

POLITECNICO DI MILANO

SCHOOL OF INDUSTRIAL AND INFORMATION
ENGINEERING
MASTER OF SCIENCE IN AUTOMATION AND
CONTROL ENGINEERING



Human arm impedance estimation: techniques and
simulations

Supervisor: Prof. Luca BASCETTA

Master Graduation Thesis by:
Maksim Olegovich OBRAZTSOV
914838

Academic Year 2018-2019

Dedications

I would like to dedicate this project to my beloved mother and to my dearest friend S. who helped me to stay firm, stay positive and made me not feel completely alone and abandoned in these difficult times.

Acknowledgments

The list of people who dedicated their time to assist with various tasks that were tackled during the year of working on this project: Professor Luca Bascetta, my supervisor; Doctor Hang Su, who helped operating the laboratory equipment; the people of NEARLAB, who provided the necessary equipment for the experiments. Thank you.

A shout-out to the friends I made during my studies in Politecnico: Andrea, Giovanni, Ludovico, Mehdi and Nicola, who managed to graduate before me, but kept in touch.

An unfortunate, yet an important mention of the SARS-CoV-2 pandemic, which prevented the successful execution of all the planned activities and experiments. Despite that, the project was finished on time.

Contents

Introduction	1
Objectives and methods	2
Thesis structure	2
1 Digital human modeling: an overview	3
1.1 ‘Black box’ modeling	4
1.1.1 Parametric models	6
1.1.2 Non-parametric models	6
1.1.3 Challenges	8
1.2 Reductionist modeling	9
1.2.1 The skeletal system	11
1.2.2 Muscle components	14
1.2.3 Neural components	16
1.2.4 Challenges	18
1.3 Conclusions	19
2 Modeling of human arm dynamics	21
2.1 Reduced-complexity model	21
2.2 Nonlinear, simplified model	24
2.2.1 Remark on endpoint rotational stiffness	26
2.2.2 Remark on polar coordinates	27
2.3 EMG-driven ‘black box’ approach	29
2.4 EMG-driven genetic model	30
2.5 Statistical EMG-driven modeling	30
2.6 A structural EMG-driven model	31
2.7 Conclusions	33
3 Simulations	35
3.1 Experiment 1: simplified linear identification	36
3.1.1 Motion	36
3.1.2 Identification	36

CONTENTS

3.2	Experiment 2: complete linear identification	39
3.2.1	Motion	39
3.2.2	Identification	39
3.2.3	Results	41
3.3	Experiment 3: simplified quadratic identification	43
3.3.1	Motion	43
3.3.2	Identification	43
3.3.3	Results	44
3.4	Experiment 4: complete quadratic identification	45
3.4.1	Motion generation	45
3.4.2	Identification	45
3.4.3	Results	46
3.5	Experiment 5: polar model exploration	47
3.5.1	Motion	47
3.5.2	Identification and results	48
	Discussion	53
	Future development	53
	Conclusions	54
	A Proofs	57
	Bibliography	59

List of Figures

1.1.1 Schematic diagram of the system identification context	5
1.2.1 Neuromusculoskeletal interaction diagram [1]	11
1.2.2 D-H model of a human arm	12
1.2.3 Triangle model of a human arm	13
1.2.4 Muscle paths example [1]	14
1.2.5 Muscle model representation	15
1.2.6 Schematic of actions of muscle cross-bridge	16
1.2.7 Effects of muscle on SEMG signal-force relationship [2]	17
1.2.8 A highly schematized flow diagram of the hierarchical and parallel aspects of the motor system [3]	18
2.2.1 Postural component of the model	25
2.2.2 Human arm triangle	26
2.6.1 The musculoskeletal model [4]	32
3.1.1 Linear motion position and velocity	37
3.1.2 Perturbations	37
3.1.3 Stiffness estimation	38
3.1.4 Damping estimation	39
3.1.5 Percentage estimation error	40
3.2.1 Circular motion position and velocity	41
3.2.2 Time history of the simulation	42
3.2.3 Results	42
3.2.4 Stiffness to position relation	43
3.3.1 Time history of the simulation	44
3.3.2 Results	44
3.4.1 Time history of the simulation	46
3.4.2 Results	46
3.4.3 Stiffness to position relation	47
3.5.2 Cartesian fit quality	48
3.5.3 Polar fit quality	49

LIST OF FIGURES

3.5.4 Cartesian fit quality	49
3.5.5 Polar fit quality	50

List of Tables

1.1	D-H Parameters of human right arm kinematic model	12
3.1	Arm stiffness coefficients	35
3.2	Arm damping coefficients	36
3.3	Linear motion parameters	36

Abstract

Human-robot interaction becomes an increasingly important field of research as robots permeate our life, but how to estimate the effect that the humans have on the system? Through the modeling of the human. Of many ways to model the human, this thesis explores the biomechanical modeling of the human arm.

For the biomechanical modeling of the human in general, 'black box' approaches are usually utilized. The reductionist models exist, but they are used to either design the 'black box' models or to support those at the base level. Various studies dedicated to the modeling of the human arm are summarized by this thesis. There are two types of studies: ones that estimate the human arm impedance using the spatial data and muscle activation readings and those that map surface EMG signals of the arm to the forces generated at the endpoint. The former approaches seem more effective.

The summarized studies received some remarks. One study in particular received more elaborate suggestions that were put to the test in simulations. The simulations showed promising results, providing ground for the future research on the topic. Further suggestions were given to ensure advances in the future works.

Keywords: human-robot interaction; human arm impedance; human modeling; survey; least squares estimation.

Sommario

L'interazione uomo-robot diventa un campo di ricerca sempre più importante poiché i robot permeano la nostra vita, ma come possibile stimare l'effetto che gli umani hanno sul sistema? Attraverso la modellistica dell'umano. Di molti modi per modellare l'essere umano, questa tesi esplora la modellazione biomeccanica del braccio umano.

Per la modellistica biomeccanica dell'essere umano in generale, vengono generalmente utilizzati approcci a "black box". Esistono modelli riduzionisti, ma vengono utilizzati per progettare i modelli "black box" o per supportare quelli a livello base. Vari studi dedicati alla modellazione del braccio umano sono riassunti da questa tesi. Esistono due tipi di studi: quelli che stimano l'impedenza del braccio umano usando i dati spaziali e le letture di attivazione muscolare e quelli che mappano i segnali EMG di superficie del braccio alle forze generate all'endpoint. I primi approcci sembrano più efficaci.

Gli studi riassunti hanno ricevuto alcune osservazioni. Uno studio in particolare ha ricevuto suggerimenti più elaborati che sono stati messi alla prova nelle simulazioni. Le simulazioni hanno mostrato risultati promettenti, fornendo terreno per le future ricerche sull'argomento. Sono stati dati ulteriori suggerimenti per garantire progressi nei lavori futuri.

Parole chiave: interazione uomo-robot; impedenza del braccio umano; modellistica umana; indagine; stima dei minimi quadrati.

Introduction

Human-robot interaction is a tremendously important concept that was guiding (and guided by) many fields of science. Robots are starting to participate in many aspects of human lives and nowadays they are not limited to restricted cells at the factory anymore. The robots can work together with an operator not only in the same room, but also on the same task. Moreover, even people without the technical knowledge can engage with robotic systems as users. All that calls for a careful design of the human-robot interaction. But even though we can use modeling to predict the behavior of the robot and the environment, how can the human be accounted for in the control algorithms of the robot?

Currently, a complete model of the human is still an unsolved mystery. There are just too many aspects of human, with some of them being: varying shapes and sizes that fluctuate around the specific proportions, dynamic properties of the human body, the effects of aging, the neural system of the human and human behavior. While more physical aspects, such as mechanical properties of human limbs, can be described using 'black box' approaches, the more human neural control system is involved, the more uncertainty is introduced in the human interactions, calling for statistical or probabilistic models. Still, approximations and 'black box' models are still important, since they might give insight on how the real system works and some of such models still can be very helpful in practical applications. Such advances will surely help us accelerate the understanding of the human nature.

However, as the research in the field has advanced, it might be difficult to keep track of all the new technologies and methods introduced by scientists all around the world. When many complex papers are published on the same topic, it might be easy to miss an approach that is distinct and may inspire a breakthrough research.

Objectives and methods

The main objective of this thesis project is to ease the burden of information gathering for the future researches. The work will be dedicated to human modeling, with main focus on the estimation of the properties of the human arm. Focusing only on a particular topic will help to highlight the differences between the approaches that has been developed so far.

The objective will be accomplished by searching for the published, relevant papers using the online search engines and reviewing them. For each study, a brief explanation of the theoretical base will be made and the study's conclusions will be provided. Then, a remark from the author of this project will follow. The remark might vary from a short comment on the possible improvement to an elaborated proposition on the way the study could have been improved. For some studies, the suggestions will be modeled and showcased in simulations with help of the MATLAB software.

Thesis structure

The thesis is organized in the following manner:

Chapter 1: an overview of the digital human modeling field, starting from the general picture and then shifting focus towards the fundamental knowledge behind the analysis and modeling of the neuromusculoskeletal structures. The purpose of this chapter is to introduce to the reader relevant techniques and sources of information on the field as well as serve as the main reference point for the further chapters.

Chapter 2: an overview of the specific studies dedicated to the biomechanical modeling of the human arm. Among the provided studies, comparisons will be drawn and suggestions on the potential improvements will be given.

Chapter 3: details and explanations of the simulations based on the knowledge mentioned in the previous chapters that will be used to test the suggestions that were given in the previous chapter.

Then, a discussion chapter will follow. In that chapter the impact of the project will be reflected upon and the potential improvements will be proposed.

Chapter 1

Digital human modeling: an overview

Nowadays, the need of digital modeling and simulation of processes quickly rises, as being able to predict the outcome of the processes lets humans avoid unnecessary damages or costs and can give them an insight on how to pick the right action. Because humans often play active roles in said processes and influence their outcome, it becomes clear that it is also important to use simulations that take humans into account. Thus, the conglomeration of knowledge on how to create a useful digital representation of the human and use it in a relevant simulation is what can be called the field of digital human modeling.

Digital human modeling is a young and a very broad field that is adjacent to many fields of science. The abundance of ways to model a human calls for a categorization, which, of course, cannot be absolute. Nevertheless, human modeling can be represented by the five categories [5].

- **Anthropometric models** are focused on representing the sizes and shapes of a human body and its motion. Mainly used to design workspaces with an emphasis on used space and reachability of the workspace functions.
- **Models for production design** are less detailed than anthropometric models and focus on human motion and behaviour with relation to a working process or their workstation. These models can be used to inspect the ergonomics of a workspace or to consider factory planning.
- **Biomechanical models** represent human body as a mechanical system and compute its dynamic behaviour. They can be used to consider

ergonomics, safety or to simulate any mechanical interaction between human and a relevant structure.

- **Anatomical models** focus on anatomy of a human body and what contains within it's volumes. The models can be used for surgery planning, injury simulation or motion capture.
- **Cognitive models** simulate the information flow between sensory organs of a human and their actuators. Simulating the behaviour of the human has many applications where human performance or human factor are important.

An approach to modeling can be defined by how much information is put into a model structure and it's parameters.

Because of the abundance of approaches to modeling, let's define their range by explaining the two distinctive extremities: a 'black box' approach [6] versus an 'a priori' or a reductionist approach [7, 8].

This project will mainly consider the biomechanical modeling and provide relevant examples from the field. The goal of this chapter is to provide the theoretical basis of the field in question. This information will be used in the following chapters.

1.1 'Black box' modeling

In a 'black box' modeling approach, a system is considered a transformation between the input and the output data and modeled as such. This formulation coincides with a general definition of system identification – the process of building mathematical models of systems based on measurements of their inputs and outputs. To create a 'black box' model means to solve a system identification problem, which can be illustrated by a Figure 1.1.1. The solution of such a problem can be decomposed into the following steps:

- **Choose an objective.** Considering the diagram in Figure 1.1.1, the objective can be to identify the:
 - *Deterministic model:* P , a relationship between $u(t)$ and $z(t)$, assuming that the noise signal $w_1(t)$ is zero. It is chosen when it is necessary to understand the function of the system.
 - *Noise model:* F_n , a relationship between the output and a noise model $W_2(t)$, given observations of the output. It is chosen when the inputs cannot be measured.

