehabilitation setup. Motion is focused on forearm pronation supination. Game is based on to carry a hook to a place where target bucket should be hooked and carried. Motion is forced under a spring that is fixed to a wall and deflected with motion, which tries to rotate back pendulum according to virtual spring coefficient (Figure 5.9).
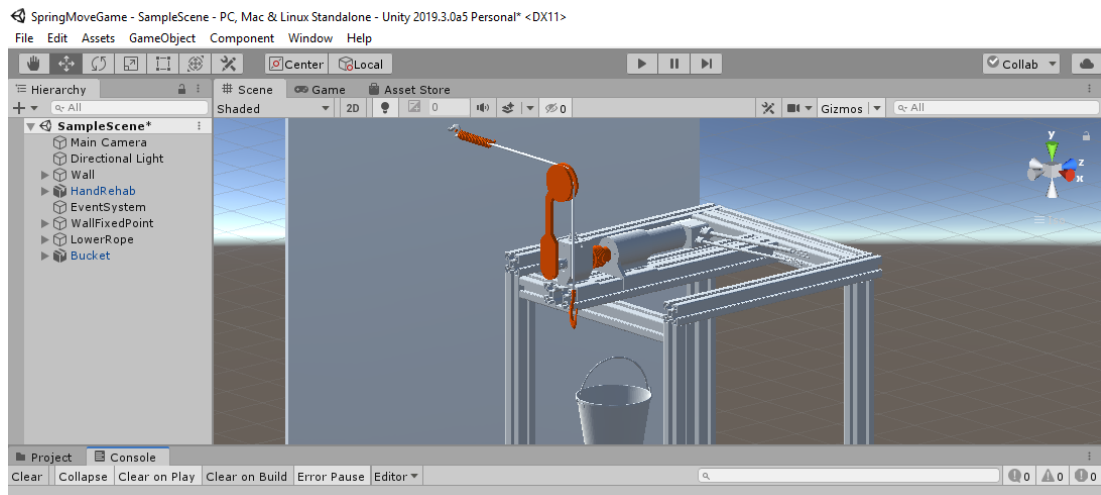
Figure 5.9 Spring game scene in Unity 3D.

Similar to the previous section, difficulty levels of this game are designed by considering rehabilitation modes as well. Game starts with passive motions, which focuses on creating plasticity as discussed in passive mode. In these levels of the game, user should just hold the handle, all movements are carried out by the control structure of the system itself. This helps to increase awareness of movement and create plasticity on the brain. In the next levels, active assist modes are taken into action. Movement is fulfilled via support of control structure as in free move case without virtual spring due to the loss of muscle activities of user. In these levels, position should be controlled as well due to the fact that motion resembles free-move. As rehabilitation continues, the next step of rehabilitation modes, active partial assist mode, comes into action. In this mode, free-move alike motion should be considered again without extra support to user motion. In the next levels, the system actually is adopted to active resist modes, where virtual spring creates finite amount of counter force to user. From small amount of spring constant to higher amount, user should be resisted in order to increase muscle activities, since it is named as active-resist mode. Also, with active resistance is started, motion is also kept in hooking action, which is considered as hooking time in order to increase ability to keep force at one position. In game visualization at these levels, the buckets cannot be hooked on the hook until the hooking time is elapsed.
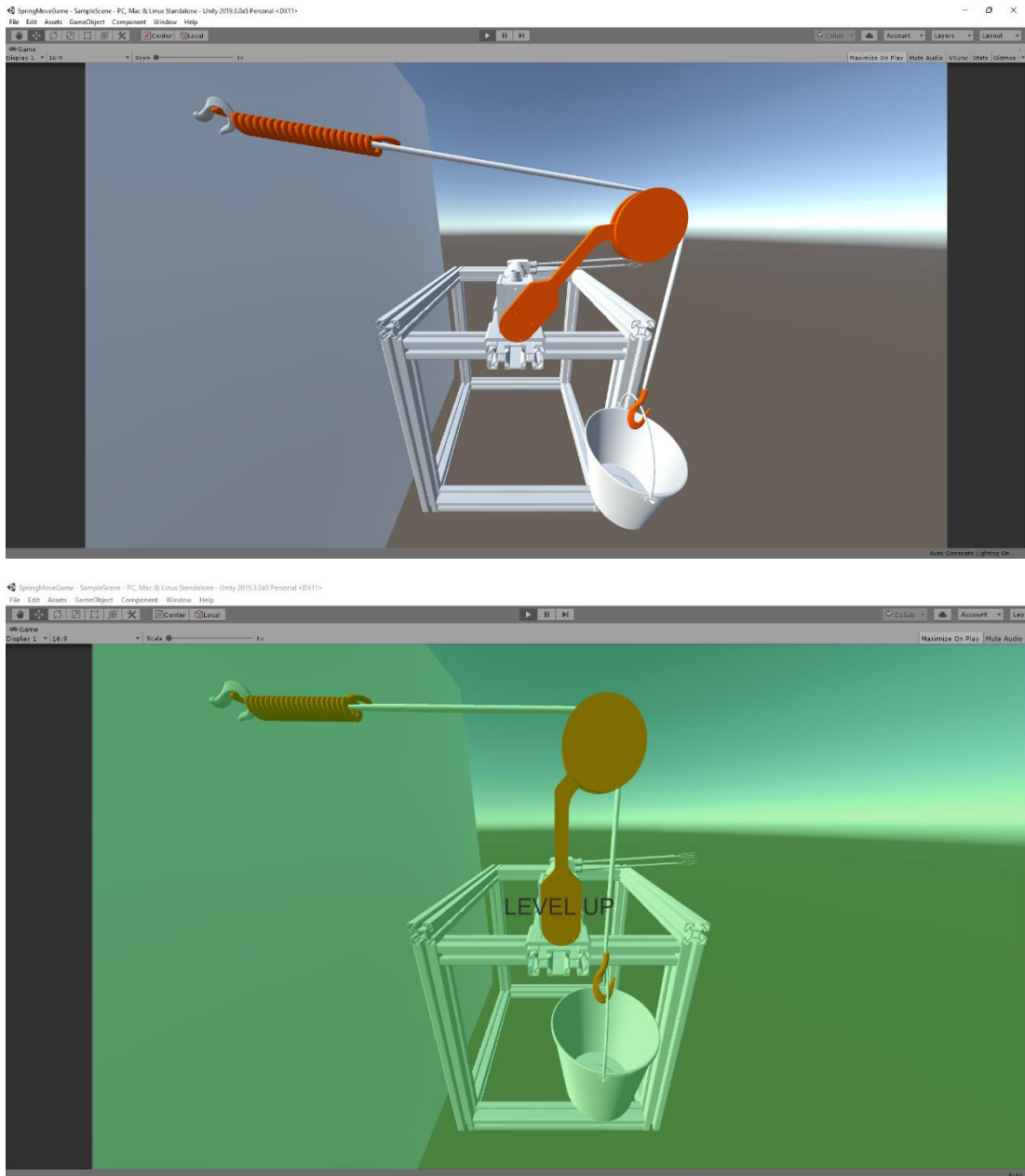
Figure 5.10 a) Spring game hooking bucket b) Spring game level achieved.

