VUB VRIJE UNIVERSITEIT BRUSSEL

VUB BRUBOTICS
HUMAN ROBOTICS
RESEARCH CENTER

UNIVERSITEIT
GENT

Master thesis submitted in order to be awarded the Master's Degree in Biomedical engineering

MUSCULOSKELETAL MODELING WITHIN THE CONTEXT OF EXOSKELETAL ASSISTANCE

Zarah Van Schoor

2022/2023

Promotor:	prof. dr. ir. Tom Verstraten
Supervisors:	dr. ir. Kevin Langlois
	dr. Sander De Bock

Engineering

Acknowledgments

First of all I would like express my gratitude to my promotor professor dr. ir. Tom Verstraten for the opportunity to conduct this research, as well as for the interesting tips, conversations and feedback.

Secondly I want to thank my supervisors dr. ir. Kevin Langlois and dr. Sander De Bock for all the help during this thesis and for always being available for any questions I had.

I also express my thanks to dr. Arthur van der Have for aiding me with understanding how to use the OpenSim program.

And lastly I want to extend my thanks to ir. Christopher Van Vlerken for his contribution.

Abstract

The objective is to construct a musculoskeletal model of the upper extremities, focusing on the shoulder joint, to accurately estimate forces acting on the joints. This model can advance insights in human-robot interaction and facilitate the development of more effective exoskeletons. In this thesis the model is used to investigate the impact of exoskeletal assistance through force versus through torque.

To investigate this, an existing model in OpenSim, that meets the requirements, is used. A flexion-extension movement of the shoulder is performed in real-life and recorded with XSens DOT IMUs. From this the joint coordinates are computed and loaded in OpenSim. First a validation is performed by comparing measured and computed muscle activity. After validation, simulations are done with different levels of exoskeletal assistance, resistance, and no assistance. This both with force and then torque as method of assistance. For each of these conditions the reaction forces at the shoulder are computed.

These simulations showed that reaction forces are higher when applying an external torque, and lower when applying an external force, compared to when applying no external load. The reaction forces in x-, y- and z-direction for when the humerus reaches an angle of 90° are shown in Figure 1, both for force (left) and torque (right) as assistance method. Reaction moments showed little difference between the different simulations, but for external torque the reaction moment slightly increased. When applying a resisting load, the reaction force increased, the reaction moment also increased, but the increase was small.

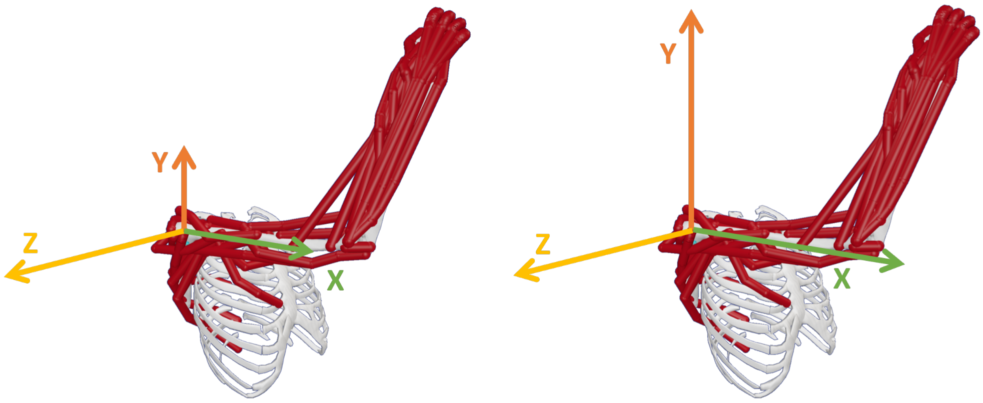


Figure 1: Reaction forces at the shoulder joint when the humerus reaches an angle of 90°. On the left for assistance through force, and on the right for assistance through torque.

Contents

Acknowledgments iii

Abstract v

1 Introduction 1

1.1 Social impact 1

1.2 Research question 1

1.3 Thesis outline 3

2 The shoulder joint complex 5

2.1 Shoulder joints 5

2.2 Shoulder motion 5

2.3 Shoulder muscles 7

2.3.1 Muscles that position the pectoral girdle 7

2.3.2 Muscles that move the glenohumeral joint 7

2.4 Shoulder ligaments 8

2.5 Musculoskeletal model requirements 8

3 Musculoskeletal modeling 13

3.1 Musculoskeletal modeling software 13

3.2 Overview of existing upper-limb OpenSim models 14

3.2.1 Arm26 14

3.2.2 Dynamic arm simulator (DAS3) 14

3.2.3 Upper extremity kinematic model 15

3.2.4 Upper extremity dynamic model 16

3.2.5 Model of the scapulothoracic joint 17

3.2.6 Thoracoscapular shoulder model 17

3.3 Review of OpenSim models 20

3.4 Simulations in OpenSim 21

4 Methods 23

4.1 Validation 23

4.2 Torque versus force assistance 25

4.2.1 Motion description 25

4.2.2 Robotic assistance description 29

5 Results 33
5.1 Validation 33
5.2 Torque versus force assistance 39

6 Discussion and future work 45
6.1 Conclusion 45
6.2 Limitations and future work 45

Appendix 47

Bibliography 51

Chapter 1

Introduction

When performing daily activities such as walking, running, taking something from a shelf, brushing your hair, and so on, we don't think about the acceleration and/or deceleration of our body segments and how our muscles are controlled to obtain the required joint movements for the intended motion. However, the underlying mechanism is highly complex and, to improve applications in health science and engineering, for instance to design exoskeletons, gaining an understanding of this mechanism is important [2].

1.1 Social impact

Exoskeletons are being researched more and more, and the sight of employees wearing exoskeletons when for example working in warehouses or on construction sites might not be unthinkable in the near future. But also in the field of rehabilitation exoskeletons could be an added value. But, if people use exoskeletons in repetitively, it is important to know that the exoskeleton is transferring the assistance as efficiently and optimally as possible, and foremost that it is not doing any harm by seriously increasing the joint load. This is mainly important when an exoskeleton is used in long term.

1.2 Research question

Usually the goal of exoskeletons is to reduce the needed muscle force. And thus, exoskeletons are often validated by showing that the muscle activity has decreased when using the exoskeleton. Reaction forces acting on the joints however are rarely studied, with all potentially negative consequences that entails.

Generally exoskeletal support is given through force application, however, in doing so, there will always be a reaction force F_R. The hypothesis holds that the application of a torque will result in lower joint reaction loads than when applying a force, because applying a force will create a reaction force at the level of the shoulder to keep the shoulder in place, while applying a torque will create a reaction moment at the shoulder, but since the shoulder allows for rotation, the reaction moment will be zero, or very small since the shoulder is not a perfect hinge joint. This reasoning however does not include the contribution of muscles. When looking at Figure 1.1 (right) and Equations 1.1, it can be seen that when F_{exo} increases, another term has the decrease in order maintain the force balance. Usually exoskeletons are made with the goal to reduce muscle forces, which means that F_{muscle} should decrease. However, the relation between F_{exo} and F_{muscle} is not known, and thus the influence on R remains a question.

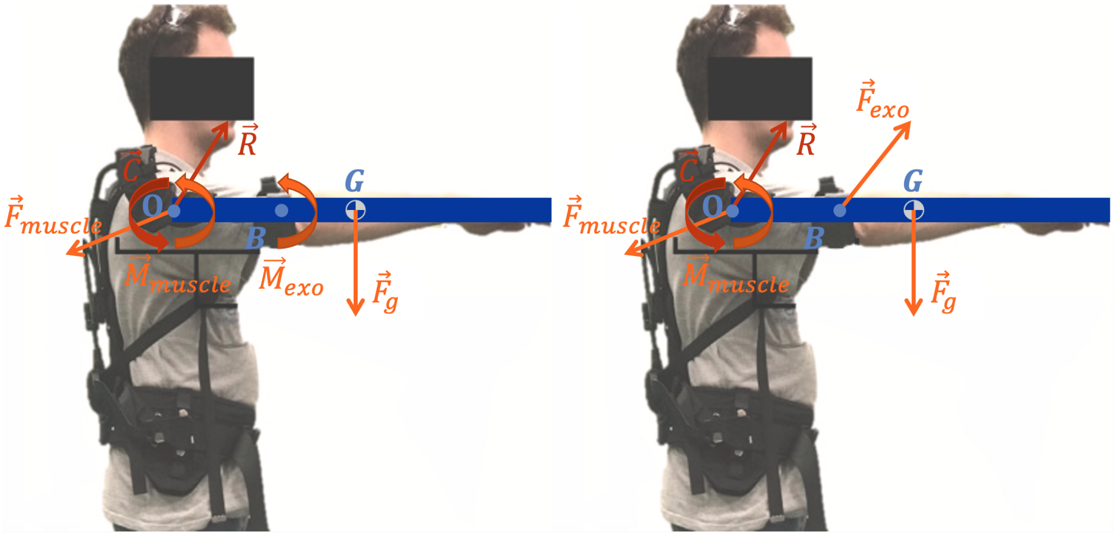


Figure 1.1: Exoskeletal assistance can be given through torque (left) and/or through force (right). The hypothesis holds that giving assistance solely through torque, will lead to lower reaction forces in the shoulder since there will be no reaction forces needed to hold the humerus in place, which will be the case when giving assistance solely through force.

$$\begin{aligned} 0 &= R^X + F^X_{exo} + F^X_{muscle} \\ 0 &= R^Y + F^Y_{exo} + F^Y_{muscle} + F_g \\ 0 &= \overline{C} + \overline{C}_{muscle} + \overline{OB} \times \overline{F}_{exo} + \overline{OG} \times \overline{F}_g \end{aligned} \tag{1.1}$$

When an external torque is applied instead of a force, the muscle force F_{muscle} should decrease, leading to a decrease in R, both forces should be be equal and opposite (in x-direction), in order to maintain the force balance.

$$\begin{aligned} 0 &= R^X + F^X_{muscle} \\ 0 &= R^Y + F^Y_{muscle} + F_g \\ 0 &= \overline{C} + \overline{M}_{exo} + \overline{C}_{muscle} + \overline{OG} \times \overline{F}_g \end{aligned} \tag{1.2}$$

There are however 2 challenges. A first challenge is the assessment of joint reaction forces at the level of the joints, which can only really be measured in an invasive way, and secondly, how to get a robot to transfer just a torque.

A way of to overcome these challenges is by using musculoskeletal models. A musculoskeletal model is a computational model that consist of a skeleton of rigid body segments (bones) connected by joints. Muscles span over the joints and can generate forces to move the body parts. The main advantage of these musculoskeletal models is that they include models of muscles and thus allow to characterize muscle forces, which are then also considered in the computation of reaction forces. These models are useful for biomechanical analyses, and simulations are often used for estimating internal forces, joint kinematics and joint moments [2] [16].

Thus, if simulations with a musculoskeletal model confirms the hypothesis, this gives extra rationale to give this research question more attention and to try it in real life.

So, the goal of this thesis is to employ a musculoskeletal model of the upper extremities, focusing on the shoulder joint, to estimate forces acting on the joints as a result of exoskeletal assistance through force or torque. This model can advance insights in human-robot interaction and facilitate the development of more effective exoskeletons.

1.3 Thesis outline

Chapter 2 provides a brief description of the shoulder followed by an introduction to OpenSim and a comparison of existing upper-limb musculoskeletal models in Chapter 3. Chapter 4 describes the performed experiments, being first a validation of the simulations by comparing measured electromyography (EMG) data with estimated muscle activation of the model for the same motion [4] [28], and an inspection of the required moment to achieve to movement computed by OpenSim and manually calculated. This is followed by a description of the experiment and data processing steps for comparing force to torque assistance. The results of the validation and experiment simulations are discussed in Chapter 5, and finally a general conclusion and possible future work is given in Chapter 6.

Chapter 2

The shoulder joint complex

This chapter will shed some light on the complex body part that the shoulder is. Sections 2.1, 2.2, 2.3 and 2.4 give some anatomical information about the shoulder joints, motion, muscles and ligaments respectively. The chapter is concluded with the requirements for a musculoskeletal model for this thesis based on this information in section 2.5.

2.1 Shoulder joints

The shoulder connects the upper arm to the thorax and is composed of the clavicle, scapula, humerus and sternum (Figure 2.1). These bones are connected through 4 joints shown in Figure 2.2 and described in more detail below [36][44].

- Glenohumeral joint: ball-and-socket joint connecting the head of the humerus into the glenoid cavity [19] of the scapula [36].
- Acromioclavicular joint: gliding synovial joint (also called gliding diarthrosis) allowing the clavicle and acromion (part of scapula) to move side-to-side, up and down, and diagonally [36].
- Sternoclavicular joint: gliding synovial joint (gliding diarthrosis) allowing the clavicle and sternum to move side-to-side, up and down, and diagonally [36].
- Scapulothoracic joint: not considered as a true since the bones are not connected by ligaments, the scapula glides along the rib cage [36].

Due to this network of bones, joints and soft tissues and the spheroid shape of the glenohumeral joint (large ball in small socket), the shoulder has a very large range of motion. But this also leads to the shoulder being prone to dislocation and other injures, which shows the importance of shoulder stability [36][44].

2.2 Shoulder motion

The wide range of motion comes at the expense of stability. The shoulder is stabilized by its surrounding skeletal muscles and various ligaments. The majority of support is coming from the supraspinatus, infraspinatus, teres minor, and subscapularis muscles and their tendons, also

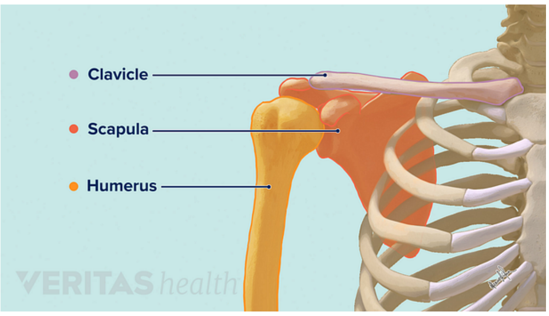


Figure 2.1: The shoulder joint complex consists of three bones: the clavicle, the scapula, and the humerus [36].

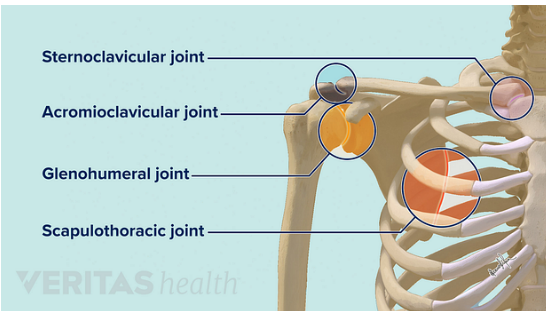


Figure 2.2: There are four joints in the shoulder joint complex [36].

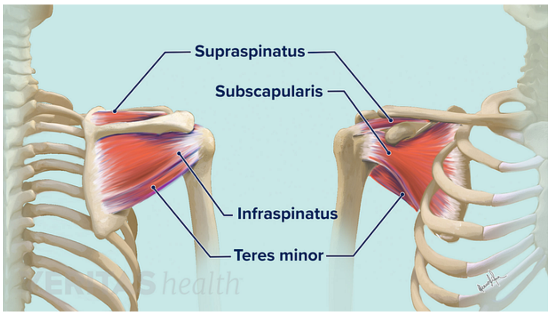


Figure 2.3: The rotator cuff is a group of four muscles that stabilize the shoulder joint [36].

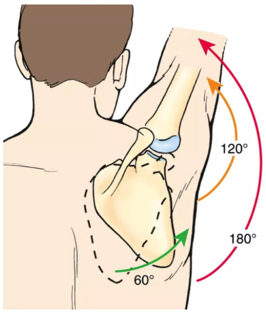


Figure 2.4: The shoulder rhythm describes the kinematic interaction between the scapula and humerus when elevating the arm [43].

know as the rotator cuff (Figure 2.3). They reinforce the joint capsule and limit the range of movement [19].

A shoulder movement is a combination of movement of the four shoulder joints. Each joint contributes continuously, but at a various rate and a different phase of the shoulder movement [43].

The kinematic interaction between the scapula and humerus when elevating the arm is described by the shoulder rhythm (also scapulohumeral or glenohumeral rhythm), which is the ratio (R) of glenohumeral (GH) to scapulothoracic (ST) rotation, R = GH/ST. A general ratio of 2:1 is used (as shown in Figure 2.4), however this is oversimplified since the ratio is dependent on several factors such as the arc and plane of shoulder elevation, the shoulder loading, and varies between individuals and ages [43] [50] [24] [33].

In general, for small shoulder angles, under 30°, the contribution of the scapulothoracic joint is very small and the motion is mainly due to the glenohumeral joint (thus a larger shoulder rhythm). As the shoulder angle increases, the contribution of the scapulothoracic joint also increases, in a non-linear trend, thus decreasing the shoulder rhythm [50] [43].

2.3 Shoulder muscles

The muscles involving the shoulder can be divided into two groups, the muscles that position the pectoral girdle (also called shoulder girdle, containing the clavicle and scapula), shown in Figure 2.5, and the muscles that move the glenohumeral joint, shown in Figure 2.6.

2.3.1 Muscles that position the pectoral girdle

The **trapezius** is a large superficial muscle that originates along the spine (on the spinous processes of vertebrae C7 to T12), the nuchal ligament and the external occipital protuberance. It inserts on the acromion (clavicle) and scapular spine. Each region (upper, middle, lower) of the trapezius can contract independently, and depending on which region is active, the trapezius can elevate, retract, depress, or rotate the scapula upward, elevate the clavicle, or extend the neck [20] [45].

The **serratus anterior** originates along several ribs and inserts on the anterior surface of the vertebral border of scapula, thus running between the scapula and the ribs. When contracting, the serratus anterior protracts the scapula, swinging the shoulder superiorly [20].

The **subclavius** is a deeper laying muscle that originates on the first rib and inserts on the inferior border of the clavicle. This muscle works in synergy with the **pectoralis minor**, originating on the anterior-superior surfaces of 3 to 4 ribs and inserting on the coracoid process of the scapula. Contraction of these muscles depresses and protracts the scapular end of the clavicle, which rotates the scapula so that the glenoid cavity moves inferiorly [20].

The **rhomboid major** and **minor** and **levator scapulae** muscles lay underneath the trapezius muscle and insert along the vertebral border of the scapula. The rhomboids originate on the spinous processes of several superior thoracic vertebrae and the levator scapulae originates on the transverse processes of the first four cervical vertebrae. They work in synergy to elevate, retract and rotate the scapula, tilting the glenoid cavity inferiorly [20] [41] [42].

2.3.2 Muscles that move the glenohumeral joint

The **deltoid** muscles (posterior, anterior and medial) originate on the clavicle and scapula (acromion and adjacent scapular spine) and inserts on the deltoid tuberosity of the humerus. It is the major shoulder abductor, but the anterior part also contributes to the shoulder flexion and medial rotation and the posterior part contributes to the extension and later rotation [20].