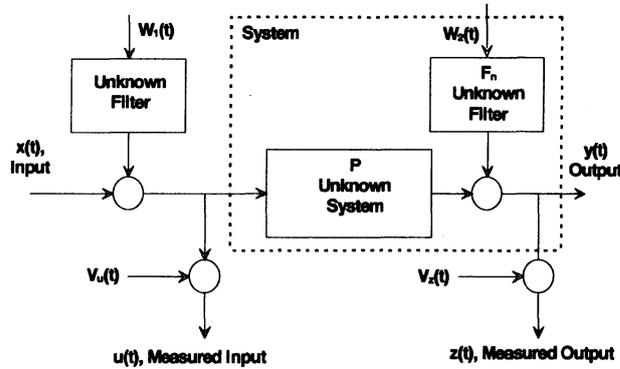


Figure 1.1.1: Schematic diagram of the system identification context

- *Complete model*: both P and F_n , to generate accurate predictions of the outputs.
- **Choose the structure.** While the ‘black box’ approach requires less a priori information to setup the model, it is still necessary to make assumptions and select an internal form of a black box. These options will be discussed in the following sections.
- **Design the inputs.** Input design is essential for getting as much information from the system as possible and it will determine the capabilities of the model to predict outputs correctly.
- **Estimate the parameters.** With structure selected and data obtained, a fitting algorithm will be executed. A suitable identification algorithm must be able to tackle the computational complexity of the model and converge to a solution using the obtained amount of input-output data.
- **Model validation.** An identified model must be able to predict well the input-output relations of the different input scenarios for the original system and, hopefully, input-output relations of the similar systems. The model is unusable if it can only predict the behaviour in scenarios that were used to identify the model.

When an objective is chosen, it is necessary to select a structure to be identified. Approaches to the identification can be divided into two groups: “parametric” and “non-parametric”.

1.1.1 Parametric models

A **parametric approach** is defined by analytic expressions with a relatively small amount of parameters. This approach is typically represented by a discrete, weighted relation between measured inputs and outputs. These relations can be linear (e.g Output Error OE, Moving Average MA, Autoregressive AR, ARMA(-X) models) or nonlinear (NARMAX model) [9].

Such models were used to study EMG signals [10], which is an essential part of muscle activation detection and estimation of various parameters of human limbs. Considering that the recordings of EMG signals can be noisy, such models can be very relevant for EMG signal processing as well.

A major advantage of parametric methods is their flexibility in incorporating highly nonlinear and non-stationary behaviour.

Disadvantages of these methods are: a necessity to establish a structure of a model a priori and large errors if the structure was chosen poorly; the results are discrete.

These properties make this approach suitable when it is possible to choose the right structure a priori, when discrete results are allowed or when the complexity of computations is important.

1.1.2 Non-parametric models

A **non-parametric** approach is to choose a generic model structure with a relatively high amount of parameters. The non-parametric relations take variety of forms.

The advantages of using the non-parametric modeling are: only a few a priori assumptions needed (e.g linearity, stationarity, etc), these assumptions can be verified experimentally; a possibility of providing the results directly in continuous time; relative straightforwardness between parameter variance and input selection [11]; possibility of providing an insight into system's order or structure.

The main disadvantage of this approach is the amount of parameters that is needed to fit model outputs to measured outputs. This can make computations more difficult (important for real-time algorithms) or make it harder to understand the contribution of a specific parameter.

This approach is suitable for a situation when little is known about the system to be identified, computational complexity is not important or all that matters is the input-output relation, with no need of considering the actual mechanisms of the modeled phenomenon.

Quasi-static linear models

Quasi-static linear models can be used when it is possible to assume that the modeled system behaves linearly about a constant operating point. For example, a second-order model of form:

$$TQ(t) = I(l)\frac{d^2q(t)}{dt^2} + B(l)\frac{dq(t)}{dt} + K(l)q(t) \quad (1.1.1)$$

has been found to predict the joint torque in a neuromuscular system, $TQ(t)$, given by small perturbations of joint position, $q(t)$. The model is quasi-linear because the inertial (I), viscous (B) and elastic (K) parameters vary significantly with operating point [11]. These models can give an insight into system properties and it is even easier for linear systems, where it is common to use Bode plots or other engineering techniques for analysis.

A model described in Section 2.2 can be an example of a quasi-static linear modeling. Even though in the example the operation point l would also depend on time (being exactly $q(t)$), in practice, during linear online estimation of the parameters as Section 3.2 shows, the model's behavior is exactly quasi-static between the two consecutive estimations. Any linearization like this would fit this approach.

Time-varying models

Time-varying models can extend the case of the quasi-linear modeling by introducing a reference trajectory. A nonlinear system is thus approximated by the linearization around the sequence of operating points. The main advantage of this approach is that important quantities, such as joint stiffness of reflex EMG, can be characterized with high accuracy and temporal resolution. However, as for the disadvantage, the resulting description will be valid only for a tested trajectory and will be difficult to generalize.

Nonlinear models

Some systems can be represented as a series of a dynamic linear (L) part and a static nonlinear (N) part. Simple models of this type include the Wiener (LN), the Hammerstein (NL) and the sandwich (LNL) model. More general dynamic nonlinear systems can be described by functional expansions, such as Volterra and Wiener series. Functional expansion methods potentially have the ability to describe systems of any complexity, but they are rarely used because of their computational difficulty, requirement of white inputs and sensitivity to measurement noises. Dynamic nonlinear systems can also

be modeled as a finite number of parallel LNL cascades or Wiener systems. For example, the paper “*EMG-force modeling using parallel cascade identification*” [12] described a process of mapping the EMG measurements to the force, induced at the wrist of the human arm. The study claimed a solid 44% of improvement in force prediction over a Hill-based model (for details, Subsection 1.2.2), which comments on how flexible the approach is.

1.1.3 Challenges

Input design is extremely important for system identification goals, as ability of identifying or validating a model largely determined by input’s properties.

For linear systems, the identification requires that the input signal must be persistently exciting and must excite all modes of the system. In reality, such an input is rarely allowable and, therefore, there will be a trade-off in the identification quality. Parametric models have an ability to use less rich inputs, given that the model structure has good compatibility with the real system. For nonlinear systems, the design is more difficult, because the outputs can be influenced by more than just input’s frequency.

Appropriate inputs will surely be limited to a certain frequency range. For that reason, the resulting model will also be limited as well. Such limitations can make these models impossible to use as submodels, as they may be incompatible with other parts of the model.

Another challenge of using a ‘black box’ approach is dealing with **noise**. Measurement or other noise components influence both input and output signals and will produce an identification error due to the additional system excitement or interfere with the identification algorithm’s work. While filtering can help to reduce the noise, it will affect the signal amplitude, which can also be important in case of nonlinear systems.

The presence of the **feedback** loop in the system is likely to complicate the identification of the open loop system. If it is impossible to introduce a perturbation within the loop, a ‘black box’ approach will yield a model that describes an overall closed loop system. In case of the neuromuscular applications with alive actors it is very important, since the mechanisms of neural feedback are complex, especially in humans.

For the same reason, an **application field**, in which model identification is used, can also add difficulty to the procedure. A field will certainly impose constraints on input design and ability to measure signals. For example, in case of modeling of a human arm, the input properties are limited by risks of injury and fatigue and some properties — like muscle contraction — cannot be measured directly.

1.2 Reductionist modeling

The idea of reductionism in modeling (also known as structural modeling or ‘a priori’ modeling) is to reduce a complex system into smaller components, related to prior knowledge (e.g Newtonian physics) and create a complete system model out of these components. Even though simplifications are the core of this approach, the resulting models can become complex enough to describe real systems through combination of individually produced submodels.

The main advantage of structural modeling is that the model itself bears a certain amount of knowledge, which allows for interpretation of how particular internal variables and parameters can be related to the properties of the model. It is often the case that a structural model is derived from a black box model.

As for the disadvantages, these models can have many parameters, distributed among numerous submodels, and it might be impossible to identify specific parameters from input-output data of the complete system. This can be countered by performing reductionist experiments, where the behaviour of a subsystem is isolated, when possible. Several modeling steps can be mentioned when the reductionist approach is chosen:

- **Choosing model complexity.** How simplified or isolated a model should be? For example, when modeling a limb, muscles can be lumped into groups. Instead, they can be treated as fibers or even be modeled on a cellular level. It is also often impractical to model structures that do not contribute to a studied behaviour, but neglecting parts of the studied system should be done carefully, especially in the neuromuscular case, where the internal mechanisms can be subtly interconnected. Neglecting contributing mechanisms would at least introduce errors, if not draw the model unusable.
- **Choosing model structure.** The bottom ‘layer’ of the structured model is likely to be a simplification of the actual system’s component. What is built upon this ‘layer’ and how simplified it is will determine how much information is put inside the model and how difficult the next step is going to be. It is important to remember that the more information is put into the model, the more likely it is to make a wrong assumption about the real structure of a system.

- **Parameter estimation.** This step is similar to the one described for a ‘black box’ approach. What is different, however, is the potential ability to estimate the parameters in chunks by isolating behaviours of the subsystem from each other.
- **Sensitivity analysis.** This step is important to check if the model parameters are relevant to the studied behaviour. For example, the Fourier series expansion can approximate periodic functions, with each new term adding to the accuracy of the approximation. Sensitivity analysis can help to understand when one can stop adding new terms — or which terms (and their parameters) can be neglected.

The sensitivity is the measure of the change in the system output (or in the parameter estimation objective function), produced by a change in parameter value(-s). A normalized sensitivity can be computed as follows [7]:

$$S_P = \left[\frac{\Delta Y}{Y} / \frac{\Delta P}{P} \right] \cdot 100\%, \quad (1.2.1)$$

where S_P is the normalized sensitivity to changes in parameter P and Y is the output. The sensitivity can be computed analytically for models with explicit analytical forms, but if the model is based on numerical methods, the sensitivity must be estimated from simulations. High sensitivity means that the parameter in question must be estimated as accurately as possible. A very low sensitivity shows that the parameter can be removed from the model with little consequence. In case of nonlinear models, the sensitivity is not constant and must be tested under the full range of operating conditions.

- **Model validation.** Ideally, a model based on a small set of data must be able to predict behaviour under a wide range of conditions. If model fails to predict outputs in validation data sets, there are several possibilities for problems. The first possibility is that the scope of the model was chosen incorrectly and some important independent variables were neglected (Step 1). The second possibility is that the model structure is incorrect, likely being a result of overfitting (Step 2). The third possibility is poor parameter estimation due to poorly designed inputs (Step 3).

Let’s consider general models and approaches in the field of neuromusculoskeletal modeling to demonstrate the reductionist approach and keep it closer to the topic of the project.

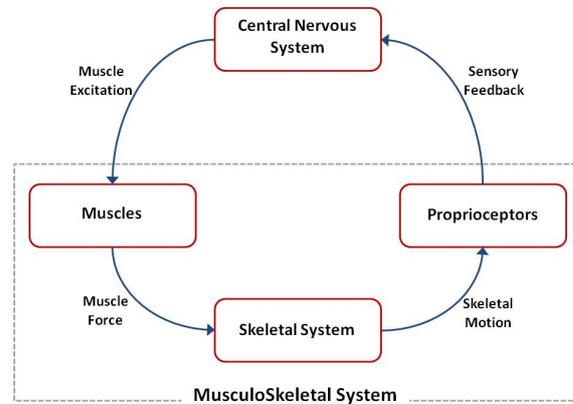


Figure 1.2.1: Neuromusculoskeletal interaction diagram [1]

1.2.1 The skeletal system

The skeletal system refers to the skeleton itself. It is composed of bones (relatively rigid bodies, segments or body segments), that are connected to each other via articulations or joints with help of soft tissues, such as cartilage. The configuration of a skeletal system determines the kinematics of a human body.

Let's see how kinematics of a skeletal system can be modeled, using the human arm as an example.