The whole procedure can be reprogrammed with mirror visualization in order to be used in the other side of motion. Spring move game in action can be seen in Figure 5.11.
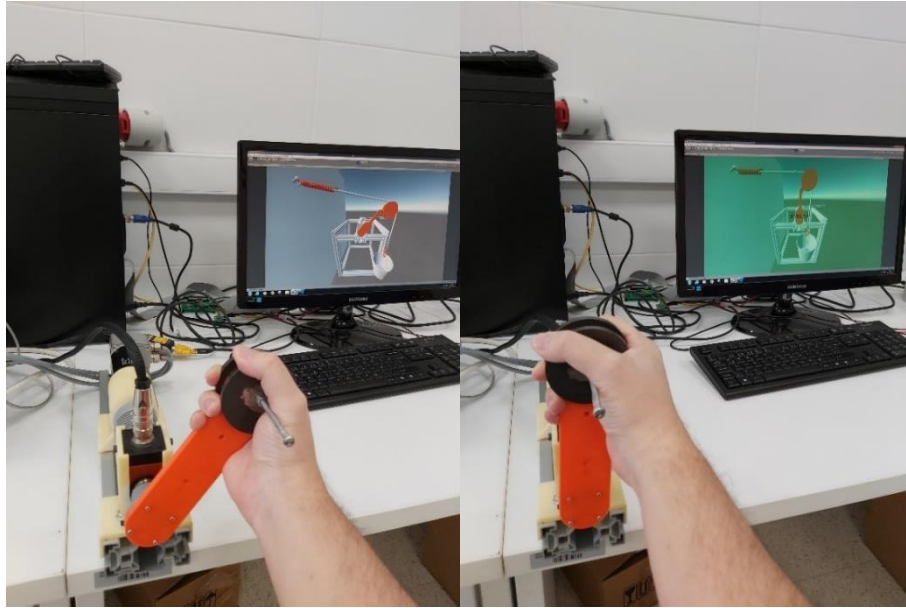
Figure 5.11 Spring move game in action.

According to the needs that are discussed above, game with 14 levels is designed and tabulated below to show utilized parameters in each level (Table 5.3). Here, it can be seen that user needs to achieve different amount of pronation-supination angles as specified in 4th column.

Table 5.3 Spring move game difficulty level parameters.

Level	Control Mode	Admittance Control Default Parameters	Angle	Hooking Time
L1	Passive Mode (Velocity Control)	$V_{system} = 5$ rpm	$\Theta = 40°$	$t_{hooking} = 0$ s
L2	Passive Mode (Velocity Control)	$V_{system} = 5$ rpm	$\Theta = 60°$	$t_{hooking} = 0$ s
L3	Active Assist Mode Tassist = 0.5Nm	$J_v = 0.5, B_v = 2, K_v = 0$	$\Theta = 40°$	$t_{hooking} = 0$ s
L4	Active Assist Mode Tassist = 0.5Nm	$J_v = 0.5, B_v = 2, K_v = 0$	$\Theta = 60°$	$t_{hooking} = 0$ s
L5	Active Partial Assist Mode	$J_v = 0.5, B_v = 2, K_v = 0$	$\Theta = 40°$	$t_{hooking} = 0$ s
L6	Active Partial Assist Mode	$J_v = 0.5, B_v = 2, K_v = 0$	$\Theta = 60°$	$t_{hooking} = 0$ s
L7	Active Resist Mode	$J_v = 0.5, B_v = 2, K_v = 3$	$\Theta = 40°$	$t_{hooking} = 0$ s
L8	Active Resist Mode	$J_v = 0.5, B_v = 2, K_v = 3$	$\Theta = 40°$	$t_{hooking} = 5$ s
L9	Active Resist Mode	$J_v = 0.5, B_v = 2, K_v = 3$	$\Theta = 60°$	$t_{hooking} = 0$ s
L10	Active Resist Mode	$J_v = 0.5, B_v = 2, K_v = 3$	$\Theta = 60°$	$t_{hooking} = 5$ s
L11	Active Resist Mode	$J_v = 0.5, B_v = 2, K_v = 4$	$\Theta = 40°$	$t_{hooking} = 0$ s
L12	Active Resist Mode	$J_v = 0.5, B_v = 2, K_v = 4$	$\Theta = 40°$	$t_{hooking} = 5$ s
L13	Active Resist Mode	$J_v = 0.5, B_v = 2, K_v = 4$	$\Theta = 60°$	$t_{hooking} = 0$ s
L14	Active Resist Mode	$J_v = 0.5, B_v = 2, K_v = 4$	$\Theta = 60°$	$t_{hooking} = 5$ s

5.6.4 Reference motion with spring game results

Designed game with difficulty levels that are tabulated in Table 5.3 is played by a healthy person in order to fulfill proof of concept. Results are tabulated in Table 5.4.