The **supraspinatus** muscle assists in the abduction of the shoulder. It originates on the supraspinous fossa of the scapula and inserts on the greater tubercle of the humerus [20].

The **subscapularis** muscle originates at the subscapular fossa of the scapula and inserts on the lesser tuberosity of the humerus. This muscle performs medial rotation at the shoulder [20].

The **teres major** performs extension, adduction, and medial rotation of the shoulder. It originates on the inferior angle of the scapula and inserts on the medial lip of intertubercular groove of the humerus [20].

The **teres minor** has its origin at the lateral border of the scapula and inserts on the greater tubercle of the humerus. Contraction initiates lateral rotation of the shoulder [20].

The **infraspinatus** muscle also performs lateral rotation of the shoulder. It is originated at the infraspinous fossa of scapula and inserts on the greater tubercle of the humerus [20].

The **coracobrachialis** originates at the coracoid process of the scapula and inserts on the medial margin of the shaft of the humerus. It is the only muscle originating on the scapula that produces flexion and adduction of the shoulder [20].

The **pectoralis major** muscle originates from the second to sixth rib, the body of the sternum and the inferior, medial portion of the clavicle and inserts on the crest of the greater tubercle and lateral lip of intertubercular groove of the humerus. It produces flexion, adduction and medial rotation of the shoulder [20].

The **latissimus dorsi** muscle originates on the spinous processes of the inferior thoracic and all lumbar vertebrae, ribs 8–12, and the thoracolumbar fascia and inserts on the floor of intertubercular groove of the humerus. It produces extension, adduction, and medial rotation of the shoulder [20].

The **biceps brachii** help shoulder flexion, and the **triceps brachii** help shoulder extension.

2.4 Shoulder ligaments

Most of the stabilizing support comes from muscles, however, ligaments aid to reinforce and stabilize the shoulder [19].

The capsular ligament or **glenohumeral** ligament consist of 3 parts (superior, inferior and medial) and connects the edge of the glenoid fossa to the neck of the humerus. From all shoulder ligaments this is the main source of stability, it reinforces the anterior aspect of the shoulder joint [37] [26]. More specifically, all 3 parts limit the external rotation of the humerus, and additionally the superior glenohumeral ligament limits inferior translation, the medial glenohumeral ligaments limits anterior translation and the inferior glenohumeral ligaments limits internal rotation and superior and anterior translation [40].

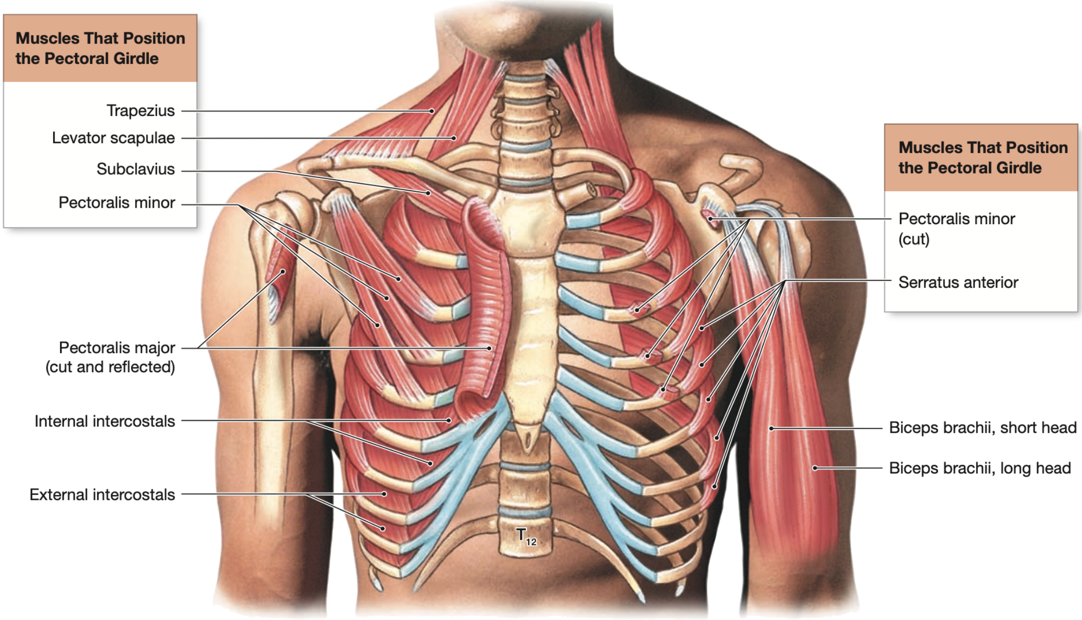
The **coracohumeral** ligament connects the base of the coracoid process to the greater tubercle of the humerus and reinforces the superior aspect of the shoulder joint. It limits the inferior translation and external rotation of the humerus [26] [35].

The **coracoacromial** ligament connects the acromion to the coracoid process of the scapula and prevents superior displacement of the humeral head [26]

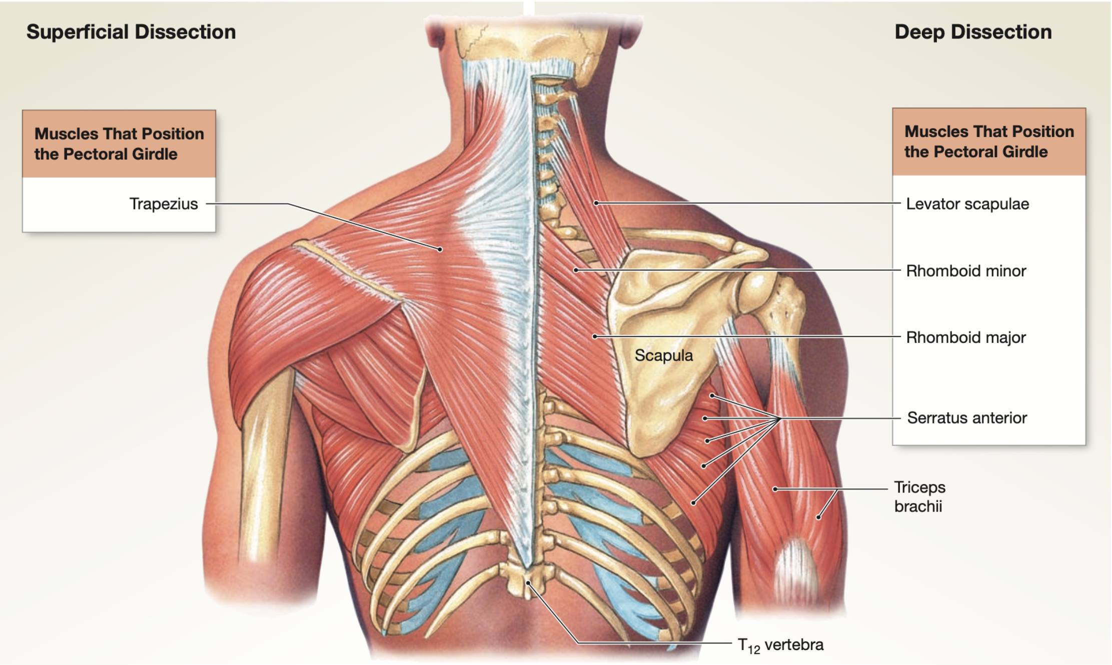
The **coracoclavicular** and **acromioclavicular** ligaments connect the clavicle to the scapula and reinforce the alignment of the clavicle with relation to the scapula [26].

2.5 Musculoskeletal model requirements

The shoulder muscles are activated in different levels for different types of movements. The muscles that are of most interest in the light of this thesis are the flexor and extensor muscles (deltoid, coracobrachialis, pectoralis major, teres major, latissimus dorsi) and the shoulder stabilizing muscles (deltoid, rotator cuff muscles: supraspinatus, infraspinatus, teres minor, subscapularis; trapezius, serratus anterior, rhomboids, pectoralis minor and major), as well as the shoulder stabilizing ligaments (glenohumeral ligaments) [23] [39] [19] [55].

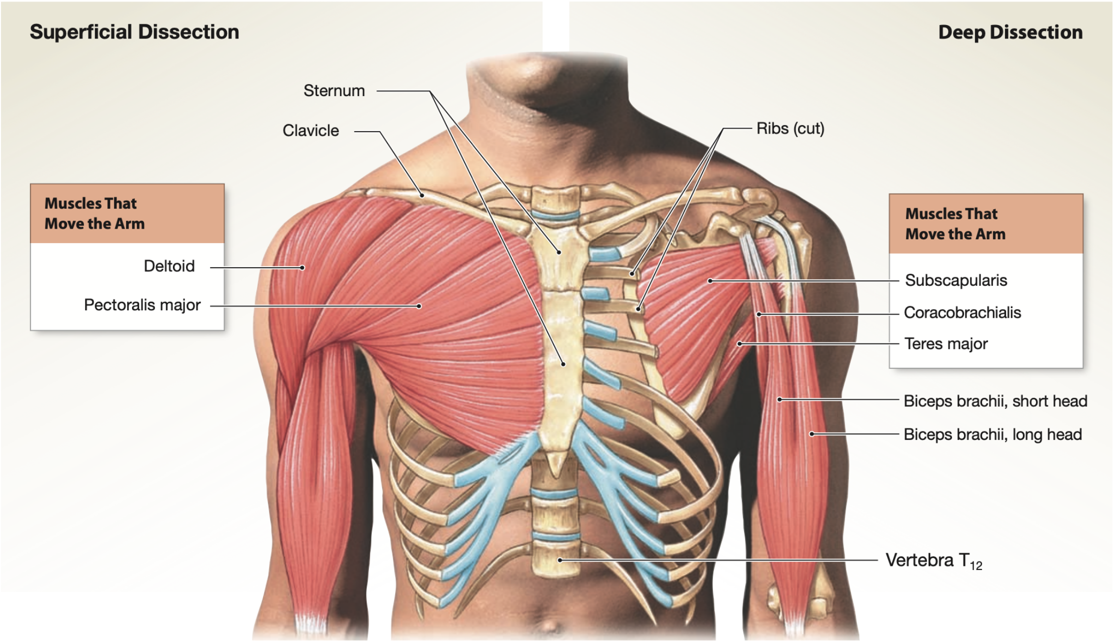

Muscles That Position the Pectoral Girdle
Trapezius
Levator scapulae
Subclavius
Pectoralis minor
Pectoralis major (cut and reflected)
Internal intercostals
External intercostals
T_{12}
Muscles That Position the Pectoral Girdle
Pectoralis minor (cut)
Serratus anterior
Biceps brachii, short head
Biceps brachii, long head

(a) Anterior view.

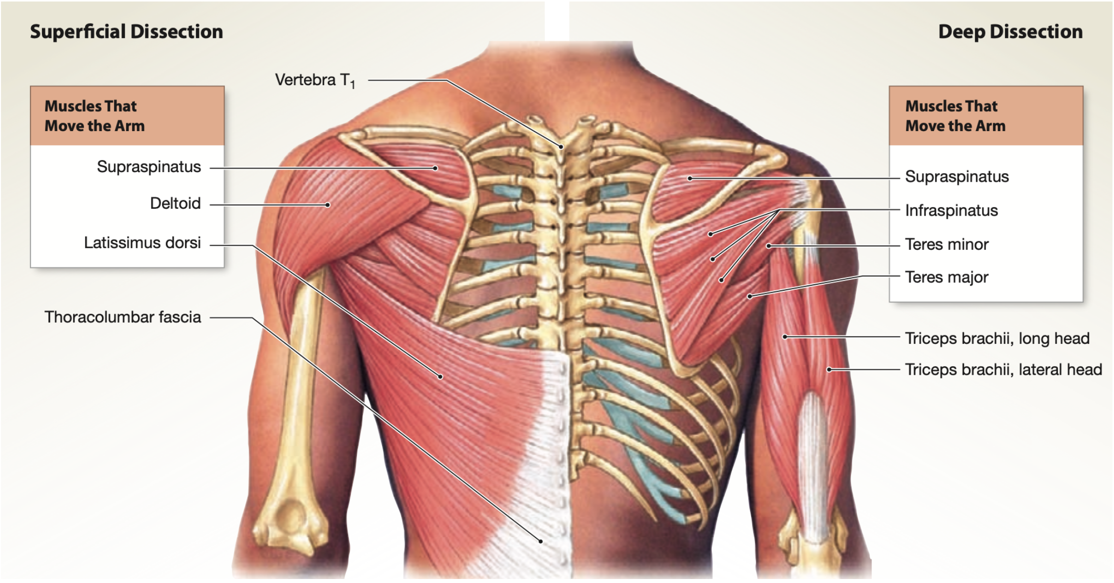


(b) Posterior view.

Figure 2.5: The pectoral girdle connects the upper extremities with the thorax and consists of two bones (clavicle and scapula) and 6 muscles (levator scapulae, pectoralis minor, rhomboid minor and major, serratus anterior, subclavius and trapezius) [20].



(a) Anterior view.



(b) Posterior view.

Figure 2.6: Muscles of the shoulder that move the arm [20].

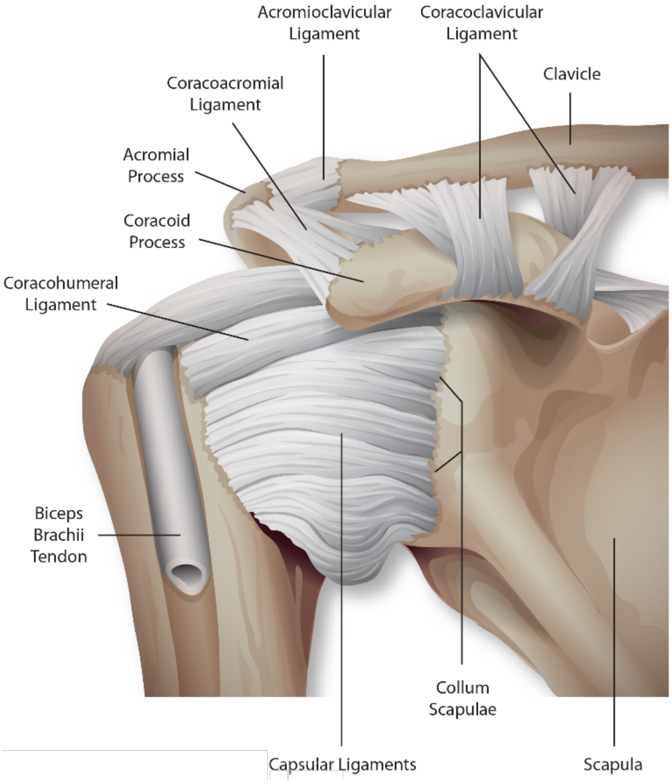


Figure 2.7: Ligaments help to reinforce and stabilize the shoulder joint, the main ligaments are the glenohumeral, coracohumeral, coracoacromial and coracoclavicular ligaments [47].

Chapter 3

Musculoskeletal modeling

This chapter focuses on musculoskeletal modeling. In Section 3.1 an overview is given of different musculoskeletal modeling software on the market, followed by an overview and comparison of existing musculoskeletal models for the OpenSim software, which the software used for this thesis, in sections 3.2 and 3.3 respectively. The chapter ends with a short explanation about the used simulation tools in Section 3.4.

3.1 Musculoskeletal modeling software

Several musculoskeletal modeling and computation software exist, such as Anybody, OpenSim, SIMM, BoB and others.

Anybody provides mechanical modeling of the living body, in particular musculoskeletal modeling with the human body as main domain of application, but the technology applies to any creature (currently living, pre-historic, or imaginary) [57].

BoB (Biomechanics of Bodies) is a group of biomechanical modeling software packages written in MATLAB M-code and is used in academia and industry for product design, sporting performance, equipment design, man/machine interactions, and many more [5].

SIMM (Software for Interactive Musculoskeletal Modeling) lets you create and analyze graphics-based models of the musculoskeletal system. Moment arms and lengths of the muscles and ligaments can be calculated as well as forces and joint moments that each muscles generates. Easy manipulation of the model allows for quick exploration of effects of changing musculoskeletal geometry [38].

The open source software *OpenSim* [10][54] is used for biomechanical modeling of humans, animals, robots and the environment and performing simulations and analyses on their interactions and movement [17]. The graphical user interface provides several tools to visualize and modify the model and to carry out the analyses. The software also provides an API (application programming interface) that makes it possible for OpenSim to communicate with other programs like C++, Python and MATLAB.

A technical note on SIMM and OpenSim states the complementary nature of the two software. Where SIMM has relatively limited tools for computing muscle excitations and analyzing results of dynamic simulations, OpenSim contains a muscle editor, model viewer, coordinate viewer, and plotting tool. And where OpenSim provides very limited editing tools (e.g. joint editor, body editor, deform editor, ...), SIMM provides these complementary capabilities. OpenSim can import and export SIMM models, so users can easily take advantage of features of both software [56].

Langholz et. al report on the comparison of Bob, Anybody and OpenSim as Bob being the most straightforward, AnyBody having the most sophisticated model which can be adjusted in great detail, and OpenSim being a beneficial tool for academic projects with a many users and papers available to support the user [29].

Mirza et.al reviews OpenSim as being a leading open-source software for human movement analysis [30].

Of the mentioned software, OpenSim is only one being freely available [29].

In this thesis the OpenSim version 4.4, released on July 24 2022, is used in combination with MATLAB.

3.2 Overview of existing upper-limb OpenSim models

There are several OpenSim models openly available, either directly when downloading the OpenSim software, or downloadable via the website simtk.org. The characteristics of the upper-limb models that could be of interest for this research are listed in Table 3.1 and a comparison is made.

3.2.1 Arm26

The first model, Arm 26, is directly available when downloading the OpenSim software. It is a very simplistic model of the right arm with only 3 muscles modeled by 6 muscle elements: triceps brachii (long, lateral and medial head), biceps brachii (long and short head) and brachialis. The model has 2 degrees of freedom: shoulder elevation (flexion) and elbow flexion [10][54]. This model was last updated in July 2008 [15].

The arm26 is, together with the Stanford VA model (see model below), used for the development of a two upper limbs model. The developed model was validated by comparing simulated and experimental muscle activation data [21].

This model was also used to evaluated the performance of a newly developed method to control the assistive torque that minimizes the human muscle activations under stability guarantees for assistive exoskeletons. [34]

3.2.2 Dynamic arm simulator (DAS3)

In 2008 Chadwick E. et al.[6] developed a first version of the dynamic arm simulator (DAS3) in the SIMM software. It is a model of the right arm and was developed to design and test neuroprostheses and consisted of the shoulder and elbow as substitute for the human arm. This model was validated by correlating the model-calculated muscle activations to EMG signals recorded from shoulder and arm muscles when performing arm movements [6]. Later this model was updated and implemented in OpenSim, in 2014, by Chadwick E. et al.[7], the model was last updated in 2016.