Skeletal kinematics

A human arm can be modeled in the **Denavit-Hartenberg convention**. The implementation of the arm on the Figure 1.2.2 was derived from the OpenSim "Stanford VA upper limb model" [13] with some changes [14]. Its parameters are presented in Table 1.1. Certain parameters were named in form of len_{body}^{symbol} , which means the value of measurement of *body* in *symbol*-direction. The default values can be acquired from the referenced OpenSim model. For symbols y' , a , d the values are provided in parentheses on the Figure 1.2.2.

Alternatively, the arm kinematics can be represented by a **Generalized Human Arm Triangle** model [14]. The purpose of this model was to improve the computational efficiency of the Stanford model, so it can be used for real-time applications and it was proven to be only slightly less accurate than the original numerical OpenSim IK solver. The approach is shown on Figure 1.2.3. The parameters:

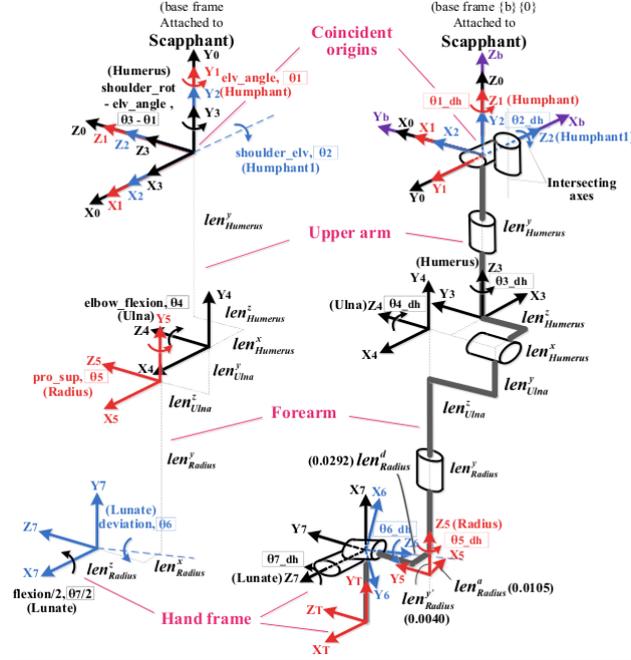


Figure 1.2.2: D-H model of a human arm

$\theta_{i,dh} (^{\circ})$	$d_i (m)$	$a_{i-1} (m)$	$\alpha_{i-1} (^{\circ})$
90.0	0.00	0.00	0.0
θ_1	0.00	0.00	0.0
θ_2	0.00	0.00	90.0
$\theta_3 - \theta_1 - 90.0$	$-len^y_{Humerus}$	0.00	-90.0
$\theta_4 + 180.0$	$len^z_{Ulna} - len^z_{Humerus}$	$-len^x_{Humerus}$	-90.0
$\theta_5 - 124.2$	$-len^y_{Ulna} - len^y_{Radius} - len^{y'}_{Radius}$	0.00	-90.0
$\theta_6 + 101.3$	$-len^d_{Radius}$	len^a_{Radius}	97.8
$\theta_7/2 + 13.7$	0.00	-0.08	-90.0

Table 1.1: D-H Parameters of human right arm kinematic model

- r**: unit direction vector of the upper arm;
- l**: unit normal vector of the generalized human arm triangle plane. The direction of l is determined by the right-hand rule, and the right-hand screw direction is the direction of elbow extension;
- α : angle between the upper arm and the lower arm;
- f**: unit direction vector of the fingers;
- p**: unit normal vector of the plane of the palm. Its direction points outward from the palm.

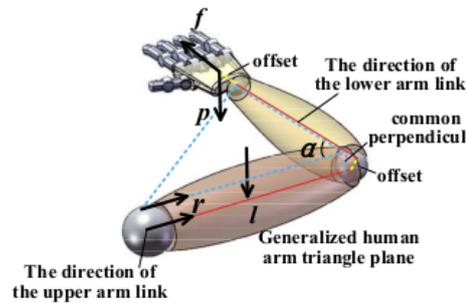


Figure 1.2.3: Triangle model of a human arm

Despite being defined by five variables, the number of independent variables is actually seven due to the normalization constraints of the four unit vectors and the perpendicular relationships between two pairs of vectors, \mathbf{r} vs \mathbf{l} and \mathbf{f} vs \mathbf{p} . Therefore, this representation is complete and the authors of the original work were able to express all seven joint coordinates of the arm configuration.

Skeletal dynamics

Considering the skeletal dynamics, it is possible to assume that skeletal system consists of rigid bodies connected by joints with certain degrees of freedom [1]. Dynamics of such a system can be modeled by methods that were developed for the modeling of robot manipulators:

$$M(\theta)\ddot{\theta} + C(\theta, \dot{\theta}) + G(\theta) = T, \quad (1.2.2)$$

where θ is the movement in joint coordinate system, $M(\theta)$ is the mass matrix, $C(\theta, \dot{\theta})$ is the Coriolis and centripetal terms, G is the gravity term and T is the torque resulting from the external forces and moments.

The skeletal system does not move by itself, but rather plays the role of a supporting structure for the muscles that generate necessary forces and torques to produce movement. That interaction couples the skeletal system with the muscle components which, besides generation of motion, have their own mass and other properties that cannot be neglected when considering the dynamics of such a system.

1.2.2 Muscle components

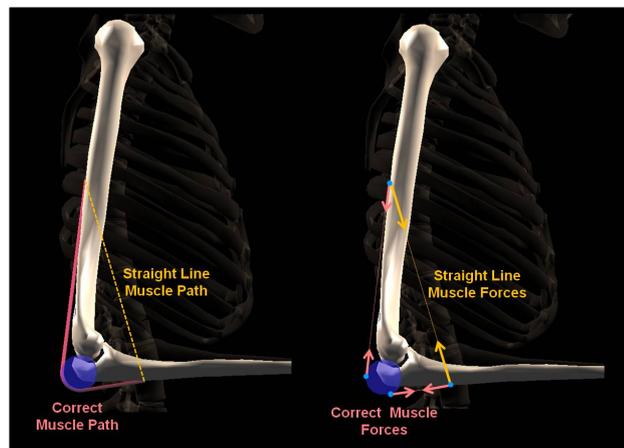


Figure 1.2.4: Muscle paths example [1]

Muscles are the source of motion. Muscles can only contract, so normally they work in pairs to be able to move limbs in opposite directions. The way a muscle is connected to a bone is described by a muscle path. The muscle path determines the direction of the forces when the muscle contracts and the muscle's length, which is related to how much force a muscle can exercise. For example, a Figure 1.2.4 shows the difference in generated forces and muscle lengths between a correct muscle path and a simplification - a modeler should be aware of this, since muscle paths' length is crucial for the computation of musculoskeletal dynamics.

To compute forces generated by a muscle contraction, it is necessary to have a model of such relationship. Several models were created, with Hill's model (and its derivatives) being more popular and with the Huxley model being an alternative.

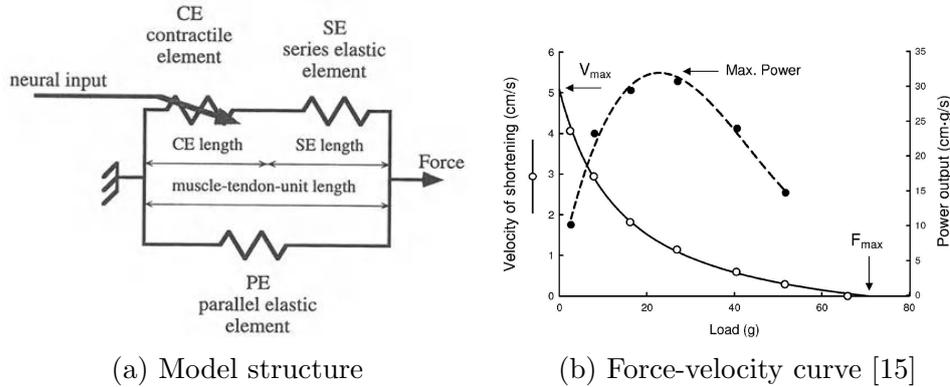


Figure 1.2.5: Muscle model representation

Hill/Hill-based muscle models

In the three-element Hill model, a muscle is represented by a combination of two nonlinear spring elements and one contractile element. The most common depiction of the Hill muscle model is shown on a Figure 1.2.5a, but in some cases the SE element goes in series with a parallel connection of CE and PE elements. In the original Hill's model, the muscle contraction velocity and the load are related by the following equation [16]:

$$(v + b)(F + a) = b(F_0 + a), \quad (1.2.3)$$

where v is the velocity of contraction, F is the load, a is the coefficient of shortening heat, $b = \frac{v_0}{F_0}$, where v_0 is the contraction speed at no load and F_0 is the maximum isometric tension. The resulting relation is hyperbolic and is shown on Figure 1.2.5b.

In Hill-based contractile element models, this relation extended to a following one [7]

$$\frac{F}{F_0} = A(u) \cdot LT(l_m) \cdot FV(\dot{l}_m) \quad (1.2.4)$$

In this equation the normalized force is related to the length-tension property of a muscle, $LT(l_m)$, to the force-velocity property, $FV(\dot{l}_m)$ and to the dynamics of muscle activation by neural inputs, $A(u)$. As seen from the equation, the force is proportional to the activation level of a muscle. Even though the actual neural inputs are difficult to measure, it is possible to use surface electromyography to estimate the level of activation - more details will be given in one of the following subsections.

It is important to note that Hill's model was derived from a 'black box' model and is essentially a curve fit. For that reason this model is suitable

to simulate a muscle mathematically, but does not consider the underlying mechanisms and might be unsuitable for some conditions under which it has not been tested.

Huxley/Zahalak muscle models

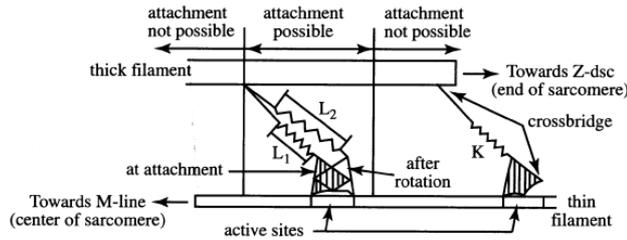


Figure 1.2.6: Schematic of actions of muscle cross-bridge

In contrast to Hill's model, a model for the underlying mechanism behind muscle force generation was proposed by Huxley in 1957 [17]. The proposed muscle model, also known as *Sliding-Filament Biophysical* model, considered the interaction between two muscle proteins, actin and myosin, that are arranged in a regular pattern in the basic unit of muscle contraction, the sarcomere, as seen on Figure 1.2.6 [18]. Sarcomeres are arranged in series to form myofibrils, which are arranged in parallel to form muscle cells or fibers, which in turn are arranged in parallel to form the bulk of the whole muscle. In the model, the internal mechanics of sarcomeres were described, as well as the overall length-tension and force-velocity relationships.

This approach has been advanced by Zahalak, who has developed a structural model describing excitation-contraction coupling and contraction dynamics [19]. By assuming a Gaussian distribution for the actin-myosin bond lengths, the new model allowed for computation of the macroscopic properties of the muscle by numerically solving ordinary differential equations, rather than partial differential equations.

However, the largest component that Zahalak model can describe is a motor unit, with basic model being a fifth order. Because a complete muscle has many motor units, such an approach would still be computationally heavy, although more detailed and accurate [20].

1.2.3 Neural components

Even though neural components are out of scope of this project, it is important to briefly describe the neural mechanisms, since muscle activation

is the result of neural system's work. For example, muscle activation level uses neural signals as inputs, as can be seen in a Hill-based muscle model described previously. Moreover, the neural system provides feedback loops for the neuromusculoskeletal system which cannot be completely ignored, since a feedback loop is a certain obstacle for the identification procedures.