Table 5.4 Spring move game results in each level.

Level	Peak Torque	Average Torque	Estimated Energy	Level Time
L1	0 Nm	0 Nm	0 Nm rad	3.10 s
L2	0 Nm	0 Nm	0 Nm rad	3.35 s
L3	1.03 Nm	0.63 Nm	4.16 Nm rad	9.60 s
L4	1.42 Nm	0.95 Nm	9.89 Nm rad	9.65 s
L5	1.45 Nm	0.83 Nm	5.60 Nm rad	10.15 s
L6	1.46 Nm	1.12 Nm	11.30 Nm rad	10.35 s
L7	4.22 Nm	2.72 Nm	18.64 Nm rad	11.30 s
L8	6.48 Nm	4.95 Nm	64.75 Nm rad	14.25 s
L9	6.22 Nm	3.87 Nm	39.21 Nm rad	10.70 s
L10	5.99 Nm	4.76 Nm	93.37 Nm rad	15.13 s
L11	5.56 Nm	3.35 Nm	24.18 Nm rad	9.55 s
L12	9.17 Nm	5.08 Nm	68.23 Nm rad	14.55 s
L13	7.80 Nm	4.96 Nm	50.77 Nm rad	9.60 s
L14	13.79 Nm	8.06 Nm	152.11 Nm rad	14.50 s

As the levels are achieved, torque intention from user increases as expectedly. Here, given energy type to the system is mainly potential energy due to the effect of virtual spring on the system. Especially, efforts are increased in the levels where the hooking time is in action.

5.7 Hand Rehabilitation Game Scenario and Application

In the light of known facts about game therapy in rehabilitation, a game design is made for finger rehabilitation. By using different effects of admittance control which are revealed in previous section, level of games are designed and their effects are bind with rehabilitation modes as mentioned in Sub-section 1.1.

Since the finger rehabilitation system is a single degree of freedom six-bar mechanism, output motion is constraint to one path which is based on real path that is extracted from the motion capture system. For this reason, controlled character should also be in constrained motion.

5.7.1 Fish feeds tiddler game scenario

The concept of the game is based on a fish which tries to feed tiddler under water. For reaching forage, the fish should follow a path which is exactly the same as the handle of the hand rehabilitation system. Once the fish catches the forage, it swims back to the tiddler and feed it. According to the rehabilitation procedure, there might be some resistances in the system as well. Initially, the system provides some extra assistive motion, where water color is clear blue (Figure 5.12). Water effects and fish are extracted from Unity's Asset Store.

Figure 5.12 Fish feeds tiddler with extra assistance scene.

In the next levels of rehabilitation, extra resistance is presented in the system in terms of admittance dynamics and/or extra resistance offset. In such cases, color of the water becomes blurred and close to yellow, which represents oily resistive water (Figure 5.13).

Figure 5.13 Fish feeds tiddler with extra resistance scene.

In the last levels where virtual spring comes into action, effect of virtual spring is visualized with an extra water flow on the fish. In such cases, virtual spring forces user with the handle along with the power of visualization in the game (Figure 5.14).

Figure 5.14 Fish feeds tiddler with virtual spring effect scene.

As it is one of the main goals of this thesis, game levels are based on the rehabilitation modes as discussed in sub-section 1.1. In first levels of the game, passive mode tries to motivate user and creates plasticity in user's brain. In levels that are related with passive mode, user just place his/her hand on the handle and control system is just processed with velocity control since interaction control is not needed. In the next levels of the game, active assist mode is processed. In these levels, different levels of active assistance with different levels of dynamic virtual gains are used in the game from the most assistive one to the least assistive. Active partial assist mode is in the next levels where only dynamic virtual gains are altered. In the next levels, rehabilitation actually aims to increase muscle activities of user, where active resistance comes into action. In these levels, user tries to beat virtual dynamics as well as extra resistance torque on control system. As discussed above, in game scenes, the water color is changed to blur one, which represents extra dynamics stems from oil and dirt on the water. In final levels, controller scheme as well as the game forces the user with extra spring-related torque. Here, with different levels, forage is also placed in different distances, due to the fact that it is a relatively hard to beat spring force. Also, just reaching the position will not be enough to get the forage, additionally, user must wait for 2 seconds in order to catch it, which causes to stabilize fingers under finite amount of force for a while.

After the forage is caught by the fish, it returns forage back to the tiddler automatically by switching control mode to position control. In such cases, Unity script sends catching information to Simulink structure over UDP. For this purpose, Simulink block of Real-time UDP Receive is used as seen in Figure 5.15. Drive mode is switched according to the information that receives from Unity.
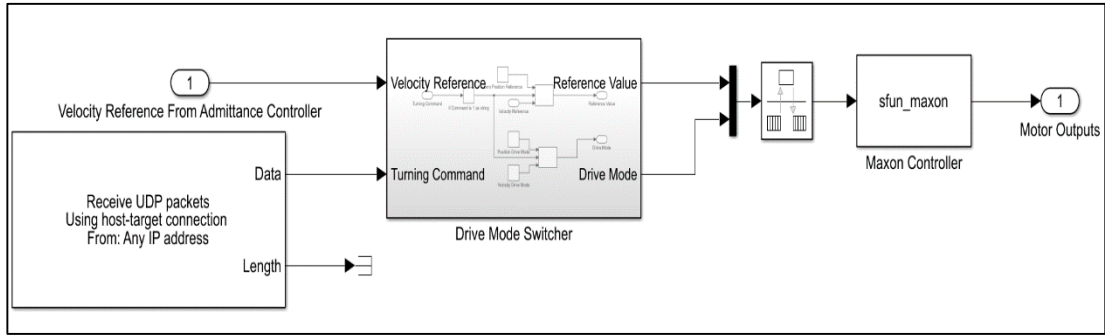
Figure 5.15 Simulink UDP receive block with drive mode switcher.