The sterno-clavicular, acromio-clavicular and glenohumeral joints are each represented by three orthogonal hinges, this accounts for 9 degrees of freedom, elbow flexion and forearm pronation-supination add 2 more degrees of freedom, which makes a total of 11 degrees of freedom. 138 muscle elements are used to represent 29 muscles concentrated around the shoulder. In this model, the shoulder girdle is modeled to be independently controlled, meaning that the movement of the scapula and clavicle is not kinematically controlled, either by measured motions or by describing the shoulder rhythm, but by muscle forces [7] [6].

The independent control of the shoulder girdle in the light of this thesis is not preferable.

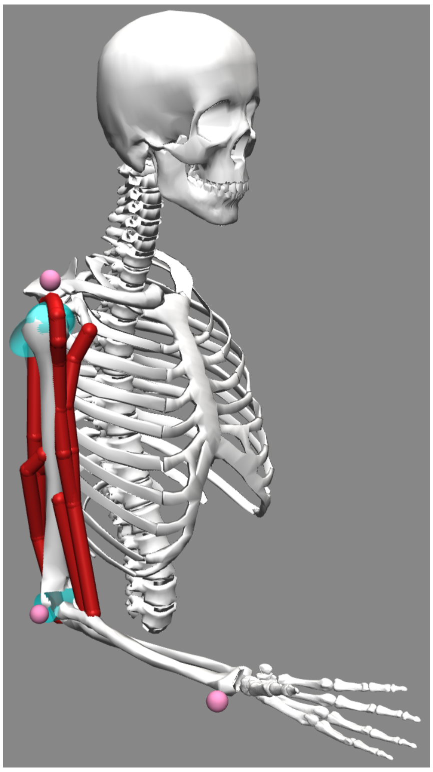


Figure 3.1: Arm 26 model [10][54]. Consisting of a thorax, spine and head as base, a humerus and a radius, ulna and hand modeled as one body; 6 muscles; 2 DOF and 3 markers (pink dots). The blue surfaces are virtual wrap objects to prevent a muscle to pass through a bone when a muscle wraps over a bone surface.

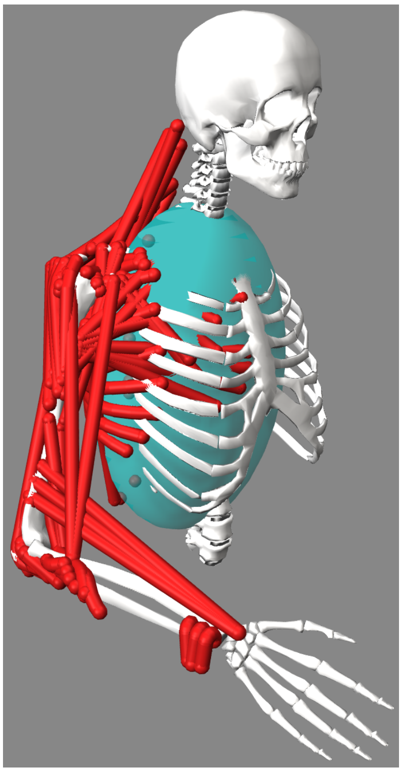


Figure 3.2: DAS3 model [7]. Bodies: base (head, thorax, spine), clavicle, scapula, humerus, ulna, radius, hand; 138 muscle elements representing 29 muscles; 11 DOF, wrapping ellipsoid (blue) over which the muscles move.

The DAS model is used in research to investigate the influence of humeral elevation and plane of elevation on the stabilizing function of the shoulder muscles [1].

3.2.3 Upper extremity kinematic model

The upper extremity kinematic model (Stanford VA upper limb model) was developed by Stanford researchers in 2005. Previous developed models were mostly used to analyze individual regions such as the shoulder, elbow or wrist. However many upper extremity muscles cross over multiples joints, which indicated the need for an integrated model of the upper extremity to better understand the coupling between joints. Stanford researchers used already developed models of the shoulder, elbow and wrist as a foundation, adding new representations of the shoulder, thumb and index finger. The accuracy of the model was assessed by comparing the computed moment arms to experimental data. The model matched well with published literature data [27].

This model can only be used for kinematic analyses since no inertial parameters for the bodies are included. The model is of the right arm and consists of 50 muscle elements representing 32 muscles (muscles with multiple tendons or with wide attachments are represented by multiple compartments) and has 15 degrees of freedom. The motion of the shoulder girdle is described by simplified regression equations from de Groot and Brand[9], to only vary with the thoracohumeral (shoulder elevation) angle[27].

3.2.4 Upper extremity dynamic model

The upper extremity dynamic model is an evolution of the model mentioned above and was developed by the same authors. In 2015 the first version of this model was developed as a tool to evaluate differences in simulation outcomes between software platforms. Inertial parameters were added to the model to make dynamic analyses possible, the number of degrees of freedom was reduced to 7 by fixing the hand in a grip posture and, as in the previous model, each arm has 50 muscle-tendon actuators to represent the 32 muscles.

In 2019 this model was updated to evaluate how glenohumeral joint stability is influenced by task target and task type (unimanual and bimanual pushing and pulling). Joint definitions, muscle paths, and muscle properties were mirrored to the contralateral side to extend the model to a bimanual model, so in total this model has 14 DOF and 100 muscle elements. In addition to the muscles this model also includes 4 ligament elements representing 2 shoulder ligaments, namely the coracohumeral ligament and the glenohumeral ligament (superior, medial and inferior part). The model was last updated in February 2021 [49][32].

The MOBL_ARMS model was, among others, used in research to investigate the glenohumeral muscle and joint contact forces in different keyboard setups [22] and to estimate the upper limb angular kinematics and net joints moments as a result of a wheelchair propulsion movement [25].

Richardson RT et al. [46] reported increasing scapulothoracic angle errors with humerothoracic elevation. They advise to exercise caution when using the MOBL_ARMS model for applications requiring precise scapulothoracic kinematics [46]. However, this paper was published before the MOBL_ARMS model was updated in 2019.

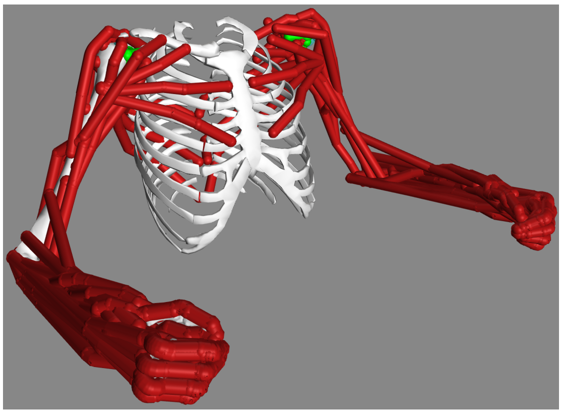


Figure 3.3: MOBL-ARMS model [49][32]. Body parts: thorax, clavicle (2), scapula(2), humerus (2), ulna (2), radius (2), proximal row of wrist bones (2) and hand (2); 50 muscle elements representing 32 muscles on each side (red); 4 ligament elements representing 2 ligament on each side (green); 7 DOF on each side.

3.2.5 Model of the scapulothoracic joint

The model of the scapulothoracic joint is rigid-body model of the right shoulder and arm developed in 2015 by Seth A. et al. [52]. This model was developed to accurately describe the kinematics of the scapula relative to the thorax. This was done by parameterizing the scapulothoracic motion by 4 joint coordinates: (1) elevation/depression and (2) abduction/adduction on an ellipsoidal thoracic surface, (3) upward rotation of the scapula and (4) internal rotation or 'winging'. This model is purely skeletal, thus contains no muscles, and has 11 degrees of freedom.

The model was evaluated for its accuracy by comparing computed marker locations to measured marker data from motion-capture experiments, to properly compare the marker locations the upper extremity model was scaled to the test subject.

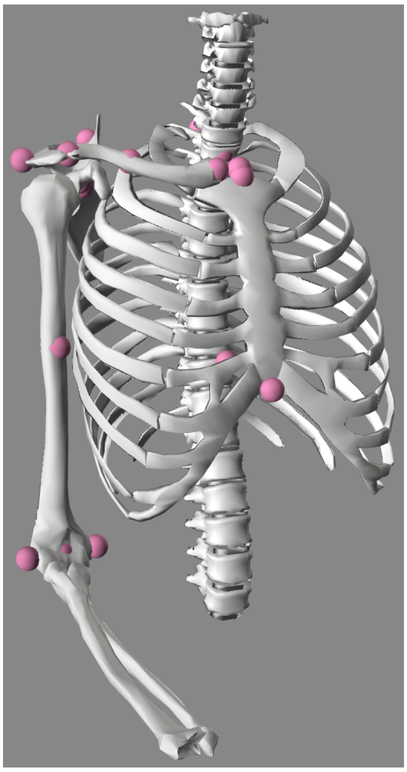


Figure 3.4: Scapulothoracic joint model [52]. Body parts: thorax, clavicle, scapula, humerus, ulna, radius, hand; 21 markers (pink dots); 11 DOF.

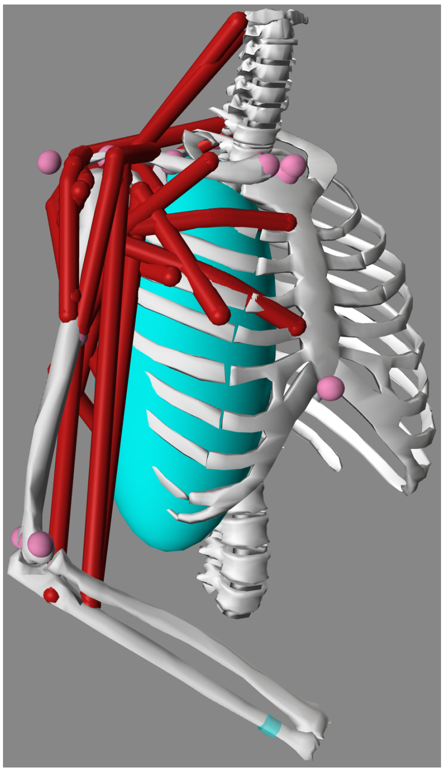


Figure 3.5: Thoracoscapular muscle model [53]. Body parts: thorax, clavicle, scapula, humerus, ulna, radius, hand; 33 muscle elements representing 16 muscles;21 markers (pink dots); 11 DOF; blue wrapping surface to guide the muscle path.

3.2.6 Thoracoscapular shoulder model

In 2019 Seth A. et el. [53] developed a model that combines the scapulothoracic model, from above, with scapular muscles to drive the shoulder motions in order to asses the role of the thoracoscapular muscles in the motion of the scapula. The model includes 33 muscle elements representing 16 thoracoscapular muscles and has the same degrees of freedom as the skeletal model from above: pronation/supination of the wrist, flexion/extension of the elbow, axial rotation, adduction/abduction and extension/flexion of the arm and the 6 remaining degrees of freedom position and orientate the scapula and clavicle, of which the joint coordinates are linked, however, they are not linked one-to-one [53]. The movement of the scapula is again kinematically uncoupled, this independent movement of the scapula is not preferred for this thesis.

The model was evaluated by comparing the muscle activations from muscle-driven simulations to the EMG measured for the same tasks. An average mean absolute error (MAE) of 0.06 was reported across all model muscles, with the largest MAE of 0.15 for the latissimus dorsi during loaded (2kg) flexion. For unloaded flexion and abduction the MAE's for all muscles were below 10% [53].

Table 3.1: Overview of upper-limb model in OpenSim, the one that is used in this thesis is highlighted in blue.

OpenSim model	**Kinematic / dynamic**	**DOF**	**Muscle(s) (elements)**	**Ligaments**	**Shoulder rhythm**
Arm26	Dynamic	2: - Shoulder flex/ext - Elbow flex/ext	6 muscle elements for 3 muscles	No	No movement of scapula and/or clavicle
Upper Extremity Kinematic Model [27]	Kinematic	15: - 3 Shoulder - 1 Elbow - 1 Radio-ulnar - 4 Thumb - 4 Forefinger	50 muscle elements representing 32 muscles	No	Simplified relation with shoulder angle
Upper Extremity Dynamic Model (MoBL-ARMS) [49] [32]	Dynamic	7 each arm: - 3 Shoulder - 1 Elbow - 3 Hand/wrist	100 muscle elements (50 for each arm)	2: - Coracohumeral - Superior, medial and inferior glenohumeral	Simplified relation with shoulder angle
Dynamic Arm Simulator (DAS3) [7]	Dynamic	11: - 9 shoulder - 1 Elbow - 1 Radio-ulnar	138 muscle elements	No	Independent control of shoulder girdle
Model of the Scapulothoracic Joint [52]	Dynamic	11: - 1 Wrist - 1 Elbow - 9 Shoulder	Skeletal model	No	Kinematically uncoupled movement of scapula
Thoracoscapular shoulder model [53]	Dynamic	11: - 1 Wrist - 1 Elbow - 9 Shoulder	33 muscles and muscles groups	No	Kinematically uncoupled movement of scapula

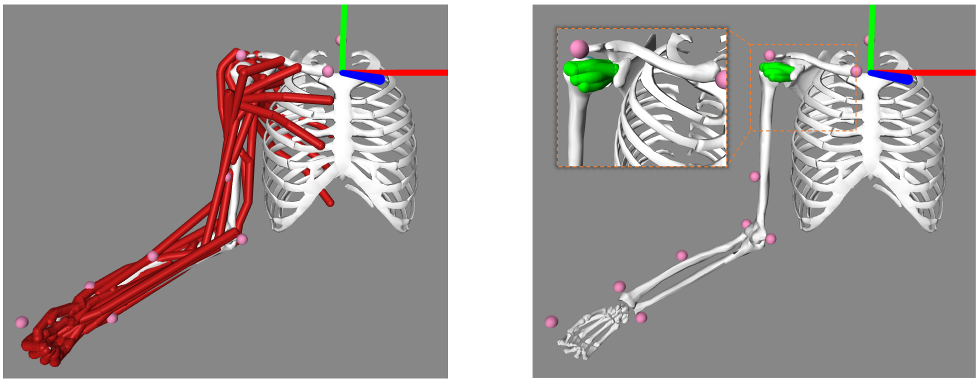
3.3 Review of OpenSim models

With the requirements for the OpenSim model, mentioned in section 2.5, in mind, some of the discussed models can already be excluded as option for the following reasons. The Arm26 is too simplistic for the purpose if this thesis, but, this also makes for a low computational cost, and is thus very suitable to learn to work with the OpenSim program and to run some fast first impression simulations, for which it will thus be used. The upper extremity kinematic model cannot be used for dynamic analyses, which is a major part of this thesis, and is thus not a suited model. Computing reaction forces in the muscles is part of the research of this thesis, so since the scapulothoracic joint model does not include any muscles, this model is also not suited. However, the model of the scapulothoracic joint could still be interesting to investigate the difference between applying a force compared to a torque purely on the level of the bones, independent of the muscle activity, which could give more insight on the function of muscles in terms of reaction forces. This could be interesting when looking at people who can no longer generate muscle activity.

The DAS3 and thoracoscapular muscle model both have muscles and dynamic parameters. The downside however, is that the shoulder girdle is modeled to be independently controlled and thus, that no shoulder rhythm is present.

The upper extremity dynamic model is a very comprehensive model, including both left and right arm, with much more muscles than of interest for this thesis. A big advantage of this model is that it is the only model that includes two important shoulder ligaments and that the movement of the shoulder girdle is constraint by the shoulder rhythm.

The upper extremity dynamic model best fits the model requirements for this research, despite being much more extensive than necessary. In addition, in the downloaded folder of this model a unimanual model of the right arm is present and the muscles that are not of immediate interest, such as muscles at the level of the elbow or hand, will not negatively influence further simulations. On the contrary, it can only make the results more realistic. The final model is shown in Figure 3.6.



(a) Muscles.

(b) Ligaments.

Figure 3.6: Unimanual MoBL-ARMS model that is used for simulations in this thesis [49][32]. This model contains 32 muscles and 2 ligaments and has 7 DOF.

3.4 Simulations in OpenSim

To investigate the difference in reaction forces in the shoulder when using torque or force as exoskeletal assistance, the model needs external load input elements. In the simulations, the assistance purely through force is applied at the center of mass (COM) of the humerus as a 2D vector in the plane of elevation, this is done for simplicity and equality between simulations. The assistance purely through torque is applied around the transverse axis, and since the torque is applied as a free vector, the point of application is arbitrary.

Muscle forces and activations

Muscles are represented by musculotendon actuators which consist of an active contractile element, a passive elastic elements and an elastic tendon element. During non-isometric contraction the muscle force varies with its length. The active-force-length curve ($f^L(\widetilde{l}^M)$) represents the non-linear variation of the maximal force a muscle can develop, depending on its length. The force a muscle develops also depend on the rate at which its length changes, this is represented in the force-velocity curve ($f^V(\widetilde{v}^M)$). The passive-force-length curve ($f^PE(\widetilde{l}^M)$) represents the force a muscle develops when stretched beyond a threshold length. The tilde denotes the force, length and velocity normalized by the peak force (f_0^M), the length at peak force (l_0^M) and the maximal velocity (v_{max}^M) respectively. These curves form a force-length-velocity surface:

$$f^M = f_O^M * (a * f^L(\widetilde{l}^M) * f^V(\widetilde{v}^M) + f^PE(\widetilde{l}^M)) \tag{3.1}$$

with a the muscle activation, ranging from a_{min} to 1 [14].

Muscle forces needed to achieve a certain movement are computed for the loaded motion with the static optimization tool. It uses the known motion, to solve the equations of motion for the unknown generalized forces, while being constrained by force-length-velocity properties and minimizing the sum of squared activations. And although a tendon can stretch way beyond its slack length, in static optimization the tendon is assumed to be inextensible [12][14].

Joint reaction forces

The reaction forces acting on the shoulder are computed with the JointReaction analysis tool. It calculates the resultant forces and moments transferred between two consecutive bodies as a result of all loads acting on the model, including muscles, motors, and all other actuators. The reaction load acts at the joint center of both the parent and child bodies. This function takes the names of the joints of interest, the body on which the reaction is applied (parent or child), the frame in which to express the reaction (parent or child) and a storage file containing the muscle forces computed with the static optimization tool, as input [13].

Validation of simulations

The simulated data needs to be validated. This is done by comparing the behavior of the measured EMG signal and the computed muscle activations. When giving different amplitudes of assistance to a person, the EMG activation of the muscles will decrease when increasing the level of assistance, the same behavior should be seen in the computed muscle activations.

Another way of validating the simulations and added external force and torque, is by comparing the influence of the external force on the reaction moments in the shoulder computed with the inverse dynamics tool and a manually calculated estimation. The inverse dynamics tool of OpenSim determines the forces and moments needed to produce a given motion for each joint, taking into account the applied external forces and/or torques. The inverse dynamics tool ignores the contribution of muscles [11].

Chapter 4

Methods

This chapter first explains the validation method and the taken data processing steps in order to achieve the needed data in Section 4.1. This is followed by the description of the performed motion and robotic assistance experiments and how this data is processed to the wanted format in Section 4.2.