Motor unit

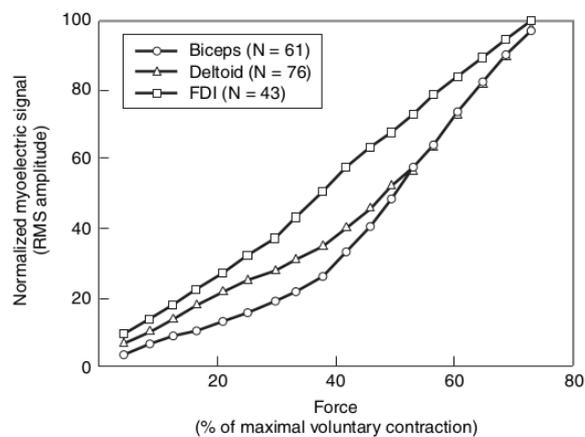


Figure 1.2.7: Effects of muscle on SEMG signal-force relationship [2]

Mentioned previously, a motor unit is a basic level of nervous system organization of the muscle, consists of the lower motor neuron, its axon and the muscle fibers it innervates. Amount of muscles fibers per motor unit varies greatly in the human body, with lower fiber-per-unit ratios being good for fine motor tasks (e.g 3 to 1 ratio in the extraocular muscles) and with higher ratios being good for strength production (e.g 2000 to 1 in a gastrocnemius muscle of the leg) [3].

When a motor unit is about to be activated, the ionic barrier of the muscle tissue breaks down and the signal is sent throughout the entire system in both directions from the motor end plate to the tendinous attachments at both ends. This creates the motor unit action potential (MUAP) and the muscle contracts.

This behaviour provides a basis for the electromyography (EMG). Because it is problematic to measure the generated potentials directly from the muscle fibers, surface electromyography (SEMG) is usually adopted to measure it extracellularly. The potential generated within muscles has amplitude of millivolts and, due to the electrical impedance of fat and skin, reaches the

surface of the skin with an amplitude of microvolts. As the exertional demands increase, the firing rate of motor units increase in the range from 8 to 50 Hz, which results in higher SEMG readings. As shown on Figure 1.2.7, *Force* has roughly linear relationship with the myoelectric signal (*FDI* - first dorsal interosseous muscle. *N* - average number of isometric contractions for each muscle group).

Nervous system

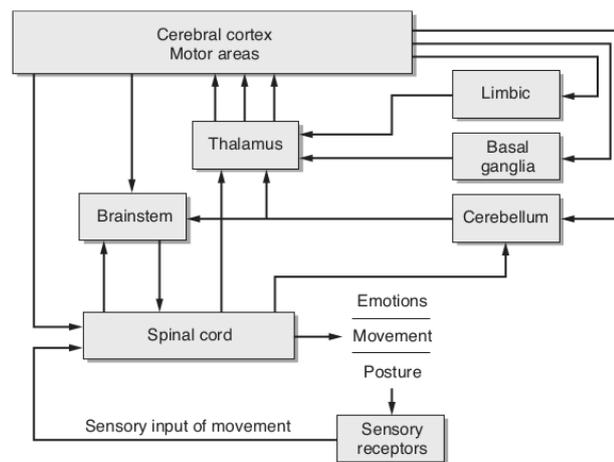


Figure 1.2.8: A highly schematized flow diagram of the hierarchical and parallel aspects of the motor system [3]

As can be seen on Figure 1.2.8, the nervous system is quite complex. For the movement, it means that it is possible to differentiate between voluntary and non-voluntary (reflex) movements (muscle co-contractions). Both can be characterized by the reaction times, which introduce delays to the feedback system: from 150 to 200 *ms* for voluntary and from 30, *ms* for primary reflexes to 100-200, *ms* for secondary or postural reflexes. [21, 22, 23]

1.2.4 Challenges

Because, at some point, structural models require identification, this approach shares some of its challenges with the ‘black box’ approach. Namely, **input design** is still very important and picking the right **model structure** basically defines the reductionist approach.

However, there is another challenge that is more relevant to the reductionist approach - it is the **current amount of knowledge about the modeled**

system. A good example of this is the comparison between Hill and Huxley models. Hill model uses the results from the ‘black box’ modeling and is certainly a simplification of a real system. As the knowledge about muscle structure grew, the Huxley model appeared. Structural modeling uses fundamental knowledge, but as time goes on, fundamentals get reevaluated and extended. This means that paths to structural modeling are discovered as the science advances.

1.3 Conclusions

In this chapter, a brief overview of the digital human modeling field was provided. ‘Black box’ and reductionist approaches were explained, with some of their methods and facts being described in detail and others being only mentioned. This information will help to explain phenomena that is going to be considered or utilized in the following chapters.

Chapter 2

Modeling of human arm dynamics

This chapter will reference and explain a number of studies concerning the modeling of the human arm dynamics. These studies will be briefly described, commented on and then compared, to provide a detailed and diverse picture of the current state of the art.

2.1 Reduced-complexity model

In the paper “*Reduced-complexity representation of the human arm active endpoint stiffness for supervisory control of remote manipulation*” [24], A. Ajoudani presented an approach to modeling of the human arm impedance with focus on the arm endpoint stiffness, optimized for the real-time stiffness prediction.

The model was based on Hill’s musculoskeletal model and the contractile element was considered dominant under the assumption that during motions the musculotendon unit is not fully stretched. In this case the relation between the joint stiffness (K_J) and the active muscle stiffness (\hat{K}_m) would be:

$$K_J(p, q) = J_m^T(q) \hat{K}_m(p) J_m(q), \quad (2.1.1)$$

where p is the muscle activity, q is the joint angle vector, T and J_m are the transpose operator and the muscle Jacobian.

The muscle Jacobian is a transformation relating musculotendon length changes

to the joint angle variations:

$$J_m = \begin{bmatrix} \frac{\partial I_1(q)}{\partial q_1} & \frac{\partial I_1(q)}{\partial q_2} & \dots & \frac{\partial I_1(q)}{\partial q_n} \\ \frac{\partial I_2(q)}{\partial q_1} & \frac{\partial I_2(q)}{\partial q_2} & \dots & \frac{\partial I_2(q)}{\partial q_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial I_m(q)}{\partial q_1} & \frac{\partial I_m(q)}{\partial q_2} & \dots & \frac{\partial I_m(q)}{\partial q_n} \end{bmatrix} \quad (2.1.2)$$

This relation was used under assumption that there were no torques at the arm joints. Since eight dominant muscles were considered, $\hat{K}_m \in \mathbb{R}^{8 \times 8}$. To map muscle stiffness to the arm model with seven joint coordinates, the muscle Jacobian would be $J_m \in \mathbb{R}^{8 \times 7}$. To compute the Cartesian stiffness, the following relation was used:

$$K_c(p, q) = J^{+T}(q)[K_J(p, q) - G_J(q)]J^+(q), \quad (2.1.3)$$

where $J^+ = K_J^{-1}J^T(JK_J^{-1}J^T)^{-1}$ is the pseudo-inverse of the arm Jacobian and $G_J(q)$ is the term accounting for external and gravity forces. During experiments the external forces were zero and the gravity was neglected, thus making the term irrelevant for the study. To track the joint coordinates, the Generalized Human Arm Triangle [14] model was chosen, that is shown on Figure 1.2.3 and was described in the previous chapter. Using the retrieved joint coordinate values, muscle lengths were computed with parameters taken from the ‘‘Stanford VA Upper Limb Model’’ [13].

The estimated muscle stiffness was modeled as $\hat{K}_m = a_{cc}(p)K_s$, with K_s being a diagonal [25] matrix of muscle contributions to the stiffness and $a_{cc}(p)$ being a scalar, time-varying function of the muscular activities, which were measured using the SEMG method. The only muscles that were measured were the biceps brachii (P_B) and the triceps brachii (P_T), thus:

$$a_{cc}(p) = c_1 + c_2(P_B + P_T) \quad (2.1.4)$$

The identification of the proposed model was done in multiple steps:

1. The forces and the positions of the arm endpoint were recorded and translated into the frequency domain.
2. The relation between inputs and outputs was modeled as a linear non-parametric black box, a MIMO transfer function:

$$\begin{bmatrix} F_x(f) \\ F_y(f) \\ F_z(f) \end{bmatrix} = \begin{bmatrix} G_{xx}(f) & G_{xy}(f) & G_{xz}(f) \\ G_{yx}(f) & G_{yy}(f) & G_{yz}(f) \\ G_{zx}(f) & G_{zy}(f) & G_{zz}(f) \end{bmatrix} \begin{bmatrix} x(f) \\ y(f) \\ z(f) \end{bmatrix} \quad (2.1.5)$$

$$G_{ij}(s) = I_{c_{ij}}s^2 + B_{c_{ij}}s + K_{c_{ij}},$$

where $G_{ij}(s)$ is a SISO transfer function and I_c , B_c and K_c denote the endpoint inertia, viscosity and stiffness matrices, respectively. Because the study focused on stiffness, inertia and viscosity parameters were unused.

3. The proposed model was fit to the obtained model by minimizing the Frobenius norm:

$$\|J_m^T(q)a_{cc}(p)K_sJ_m(q) - J^T(q)K_c(p, q)J(q)\|, \quad (2.1.6)$$

with $J \in \mathbb{R}^{3 \times 7}$ and $K_c \in \mathbb{R}^{3 \times 3}$, since only the translational movements were considered. Once fit, the model was ready to be used for real-time stiffness prediction. The model fit was reported to have an average error of approximately 12%.

The teleimpedance control experiments showed a successful application of described model to the stiffness modulation for the drilling operation. The study suggested that, considering the stiffness ellipsoid, the muscle activation was mostly contributing to its volume and the arm configuration was defining its orientation.

It's important to note that this study was soon extended by authors in the paper "*Online Model Based Estimation of Complete Joint Stiffness of Human Arm*" [26]. While the model used for the estimation was the same as in the Equation 2.1.1, with K_m being defined in the same way, the estimation of the MIMO system was replaced by the estimation of the human arm joint stiffness K_J through the means of the nullspace compliance. After the K_J was estimated with a new method, the authors proceeded to estimating the unknowns of the same K_m expression through the Frobenius norm. The algorithm remained to be a two-step algorithm, but the authors adapted the second step to be available for the online use.

To summarize, the core proposal of this study is to model human arm's stiffness as a combination of the constant muscle contribution matrix K_s , related only to the muscle fibers properties, and the varying function a_{cc} , related only to the muscle activation levels. With that, influence of posture on the stiffness is accounted for with muscle Jacobian J_m , computed by a private and elaborate OpenSim model of the human arm.

While the model proved to be efficient enough for the real-time control applications, the chosen equipment was quite sophisticated, as using both EMG measuring and motion capture devices might not be convenient for some practical applications.

2.2 Nonlinear, simplified model

A different approach to model the human arm impedance was used in the paper “*Ensuring safety in hands-on control through stability analysis of the human-robot interaction*” [27]. The following assumptions were used:

- The hand performs only translational planar motions (for simplicity of the analysis)
- The human arm mechanical impedance in the motion plane can be decomposed into two contributions, along the x and y directions
- The human arm impedance is isotropic

The considered model was:

$$F_h(q_h, t) = M_h(q_h, t)\ddot{q}_h + D_h(q_h, t)\dot{q}_p + K_h(q_h, t)\Delta q_h, \quad (2.2.1)$$

where F_h was the force exerted by the human arm at the hand frame in the x and y directions, M_h , D_h and K_h – the time-varying and configuration dependent equivalent mass, damping and stiffness matrices related to the x and y directions respectively, and $q_h = [x \ y]^T$ in the human shoulder frame. Then the inertial component was neglected due to low accelerations of the task, the damping matrix would be set:

$$D_h(q_h, t) = \begin{bmatrix} \delta_{xx}\sqrt{K_{h_{xx}}(q_h, t)} & 0 \\ 0 & \delta_{yy}\sqrt{K_{h_{yy}}(q_h, t)} \end{bmatrix} \quad (2.2.2)$$

The stiffness was proposed to be of the following form:

$$\begin{aligned} K_h(q_h, t) &= K_{post}(q_h, t) + K_{cont}(t) \\ K_{cont}(t) &= \bar{K}_{cont}\sigma(t - \tau) \\ K_{post}(q_h, t) &= b_1 \begin{bmatrix} x^2(t - \tau) \\ y^2(t - \tau) \end{bmatrix} + b_2 q_h(t - \tau) + b_3, \end{aligned} \quad (2.2.3)$$

where K_{post} is the postural stiffness, K_{cont} is the stiffness component related to the co-contraction of muscles with \bar{K}_{cont} being the maximum co-contraction stiffness value, $\sigma : \mathbb{R} \rightarrow [0, 1]$ is a nominalized muscular co-contraction index, and τ is a delay accounting for involuntary reaction time. In the study, two sets of stiffness parameters were estimated, one set from a motion along each axis; then the produced models are appropriately combined to account for a free planar motion. The further developments [28]

proposed a more complex model for the stiffness components:

$$\begin{aligned}
 K_{cont}(t) &= \begin{bmatrix} k_{\sigma_{xx}} & 0 \\ 0 & k_{\sigma_{yy}} \end{bmatrix} \sigma(t) \\
 K_{post}(q_h, t) &= \begin{bmatrix} K_{h_{xx}}(q_h, t) & 0 \\ 0 & K_{h_{yy}}(q_h, t) \end{bmatrix} \\
 K_{h_{**}}(q_h, t) &= k_{*0} + [k_{*1} \quad k_{*2}] q_h + q_h^T \begin{bmatrix} k_{*3} & k_{*4} \\ k_{*4} & k_{*5} \end{bmatrix} q_h
 \end{aligned} \tag{2.2.4}$$

where * can stand for x or y .