If the forage is caught by the fish, Unity script send character '1' over UDP, whose corresponding ASCII code is '49'. This means that if data comes from UDP is equal to '49', controller should be switched from velocity to position where reference value is starting position, 0. If broadcasting UDP message from Unity is other that '49', control structure should process its admittance controller as the main controller, where inner loop is the velocity controller (Figure 5.16). Velocity reference is received from admittance control structure as it is.

Figure 5.16 Drive mode switcher blocks.

According to the needs of rehabilitation modes, game with 18 levels is designed and tabulated below to show utilized parameters in each of the levels (Table 5.5).

Table 5.5 Fish feeds tiddler game difficulty level parameters.

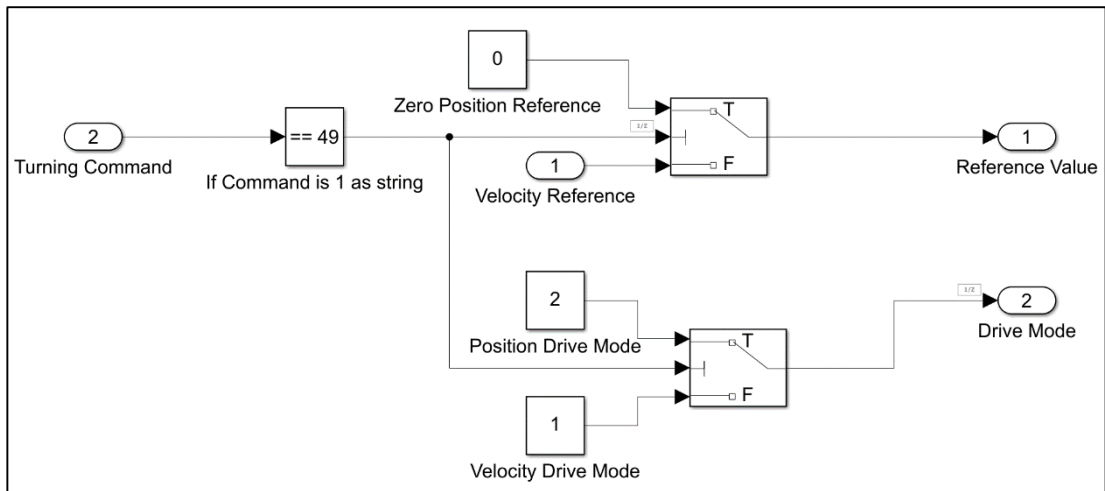| Level	Control Mode	Controller References	Admittance Control Default Parameters	Angle
L1	Passive Mode (Velocity Control)	$V_{ref} = 15$ rpm	$J_v = 0$, $B_v = 0$, $K_v = 0$	Unrelated
L2	Passive Mode (Velocity Control)	$V_{ref} = 10$ rpm	$J_v = 0$, $B_v = 0$, $K_v = 0$	Unrelated
L3	Active Assist Mode	$T_{assist} = 0.4$ Nm	$J_v = 0.1$, $B_v = 0.25$, $K_v = 0$	$\Theta = 137°$
L4	Active Assist Mode	$T_{assist} = 0.4$ Nm	$J_v = 0.25$, $B_v = 0.5$, $K_v = 0$	$\Theta = 137°$
L5	Active Assist Mode	$T_{assist} = 0.2$ Nm	$J_v = 0.1$, $B_v = 0.25$, $K_v = 0$	$\Theta = 137°$
L6	Active Assist Mode	$T_{assist} = 0.2$ Nm	$J_v = 0.25$, $B_v = 0.5$, $K_v = 0$	$\Theta = 137°$
L7	Active Partial Assist Mode	$T_{assist} = 0$ Nm	$J_v = 0.1$, $B_v = 0.25$, $K_v = 0$	$\Theta = 137°$
L8	Active Partial Assist Mode	$T_{assist} = 0$ Nm	$J_v = 0.25$, $B_v = 0.5$, $K_v = 0$	$\Theta = 137°$
L9	Active Resist Mode	$T_{resist} = 0.5$ Nm	$J_v = 0.1$, $B_v = 0.25$, $K_v = 0$	$\Theta = 137°$
L10	Active Resist Mode	$T_{resist} = 0.5$ Nm	$J_v = 0.25$, $B_v = 0.5$, $K_v = 0$	$\Theta = 137°$
L11	Active Resist Mode	$T_{resist} = 0.5$ Nm	$J_v = 0.1$, $B_v = 0.25$, $K_v = 0$	$\Theta = 137°$
L12	Active Resist Mode	$T_{resist} = 0.5$ Nm	$J_v = 0.25$, $B_v = 0.5$, $K_v = 0$	$\Theta = 137°$
L13	Active Resist Mode	$T_{resist} = 0$ Nm	$J_v = 0.1$, $B_v = 0.25$, $K_v = 0.5$	$\Theta = 70°$
L14	Active Resist Mode	$T_{resist} = 0$ Nm	$J_v = 0.1$, $B_v = 0.25$, $K_v = 0.5$	$\Theta = 137°$
L15	Active Resist Mode	$T_{resist} = 0$ Nm	$J_v = 0.1$, $B_v = 0.25$, $K_v = 0.75$	$\Theta = 70°$
L16	Active Resist Mode	$T_{resist} = 0$ Nm	$J_v = 0.1$, $B_v = 0.25$, $K_v = 0.75$	$\Theta = 137°$
L17	Active Resist Mode	$T_{resist} = 0$ Nm	$J_v = 0.1$, $B_v = 0.25$, $K_v = 1.0$	$\Theta = 70°$
L18	Active Resist Mode	$T_{resist} = 0$ Nm	$J_v = 0.1$, $B_v = 0.25$, $K_v = 1.0$	$\Theta = 137°$