4.1 Validation

To validate the simulations in OpenSim, EMG signals are recorded for the same motion in with three levels of assistance: no support, low force support and high force support. This EMG data is compared to the computed muscle activations and forces from OpenSim.

A second validation step is to compare the moment required to lift the arm computed by OpenSim to the manually calculated moment.

Electromyography

The muscle activations of five unilateral muscles of the right, dominant, hand were recorded (2000Hz, Cometa, Bareggio, Italy [8]) according to the Surface Electromyography for the Non-Invasive Assessment of Muscles (SENIAM [51]) guidelines. The activity of the following muscles was measured:

- Deltoid anterior (shoulder flexion)
- Deltoid medialis (shoulder abduction)
- Deltoid posterior (shoulder extension)
- Biceps brachii (shoulder flexion)
- Long head of triceps (shoulder extension)

The raw EMG data is bandpass filtered with a 4th order Butterworth filter from 10 to 500Hz, full-wave rectified, and smoothed with a low-pass filter (4th order Butterworth) at 6Hz.

Synchronization

To synchronize the EMG and motion signals, a short impulse is given to one of the DOT sensors and a reference EMG sensor simultaneously. This impulse is used to align the measurements, Figure 4.1 shows the peaks in acceleration for both the Xsens DOT and EMG sensors.

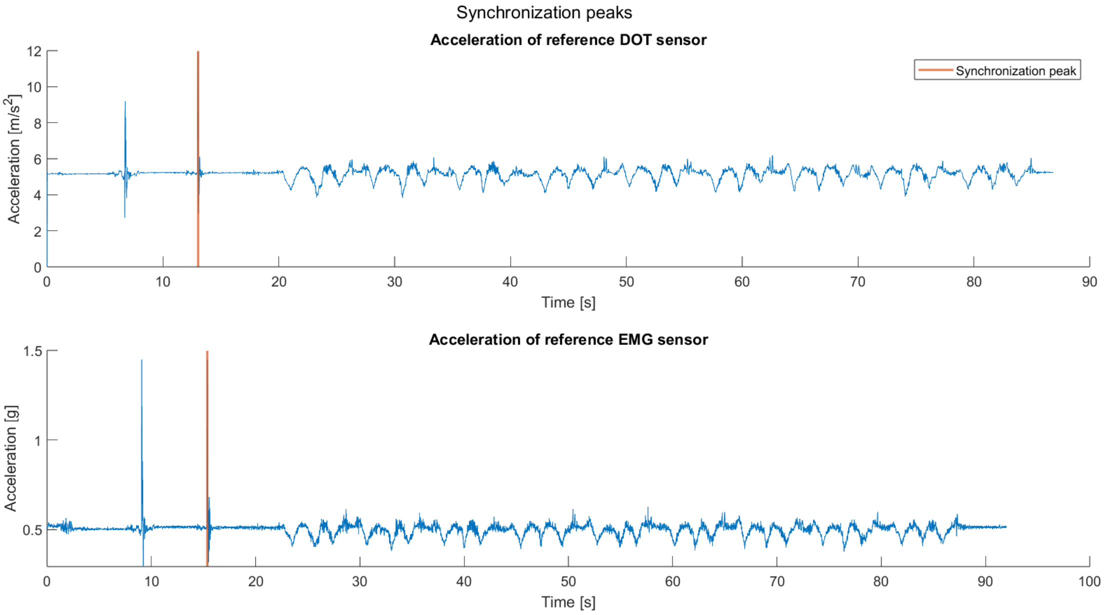


Figure 4.1: A short and quick impulse is given to an IMU and EMG sensor simultaneously. The peaks in acceleration (red line) seen on the graphs are used to align the measurements.

The different signals are measured at a different frequency: Xsens DOT measures at a rate of 60Hz, the EMG sensors at a frequency of 2000Hz. So to have the data at the same frequency, the EMG data is re-sampled to a frequency of 60 Hz.

Inverse Dynamics

In order to estimate the required moment around the shoulder, the total mass of the arm (m_{arm}) and the center of mass (COM) of the arm (COM_{arm}) are calculated. The COM of the arm is calculated when the humerus is horizontal, by taking the sum of the multiplications of the COM of each body part with its perpendicular distance from the shoulder joint, and dividing this sum by the total mass of the arm:

$$\begin{aligned} COM_{arm} &= \frac{COM_{hum} * m_{hum} + COM_u * m_u + COM_r * m_r + COM_{hand} * m_{hand}}{m_{hum} + m_u + m_r + m_{hand}} \\ &= \frac{0.14m * 1.99kg + 0.32m * 1.11kg + 0.36m * 0.23kg + 0.38m * 0.58kg}{1.99kg + 1.11kg + 0.23kg + 0.58kg} \\ &= 0.24m \end{aligned} \tag{4.1}$$

The masses and origins of each body part are read out from the OpenSim MOBL_ARMS model. To then find the position of the COM, the origin of each body part is summed with the longitudinal component of its COM multiplied with $cos(67°)$ to have the horizontal component, 67° is the angle of the elbow when the humerus is horizontal. For the humerus, this multiplication is not needed since it is already horizontal. The positions of the origins and COMs are shown in Figure 4.2.

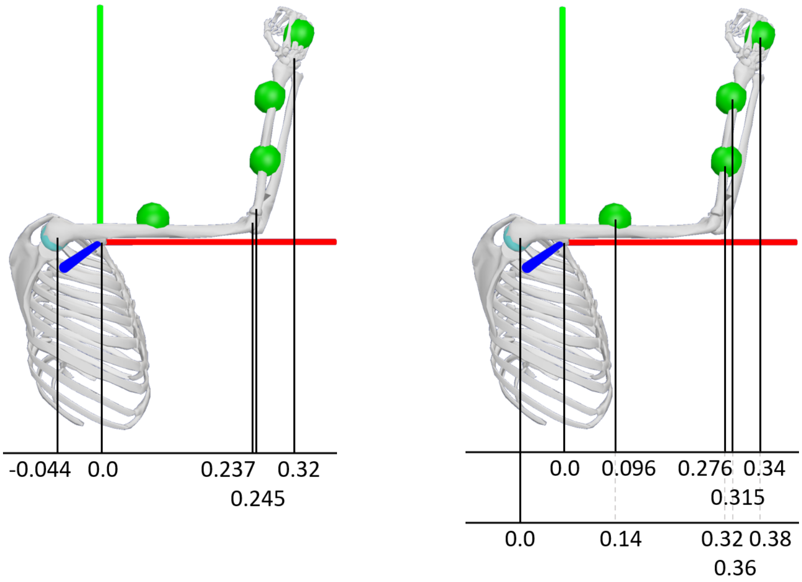


Figure 4.2: To estimate the center of mass (COM) of the right arm, the positions of the COMs (green spheres) of the separate body parts are calculated. The positions of the origins of each body part is outputted from OpenSim (left). The positions of the COMs of the body parts are calculated by summing the position of the origin and the horizontal component of the position of the COM.

The final resuired moment around the shoulder is then found by multiplying the total mass of the arm with its COM and the sinus of the shoulder elevation angle:

$$M = m_{arm} * COM_{arm} * sin(\alpha) \tag{4.2}$$

with α being the shoulder elevation (explained in the section below).

To include the external force or torque, the required moment is calculated as follows:

$$\begin{aligned} M_{force} &= M - 0.14m * F_{ext} \\ M_{torque} &= M - T_{ext} \end{aligned} \tag{4.3}$$

4.2 Torque versus force assistance

4.2.1 Motion description

A subject was asked to perform a flexion-extension motion of the shoulder. As start position the subject is sitting with his back against the backrest of a chair with his upper-arm straight down. After starting the measuring devices, the subject slowly moves his arm forward and upward until his arm is about 90° up and then lowers his arm again, the participant was instructed to keep the elbow angle static and not fully extended to avoid overextension. One cycle of flexion-extension took 7 seconds, the movement is shown in Figure 4.3. This motion is first performed once and recorded with the SAFeR [48] robot. The SAFeR robot then executes this motion in a cycle. The experiment is performed with different levels of assistance. The motion with assistance was repeated 30 times per assistance level (low assistance, high assistance), for the case without robotic assistance, so without the SAFeR robot, the motion was repeated 9 times.

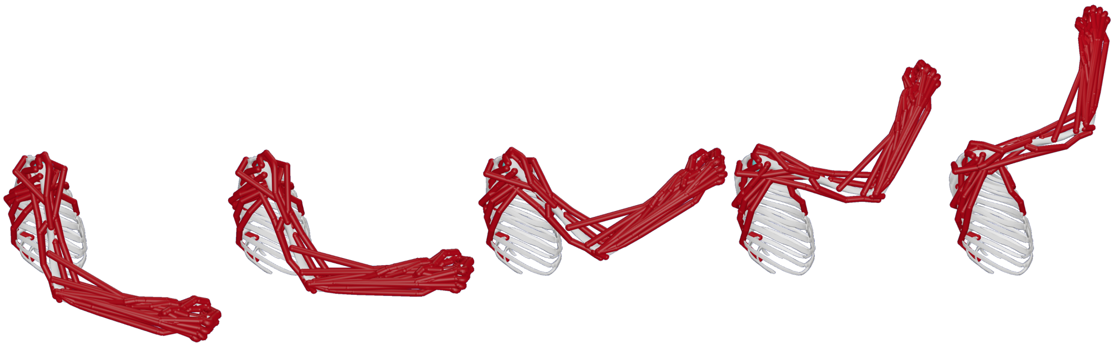


Figure 4.3: Flexion-extension motion of the shoulder. This motion is performed 9 to 30 times within one set and one repetition took about 7 seconds. Three sets of measurements are performed: once without robotic assistance, once with low robotic assistance and once with high robotic assistance.

The kinematic data of the performed movement was recorded with Xsens DOT sensors, which are wearable sensors providing 3D angular velocity, 3D acceleration and 3D orientation (Euler angles are in Z-Y-X sequence). Three sensors were used, one on the upper-arm, one on the fore-arm and one on the sternum.

From Xsens DOT angles in local sensor frame to angles in a reference frame

The Xsens DOT sensors can give their orientation output as quaternions or as Euler angles. These are used to compute the rotation matrix, which rotates a vector from the sensor coordinate system (S) to the local earth-fixed reference coordinate system (G), known as East-North-Up (ENU) [3].

To calculate the angles that rotate a vector from one body segment (L) to another, or to a reference frame (R), Equation (4.4) is used, the different frames are shown in Figure 4.4.

$$ {}^{R}R_{L} = {}^{R}R_{G} * {}^{G}R_{L} = ({}^{G}R_{R})^{T} * {}^{G}R_{L} \tag{4.4}$$

From reference frame Euler angles to joint coordinates in OpenSim

In the MOBL_ARMS model the shoulder orientation is controlled by first defining the plane of elevation, which is rotated by the angle of elevation, γ, around the fixed Y-axis (not physically moving the arm but only the plane in which the arm is elevated), followed by the elevation itself, namely α, around the moving X-axis. Note that the Y-axis in the MOBL_ARMS model corresponds to the X-axis of the XSens DOT reference frame, the X-axis to the Z-axis, and the Z-axis to the Y-axis, see Figure 4.5. Thus, the rotation matrix that rotates the arm to a certain position, expressed in the coordinate system of OpenSim, is composed of first a rotation matrix that rotates the arm around the X-axis the plane of elevation around the Y-axis with an angle γ:

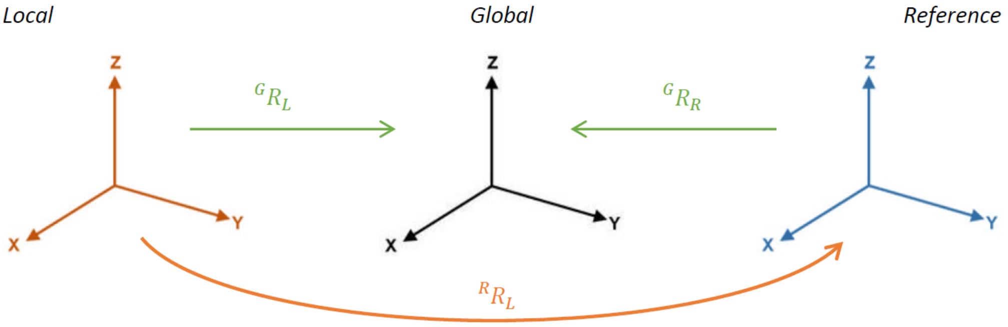


Figure 4.4: All DOT sensors have their own, Local, coordinate system. The rotation matrix, R, formed by the measured Euler angles, rotates a vector from the Local DOT frame to the Global East-North-Up (ENU) frame. To rotate a vector from its DOT frame to another reference DOT frame, the vector is first rotated to the Global frame, followed by rotating it to the DOT Reference frame with the inverted rotation matrix.

$$\begin{bmatrix} c(\gamma) & 0 & s(\gamma) \\ 0 & 1 & 0 \\ -s(\gamma) & 0 & c(\gamma) \end{bmatrix} * \begin{bmatrix} -1 & 0 & 0 \\ 0 & c(\alpha) & -s(\alpha) \\ 0 & s(\alpha) & c(\alpha) \end{bmatrix} = \begin{bmatrix} -c(\gamma) & s(\gamma)*s(\alpha) & s(\gamma)*c(\alpha) \\ 0 & c(\alpha) & -s(\alpha) \\ +s(\gamma) & c(\gamma)*s(\alpha) & c(\gamma)*c(\alpha) \end{bmatrix} \tag{4.5}$$

To find γ and α, the following reasoning is used. For both sets of rotations, Euler angles from XSens and joint coordinates in OpenSim, the final position of the arm should be the same, meaning that multiplication of the rotation matrices with the start position of the arm has to give the same end position. Since the default, a.k.a. start position, in OpenSim is defined as the arm straight down, the start position in the Xsens coordinate frame is $(-1, 0, 0)$, and in OpenSim coordinate frame $(0, -1, 0)$.

Rotation of the arm in OpenSim coordinate system:

$$\begin{bmatrix} -c(\gamma) & s(\gamma)*s(\alpha) & s(\gamma)*c(\alpha) \\ 0 & c(\alpha) & -s(\alpha) \\ s(\gamma) & c(\gamma)*s(\alpha) & c(\gamma)*c(\alpha) \end{bmatrix} * \begin{bmatrix} 0 \\ -1 \\ 0 \end{bmatrix} = \begin{bmatrix} -sin(\gamma)*sin(\alpha) \\ -cos(\alpha) \\ -cos(\gamma)*sin(\alpha) \end{bmatrix} \tag{4.6}$$

Rotation of the arm in Xsens coordinate system:

$${}^{R}R_{L} * \begin{bmatrix} -1 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} -R1 \\ -R2 \\ -R3 \end{bmatrix} \tag{4.7}$$

Since the X,Y,Z axes in OpenSim correspond to the Z,X,Y axes in XSens frame, the following equations are found:

$$\begin{cases} R1 = cos(\alpha) \\ R2 = cos(\gamma)*sin(\alpha) \\ R3 = sin(\gamma)*sin(\alpha) \end{cases} \tag{4.8}$$

From which the joint coordinate α for OpenSim can be calculated:

$$\{ \ \alpha = acos(R1) \tag{4.9}$$

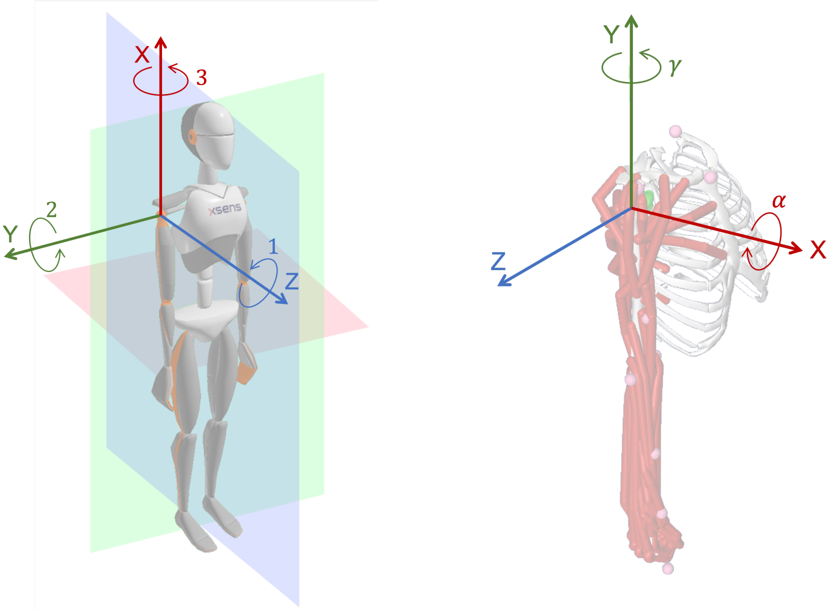


Figure 4.5: The shoulder position calculated from the Xsens DOT sensors is expressed by Euler angles (ZYX sequence) in the coordinate system of the DOT placed on the sternum (left model [18]), while the shoulder position of the MOBL_ARMS model in OpenSim is expressed by two angles in the coordinates system shown in the right model: γ, which rotates the plane of elevation around the fixed Y-axis (not physically moving the arm but only the plane in which the arm is elevated) and α, which elevates the arm by rotating around the moving X-axis.

The elevation angle γ is found by using the following formula's:

$$\begin{cases} ||\mathbf{a} \times \mathbf{b}|| = ||\mathbf{a}|| * ||\mathbf{b}|| * sin(\gamma) & \rightarrow sin(\gamma) = \dfrac{||\mathbf{a} \times \mathbf{b}||}{||\mathbf{a}|| * ||\mathbf{b}||} \\ \mathbf{a} \cdot \mathbf{b} = ||\mathbf{a}|| * ||\mathbf{b}|| * cos(\gamma) & \rightarrow cos(\gamma) = \dfrac{\mathbf{a} \cdot \mathbf{b}}{||\mathbf{a}|| * ||\mathbf{b}||} \end{cases}$$
$$\Rightarrow tan(\gamma) = \frac{sin(\gamma)}{cos(\gamma)} = \frac{\dfrac{||\mathbf{a} \times \mathbf{b}||}{||\mathbf{a}|| * ||\mathbf{b}||}}{\dfrac{\mathbf{a} \cdot \mathbf{b}}{||\mathbf{a}|| * ||\mathbf{b}||}} = \frac{||\mathbf{a} \times \mathbf{b}||}{\mathbf{a} \cdot \mathbf{b}} \rightarrow \gamma = atan2\left(\frac{||\mathbf{a} \times \mathbf{b}||}{\mathbf{a} \cdot \mathbf{b}}\right) \tag{4.10}$$

a represent a vector in the direction of the arm, with as magnitude 1, **b** represents a unit vector in the y-direction (vertical).