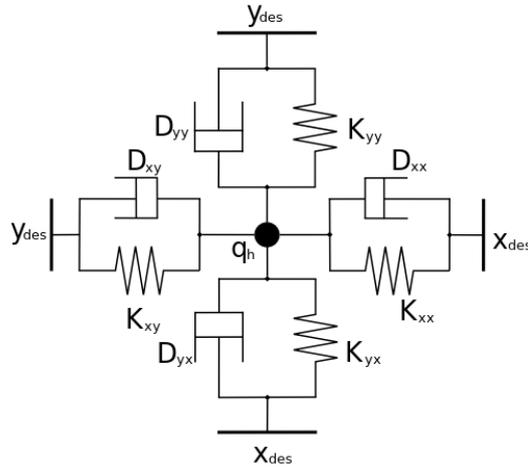


Figure 2.2.1: Postural component of the model

To summarize, the study proposed a way to model the overall impedance of the human arm directly in Cartesian space as a second order (counting inertial component) dynamic system. The postural component of the human arm impedance can be represented as a set of virtual spring-and-damper systems, as shown in the Figure 2.2.1. In the figure, values of stiffness and damping change as q_h changes and $q_{des} = [x_{des}; y_{des}]$ represents the desired position of the endpoint of the arm, which is chosen by human's neural control system. Because q_{des} is not measurable, to perturb the system it is necessary to exploit human reaction times and make assumptions about the behavior of q_{des} during a perturbation be able to compute Δq_h . Additionally, tracking of muscle activation can be employed. In the study, this was exactly the case, as parameters of the co-contraction stiffness component were identified beforehand and then used to adjust the recorded force values. Then, the rest of the parameters were identified via the least squares

method.

The described approach is relatively simple, because it works directly in Cartesian coordinates and does not require to track the configuration of the arm except for it's endpoint. However, because it essentially provides a position-to-stiffness map, the traversed area used for model parameters estimation can be described as quite modest.

2.2.1 Remark on endpoint rotational stiffness

From the Figure 2.2.1, one may wonder if it is possible to attach a virtual rotational spring-and-damper system to the human arm endpoint in order to account for the rotational movement and moments. The answer depends on how the endpoint rotation is defined. As the arm endpoint is chosen roughly at the position of the hand, it can be considered sensible to define the endpoint rotation as the orientation of the hand. With the proper equipment, it is possible to record the moments that were provides by a perturbed hand and thus establish a relationship between the time histories of hand rotation and reaction moments, putting more information into the model and, hopefully, to reduce errors due to the neglected interactions.

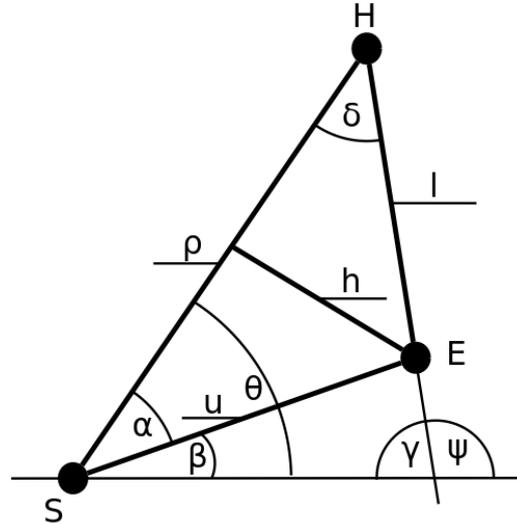


Figure 2.2.2: Human arm triangle

In the study, a special armband was used to lock the rotation of the wrist. In that regard, a human arm can be represented as a 2-link manipulator, as shown in the Figure 2.2.2. In the figure, points **S**, **E** and **H** represent the position of the Shoulder, Elbow and Hand respectively. As the wrist is

locked, it's orientation would coincide with the orientation of the forearm, which would be the angle ψ . Knowing the position of the hand and the elbow, it is trivial to find the angle.

$$\psi = \text{atan2}(y_H - y_E, x_H - x_E), \quad (2.2.5)$$

where $\text{atan2}(y, x)$ represents a function that properly handles the computation of the angle between the direction to a point with coordinates x and y and the x axis, resulting in angle $\psi \in [-\pi, \pi]$.

It is still possible, however, if only position of the hand in the shoulder frame is known. From just x_H and y_H , ρ and θ are known (for details, look in Subsection 2.2.2). The lower arm and the upper arm lengths l and u vary from human to human, but can be measured.

$$\gamma = \pi - \delta - \theta \quad (2.2.6)$$

$$\delta = \arccos\left(\frac{\rho^2 + l^2 - u^2}{2\rho \cdot l}\right) \quad (2.2.7)$$

$$\begin{aligned} \psi &= \pi - \gamma \\ &= \delta + \theta \\ &= \arccos\left(\frac{\rho^2 + l^2 - u^2}{2\rho \cdot l}\right) + \theta \end{aligned} \quad (2.2.8)$$

The expression for ψ can get simpler if additional knowledge is used - such as knowledge on human proportions. In the simplified case, if lengths of lower and upper arm are considered equal, the expression for ψ becomes:

$$\psi = \theta + \arccos\left(\frac{\rho}{2l}\right) \quad (2.2.9)$$

Because human shoulder, elbow and hand provide three points to construct a plane, these considerations would be true in 3D as well. The similar approach used to compute arm kinematics was shown in Figure 1.2.3, when the *Generalized Human Arm Triangle* was mentioned.

After the orientation of the endpoint is computed, it is possible to introduce a model similar to Equation 2.2.1, but for the appropriate moment:

$$M_h(\psi, t) = D_{rot_h}(\psi, t)\dot{\psi} + K_{rot_h}(\psi, t)\Delta\psi \quad (2.2.10)$$

2.2.2 Remark on polar coordinates

The described model is defined in the Cartesian space. As seen in Subsection 2.2.1, knowing the distance to the endpoint and the direction has proven useful. Moreover, the relations between the Cartesian position in the shoulder

frame and the forces may seem unnatural. In case of the polar coordinates formulated in the shoulder frame, a radius ρ , a distance to the arm endpoint, would relate to a radial force, which can describe such natural working operations as pushing or pulling. An angular coordinate, θ , would relate to a moment that an arm can enforce to stop the endpoint from moving sideways. It can also be useful to see if a linear model of the similar structure that was described in Section 2.2 would be simpler or more accurate.

Let's consider a following model:

$$F_{pol}(q_{pol}, t) = D_{pol}(q_{pol}, t)\dot{q}_{pol} + K_{pol}(q_{pol}, t)\Delta q_{pol} \quad (2.2.11)$$

$$q_{pol} = [\rho \quad \theta]^T, F_{pol} = [F_\rho \quad M]^T$$

To transform the data from Cartesian coordinates to polar, the following equations can be used:

$$\rho = \sqrt{x_h^2 + y_h^2} \quad (2.2.12)$$

$$\theta = \text{atan2}(y_h, x_h), \quad (2.2.13)$$

where $\theta \in [-\pi, \pi]$. To compute the perturbations in polar coordinates, a motion without perturbations can be computed in Cartesian coordinates, then transformed to the polar coordinates as shown above and then subtracted from the q_{pol} motion.

To compute proper outputs, first an amplitude and an angle of the force vector ϕ can be obtained. Then the radial and tangential components of force are computed for the given angle θ . Finally, the moment is computed for the given ρ :

$$|F_h| = \sqrt{F_{h_x}^2 + F_{h_y}^2}$$

$$\phi = \text{atan2}(F_{h_y}, F_{h_x})$$

$$\psi = \phi - \theta$$

$$F_\rho = |F_h| \cdot \cos(\psi) \quad (2.2.14)$$

$$M = |F_h| \cdot \rho \cdot \sin(\psi) \quad (2.2.15)$$

Whether such representation will prove to be practical or not, there is an important point to be made: while the raw recorded data is always processed, sometimes it is possible to extract even more information and use it to either complicate the model or to gain more insight.

2.3 EMG-driven 'black box' approach

In the thesis "*EMG-driven Human Modelling to Enhance Human-Robot Interaction* [29], several system identification techniques were tested and compared with each other. Namely, the utilized techniques were: Linear Time-Invariant (LTI) modeling, Linear Parameter Varying (LPV) modeling in both Local Identification Framework (LIF) and Global Identification Framework (GIF), and nonlinear Hammerstein modeling.

The study considered the EMG-Force relation. The inputs were the processed surface EMG signals, measured from the muscles of the right arm: the biceps brachii, triceps brachii, flexor carpi radialis and brachioradialis. The output was the absolute value of the force, that was recorded by a 6-DoF force sensor, mounted on the KUKA robot manipulator. Only the rotation of the elbow joint was considered.

The **Linear Time-Invariant** model was defined as following:

$$\begin{aligned}x(k+1) &= Ax(k) + Bu(k) \\y(k) &= Cx(k) + Du(k)\end{aligned}\tag{2.3.1}$$

A family of LTI models was derived for a set of elbow joint angles $q(k)$.

The **Linear Parameter Varying** model was defined like this:

$$\begin{aligned}x(k+1) &= A(q)x(k) + B(q)u(k) \\y(k) &= C(q)x(k),\end{aligned}\tag{2.3.2}$$

where matrices A, B and C were defined as affine functions:

$$\begin{aligned}A(q(k)) &= A_0 + A_1q(k) \\B(q(k)) &= B_0 + B_1q(k) \\C(q(k)) &= C_0 + C_1q(k)\end{aligned}\tag{2.3.3}$$

LIF-LPV models were derived from sets of LTI models. On the contrary, GIF-LPV models were obtained using the MATLAB's Prediction-Based Subspace Identification (PBSID) toolbox [30] and the LPVcore toolbox [31]. Finally, the Hammerstein model was identified via MATLAB's System Identification Toolbox function *nlhw*. [32]

In conclusion, the author suggested using LTI modeling or GIF-LPV modeling to accurately estimate the EMG-Force relations, with Hammerstein model being the most accurate (but not control-oriented) and LIF-LPV being the least accurate.

Overall, the study provided a very detailed information on different estimation methods of the EMG-Force relations and its validation. However, the study was not concerned with how the force vector is aligned in space, which can be crucial for the practical applications.

2.4 EMG-driven genetic model

The study “*sEMG-based Endpoint Stiffness Estimation of Human Arm using Gene Expression Programming*” [33] the authors used an evolutionary algorithm to model the relationship between surface EMG measurements and stiffness of the human arm.

In order to do that, first the impedance model of the human arm is defined as follows:

$$M_e \ddot{X} + B_e \dot{X} + K_e(X - Xu) = F, \quad (2.4.1)$$

where $M_e, B_e, K_e \in R^{3 \times 3}$ represented the inertia, damping and stiffness matrices; X was the arm endpoint position and F - the force vector. Similarly to the approach described in Section 2.2, the unknown parameters were estimated using the least squares algorithm. The parameters were estimated for the set of 3 different co-contraction value ranges: a relaxed arm, 30-50% and 70-90% of processed sEMG reading values.

Then, after the parameters are estimated, a genetic model was trained to predict the diagonal stiffness matrix coefficients.