5.7.2 Fish feeds tiddler game results

All difficulty levels are played by a healthy person in order to verify proof of concept. Results are tabulated as in Table 5.6.

Table 5.6 Fish feeds tiddler game results in each level.

Level	Peak Torque	Average Torque	Estimated Energy	Level Time
L1	0 Nm	0 Nm	0 Nm rad	3.10 s
L2	0 Nm	0 Nm	0 Nm rad	3.35 s
L3	1.39 Nm	0.37 Nm	22.97 Nm rad	7.15 s
L4	1.70 Nm	0.56 Nm	34.91 Nm rad	8.13 s
L5	1.83 Nm	0.42 Nm	25.78 Nm rad	7.40 s
L6	1.22 Nm	0.56 Nm	34.59 Nm rad	8.75 s
L7	0.99 Nm	0.39 Nm	24.22 Nm rad	8.17 s
L8	1.21 Nm	0.63 Nm	38.79 Nm rad	9.25 s
L9	1.41 Nm	0.82 Nm	50.75 Nm rad	9.25 s
L10	2.50 Nm	1.35 Nm	83.53 Nm rad	11.03 s
L11	2.19 Nm	1.09 Nm	67.89 Nm rad	10.72 s
L12	2.19 Nm	1.56 Nm	96.73 Nm rad	11.90 s
L13	2.02 Nm	0.72 Nm	23.88 Nm rad	4.08 s
L14	2.90 Nm	1.63 Nm	75.20 Nm rad	6.05 s
L15	2.80 Nm	1.43 Nm	32.40 Nm rad	5.25 s
L16	3.62 Nm	2.24 Nm	102.35 Nm rad	6.08 s
L17	4.07 Nm	2.05 Nm	47.21 Nm rad	4.60 s
L18	6.03 Nm	3.49 Nm	154.91 Nm rad	6.10 s

As levels are achieved by the user, torque intention increases. From level 3 to level 13, free motion with assistance or resistance is repeated three times. In these levels, estimated energy is mainly kinetic. After level 13, motion is constraint to virtual spring and there is no repeated motion so this makes duration of the levels smaller. In these levels, estimated energy type is kinetic and potential together with the latter is dominant due to two seconds holding duration. Total locomotion of the handle affects effort dramatically.

The games that aim different functionality give promising results in order to be used in rehabilitation procedures. As future works, proposed system can be tested by a different group of people in order to get feedbacks for improvements. Different games can be designed and these can aim different functionalities by considering rehabilitation modes. At the same time the main aim, the treatment, can be hidden under the game algorithm so that the system can be used as a huge part of the treatment during entertainment. Overall methodology is ready to be used in designing and developing a single degree of freedom rehabilitation mechanism to be implemented to the game rehabilitation system.

6. CONCLUSION

In this thesis, an effective kinematic synthesis and an admittance control methodology is introduced for rehabilitation robotics. Procedure of the methodology stems from the motion capture of a human limb in order to collect a 3D position data and then reduce it to a planar 2D path which is the task of a planar single degree of freedom mechanism. Body guidance method for knee orthosis in sagittal plane is performed to follow required path and orientation. Variable admittance control structure is constructed to be used in a readily built hand rehabilitation device. Free move and spring related move structures are performed in variable admittance controller, which are bind to the individual games in order to increase cognitive effects.

The procedure is started by using Optitrack motion capture system. Finger movement is collected via the system in 3D, then with proposed procedure, path is extracted to a 2D planar motion. Two different data is compared and the procedure is confirmed. The same path extraction methodology is performed with different body parts; hand during combing hair motion and ankle during walking. 3D paths are reduced to a planar path in order to be used as a task for low degree of freedom rehabilitation systems.

As a next step, the path data is used in synthesizing single degree of freedom four-bar mechanisms. In this work, walking data is used and knee orthosis is synthesized as a case study. Body guidance kinematic synthesis is revealed and numeric application is performed.

Variable admittance control structure is applied to readily built hand rehabilitation system. Performance of controller is tested with different rehabilitation procedures as active assist and active resist in order to be part of rehabilitation treatment. Additionally, performance of auto-tuned Maxon Motor is presented as well due to its

effect on the overall results. Since the performance of the system in terms of velocity control is sufficient, parameters are left as they are in auto-tuned.

As final step, the system with tuned admittance control parameters are visualized and controlled along with virtual game environment. This application aims to increase cognitive effects on user. Different games including levels that are related with rehabilitation modes are designed and performed with one healthy middle aged subject.

With this thesis, overall procedure from scratch is revealed and different body parts are used in data collection, path extraction, kinematic syntheses and game rehabilitation with different rehabilitation modes. Overall results show that proposed methodologies are useful for designing a rehabilitation system and given rehabilitation mode controllers with experimental studies show promising results for a single degree of freedom mechanisms to be used in game therapy.

REFERENCES

1. Singh A, Tetreault L, Kalsi-Ryan L, Nouri A, Fehlings MG. Global prevalence and incidence of traumatic spinal cord injury. Clinical Epidemiology. 2014;(6):309-331.