The third degree of freedom of the shoulder is the shoulder rotation. This angle is found by calculating the rotation matrix that goes from the moved humerus position to the initial humerus position. The start position is taken as zero rotation of the humerus. From this rotation matrix the Euler angles (Z-Y-X sequence) are retrieved and the rotation around the X-axis is the rotation of the humerus around its longitudinal axis.

$$^{hum0}R_{hum} = {}^{hum0}R_G * {}^{G}R_{hum} = ({}^{G}R_{hum0})^T * {}^{G}R_{hum} \tag{4.11}$$

The angle of elbow flexion is found by calculating the angle between the longitudinal axis of the humerus and the longitudinal axis of the forearm. These vectors are expressed in the global ENU frame and are found by multiplying the rotation matrices of the IMUs with (1,0,0), since the x-axes are aligned with the longitudinal axes. Then, analogous to the calculation of the elevation angle, the elbow angle is:

$$atan2\left(\frac{||\,^{G}R_{hum} * (1,0,0) \times \,^{G}R_{fore} * (1,0,0)||}{^{G}R_{hum} * (1,0,0) \cdot \,^{G}R_{fore} * (1,0,0)}\right) \tag{4.12}$$

The joint angles for the first five repetitions of the motion with high assistance are shown in Figure 4.6.

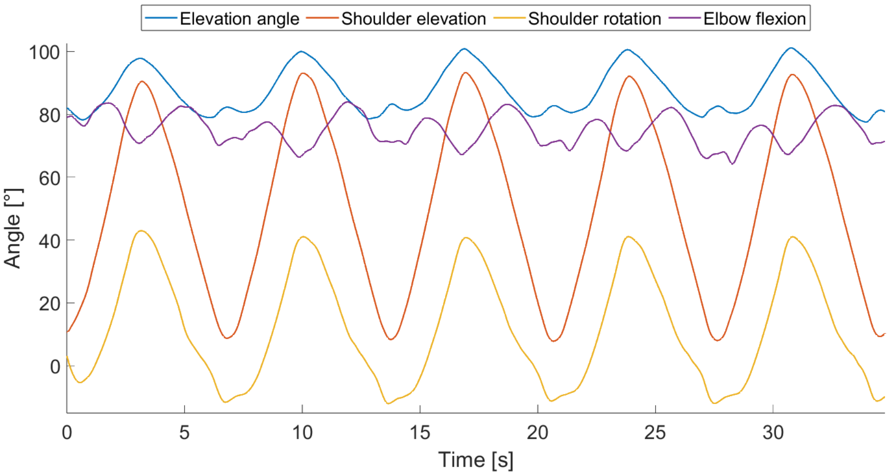


Figure 4.6: Joint coordinates computed from DOT angle measurements for the MOBL_ARMS model in OpenSim (only the first five repetitions are shown): *Elevation angle*, describing the plane of elevation (0° is abduction, 90° is flexion), *Shoulder elevation*, determines the degree of elevation (0° is vertical down, 90° is horizontal, 180° is vertical up), *Shoulder rotation*, rotation of humerus around its longitudinal axis, and *Elbow flexion*.

Motion file as input in OpenSim

For the final movement that is implemented in OpenSim, the mean of all cycles is taken. To achieve this the minima of the shoulder elevation between each cycle are taken, and used as splitting point. Then each separated cycle is re-sampled to have 500 data points, after which the mean is taken per timestamp. The final joint coordinates are shown in Figure 4.7.

4.2.2 Robotic assistance description

The reaction forces and moments felt by the SAFeR robot are recorded and expressed in the global frame of the robot, as shown in Figure 4.8. The measured data is low pass filtered (4th order Butterworth, 0.5Hz) to smoothen the signal, the results are shown in Figure 4.10.

For simplicity the assistance in the simulation is applied in 2D, in the plane of elevation. Since the coordinate frame in which the reaction forces of the SAFeR robot are measured is not the same as the one used in OpenSim, the force and torque vectors need to be transformed. The

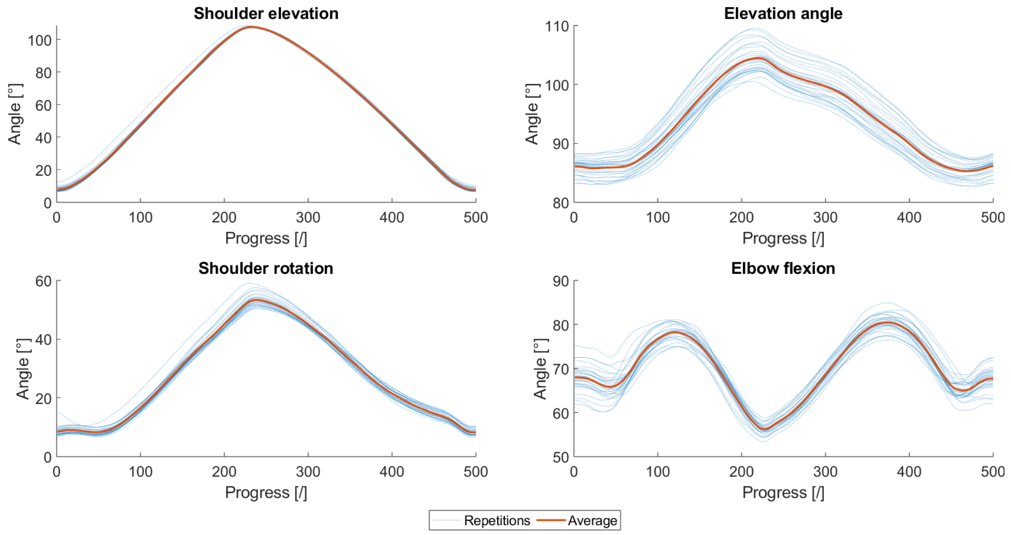


Figure 4.7: For each joint coordinate, each repetition is re-sampled to have 500 data points, then the mean over all repetitions is taken per timestamp. These are the final joint coordinates used as input in OpenSim.

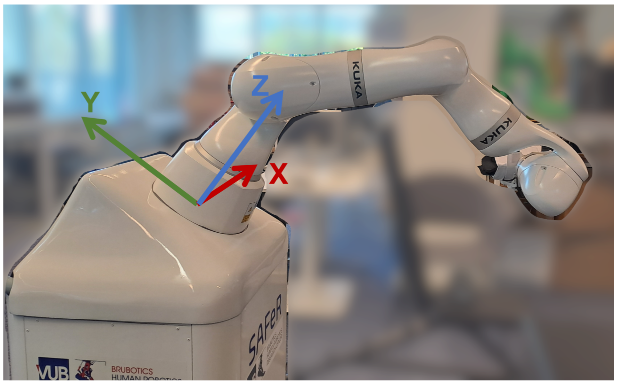


Figure 4.8: Base frame of the SAFeR robot, in which the resistance felt by the end-effector is expressed. The Y-axis makes an angle of 30° with the horizon.

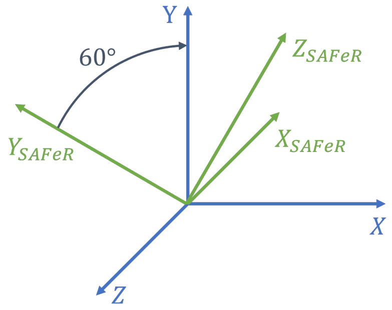


Figure 4.9: Coordinate frames of the SAFeR robot (green), and the global coordinate frame of the OpenSim model (blue). The frame of the OpenSim model is found by rotating the SAFeR frame by 60° around its X-axis, followed by rotating it -90° around its moved Y-axis.

Y_{SAFeR}-axis of the SAFeR robot makes an angle of 60° with the Y-axis of the global frame of the OpenSim model, as shown in Figure 4.9. Thus, the force vector measured by the SAFeR robot is rotated $(-60°)$ degrees around the X_{SAFeR}-axis (rotation matrix R), and then the X and Y components of the force vector and the Z component of the torque vector, as input for

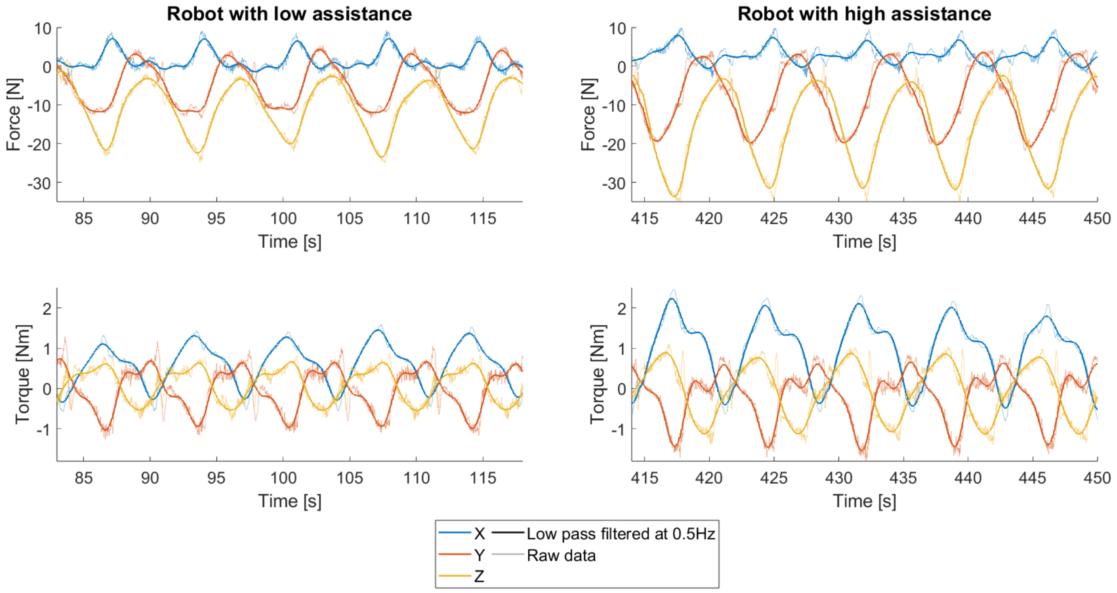


Figure 4.10: Five repetitions of reaction forces and moments measured by the SAFeR robot, expressed in the global frame of the robot, for low (left) and high (right) assistance.

OpenSim, are found as follows:

$$\begin{aligned} F_{exo}^{X} &= -R * R_{SAFeR} * \bar{1}_Z \\ F_{exo}^{Y} &= -R * R_{SAFeR} * \bar{1}_Y \\ T_{exo}^{Z} &= -C_{SAFeR} * (-\bar{1}_X) \end{aligned} \tag{4.13}$$

The negative signs are there because the assistive force should compensate for the measured reaction force and moment, and thus have to be in the opposite direction. The result is shown in Figure 4.11. The assistance through force is applied at the center of mass (COM) of the humerus, the assistance through torque is applied as a free vector.

Just as for the motion input file, the force is averaged over all cycles, the final values for the external loads are shown in Figure 4.12.

Synchronization

To align the robot assistance data with the DOT measurements, the position output of the SAFeR robot is aligned with the motion of the DOT sensors.

The robot and XSens DOT sensors measure at different frequencies, the XSens DOT sensors measure at a rate of 60Hz, while the data exported by the SAFeR robot has a frequency of 40Hz.

So, to have the data at the same frequency, the robotic assistance data is re-sampled to a frequency of 60Hz by linear interpolation.

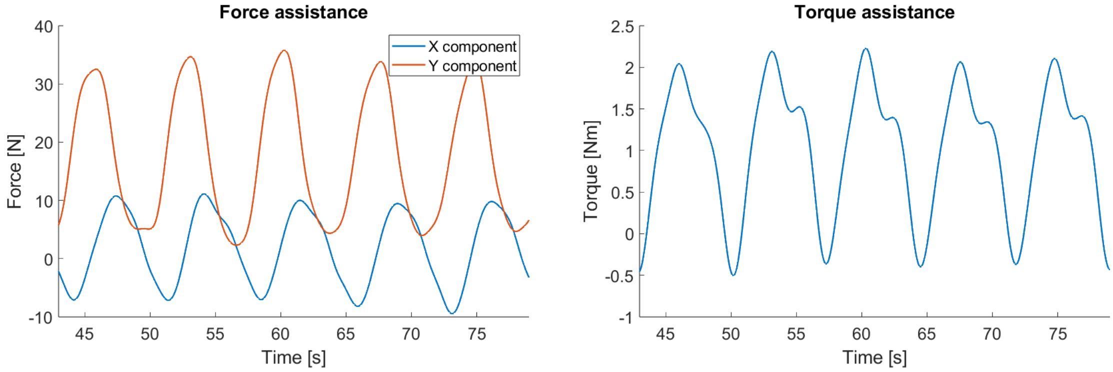


Figure 4.11: The External Load components, for the high assistance case, found after converting the recorded reaction forces and moments of the SAFeR robot to the global frame of the model. For the simulations in OpenSim, the External Load is considered to be 2D, in the plane of elevation.

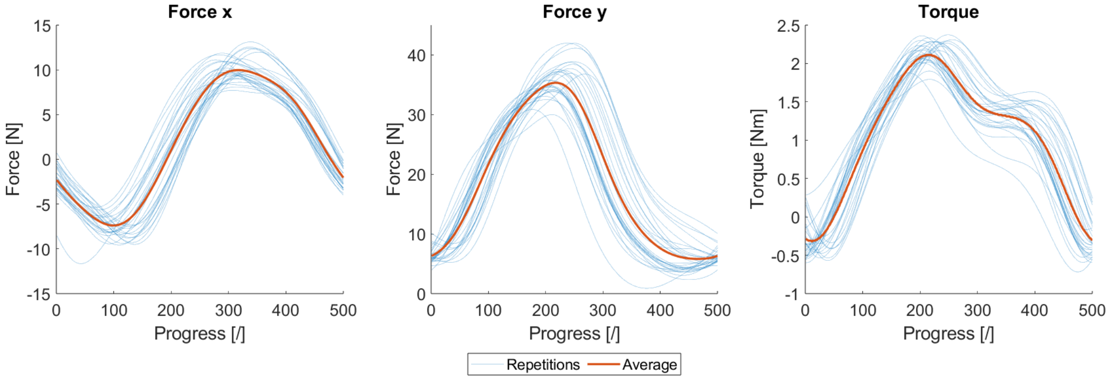


Figure 4.12: Averaged external load over all repetitions, used as external load input in OpenSim.

Chapter 5

Results

This chapter starts with the validation outcomes of comparing the measured EMG signals with the computed muscle activations, for the same muscles. After this, the consequence of using static optimization for a dynamic motion is examined, this showed that the inertia force and torque are negligible. A second validation step includes the inspection of the required moment at the level of the shoulder, as a result of no external load, external force and the external torque. This is all part of Section 5.1.

The second part of this chapter, Section 5.2, contains the reaction force and moment results as a result of applying an external force versus torque.

5.1 Validation

Muscle activation comparison

The simulations are validated by comparing the activation patterns of the computed muscle activations via OpenSim simulations, to the measured EMG.

First, the measured EMG data is inspected (Figure 5.1). These measurements show that the deltoid muscles have the highest activation, which is logic since they are the principle muscles for lifting the arm. And, as expected, the EMG amplitudes decrease with increasing level of assistance.

Since the deltoid muscles are the main muscles for moving and stabilizing the shoulder joint, they will be used for further investigation and comparisons.

When looking at the computed muscle activations of the three deltoid muscles from OpenSim, in Figure 5.2, the same reduction in muscle activation is seen with increasing level of assistance.

Inspecting the computed muscle activations of the three deltoid muscles separately, the patterns are as expected. To initialize the motion, the arm has to accelerate to initiate the movement, this shows in the increased activity of the deltoid medialis, after this initiation, the activity of the deltoid posterior increases to aid in lifting the arm and to decelerate the arm when it reaches its maximum elevation. When lowering the arm, the activity of the deltoid posterior decreases again and subsequently the activity of the deltoid medialis increases to decelerate the arm to stop the motion. This late activation behavior of the deltoid posterior can also be seen in the EMG measurements.

Figure 5.3 shows the summed activation patterns of the deltoid muscle for the different levels of assistance (no, low, high).

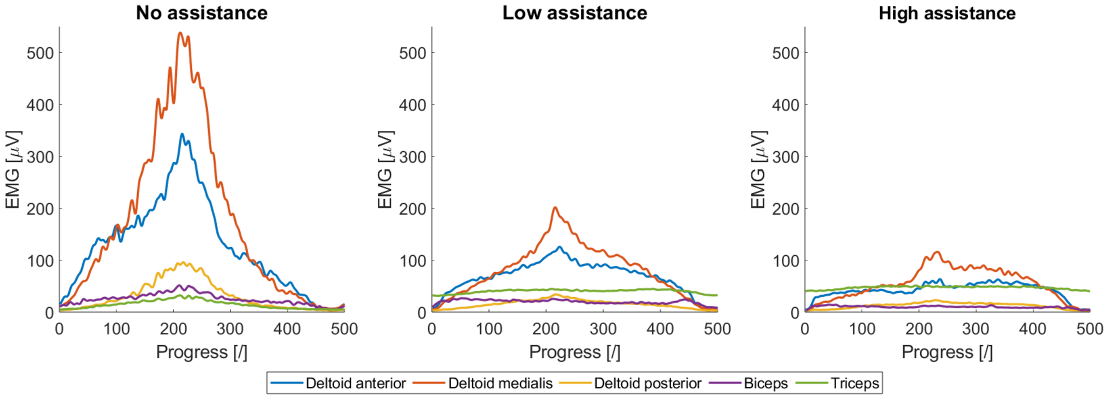


Figure 5.1: Measured EMG signals of five shoulder muscles, for the case where no (left), low (middle) and high (right) assistance was given.

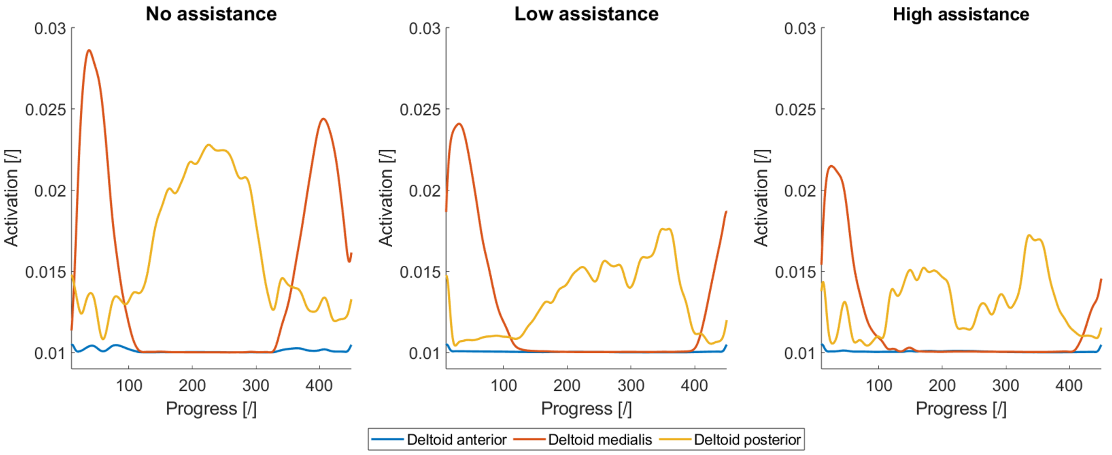


Figure 5.2: Computed muscle activations of the deltoid muscles, for the case where no (left), low (middle) and high (right) assistance was given.