The authors concluded that some groups of muscles had better correlation with the stiffness values and claimed that their non-linear algorithm outperformed linear regression models by having higher correlation and lower RMSE values. This study demonstrated the use of one of the many types of machine learning algorithms for estimation purposes. Considering that machine learning algorithms tend to demand large amounts of data, the study could have used other factors - such as position of the endpoint, for example, - to be able to draw more conclusions and comparisons.

2.5 Statistical EMG-driven modeling

In the study “*sEMG-based Estimation of Human Arm Force using Regression Model*” [34], a model that establishes a relation between surface EMG measurements and force at the endpoint of the human arm. In the paper, both Bayesian Linear Regression (BLR) and Support Vector Regression (SVR) algorithms were used.

A **Bayesian Linear Regression** model was defined like this:

$$f^t = \omega^T a, \quad (2.5.1)$$

where $f_i^t, i = 1, 2, 3$ was the theoretical force in x,y,z direction; $a \in R_{m \times 1}$ denoted the amplitude of the sEMG signals with m being the channel number of the electrodes and $\omega \in R_{m \times 1}$ denoted the unknown coefficients. It was

assumed that the noise in sEMG-force mapping of arbitrary direction was all in Gaussian distribution. The estimated force by BLR would be:

$$f_i = \omega^T a + \epsilon_i = y(a, \omega) + \epsilon_i, i = 1, 2, 3; \quad (2.5.2)$$

where $\epsilon_i \sim N(0, \beta^{-1})$, the noise with zero mean value and a variance of β^{-1} . The final result, the logarithmic probability density function of posterior distribution was:

$$\ln(\omega|F_i) = -\frac{\beta}{2} \sum_{k=1}^n \{y(a, \omega) - f_{ik}\}^2 + \frac{\alpha}{2} \omega^T \omega + const, \quad (2.5.3)$$

where F_i denoted the estimated force vector in a certain direction, n denoted the total number of samples applied for the computation, α and $const$ were constants. The maximization of the LPDF value would provide the set of estimated parameters.

In case of the **Support Vector Regression Algorithm**, by means of which a nonlinear mapping from input to output is found, the decision function was:

$$f_i(a) = \sum_{k=1}^n (\alpha_k^* - \alpha_k) K(a_k, a) + b, i = 1, 2, 3; \quad (2.5.4)$$

where K denoted the RBF kernel function and α_k^*, α_k, b denoted the unknown parameters. The problem would be solved by the means of dual programming for standard quadratic programming.

The authors provided experimental results that showed the high performance for both of the models. They recommended using the SVR model for the force estimation and the BLR model for the stiffness estimation based on sEMG.

2.6 A structural EMG-driven model

In the study “*Estimation of Time-varying Human Arm Stiffness Using Electromyogram Signal*” [4] it was proposed to use a complete musculoskeletal model of the human arm (Figure 2.6.1) and the surface EMG measurements to estimate the time-varying stiffness.

The EMG signals were rectified and filtered using a recursive filter as follows:

$$e_i(t) = \alpha_i EMG_i(t - d) - \beta_i e_i(t - 1) - \gamma_i e_i(t - 2), \quad (2.6.1)$$

where d_i was the electromechanical delay and $EMG_i(t)$ was the rectified EMG signal of the muscle i , with rest being the constants. Then, a nonlinear

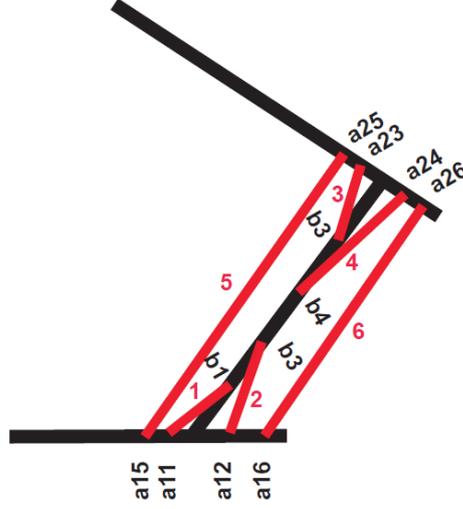


Figure 2.6.1: The musculoskeletal model [4]

function was used to map the processed EMG signal to activation level of the muscle:

$$a_i(e_i) = \frac{e_i^{A_i \cdot e_i} - 1}{e_i^{A_i} - 1}, \quad (2.6.2)$$

$$-3 \leq A < 0$$

where A_i indicated the nonlinearity index of the relation. The Hill-type model of the muscle was used to estimate muscle force around the joints:

$$F_m = a \cdot F_l(l) \cdot F_v(l) \cdot F_{max} + F_p, \quad (2.6.3)$$

where F_l was the nonlinear length-tension relation, F_v was the force-velocity relation, F_{max} was the maximum muscle force, F_p was the nonlinear passive force and l was the muscle length. The stiffness of the hand was computed as a slope of the length-tension relation as following:

$$K_e = -(J^T)^{-1} \left(\frac{\delta M}{\delta \theta} F_m + M K_m M^T + \frac{\delta J^T}{\delta \theta} F \right) J \quad (2.6.4)$$

$$M = \begin{bmatrix} \frac{\delta l_1}{\delta \theta_1} & \frac{\delta l_2}{\delta \theta_1} & 0 & 0 & \frac{\delta l_5}{\delta \theta_1} & \frac{\delta l_6}{\delta \theta_1} \\ 0 & 0 & \frac{\delta l_3}{\delta \theta_2} & \frac{\delta l_4}{\delta \theta_2} & \frac{\delta l_5}{\delta \theta_2} & \frac{\delta l_6}{\delta \theta_2} \end{bmatrix} \quad (2.6.5)$$

$$K_m = \frac{\delta F_m}{\delta l} = a \cdot F_v \cdot K_{ac}(l) + K_p(l), \quad (2.6.6)$$

where θ was a vector of joint angles (shoulder and elbow), K_e was the arm stiffness, J was the arm Jacobian, M was the muscle Jacobian, F_m was

a vector of muscle forces, K_m was the muscle stiffness and F was the applied perturbation.

The study concluded, pointing out that the proposed method was useful for tracking changes of the arm stiffness, but not the precise value of the stiffness.

2.7 Conclusions

In this chapter, various approaches to modeling of the dynamics of the human arm were described. As intuitively expected, the studies that considered more factors for the estimation process - e.g muscle co-contraction and the position of the endpoint, - seemed more promising, like the ones described in Sections 2.1 and 2.2.

On the other hand, the EMG-driven studies demonstrated how important the muscle activation is and to which extent a human can control the force exerted by their limbs. While these studies did not account for the spatial coordinates of the arm endpoint or the configuration of the arm, they provided a more complex way to account for the muscle activation, which was not found in studies, described in Sections 2.1 and 2.2. Hopefully, the complexity of the both worlds can meet in the future research in this field.

In many papers it was stated that the future development of their methods would provide way to estimate the human-arm parameters via online estimation algorithms, because the variability of the parameters from human to human makes it extremely relevant.

In the next chapter, a number of experiments will be simulated in order to test some suggestions that could have been implemented in the study, described in Section 2.2. The reason why only that study was expanded in such detail is that it is not possible to emulate the EMG readings without the proper equipment. Accidentally, it is possible to avoid modeling the source of EMG signals for the chosen study, because in the study it is done completely separately from the rest of the estimation procedures.

Chapter 3

Simulations

As stated previously, the intention of this chapter is to implement the some suggestions made in the previous chapter in simulation. Because only the study described in Section 2.2 was truly extended, this chapter will implement an online estimation algorithm, based on the chosen study, as well as explore the possibilities within the polar coordinates modeling. The algorithms will be tested with the use of *MATLAB* and *MATLAB Simulink* software.

The equations 2.2.1, 2.2.2, 2.2.4 will be used to model the human arm impedance under the same assumptions, as described in the section. The inertial component will be neglected due to the low planned accelerations. Stiffness parameters are reported in Table 3.1 and damping parameters are reported in Table 3.2.

k_{x_0}	$5.381 \cdot 10^3$	k_{y_0}	$4.497 \cdot 10^3$
k_{x_1}	$1.186 \cdot 10^4$	k_{y_1}	$1.357 \cdot 10^4$
k_{x_2}	$-1.060 \cdot 10^4$	k_{y_2}	$-1.011 \cdot 10^4$
k_{x_3}	$5.195 \cdot 10^3$	k_{y_3}	$1.226 \cdot 10^4$
k_{x_4}	$-2.164 \cdot 10^4$	k_{y_4}	$-2.380 \cdot 10^4$
k_{x_5}	$-5.539 \cdot 10^3$	k_{y_5}	$6.562 \cdot 10^3$
$k_{\sigma_{xx}}$	10.2912	$k_{\sigma_{yy}}$	19.809

Table 3.1: Arm stiffness coefficients

The ultimate goal of this section is to see if it is possible to identify the stiffness directly as a quadratic function of position in the shoulder frame of the human. Note, that in practice the coordinates of the robot endpoint would be recorded in the robot base frame, but, as shown in Appendix A,

δ_{xx}	1	δ_{yy}	1
---------------	---	---------------	---

Table 3.2: Arm damping coefficients

the change of origin (e.g to the human shoulder frame) does not change the quadratic structure of the stiffness model.

3.1 Experiment 1: simplified linear identification

3.1.1 Motion

For this task, there are two goals to achieve: one is to perturb the system well and another is to cover a suitable distance along x and y axes to map a stiffness-position relation properly.

Motion data is represented by a sequence of linear motions along x and y axes, as presented in Table 3.3 and Figure 3.1.1. Periodically, the motion stops and a perturbation is applied in turns along x or y directions. Trapezoidal and triangular perturbations are considered, as shown in Figure 3.1.2. The trapezoidal perturbation is characterized by a constant velocity of 0.14 m/s for 0.05 s, zero velocity for 0.2 s, and a negative velocity of 0.07 m/s for 0.1 s at the end. The triangular perturbation has an amplitude of 0.008 m and a duration of 0.4 s.

x, m	0.95	0.95	0.95	0.95	1.10	0.80	0.95
y, m	0.10	0.20	0.00	0.10	0.10	0.10	0.10
$Time, s$	0.00	12.50	37.50	50.00	62.50	87.50	100.00

Table 3.3: Linear motion parameters

3.1.2 Identification

A following notation will be considered:

$$q_h(t) = q_r(t) + q_p(t) \quad (3.1.1)$$

where q_h is the position of the wrist (as well as the position of the manipulator endpoint), q_p - a perturbation component of the motion and q_r is the motion,

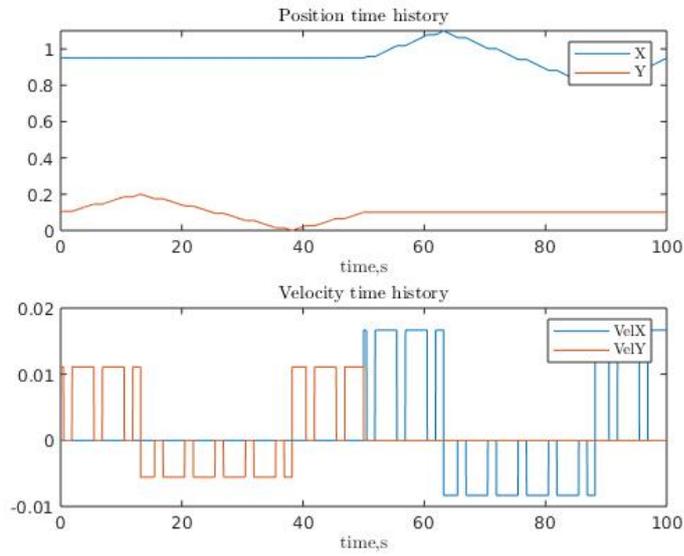
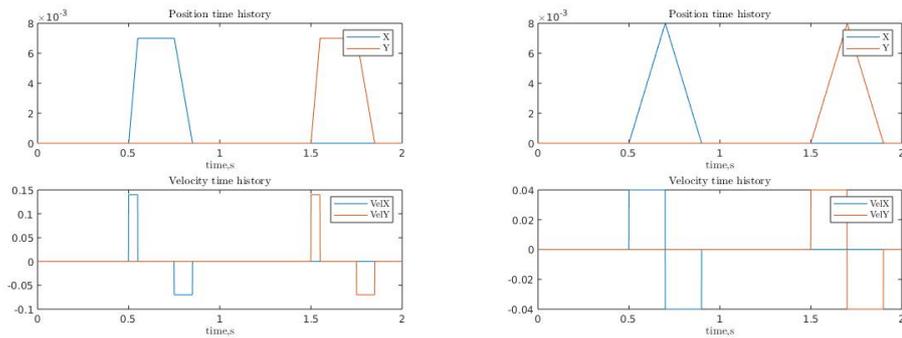


Figure 3.1.1: Linear motion position and velocity



(a) trapezoidal perturbation

(b) triangular perturbation

Figure 3.1.2: Perturbations

generated with the purpose of traversing the plane.