2. www.who.int

3. Boehme AK, Esenwa C, Elkind MSV. Stroke risk factors, genetics and prevention. Circulation Research. 2017;120(3):472-495.

4. Hachinski V, Donnan GA, Gorelick PB, et al. Stroke: working toward a prioritized world agenda. Stroke. 2010;41(6):1084-1099.

5. Orlandi G, Gelli A, Fanucchi S, Tognoni G, Acerbi G, Murri L. Prevalence of stroke and transient ischaemic attack in the elderly population of an Italian rural community. Eur J Epidemiol. 2003;18(9):879-882.

6. Venketasubramanian N, Tan LC, Sahadevan S, et al. Prevalence of stroke among Chinese, Malay, and Indian Singaporeans: a community-based tri-racial cross-sectional survey. Stroke. 2005;36(3):551-556.

7. Şensöz NP, Ülkü TB, Bölük C, Bilgiç A, et al. Stroke epidemiology in Karabük city Turkey: community based study. eNeurologicalSci. 2018;10(1):12-15.

8. Börü ÜT, Kulualp AŞ, Tarhan ÖF, Bölük C, Duman A, et al. Stroke prevalence among the Turkish population in a rural area of Istanbul: a community-based study. SAGE Open Medicine. 2018;6:1-6.

9. Öncel, Ç., Tokgöz, F., Bozkurt, Aİ. et al. Prevalence of cerebrovascular disease: a door-to-door survey in West Anatolia. Neurol Sci. 2014;35, 373–377.

10. Toksoy CK, Bölük C, Börü ÜT, Akın S, Yılmaz AY, Duman SC, Taşdemir M. Stroke prevalence in a coastal town on the black sea coast in Turkey: community based study. Neurology Research International; 2018.

11. Hebert D, Lindsay MP, McIntyre A, Kirton A, et al. Canadian stroke best practice recommendations: stroke rehabilitation practice guidelines. International Journal of Stroke. 2016;11(4):459-484.

12. Spring S. Stroke Rehabilitation. Medifocus Guide; 2009

13. Van Der Lee JH, Snels IA, Beckerman H, Lankhorst GJ, Wagenaar RC, Bouter LM. Exercise therapy for arm function in stroke patients: a systematic review of randomized controlled trials. Clinical Rehabilitation. 2001;15(1):20-31.

14. Harwin WS, Rahman T, Foulds RA. A review of design issues in rehabilitation robotics with reference to North American research. IEEE Transactions on Rehabilitation Engineering. 1995;3(1)3-13.

15. Lum PS, Burgar CG, Shor PC, Majmundar M, Van der Loos M. Robot-assisted movement training compared with conventional therapy techniques for the rehabilitation of upper-limb motor function after stroke. Archives of Physical Medicine and Rehabilitation. 2002;83(7):952-959.

16. Saglia JA, Tsagarakis NG, Dai JS, Caldwell DG. A High-performance Redundantly Actuated Parallel Mechanism for Ankle Rehabilitation. The International Journal of Robotics Research. 2009;28(9):1216–1227.

17. Poli P, Morone G, Rosati G, Masiero S. Robotic technologies and rehabilitation: new tools for stroke patients' therapy. Biomed Research International; 2013.

18. Proietti T, Crocher V, Roby-Brami A, Jarrassé N. Upper-Limb robotic exoskeletons for neurorehabilitation: a review on control strategies. IEEE Reviews in Biomedical Engineering. 2016;9:4-14.

19. Hogan N, Krebs HI. Interactive robots for neuro-rehabilitation. Restor Neurol Neurosci. 2004;22(3-5):349-358.

20. Carmichael MG, Liu D. Admittance control scheme for implementing model-based assistance-as-needed on a robot. 35th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC), Osaka; 2013:870-873.

21. Alizade R, Bayram Ç, Gezgin E. Structural synthesis of serial platform manipulators. Mechanism and Machine Theory. 2007;42(5):580-599.

22. Alizade R, Can FC, Gezgin E. Structural synthesis of Euclidean platform robot manipulators with variable general constraints. Mechanism and Machine Theory. 2008;43(11):1431-1449.

23. Alizade R, Selvi Ö, Gezgin E. Structural design of parallel manipulators with general constraint one. Mechanism and Machine Theory. 2010;45(1):1-14.

24. Alizade R, Gezgin E. Synthesis of function generating spherical four bar mechanism for the six independent parameters. Mechanism and Machine Theory. 2011;46(9):1316-1326.

25. Dede MIC, Comen D, Berker G, Erkilincoglu I. Kinematic synthesis of the motion generation of linkages. Scientific Journal of IFToMM Problems of Mechanics. 2012;48(3):13–20.

26. Gezgin E, Chang PH, Akhan AF. Synthesis of a watt II six-bar linkage in the design of a hand rehabilitation robot. Mechanism and Machine Theory. 2016;104(11):177-189.

27. Hartenberg RS, Denavit J. Kinematic synthesis of linkages. McGraw-Hill Book Co; 1964.

28. www.optitrack.com

29. Vysocky A, Novak P. Human-robot collaboration in industry. MM Science Journal. 2016;9(2):903–906.

30. Roveda L, Maskani J, Franceschi P. et al. Model-based reinforcement learning variable impedance control for human-robot collaboration. Journal of Intelligent and Robotic Systems. 2020. Available from: https://doi.org/10.1007/s10846-020-01183-3

31. Dong J, Xu J. Physical human-robot interaction force control method based on adaptive variable impedance. Journal of the Franklin Institute. 2020;357:7864-7878

32. Krebs H, Palazzolo J, Dipietro L. et al. Rehabilitation robotics: performance-based progressive robot-assisted therapy. Autonomous Robots. 2003;15:7–20. Available from: https://doi.org/10.1023/A:1024494031121

33. Casas J, Cespedes N, Múnera M, Cifuentes CA. Human-robot interaction for rehabilitation scenarios. Control Systems Design of Bio-Robotics and Bio-mechatronics with Advanced Applications. Academic Press, 2020:1-31.