From this figure, no clear correlation can be seen. This could have several causes. A possible reason is that, when using surface EMG, the measured signal is not necessarily a clean signal of one single muscle, this is due to crosstalk with neighboring muscles. This phenomenon can be seen in Figure 5.1, the measured signal pattern for all measured muscles is similar, but scaled in amplitude. The measured EMG signal is also effected by the position of the EMG sensor.

Static approximation for a dynamic motion

Since static optimization is used for a dynamic motion, the dynamic component is calculated to know whether there should be a compensation or if it is negligible.

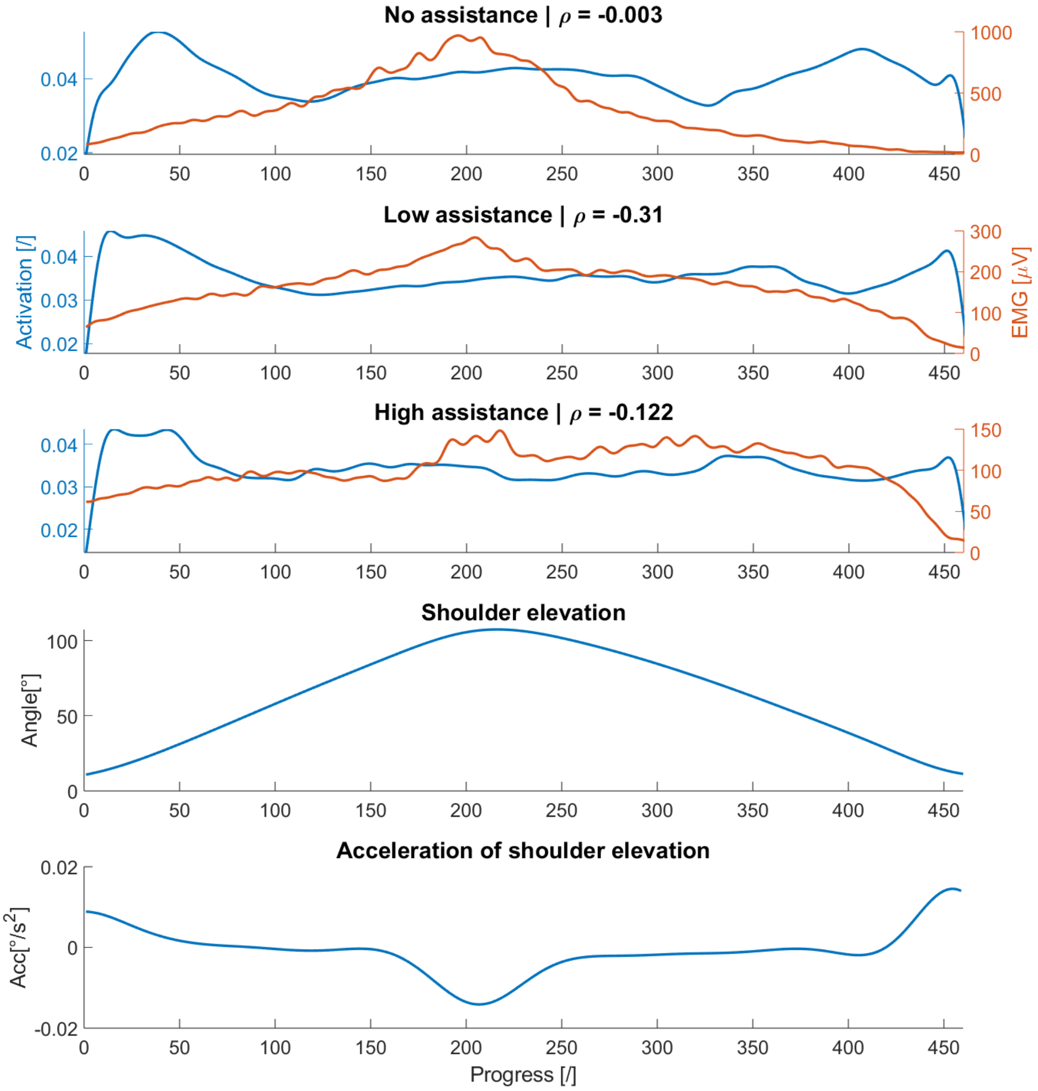


Figure 5.3: Summed activation patterns of the deltoid muscles. The simulated activations via OpenSim are in blue and the recorded EMG signals in orange. The Pearson correlation coefficient, ρ, is given per level of assistance.

The inertia force is found as:

$$\begin{aligned}\overline{F}_I = m * \overline{a}_G = m * \begin{bmatrix} L_G * sin(\alpha) \\ -L_G * cos(\alpha) \\ 0 \end{bmatrix}'' = m * \begin{bmatrix} L_G * cos(\alpha) * \dot{\alpha} \\ L_G * sin(\alpha) * \dot{\alpha} \\ 0 \end{bmatrix}' \\ = m * \begin{bmatrix} -L_G * sin(\alpha) * \dot{\alpha}^2 + L_G * cos(\alpha) * \ddot{\alpha} \\ L_G * cos(\alpha) * \dot{\alpha}^2 + L_G * sin(\alpha) * \ddot{\alpha} \\ 0 \end{bmatrix}\end{aligned} \tag{5.1}$$

For the calculation of $\overline{a}_G$, constant components (components independent of α) are omitted since they drop out after differentiation. α is the shoulder elevation angle.

For simplicity of the calculation of the moment of inertia I, the arm is approximated as 2 thin rods connected at an angle of 67°, which is the elbow angle when the humerus is horizontal. For the first rod, the mass and COM are the mass and COM of the humerus, respectively. For the second rod, the masses of the ulna, radius and hand are summed, and as COM the COM of the total forearm is used. The COM of the forearm and the needed length are found as follows:

$$\begin{aligned} COM_{fore\_arm} &= \frac{m_u * d_u + m_r * d_r + m_h * d_h}{m_u + m_r + m_h} = 0.104 \\ d_2 &= L_{hum}^2 + COM_{fore\_arm}^2 - 2 * L_{hum} * COM_{fore\_arm} * cos(180 - 67) = 0.34 \\ L_2 &= L_u + L_h = 0.315 \end{aligned} \tag{5.2}$$

with $\_u$ referring to the ulna, $\_r$ referring to the radius, $\_h$ referring to the hand, and $\_hum$ referring to the humerus. L is the length of the rod, d is the distance between the COM of the rod and the point in which the inertia is calculated, which is the shoulder joint in this case.

With this in mind, the inertia torque is:

$$\begin{aligned} \overline{T}_I &= \overline{I}_G * \frac{d\overline{\omega}}{dt} = (\overline{I}_{G1} + m_1 * d_1^2 + \overline{I}_{G2} + m_2 * d_2^2) * \frac{d\overline{\omega}}{dt} \\ &= (\frac{m_1 * L_1^2}{12} + m_1 * d_1^2 + \frac{m_2 * L_2^2}{12} + m_2 * d_2^2) * \frac{d\overline{\omega}}{dt} \\ &= (\frac{1.99 * 0.28^2}{12} + 1.99 * 0.14^2 + \frac{2.8 * 0.315^2}{12} + 2.8 * 0.34^2) * \ddot{\alpha} * \overline{1}_Z \end{aligned} \tag{5.3}$$

The results of the inertia force and torque are shown in Figure 5.4.

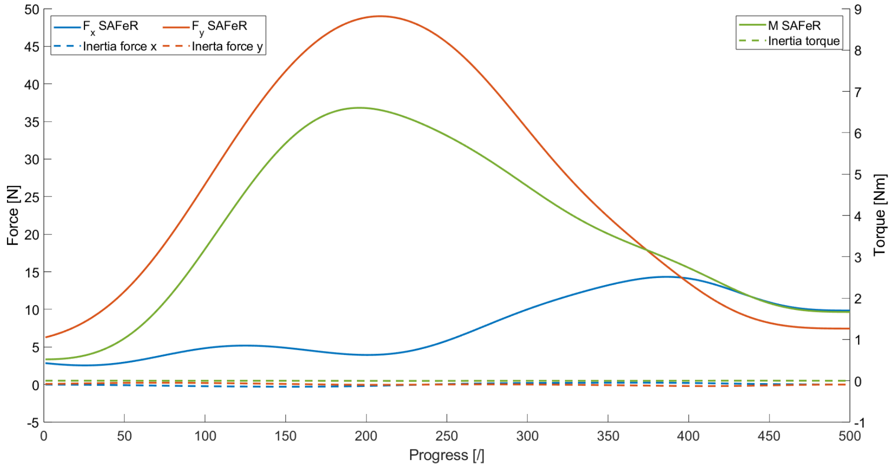
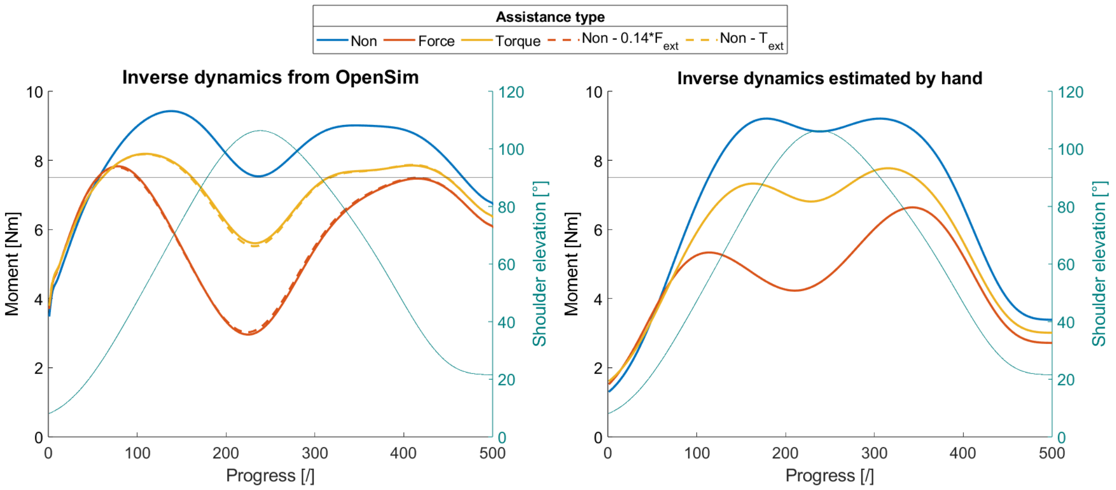


Figure 5.4: Graph of the applied external loads (solid lines) and the estimated inertia force and torque (dashed lines) of the performed flexion-extension movement. The loads are expressed in the global frame of the OpenSim model.

The absolute maximum values for the inertia force and torque are [0.28, 0.26]N and 0.0041Nm respectively. This together with Figure 5.4 shows that the inertia component is negligible and thus that using the static optimization tool for computation of the muscle forces is sufficient.

Moment comparison

To check whether the external loads are defined and applied correctly, an inverse dynamics simulation is performed and the results are compared with manual calculations. The results of both are shown in Figure 5.5.



Assistance type	No	Force	Torque	No - $0.14 * F_{ext}$	No - T_{ext}
Pearson ρ	0.83	0.21	0.48	0.9997	0.9993
MSE	4.97	5.01	4.91	0.002	0.002

Figure 5.5: Required moment around the shoulder joint to produce the wanted movement. The left graph shows the moments computed by OpenSim with the Inverse Dynamics tool, the right graph shows the moments from manual calculations. The dashed line is the moment without any external load computed by OpenSim minus the moment produced by the external load calculated by hand. The contribution of muscles is not considered in the inverse dynamics calculation. The table below the graphs shows the Pearson correlation coefficient and the Mean Squared Error (MSE) between the moments from OpenSim and the ones from the manual calculations.

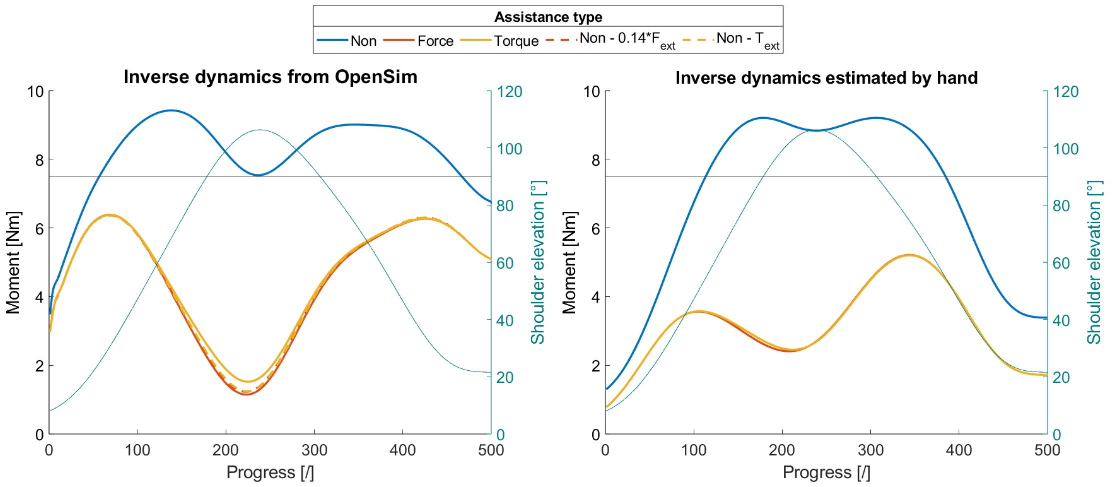
The manual calculations show that the required moment around the shoulder joint is lower when an external assistance (force or torque) is applied. When comparing these outcomes with the ones from OpenSim, they show the same behavior and a good agreement. The mean squared error and the Pearson correlation coefficient for the moment required when no assistance is provided, are 4.97 and 0.83 respectively. The same metrics for the computed moment in OpenSim with force assistance, versus the moment computed in OpenSim for no assistance minus the manually calculated moment induced by force are 0.9997 and 0.002. The metrics for the same comparison with torque assistance are 0.9993 and 0.002. Differences between the manually calculated moments calculated and the ones computed by OpenSim are due to the approximation of the center of mass of the arm and the assumption of a constant elbow angle for simplification of the calculation.

These results also show that the external force and torque are not balanced in terms of induced moment. This difference might effect the comparability of the resulting reaction forces and moments. Thus, to balance the induced moments, the external loads are balanced to induce the same torque assistance. Secondly, in the beginning of the motion the required moment is slightly increased, which is due to negative force and torque application. To account for these unwanted behaviors, the force and torque are recalculated as follows:

$$\begin{aligned} {F'_{exo}}^{X} &= F^{X}_{exo\_rescaled} + (R * \frac{T^{Z}_{exo}}{0.14}) * \overline{1}_{X} \\ {F'_{exo}}^{Y} &= F^{Y}_{exo} + (R * \frac{T^{Z}_{exo}}{0.14}) * \overline{1}_{Y} \\ {T'_{exo}}^{Z} &= T^{Z}_{exo\_rescaled} + 0.14 * F^{x}_{exo} \end{aligned} \tag{5.4}$$

with $_{\_rescaled}$, being the assistance rescaled between 0 and their maximum value. $\frac{T^{Z}_{exo}}{0.14}$ gives the moment arm in the local reference frame of the humerus, so to have the X and Y components in the global frame, this term is transformed from local to global reference frame with the rotation matrix R. F^{x}_{exo} is the force component perpendicular to the humerus expressed in the local frame.

After re-scaling and balancing out the external loads, the reduction in required moment to produce the movement is similar for force and torque assistance, see Figure 5.6. The agreement between the moments computed by OpenSim and the ones manually calculated are again good, this shows in the Pearson correlation coefficients and mean squared errors reported in the table under Figure 5.6.



Assistance type	No	Force	Torque	No - 0.14 $* F_{ext}$	No - T_{ext}
Pearson ρ	0.83	0.24	0.29	0.9998	0.9995
MSE	4.97	5.00	4.88	0.0026	0.016

Figure 5.6: Required moment around the shoulder joint to produce the wanted movement. The external loads are balanced to produce the same moment. The left graph shows the moments computed by OpenSim with the Inverse Dynamics tool, the right graph shows the moments from manual calculations. The dashed line is the moment without any external load computed by OpenSim minus the moment produced by the external load calculated by hand. The contribution of muscles is not considered in the inverse dynamics calculation. The table below the graphs shows the Pearson correlation coefficient and the Mean Squared Error (MSE) between the moments from OpenSim and the ones from the manual calculations.

5.2 Torque versus force assistance

Since the results from the Reaction Load analysis in OpenSim contained a considerable amount of outliers, a moving median is used to replace outliers with a linear interpolated value. An outlier is defined as a value more than three times the local scaled Mean Absolute Deviation away from the local median [31].

The reaction forces, R, at the level of the shoulder joint are expressed in the global frame of the model and are shown in Figure 5.7. When comparing the reaction forces when no load is applied to when an external force is applied, it can be seen that the reaction force in the Y-direction is lowering, with increasing external force. Also the reaction force in the X-direction is decreasing, albeit very little, since the external force is smaller in the X-direction than in the X-direction. In the Z-direction there is little difference since in that direction no force was applied. When looking back at the equations from the introduction, (Eq. 5.5), it can be seen that when F_{exo} increases, R and/or F_{muscle} have to decrease. Since the relation between F_{exo} and F_{muscle} is not know, it is possible that the reaction forces also decrease.

When assistance is given through torque, the expectation was that there would be no increase in reaction forces, however, simulations show the opposite effect, see Figure 5.7. The reaction force in X- and Y-direction increases with increasing torque assistance. This can be explained when looking at Equation 5.5, when torque assistance is given, F_{muscle} decreases, so R has to increase in order to maintain the force balance. The reaction force in Z-direction remains mostly the same, which is logic since the torque is applied around the Z-axis.

$$\begin{aligned}
&\text{Force assistance: } \begin{cases} 0 &= R^X + F^X_{exo} + F^X_{muscle} \\ 0 &= R^Y + F^Y_{exo} + F^Y_{muscle} + F_g \end{cases} \\
&\text{Torque assistance: } \begin{cases} 0 &= R^X + F^X_{muscle} \\ 0 &= R^Y + F^Y_{muscle} + F_g \end{cases}
\end{aligned} \tag{5.5}$$

When looking at the reaction moments, $\overline{C}$, again expressed in the global frame of the model, (Figure 5.8) it can be seen that the reaction moments are small and similar for all three levels of assistance. This is logic since the applied torque or induced moment via the applied force, results in the upward motion of the arm. Relating this back to the Equations 5.6, it would mean that the increase in assistance is almost fully counteracted by the decrease in muscle forces, which leads to a decrease in muscle moment.

$$\begin{aligned}
\text{Force assistance: } 0 &= \overline{C} + \overline{OB} \times \overline{F}_{exo} + \overline{C}_{muscle} + \overline{OG} \times \overline{F}_g \\
\text{Torque assistance: } 0 &= \overline{C} + \overline{M}_{exo} + \overline{C}_{muscle} + \overline{OG} \times \overline{F}_g
\end{aligned} \tag{5.6}$$

Figure 5.9 shows the summed ($\sqrt{x^2 + y^2 + z^2}$) reaction forces and moments. Here the decreased reaction force in case of force assistance, and the increased reaction force in case of torque assistance is clearly visible.