Every time a perturbation starts along x or y direction, the corresponding x_p , \dot{x}_h and F_{h_x} or y_p , \dot{y}_h and F_{h_y} vectors are stored, with x_p and y_p being the shift in position due to a perturbation. To perform a least squares identification, a regressor is computed:

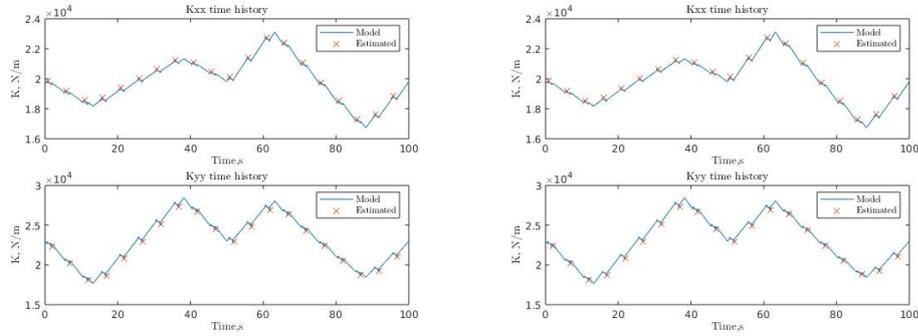
$$\phi_x = \begin{bmatrix} x_p \\ \dot{x}_h \end{bmatrix}, \phi_y = \begin{bmatrix} y_p \\ \dot{y}_h \end{bmatrix}$$

So that the vectors of parameters will be:

$$[\hat{K}_{h_{xx}} \quad \hat{D}_{h_{xx}}] = \left(\frac{1}{N} \sum \phi_x \cdot \phi_x^T \right)^{-1} \left(\frac{1}{N} \sum \phi_x \cdot F_{h_x} \right),$$

$$[\hat{K}_{h_{yy}} \quad \hat{D}_{h_{yy}}] = \left(\frac{1}{N} \sum \phi_y \cdot \phi_y^T \right)^{-1} \left(\frac{1}{N} \sum \phi_y \cdot F_{h_y} \right),$$

When next perturbation starts, new data vectors are collected and process is repeated. The results of this estimation are shown in Figures 3.1.3, 3.1.4, 3.1.5. The Figure 3.1.5 shows the relative error, computed as follows:



(a) trapezoidal perturbation

(b) triangular perturbation

Figure 3.1.3: Stiffness estimation

$$Error\% = 100 \cdot \left| \frac{K_{h_{**}} - K_{h_{**est}}}{K_{h_{**}}} \right| \quad (3.1.2)$$

Considering that results for triangular perturbations and trapezoidal perturbations are almost identical, triangular perturbations will be preferred from now on.

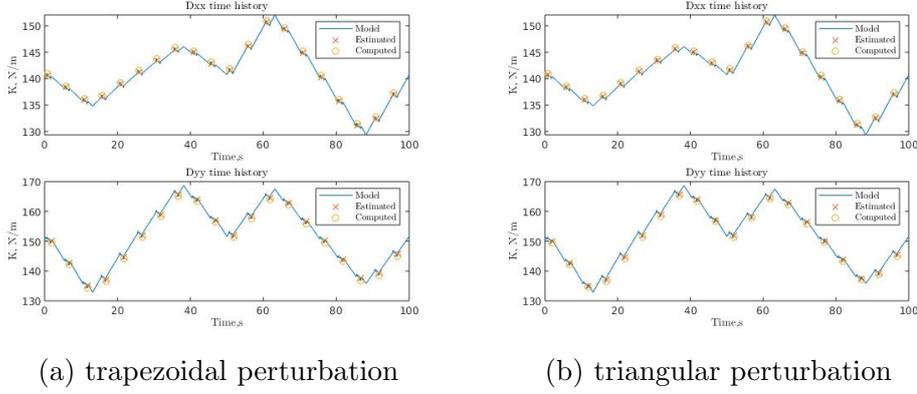


Figure 3.1.4: Damping estimation

3.2 Experiment 2: complete linear identification

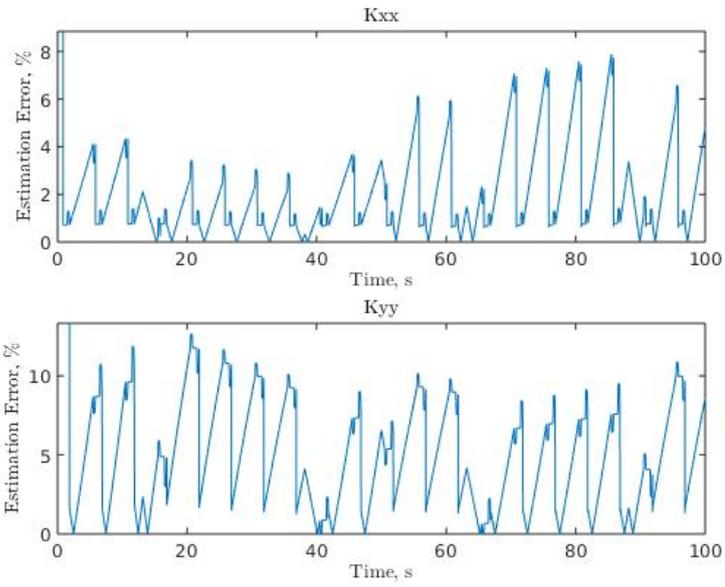
3.2.1 Motion

Here, a circular trajectory is used, centered at $O\{0.95, 0.1\}$ and with radius $R = 0.1\text{ m}$, similar to one used in Section 2.2.

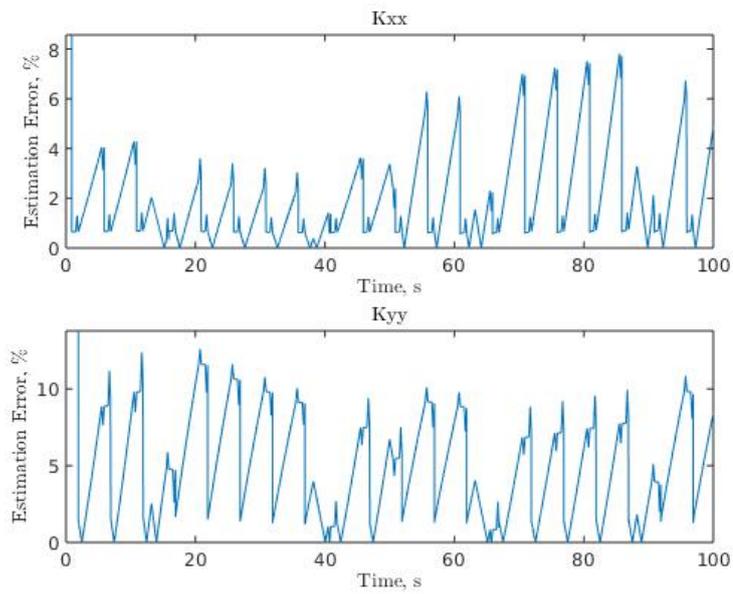
To perturb the human arm model, a sequence of three perturbations was used. Each perturbation has a triangular shape, an amplitude of 0.008 m and a duration of 0.4 s . Every time a sequence of perturbations starts, the q_r motion stops and 3 perturbations with triangular shape are injected. First perturbation is along x axis, second perturbation is along y axis and is negative. The third perturbation is a combination of the first two. Perturbations are separated by a 0.5 s pause. After the third perturbation the circular motion continues. There are 20 perturbation sequences in total, evenly distributed across the motion duration of 100 seconds.

3.2.2 Identification

When a perturbation starts along any axis, data is recorded every 0.001 s and put in a vector, until the perturbation ends. The recorded data is represented by a shift in position due to a perturbation, x_p and y_p ; human arm velocity, \dot{x}_h and \dot{y}_h ; human arm model forces, F_{hx} and F_{hy} . After the third perturbation, the data is used to compute stiffness and damping coefficients



(a) trapezoidal perturbation



(b) triangular perturbation

Figure 3.1.5: Percentage estimation error

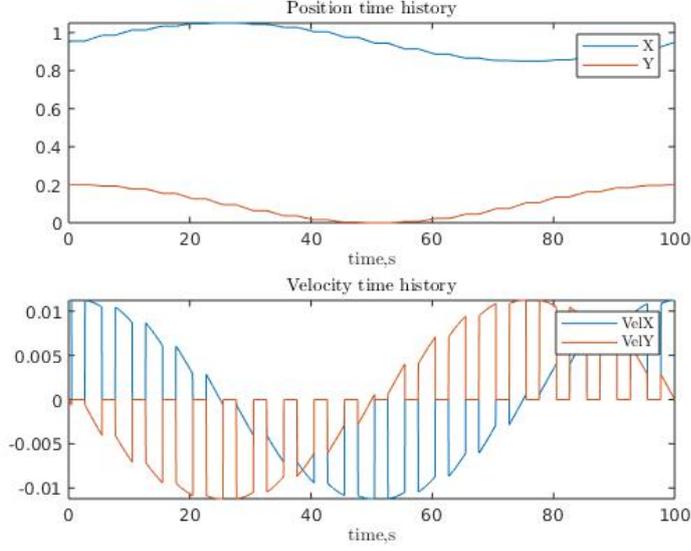


Figure 3.2.1: Circular motion position and velocity

via least squares algorithm. The following regressor vector is computed:

$$\phi_x = \begin{bmatrix} x_p \\ y_p \\ \dot{x}_h \end{bmatrix}, \phi_y = \begin{bmatrix} x_p \\ y_p \\ \dot{y}_h \end{bmatrix}$$

And so, parameter vectors are computed as following:

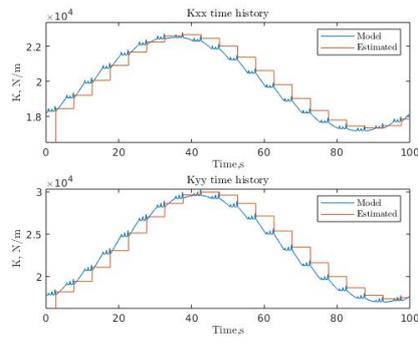
$$[\hat{K}_{h_{xx}} \quad \hat{K}_{h_{xy}} \quad \hat{D}_{h_{xx}}] = \left(\frac{1}{N} \sum \phi_x \cdot \phi_x^T \right)^{-1} \left(\frac{1}{N} \sum \phi_x \cdot F_{h_x} \right),$$

$$[\hat{K}_{h_{yx}} \quad \hat{K}_{h_{yy}} \quad \hat{D}_{h_{yy}}] = \left(\frac{1}{N} \sum \phi_y \cdot \phi_y^T \right)^{-1} \left(\frac{1}{N} \sum \phi_y \cdot F_{h_y} \right),$$

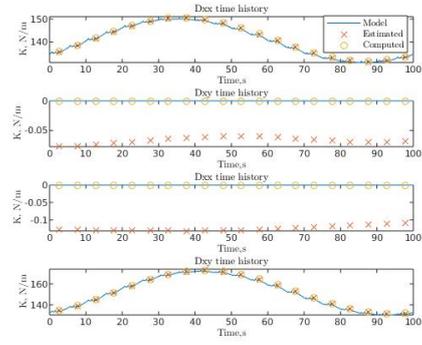
When the next perturbation starts, new data vectors are collected and process is repeated.

3.2.3 Results

This method yields the results as shown in Figures 3.2.2a, 3.2.2b, 3.2.3a, 3.2.3b, 3.2.4.