34. Mota P, Rognon JP, Le-Huy H. Digital position servo system: a state variable feedback system. IEEE Transactions on Industry Applications. 1984;IA-20(6):1473-1481. Available from: doi:10.1109/TIA.1984.4504630

35. Sepe RB, Lang JH. Real-time adaptive control of the permanent-magnet synchronous motor. IEEE Transactions on Industry Applications. 1991;27(4):706-714. Available from: doi:10.1109/28.85486

36. Ohnishi K, Murakami T. Advanced motion control in robotics. 15th Annual Conference of IEEE Industrial Electronics Society, Philadelphia, PA, USA. 1989;2:356-359. Available from: doi:10.1109/IECON.1989.69658

37. Siciliano B, Villani L. Robot force control. Kluwer, Boston; 1991.

38. Zeng G, Hemami, A. An overview of robot force control. Robotica. 1997;15(5):473-482.

39. Dede MIC. Position/Force control of robot manipulators: Middle East Technical University, Ankara, Turkey; 2003.

40. Hogan N. Impedance control: an approach to manipulation. American Control Conference, San Diego, CA, USA;1984:304-313.

41. Whitney DE. (1977). Force feedback control of manipulator fine motions. Journal of Dynamic Systems, Measurement and Control. Jun 1977;99(2):91-97.

42. Shoushtari AL, Dario P, Mazzoleni S. A review on the evolvement trend of robotic interaction control. Industrial Robot. 2016;43(5):535-551.

43. Villani L, De Schutter J. Force Control. Springer Handbook of Robotics. Springer, Berlin, Heidelberg; 2008.

44. Song P, Yu Y, Zhang X. Impedance control of robots: an overview. 2nd International Conference on Cybernetics, Robotics and Control (CRC), Chengdu; 2017:51-55.

45. Hussain S, Xie SQ, Jamwal PK. Adaptive impedance control of a robotic orthosis for gait rehabilitation. IEEE Transactions on Cybernetics. 2013;43(3):1025-1034.

46. Tsoi YH, Xie SQ. Impedance control of ankle rehabilitation robot. IEEE International Conference on Robotics and Biomimetics, Bangkok; 2009:840-845.

47. Yang Y, Wang L, Tong J, Zhang L. Arm rehabilitation robot impedance control and experimentation. IEEE International Conference on Robotics and Biomimetics, Kunming; 2006:914-918.

48. Jamwal PK, Hussain S, Ghayesh MH, Rogozina SV. Impedance control of an intrinsically compliant parallel ankle rehabilitation robot. IEEE Transactions on Industrial Electronics. 2016;63(6);3638-3647.

49. Wu Q, Wang X, Chen B, Wu H. Development of a minimal-intervention-based admittance control strategy for upper extremity rehabilitation exoskeleton. IEEE Transactions on Systems, Man, and Cybernetics: Systems. 2018;48(6):1005-1016.

50. Ayaş MS, Altaş İH. Fuzzy logic based adaptive admittance control of a redundantly actuated ankle rehabilitation robot. Control Engineering Practice. 2017;59(2):44-54.

51. Luna CO, Rahman MH, Saad M, Archambault PS, Ferrer SB. Admittance-based upper limb robotic active and active-assistive movements. International Journal of Advanced Robotic Systems; 2015.

52. Saglia JA, Tsagarakis NG, Dai JS, Caldwell DG. Control strategies for patient-assisted training using the ankle rehabilitation robot (ARBOT). IEEE/ASME Transactions on Mechatronics. 2013;18(6):1799-1808.

53. Seraji H. Adaptive admittance control: an approach to explicit force control in compliant motion. Proceedings of the 1994 IEEE International Conference on Robotics and Automation, San Diego, CA, USA. 1994;4:2705-2712.

54. Lecours A, Mayer-St-Onge B, Gosselin C. Variable admittance control of a four-degree-of-freedom intelligent assist device. 2012 IEEE International Conference on Robotics and Automation, Saint Paul, MN. 2012:3903-3908.

55. Koçak M, Baser Ö, Gezgin E. Kinematic synthesis of a single degree of freedom mechanism to be utilized for upper extremity rehabilitation. 1st International Congress on Robotic Technology and Rehabilitation Proceedings and Abstracts; 74-84.

56. Terada H, Makino K, Ishida K, Ogura T. Development of a knee joint assistive-mechanism adapted for bilateral roll-back motion. Mechanisms and Machine Science Springer, Cham. 2019;59.

57. Olinski M, Gronowicz A, Handke A, Ceccarelli M. Design and characterization of a novel knee articulation mechanism, International Journal of Applied Mechanics and Engineering. 2016;21(3):611-622.

58. Kawamoto H, Sankai Y. Power assist system HAL-3 for gait disorder person. Computers Helping People with Special Needs. ICCHP 2002. Lecture Notes in Computer Science. Springer, Berlin, Heidelberg; 2002;2398.

59. Jezernik S, Colombo G, Keller T. Robotic orthosis Lokomat: a rehabilitation and research tool. Neuromodulation. 2003;6:108–115.

60. Oberg K. Knee mechanisms for through-knee prostheses. Prosthetics and Orthotics International. 1983;7(2):107-112.

61. Chakraborty JK, Patil KM. A new modular six-bar linkage trans-femoral prosthesis for walking and squatting. Prosthetics and Orthotics International. 1994;18(2):98-108.