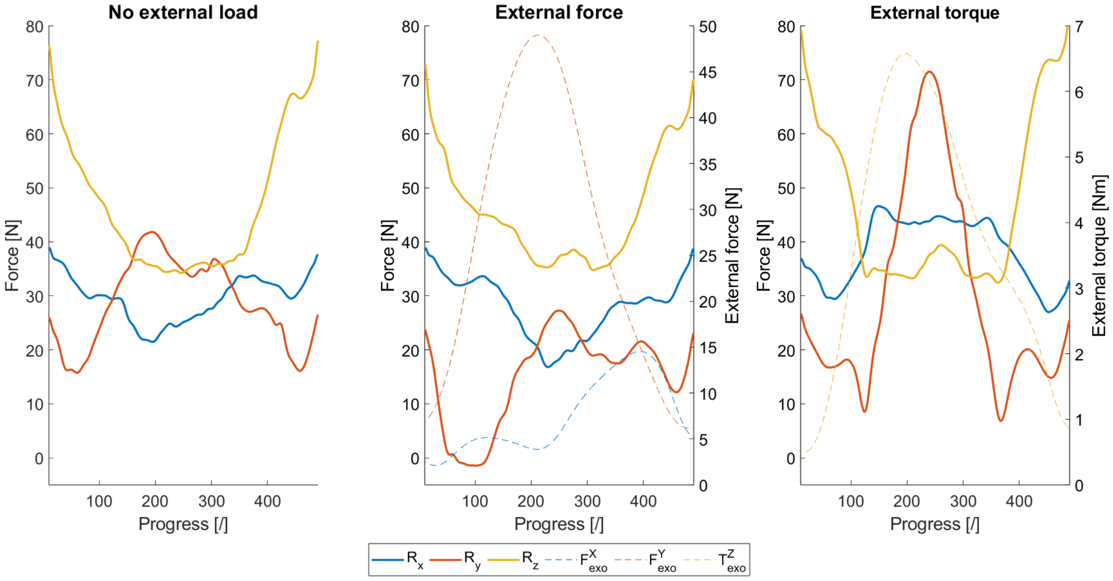


Figure 5.7: Reaction forces acting on the shoulder joint when performing a flexion extension motion with no external load (left), external force (middle) and external torque (right). The solid lines represent the reaction forces, the dashed line are the applied external force or torque. The reaction forces are expressed in the global frame of the model.

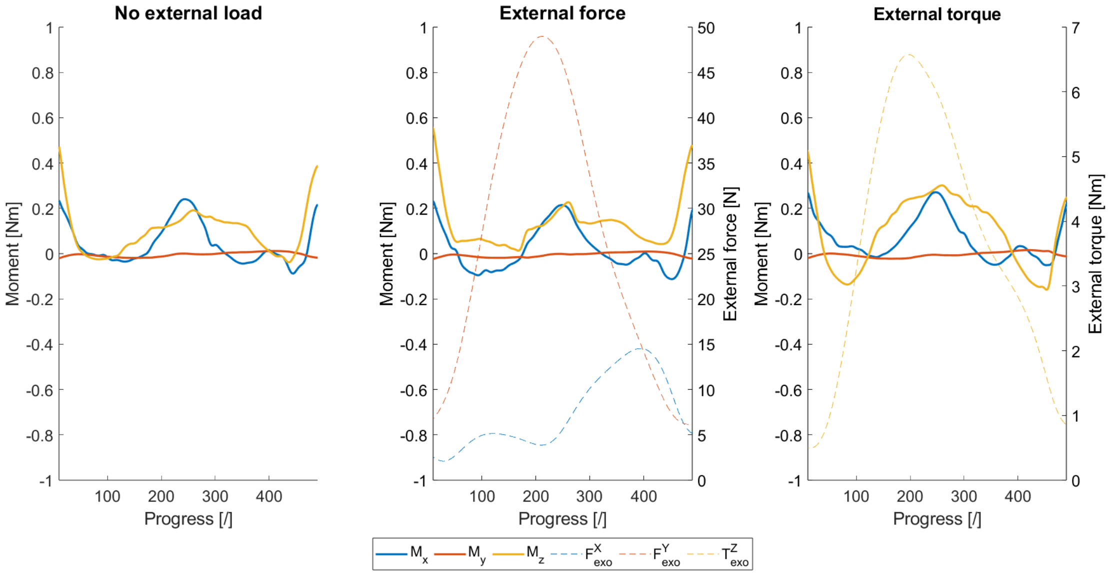


Figure 5.8: Reaction moments acting on the shoulder joint when performing a flexion extension motion with no external load (left), external force (middle) and external torque (right). The solid lines represent the reaction forces, the dashed line are the applied external force or torque. The reaction moments are expressed in the global frame of the model.

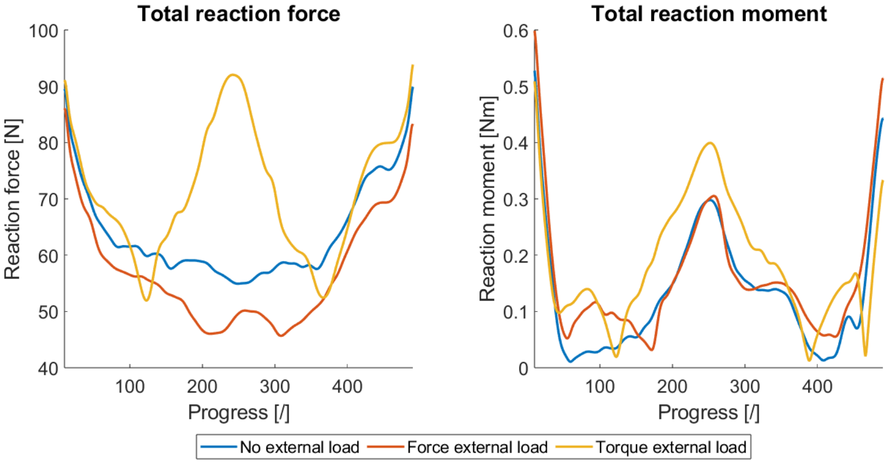


Figure 5.9: Summed reaction forces and moments for three levels of assistance: no assistance, force assistance and high assistance. With respect to the reaction forces for no assistance, the reaction force increases with torque assistance, and decreases with force assistance. The reaction moment is similar for the three levels of assistance.

The goal of an exoskeleton is to reduce the muscle activity. So, more assistance means more reduction in muscle activity. However, too much assistance can also have adverse effects. Simulations are done with five levels of assistance: no assistance, half assistance, assistance, twice assistance and minus assistance (resistance). As can be seen in Figure 5.10, for force assistance, the reaction force decrease with increasing assistance, up until a point where too much assistance is given (named 2xhigh in the figure). From there the reaction forces increase in absolute value. The X-component is higher compared to the other three levels of assistance (no, low, high). The Y-component keeps decreasing, but crosses zero and thus starts to increase in the negative direction. In the Z-direction little difference is seen between these four levels of assistance. When looking at the reaction moments, little difference is seen between the different levels of assistance. But still there is a small increase in reaction moment in the negative and positive direction of the X- and Z-component respectively. The reaction moment around the Y-axis remains near zero for all levels of assistance.

Then there is still the case when force resistance is given instead of assistance, for example when an exoskeleton is working against you. Looking at Figure 5.10, the reaction force increases in all directions, be it in the opposite direction compared to when assistance is given. For the reaction moment the differences are small, but the reaction moment round the X-axis is slightly increased. The reaction moment around the Z-axis is decreased, or increased when it crosses zero.

For the case of torque assistance, the same comparison of assistance levels is made, the results are shown in Figure 5.11. The reaction force in the X-direction increases with increasing level of assistance, as is the case in the Y- and Z-direction. The reaction moment around the X-axis is slightly increased with increasing level of assistance, mainly in the middle of the motion, since there the assistance is highest. The reaction moment around the Z-axis increases in absolute value for increasing level of assistance, so at the start and end of the motion it increases negatively and in the middle of the motion it increases positively.

When torque resistance is given, the X-component of the reaction force is decreased. As the muscle forces have to increase to overcome the resistance, the reaction force has to lower in order

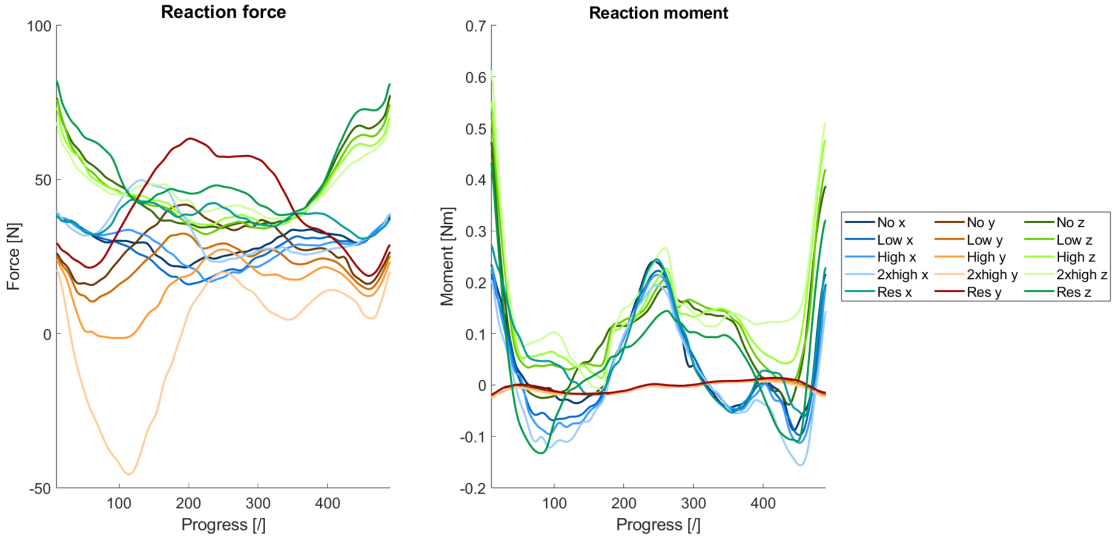


Figure 5.10: Reaction forces and moments when difference levels of force assistance are given: no, low, high, 2xhigh, resistance. The reaction forces generally decrease for increasing assistance, until the assistance is too much, then the reaction force is increased. For resisting force the reaction force is increased in absolute value. Reaction moments are slightly increased in the x- and z-direction with increasing level of assistance. With force resistance the reaction moment is slightly increased around the x-axis, and decreased around the z-axis.

to maintain the force balance. For the Y-direction, the reaction moment is increased in the same direction as when too much assistance is given, this would mean that the reaction force and muscle force are opposite in the y direction, in that case, when the muscle force increases, so thus the reaction force. The same behavior of increased reaction force is seen in the Z-direction.

For the reaction moments, the moment around the X-axis is slightly decreased, so negatively increased when it crosses zero. The moment around the Z-axis is flattened, so increased in the beginning and end of the motion, and decreased at the middle.

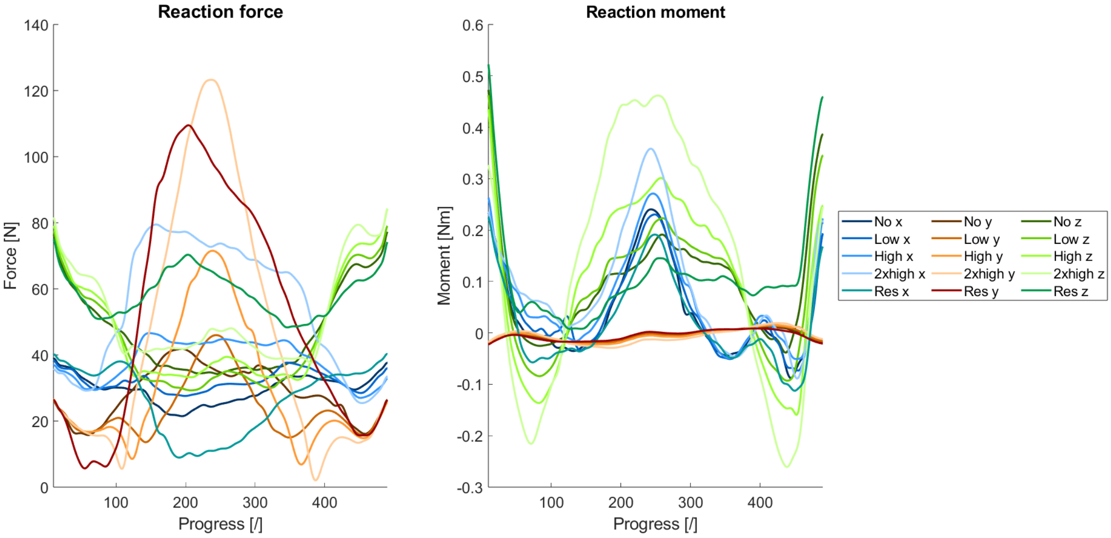


Figure 5.11: Reaction forces and moments when difference levels of torque assistance are given: no, low, high, 2xhigh, resistance. The reaction forces generally increase for increasing assistance. For resisting torque the reaction force is increased in y-and z-direction, and decreased in x-direction. Reaction moments are increased around the x- and z-axes. With torque resistance the reaction moment around the z-axis is flattened, the reaction moment around the x-axis is decreased.

Chapter 6

Discussion and future work

6.1 Conclusion

Simulations showed that providing assistance through force leads to a reduction in reaction forces while the reaction moment stays similar compared to no assistance. When providing assistance through torque, the reaction forces increased, as did the reaction moments, albeit slightly. When resistance was given, the reaction forces generally increased, in positive or negative direction, both for force and torque resistance. The same behavior is seen for the reaction moments, although the difference compared to no external load was small.

6.2 Limitations and future work

Some limitation, and therefore also possible improvements for future work, of the performed experiments and simulations.

A first limitation is the applied external load. For simplicity the applied external force and torque where in 2D and around one axis respectively. To achieve a more accurate result, the force and torque should be applied in all directions, to be as close to the reality as possible.

A second limitation concerns the point of application in case of external force assistance. In this thesis the center of mass of the humerus was used, however, this is an estimation of the point of application in reality. This also goes together with the fact that in reality the force is applied over a surface, while in the simulations it is applied in a single point. For future simulations this point of application should be more close to the point of application in reality, the force could even be divided over several points to approximate the applied force over a surface.

This leads to the next limitation, which is the fact that for this experiment the SAFeR robot was used to provide assistance. When using a wearable exoskeleton, there is no connection to the outside world, meaning that there are at least 2 connection points, which exert equal but opposite force. The SAFeR robot on the other hand, is connected to the outside world, and thus takes up this equal but opposite force. To mimic the use of an exoskeleton in a simulation as close as possible, a second force could be applied to account for the second connection point.

A last limitation is the agreement between the test subject and the OpenSim model in terms of dimensions and weight. To further improve simulations outcomes, the used musculoskeletal model could be scaled to the anthropometric measures of the test subject.

In this study the motion was measured with Xsens DOT sensors. Calculations of joint angles with these sensors are sensitive to position and orientation of sensor placement, and thus an exact replication of the performed motion is prone to deviations. To improve the replicated motion, the motion could for example be measured with a motion capture system.

Lastly, this study could be performed on multiple test subjects and with a real wearable exoskeleton to further improve outcomes. As well ass performing other motions than only a flexion extension motion.

Appendix

MATLAB scripts

MakeMotionFile.m

Adapted from matlab script provided by Xsens Support and the Developer toolkit [58].

```
function [] = MakeOpenSimFile(coord,time)

%%% Generate motion file
filepath = '...'; % Path to save file
motion_name = inputdlg('Give a name for the motion');
filename = strcat(motion_name,'.mot');

jointNameStrings = ['sternoclavicular_r2 sternoclavicular_r3
   unrotscap_r3 unrotscap_r2 acromioclavicular_r2
   acromioclavicular_r3 acromioclavicular_r1 unrothum_r1
   unrothum_r3 unrothum_r2 elv_angle shoulder_elv shoulder1_r2
   shoulder_rot elbow_flexion pro_sup deviation flexion'];

%%% Make file
data_matrix = [];
data_matrix(:,1) = time;                % Time

% jointcoordinates
data_matrix(:,2) =-0.242*coord(:,1);    % sternoclavicular_r2
data_matrix(:,3) =0.1025*coord(:,1);    % sternoclavicular_r3
data_matrix(:,4) =-0.1025*coord(:,1);   % unrotscap_r3
data_matrix(:,5) =0.242*coord(:,1);     % unrotscap_r2
data_matrix(:,6) =-0.049*coord(:,1);    % acromioclavicular_r2
data_matrix(:,7) =0.396*coord(:,1);     % acromioclavicular_r3
data_matrix(:,8) =0.178*coord(:,1);     % acromioclavicular_r1
data_matrix(:,9) =-0.178*coord(:,1);    % unrothum_r1
data_matrix(:,10) =-0.396*coord(:,1);   % unrothum_r3
data_matrix(:,11) =0.049*coord(:,1);    % unrothum_r2
data_matrix(:,12) = coord(:,2);         % elv_angle
data_matrix(:,13) = coord(:,1);         % shoulder_elv
data_matrix(:,14) =-coord(:,2);         % shoulder1_r2
data_matrix(:,15) = coord(:,3);         % shoulder_rot
data_matrix(:,16) = coord(:,4);         % elbow_flexion
data_matrix(:,17) = 0;                    % pro_sup
data_matrix(:,18) = 0;                    % deviation
data_matrix(:,19) = 0;                    % flexion

[numRows,numCols] = size(data_matrix);
ncols = num2str(numCols);
nrows = num2str(numRows);

% Create base joint angle matrix
fileID = fopen(strcat([filepath,filename{1}]),'w');
fprintf(fileID,strcat([motion_name{1},'\nnRows=',nrows,'\nnColumns
    =',ncols,'\n\n']));
fprintf(fileID,'# SIMM Motion File Header:\n');
nrange = num2str([time(1) time(length(time))]);
otherdata = num2str(1);

fprintf(fileID,strcat(['name ',motion_name{1}, '\ndatacolumns ',
    ncols,'\ndatarows ', nrows,'\notherdata ',otherdata,'\nrange ',
    nrange,'\nendheader\n']));
fprintf(fileID,strcat(['time ',jointNameStrings,'\n']));
fprintf(fileID,[repmat('%5.4f\t',1,size(data_matrix,2)),'\n'],
    data_matrix.');
fprintf(fileID,'\n');
fclose(fileID);

%%% Motion file created
text = 'Generated motion file saved to ' + convertCharsToStrings(
    strcat(filepath,filename));
display(text);

end
```

MakeExternalLoadFile.m

Adapted from matlab script provided by Xsens Support and the Developer toolkit [58].

```
function [] = MakeExternalLoadFile(assistance,angles,origin_hum,
    time,type)

import org.opensim.modeling.* % Import OpenSim Libraries

% OpenSim model
[filepath_OpenSim,filename_OpenSim,fileext_OpenSim] = fileparts('
    ...\MOBL_ARMS_41.osim'); % Path to where MOBL_ARMS model is
    saved
myModel = Model(strcat(filepath_OpenSim,'\',filename_OpenSim,
    fileext_OpenSim));
COM = myModel.get_BodySet().get('humerus').getMassCenter.getAsMat;
```

```
motion_name = inputdlg('Give a name for the force file');
filename = strcat(motion_name,'.mot');
filepath = '...'; % Path to save file

% Force data to file
data_matrix = [];
data_matrix(:,1) = time;

if strcmp(type,'force')
    % Calculate COM position in ground frame
    p = [];

    com = [0;COM(2);COM(3)];

    for i=1:length(time)
        R_ax = axang2rotm([0.0048 0.999089 0.0424 deg2rad(angles(i
            ,2))]);
        rot_ax = R_ax*[-0.998261;0.0023;0.058898];
        axang = [rot_ax' deg2rad(angles(i,1))];
        R_elv = axang2rotm(axang);
        R_rot_ax = R_elv*[0.0048;0.999089;0.0424];
        R_rot = axang2rotm([R_rot_ax' deg2rad(angles(i,3))]);
        R = R_rot*R_elv;
        P = R*com;
        p = [p;P'];
    end

    % Assistance data to new file in COM of humerus
    data_matrix(:,2) = assistance(:,1); % Force x
    data_matrix(:,3) = assistance(:,2); % Force y
    data_matrix(:,4) = 0;               % Force z

    data_matrix(:,5) = p(:,1)+origin_hum(1:length(time),1);
       % Point x
    data_matrix(:,6) = p(:,2)+origin_hum(1:length(time),2);
       % Point y
    data_matrix(:,7) = p(:,3)+origin_hum(1:length(time),3);
       % Point z

    jointNameStrings = ['humerus_force_vx humerus_force_vy
       humerus_force_vz humerus_force_px humerus_force_py
       humerus_force_pz'];

elseif strcmp(type,'torque')
    % Assistance data to new file
    data_matrix(:,2) = 0;          % Torque x
    data_matrix(:,3) = 0;          % Torque y
    data_matrix(:,4) = assistance; % Torque z
    jointNameStrings = ['humerus_torque_x humerus_torque_y
       humerus_torque_z'];

else
    error('Please indicate whether the assistance is through force
        of through torque.')
end

[numRows,numCols] = size(data_matrix);
ncols = num2str(numCols);
nrows = num2str(numRows);

file_ID = fopen(strcat([filepath,filename{1}]),'w');
fprintf(file_ID,strcat([motion_name{1},'\nversion=1','\nnRows=',
   nrows,'\nnColumns=',ncols,'\ninDegrees=no','\n\nendheader\n']))
   ;
fprintf(file_ID,strcat(['time ',jointNameStrings,'\n']));
fprintf(file_ID,[repmat('%5.4f\t',1,size(data_matrix,2)),'\n'],
   data_matrix.');
fprintf(file_ID,'\n');
fclose(file_ID);

%%% Motion file created
text = 'Generated force file saved to ' + convertCharsToStrings(
   strcat(filepath,filename));
display(text);

end
```

Bibliography

[1] Dick Ameln et al. "The Stabilizing Function of Superficial Shoulder Muscles Changes Between Single-Plane Elevation and Reaching Tasks". In: *IEEE Transactions on Biomedical Engineering* PP (June 2018), pp. 1–1. DOI: `10.1109/TBME.2018.2850522`.

[2] Michael Skipper Andersen. "4 - Introduction to musculoskeletal modelling". In: *Computational Modelling of Biomechanics and Biotribology in the Musculoskeletal System (Second Edition)*. Ed. by Zhongmin Jin, Junyan Li, and Zhenxian Chen. Second Edition. Woodhead Publishing Series in Biomaterials. Woodhead Publishing, 2021, pp. 41–80. ISBN: 978-0-12-819531-4. DOI: `https://doi.org/10.1016/B978-0-12-819531-4.00004-3`. URL: `https://www.sciencedirect.com/science/article/pii/B9780128195314000043`.

[3] Xsens Technologies B.V. *Xsens DOT User Manual*. English. Version G. XSens. June 2022. 34 pp.

[4] Erica Beaucage-Gauvreau et al. "Validation of an OpenSim full-body model with detailed lumbar spine for estimating lower lumbar spine loads during symmetric and asymmetric lifting tasks". In: *Computer Methods in Biomechanics and Biomedical Engineering* 22.5 (2019). PMID: 30714401, pp. 451–464. DOI: `10.1080/10255842.2018.1564819`. eprint: `https://doi.org/10.1080/10255842.2018.1564819`. URL: `https://doi.org/10.1080/10255842.2018.1564819`.

[5] *Biomechanics of Bodies*. URL: `https://www.bob-biomechanics.com`. accessed: 01.05.2023.

[6] Dimitra Blana et al. "A musculoskeletal model of the upper extremity for use in the development of neuroprosthetic systems". In: *Journal of biomechanics* 41 (Feb. 2008), pp. 1714–21. DOI: `10.1016/j.jbiomech.2008.03.001`.

[7] Edward Chadwick et al. "Real-Time Simulation of Three-Dimensional Shoulder Girdle and Arm Dynamics". In: *IEEE transactions on bio-medical engineering* 61 (July 2014), pp. 1947–1956. DOI: `10.1109/TBME.2014.2309727`.

[8] *Cometa*. URL: `https://www.cometasystems.com/`. accessed: 02.05.2023.

[9] J.H. de Groot and R. Brand. "A three-dimensional regression model of the shoulder rhythm". In: *Clinical Biomechanics* 16.9 (2001), pp. 735–743. ISSN: 0268-0033. DOI: `https://doi.org/10.1016/S0268-0033(01)00065-1`. URL: `https://www.sciencedirect.com/science/article/pii/S0268003301000651`.

[10] Scott Delp et al. "OpenSim: Open-Source Software to Create and Analyze Dynamic Simulations of Movement". In: *Biomedical Engineering, IEEE Transactions on* 54 (Dec. 2007), pp. 1940–1950. DOI: `10.1109/TBME.2007.901024`.

[11] OpenSim Documentation, ed. *Getting Started with Inverse Dynamics*. URL: `https://simtk-confluence.stanford.edu:8443/display/OpenSim/Getting+Started+with+Inverse+Dynamics`. accessed: 29.07.2023.

[12] OpenSim Documentation, ed. *How Static Optimization Works.* URL: https://simtk-confluence.stanford.edu:8443/display/OpenSim/How+Static+Optimization+Works. accessed: 07.04.2023.

[13] OpenSim Documentation, ed. *Joint Reactions Analysis.* URL: https://simtk-confluence.stanford.edu:8443/display/OpenSim/Joint+Reactions+Analysis. accessed: 08.03.2023.

[14] OpenSim Documentation, ed. *Millard 2012 Muscle Models.* URL: https://simtk-confluence.stanford.edu:8443/display/OpenSim/Millard+2012+Muscle+Models. accessed: 02.05.2023.

[15] OpenSim Documentation, ed. *Musculoskeletal Models.* URL: https://simtk-confluence.stanford.edu:8443/display/OpenSim/Musculoskeletal+Models. accessed: 04.12.2022.

[16] OpenSim Documentation, ed. *Tutorial 1 - Intro to Musculoskeletal Modeling.* URL: https://simtk-confluence.stanford.edu:8443/display/OpenSim/Tutorial+1+-+Intro+to+Musculoskeletal+Modeling. accessed: 24.11.2022.

[17] OpenSim Documentation, ed. *Welcome to OpenSim.* URL: https://simtk-confluence.stanford.edu:8443/display/OpenSim/Welcome+to+OpenSim. accessed: 04.12.2022.

[18] J. M. Konrath E. Ribera D'alcala J. A. Voerman and A. Vydhyanathan. "Xsens DOT Wearable Sensor Platform White Paper". In: (Feb. 2021).

[19] Edwin F. Bartholomew Frederich H. Martini Judi L. Nath. *Fundamentals of Anatomy & Physiology.* 10th ed. Pearson, 2015, pp. 276–278. ISBN: 978-0-321-90907-7.

[20] Edwin F. Bartholomew Frederich H. Martini Judi L. Nath. *Fundamentals of Anatomy & Physiology.* 10th ed. Pearson, 2015, pp. 358–363. ISBN: 978-0-321-90907-7.

[21] Laura Gastaldi et al. "Upper Limbs Musculoskeletal OpenSim Model: Customization and Assessment". In: Jan. 2021, pp. 162–170. ISBN: 978-3-030-55806-2. DOI: 10.1007/978-3-030-55807-9_19.

[22] Milad Gholam et al. "Investigating Glenohumeral Joint Contact Forces and Kinematics in Different Keyboard and Monitor Setups using Opensim". In: *Journal of Biomedical Physics and Engineering* (2022). ISSN: 2251-7200. eprint: https://jbpe.sums.ac.ir/article_48612_3e04337e8659ea45c0cab44806ce8671.pdf. URL: https://jbpe.sums.ac.ir/article_48612.html.

[23] Jon Giese, ed. *Stabilizing and Strengthening the Shoulder.* URL: https://www.racmn.com/blog/stabilizing-and-strengthening-the-shoulder. last updated: 26.05.2022.

[24] J. Giphart et al. "Effect of Plane of Arm Elevation on Glenohumeral Kinematics A Normative Biplane Fluoroscopy Study". In: *The Journal of bone and joint surgery. American volume* 95 (Feb. 2013), pp. 238–45. DOI: 10.2106/JBJS.J.01875.

[25] Kristy Godoy and Luciano Menegaldo. "UPPER LIMB BIOMECHANICS USING OPENSIM DURING WHEELCHAIR PROPULSION". In: Jan. 2018. DOI: 10.26678/ABCM.ENEBI2018.EEB18-0214.

[26] Oliver Jones. *The Shoulder Joint.* URL: https://teachmeanatomy.info/upper-limb/joints/shoulder/. last updated: 27.09.2022.

[27] Scott L. Delp Katherine R.S. Holzbaur Wendy M. Murray. "A Model of the Upper Extremity for Simulating Musculoskeletal Surgery and Analyzing Neuromuscular Control". In: *Annals of Biomedical Engineering* 33.6 (June 2005), pp. 829–840. DOI: 10.1007/s10439-005-3320-7.

[28] Yujun Lai et al. “Preliminary Validation of Upper Limb Musculoskeletal Model using Static Optimization”. In: vol. 2021. Nov. 2021. DOI: 10.1109/EMBC46164.2021.9629494.

[29] Janna Langholz, Gunnar Westman, and Magnus Karlsteen. “Musculoskeletal Modelling in Sports - Evaluation of Different Software Tools with Focus on Swimming”. In: *Procedia Engineering* 147 (Dec. 2016), pp. 281–287. DOI: 10.1016/j.proeng.2016.06.278.

[30] Mirza Nada Masood, Ali Adnan, and Ishak Mohamad Khairi. “Interactive Software-based Modeling for Gait Analysis of Musculoskeletal Structures”. In: *Makara Journal of Technology* 26 (2 2022). DOI: 10.7454/mst.v26i2.1542.

[31] MathWorks, ed. *filloutliers*. URL: https://nl.mathworks.com/help/matlab/ref/filloutliers.html#bvlnf4n-1-movmethod. accessed: 10.08.2023.

[32] Daniel C. McFarland et al. “Spatial Dependency of Glenohumeral Joint Stability During Dynamic Unimanual and Bimanual Pushing and Pulling”. In: *Journal of Biomechanical Engineering* 141.5 (Mar. 2019). 051006. ISSN: 0148-0731. DOI: 10.1115/1.4043035. eprint: https://asmedigitalcollection.asme.org/biomechanical/article-pdf/141/5/051006/6389652/bio_141_05_051006.pdf. URL: https://doi.org/10.1115/1.4043035.

[33] Kevin J. McQuade and Gary L. Smidt. “Dynamic Scapulohumeral Rhythm: The Effects of External Resistance During Elevation of the Arm in the Scapular Plane”. In: *Journal of Orthopaedic & Sports Physical Therapy* 27.2 (1998). PMID: 9475136, pp. 125–133. DOI: 10.2519/jospt.1998.27.2.125. eprint: https://doi.org/10.2519/jospt.1998.27.2.125. URL: https://doi.org/10.2519/jospt.1998.27.2.125.

[34] Sariah Mghames et al. “A Neuromuscular-Model Based Control Strategy to Minimize Muscle Effort in Assistive Exoskeletons”. In: vol. 2019. June 2019, pp. 963–970. DOI: 10.1109/ICORR.2019.8779456.

[35] Dimitrios Mytilinaios MD PhD Nicola McLaren MSc. *Glenohumeral joint*. URL: https://www.kenhub.com/en/library/anatomy/the-shoulder-joint. last reviewed: 26.09.2022.

[36] David Novak. *Guide to Shoulder Anatomy*. URL: https://www.sports-health.com/sports-injuries/shoulder-injuries/guide-shoulder-anatomy. published: 06.13.2019.

[37] David Novak. *Soft Tissues of the Shoulder*. URL: https://www.sports-health.com/sports-injuries/shoulder-injuries/soft-tissues-shoulder. published: 06.13.2019.

[38] Kenny Smith Peter Loan Scott Delp and Krystyne Blaikie. *SIMM 4.0 for Windows®. Software for Interactive Musculoskeletal Modeling - User's Manual*. English. June 2004. 260 pp.

[39] Physiopedia. *Dynamic Stabilisers of the Shoulder Complex — Physiopedia*. [Online; accessed 19-November-2022]. 2022. URL: https://www.physio-pedia.com/index.php?title=Dynamic_Stabilisers_of_the_Shoulder_Complex&oldid=293191.

[40] Physiopedia. *Glenohumeral Joint — Physiopedia*. [Online; accessed 19-November-2022]. 2021. URL: https://www.physio-pedia.com/index.php?title=Glenohumeral_Joint&oldid=278612.

[41] Physiopedia. *Levator Scapulae — Physiopedia*. [Online; accessed 29-November-2022]. 2022. URL: https://www.physio-pedia.com/index.php?title=Levator_Scapulae&oldid=298632.

[42] Physiopedia. *Rhomboids — Physiopedia*. [Online; accessed 29-November-2022]. 2022. URL: https://www.physio-pedia.com/index.php?title=Rhomboids&oldid=298681.

[43] Physiopedia. *Scapulohumeral Rhythm — Physiopedia.* [Online; accessed 1-November-2022]. 2021. URL: https://www.physio-pedia.com/index.php?title=Scapulohumeral_Rhythm&oldid=264640.

[44] Physiopedia. *Shoulder — Physiopedia.* [Online; accessed 29-November-2022]. 2022. URL: https://www.physio-pedia.com/index.php?title=Shoulder&oldid=320813.

[45] Physiopedia. *Trapezius — Physiopedia.* [Online; accessed 29-November-2022]. 2022. URL: https://www.physio-pedia.com/index.php?title=Trapezius&oldid=298339.

[46] R. Richardson et al. "Errors Associated With Utilizing Prescribed Scapular Kinematics to Estimate Unconstrained, Natural Upper Extremity Motion in Musculoskeletal Modeling". In: *Journal of Applied Biomechanics* 33 (June 2017), pp. 1–18. DOI: 10.1123/jab.2016-0346.

[47] MD Ross Hauser, ed. *Physical therapy and exercise for shoulder pain. When it works, when it does not work.* URL: https://www.caringmedical.com/prolotherapy-news/physical-therapy-exercise-shoulder-pain-works-work/. last updated: 24.08.2022.

[48] *SAFER: Rehabilitation Robot. A Rehabilitation Robot that can sense & feel the patient.* URL: https://www.brubotics.eu/projects/safer-rehabilitation-robot. accessed: 20.04.2023.

[49] Katherine R. Saul et al. "Benchmarking of dynamic simulation predictions in two software platforms using an upper limb musculoskeletal model". In: *Computer Methods in Biomechanics and Biomedical Engineering* 18.13 (2015). PMID: 24995410, pp. 1445–1458. DOI: 10.1080/10255842.2014.916698. eprint: https://doi.org/10.1080/10255842.2014.916698. URL: https://doi.org/10.1080/10255842.2014.916698.

[50] Jason Scibek and Christopher Carcia. "Assessment of scapulohumeral rhythm for scapular plane shoulder elevation using a modified digital inclinometer". In: *World journal of orthopedics* 3 (June 2012), pp. 87–94. DOI: 10.5312/wjo.v3.i6.87.

[51] *SENIAM.* URL: http://www.seniam.org/. accessed: 10.04.2023.

[52] Ajay Seth et al. "A Biomechanical Model of the Scapulothoracic Joint to Accurately Capture Scapular Kinematics during Shoulder Movements". In: *PLOS ONE* 11.1 (Jan. 2016), pp. 1–18. DOI: 10.1371/journal.pone.0141028. URL: https://doi.org/10.1371/journal.pone.0141028.

[53] Ajay Seth et al. "Muscle Contributions to Upper-Extremity Movement and Work From a Musculoskeletal Model of the Human Shoulder". In: *Frontiers in Neurorobotics* 13 (Nov. 2019). DOI: 10.3389/fnbot.2019.00090.

[54] Ajay Seth et al. "OpenSim: Simulating musculoskeletal dynamics and neuromuscular control to study human and animal movement". In: *PLOS Computational Biology* 14.7 (July 2018), pp. 1–20. DOI: 10.1371/journal.pcbi.1006223. URL: https://doi.org/10.1371/journal.pcbi.1006223.

[55] Shoulderdoc.co.uk, ed. *Shoulder Ligaments.* URL: https://www.shoulderdoc.co.uk/article/1179#:~:text=Glenohumeral%5C%20Ligaments%5C%20(GHL)%5C%3A&text=In%5C%20the%5C%20shoulder%5C%2C%5C%20the%5C%20joint,middle%5C%20and%5C%20inferior%5C%20glenohumeral%5C%20ligaments. last updated: 09.09.2022.

[56] *SIMM 5.0 and OpenSim 2.0.* English. Tech. rep. Mar. 2010. 3 pp.

[57] *Who is AnyBody Technology?* URL: https://www.anybodytech.com. accessed: 01.05.2023.

[58] XSens, ed. *Software & documentation.* URL: `https://www.movella.com/support/software-documentation?hsCtaTracking=39d661fa-2ea8-4478-955e-01d0d8885f14%7C3ad1c7d6-9c3a-42e9-b424-5b15b9d0924e`. accessed: 19.10.2022.