62. Radcliffe CW. Four-bar linkage prosthetic knee mechanisms: kinematics, alignment and prescription criteria. Prosthetics and Orthotics International. 1994;18(3):159-173.

63. Greene MP, et al. Four bar linkage knee analysis. Orthotics and Prosthetics. 1983;37(1):15-24.

64. Davis RB, Õunpuu S, DeLuca PA, Romness MJ. Clinical gait analysis and its role in treatment decision-making. MedGenMed. 1999;1(1) [formerly published in Medscape Orthopaedics & Sports Medicine eJournal 1998;2(5)]. Available from: http://www.medscape.com/viewarticle/440148

65. Pugh S. Total design – integrated methods for successful product engineering. Addison Wesley Publishing; 1990.

66. Simoneau GG. Kinesiology of walking. Kinesiology of the Musculoskeletal System: Foundations of Physical Rehabilitation. 2010:636.

67. Eugenio. Commanding maxon motors EPOS2 motor controller from MATLAB Available from: (https://www.mathworks.com/matlabcentral/fileexchange/53735-commanding-maxon-motors-epos2-motor-controller-from-matlab), MATLAB Central File Exchange. Retrieved May 30, 2020.

68. Koçak M, Ayar O, Gezgin E. Preliminary study on the admittance control of a hand rehabilitation system. Medical Technologies Congress (TIPTEKNO), Izmir, Turkey. 2019:1-4.

69. Lange B, Flynn S, Rizzo A. Initial usability assessment of off-the-shelf video game consoles for clinical game-based motor rehabilitation. Physical Therapy Reviews. 2009;14(5):355-363.

70. Mirelman A, Bonato P, Deutsch JE. Effects of training with a robot-virtual reality system compared with a robot alone on the gait of individuals after stroke. Stroke. 2009;40(1):169-174.

71. www.unity.com

72. https://socket.io/

73. Postel J. Transmission control protocol; 1981.
74. Postel J. User datagram protocol; 1980.
75. Comer D. Computer Networks and Internets: Prentice Hall; 2009.

CURRICULUM VITAE

Name Surname:	**Mertcan KOÇAK**
Place and Date of Birth:	**ANKARA 11.01.1991**
Address:	**İzmir Katip Çelebi Üniversitesi Mühendislik ve Mimarlık Fakültesi, Mekatronik Mühendisliği Bölümü, Balatçık Kampüsü, Çiğli/İzmir, Türkiye**
E-mail:	**kocakmertcan@gmail.com**

Education:

B.Sc. Mechanical Engineering Automotive Engineering Program in Hacettepe University, Ankara, Turkey (2009 - 2014)

List of Publications:

Doğan M, Koçak M, Kılınç ÖO, Ayvat F, Sütçü G, Ayvat E, Kılınç M, Ünver Ö, Yıldırım SA. Functional range of motion in the upper extremity and trunk joints: Nine functional everyday tasks with inertial sensors. Gait and Posture. 2019: 70: 141-147. Available from: https://doi.org/10.1016/j.gaitpost.2019.02.024

Amini A, Başlamışlı SÇ, İnce B, Koçak M. The effect of different gear ratio selection algorithms on the efficiency of conventional and parallel hybrid drivetrains. Anadolu University Journal of Science and Technology A – Applied Sciences and Engineering. 2017: 18-4.

Conference Papers:

Koçak M, Ayar O, Gezgin E. Preliminary study on the admittance control of a hand rehabilitation system. TıpTekno 2019, Kuşadası. Available from: https://doi.org/10.1109/TIPTEKNO.2019.8895238

Koçak M, Can FC, Gezgin E. Design of a graphical user interface for the structural synthesis of parallel manipulators with single platform. Interactive Collaborative Robotics. ICR 2019. Lecture Notes in Computer Science, Vol 11659. Springer, Cham(2019). Available from: https://doi.org/10.1007/978-3-030-26118-4_18

Mihçin Ş, Köse H, Çizmecioğulları S, Koçak M, Çıklacandır S, Tosun A, Akan A. Investigation of wearable motion capture system towards biomechanical modelling. IEEE International Symposium on Medical Measurements and Applicatıons (MeMeA), June 2019, İstanbul. Available from: https://doi.org/10.1109/MeMeA.2019.8802208

Doğan M, Koçak M, Sütçü G, Kılınç ÖO, Ayvat F, Ayvat E, Kılınç M, Yıldırım SA. Determination of upper extremity and trunk kinematics in washing the face activity. The International Congress on Physiotechnotherapy (ICPTT), May 2018, Sarajevo (Paper Number: 4302001).

Koçak M, Başer Ö, Gezgin E. Kinematic synthesis of a single degree of freedom mechanism to be utilized for upper extremity rehabilitation. 1st International Conference on Robotic Technology and Rehabilitation (ICRAR) (2018), İstanbul.

Amini A, Başlamışlı SÇ, İnce B, Koçak M. Combination of optimal gear selection and equivalent consumption minimization strategies in parallel hybrid vehicles. TOK'16 National Congress of Automatic Control (2016), Eskişehir, Turkey.

Başlamışlı SÇ, Koçak M, İnce B, Testik MC. Driving cycles for Turkey: Konya case study. OTEKON '16 8th Congress of Automotive Technologies (2016), Bursa (Paper No: B111).

Başlamışlı SÇ, İnce B, Koçak M, Saygılı HH. Design of energy management system algorithms for enhancing fuel economy of electric/hybrid city busses. OTEKON '16 8th Congress of Automotive Technologies (2016), Bursa (Paper No: B101).

Koçak M, Kutluay E. A matlab based tool for analysis of steering configuratıons of multiaxle vehicles. OTEKON '16 8th Congress of Automotive Technologies (2016), Bursa (Paper No: B209).