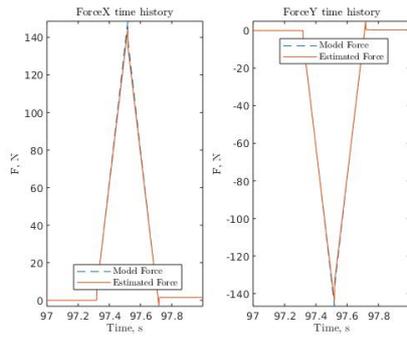


(a) Stiffness

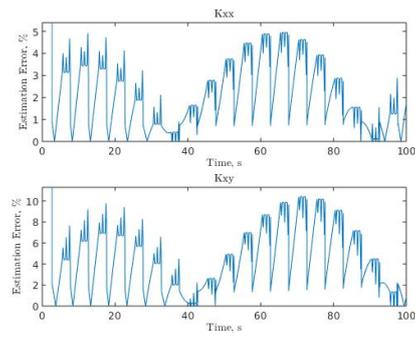


(b) Damping

Figure 3.2.2: Time history of the simulation



(a) Force prediction



(b) Error estimation percentage error

Figure 3.2.3: Results

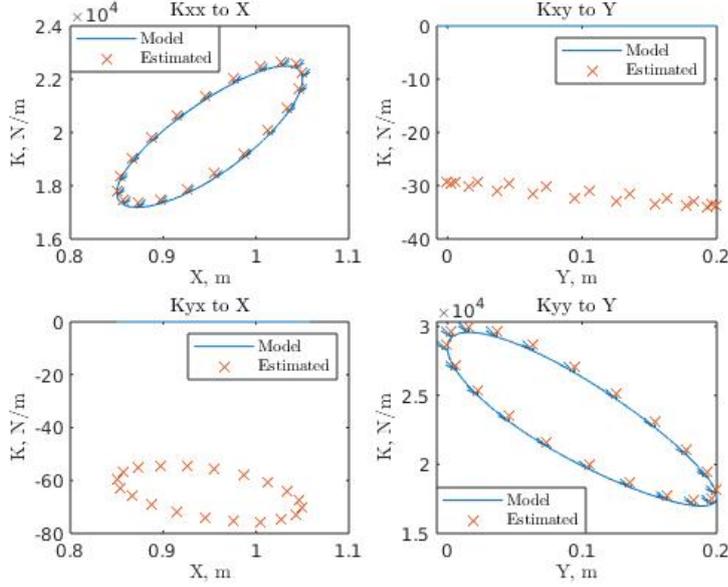


Figure 3.2.4: Stiffness to position relation

3.3 Experiment 3: simplified quadratic identification

3.3.1 Motion

Only sinusoidal trajectory in x direction is considered, with amplitude of 0.1 m and bias of 0.95 m . Due to absence of motion in y direction, the human arm stiffness model described in the first section becomes:

$$K_{h_x}(x_h, t) = k_{x_0} + k_{x_1}x_h + k_{x_3}x_h^2$$

20 triangular perturbations along x axis are evenly distributed across the motion duration of 100 seconds.

3.3.2 Identification

When a perturbation starts, data is recorded every 0.001 s and put in a vector, until the perturbation ends. The recorded data is represented by \mathbf{x}_p , \mathbf{x}_h , $\dot{\mathbf{x}}_h$ and \mathbf{F}_{h_x} . When a perturbation ends, a least squares algorithm is used, but the stored vectors are not discarded. Instead, when the next perturbation comes, they are expanded with new data. The regressor vector

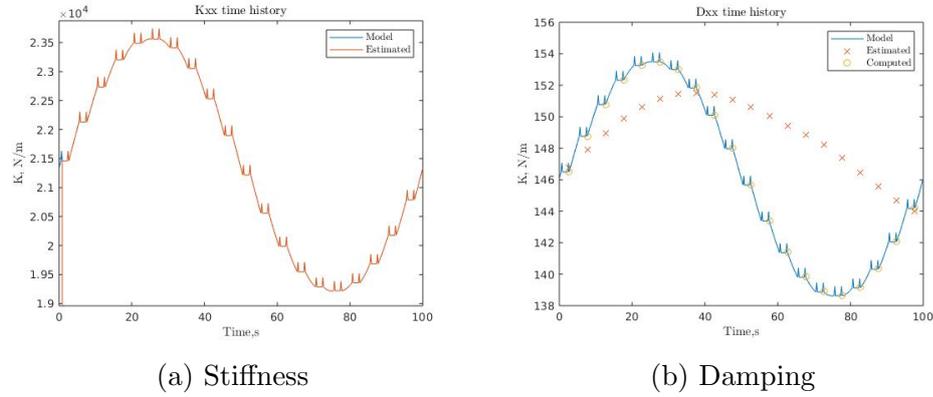


Figure 3.3.1: Time history of the simulation

in this case will be:

$$\phi = \begin{bmatrix} \mathbf{1} \cdot x_p \\ x_h \cdot x_p \\ x_h^2 \cdot x_p \\ \dot{x}_h \end{bmatrix}$$

So that the parameter vector is computed via least squares:

$$[\hat{k}_{x_0} \quad \hat{k}_{x_1} \quad \hat{k}_{x_3} \quad \hat{D}_{h_{xx}}] = \left(\frac{1}{N} \sum \phi \cdot \phi^T \right)^{-1} \left(\frac{1}{N} \sum \phi \cdot F_{h_x} \right),$$

where

$$\hat{K}_{h_x}(x_h, t) = \hat{k}_{x_0} + \hat{k}_{x_1} x_h + \hat{k}_{x_3} x_h^2$$

3.3.3 Results

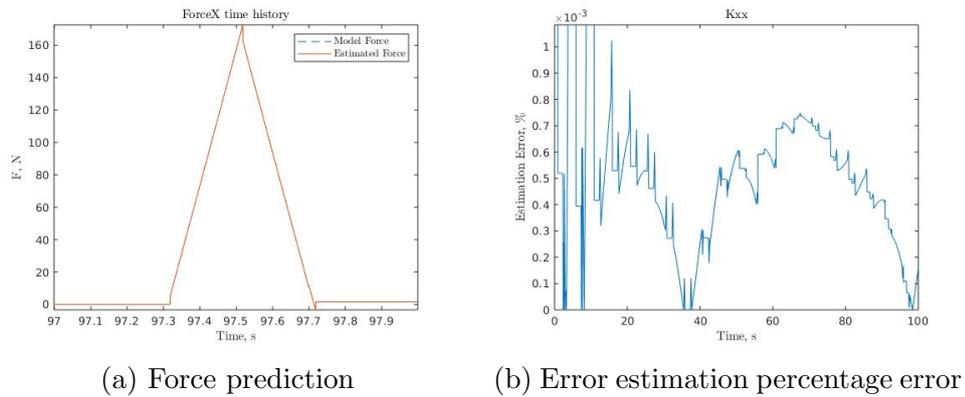


Figure 3.3.2: Results

The results of this identification are shown in Figures 3.3.1a, 3.3.1b, 3.3.2a, 3.3.2b. As seen in figure 3.3.1b, estimated damping does not follow model damping well. Damping, computed as a square root of stiffness, follows the original damping more precisely. It is important to note, however, that elimination of damping estimation from the algorithm worsens the convergence and precision of stiffness estimation.

3.4 Experiment 4: complete quadratic identification

3.4.1 Motion generation

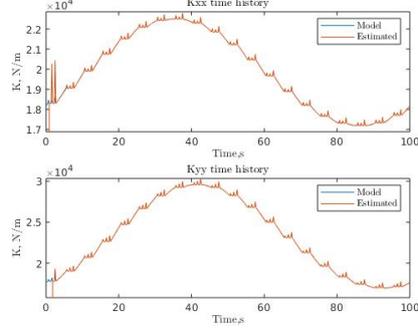
In this case the motion is identical to the Section 3.2.

3.4.2 Identification

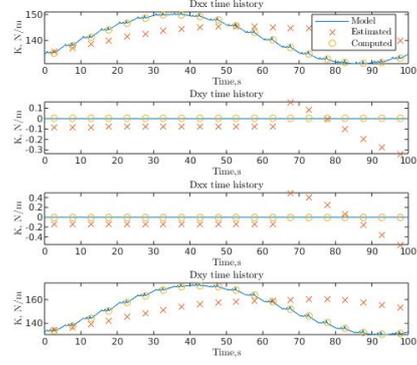
Whenever a triangular perturbation starts along any axis, the positional data is recorded every 0.001 s and put in a vector, until the perturbation ends. Recorded data is represented by x_p and y_p , x_h and y_h , \dot{x}_h and \dot{y}_h , F_{h_x} and F_{h_y} . When a perturbation ends, a least squares algorithm is used, but the stored vectors are not discarded. Instead, when the next perturbation comes, they are expanded with new data. It is possible to limit the length of the vector in order to limit the data history and make the algorithm more sensitive to the new data. The regressor vectors:

$$\phi_x = \begin{bmatrix} 1 \cdot x_p \\ x_h \cdot x_p \\ y_h \cdot x_p \\ x_h^2 \cdot x_p \\ 2x_h y_h \cdot x_p \\ y_h^2 \cdot x_p \\ \dot{x}_h \\ \dot{y}_h \end{bmatrix} \quad \phi_y = \begin{bmatrix} 1 \cdot y_p \\ x_h \cdot y_p \\ y_h \cdot y_p \\ x_h^2 \cdot y_p \\ 2x_h y_h \cdot y_p \\ y_h^2 \cdot y_p \\ \dot{x}_h \\ \dot{y}_h \end{bmatrix} \quad (3.4.1)$$

Note that the velocity data is utilized as well. Even though damping in the model is computed from stiffness, it is important to try and prevent velocity profiles to interfere with stiffness estimation, even when the estimated damping coefficients should be discarded. The parameter vector is computed via



(a) Stiffness



(b) Damping

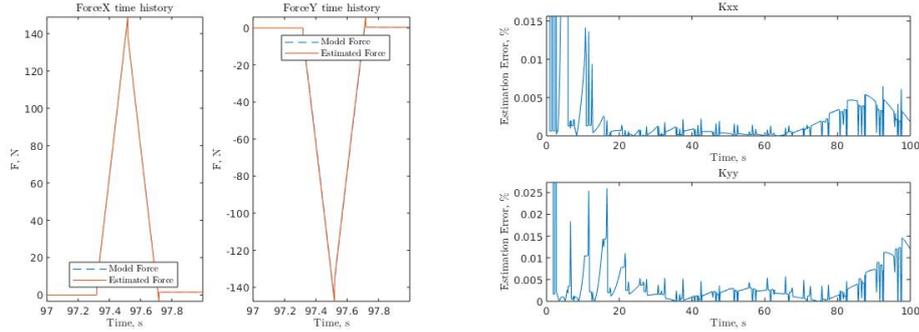
Figure 3.4.1: Time history of the simulation

least squares as follows:

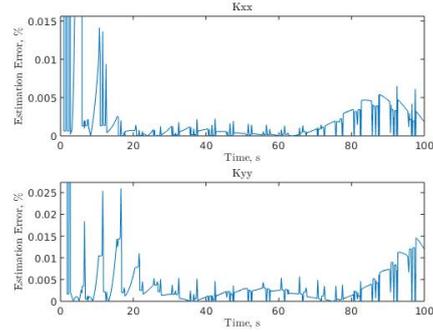
$$\begin{bmatrix} \hat{k}_{x_0} & \hat{k}_{x_1} & \hat{k}_{x_2} & \hat{k}_{x_3} & \hat{k}_{x_4} & \hat{k}_{x_5} & \hat{D}_{h_{xx}} & \hat{D}_{h_{xy}} \end{bmatrix} = \left(\frac{1}{N} \sum \phi_x \cdot \phi_x^T \right)^{-1} \left(\frac{1}{N} \sum \phi \cdot F_{h_x} \right) \quad (3.4.2)$$

$$\begin{bmatrix} \hat{k}_{y_0} & \hat{k}_{y_1} & \hat{k}_{y_2} & \hat{k}_{y_3} & \hat{k}_{y_4} & \hat{k}_{y_5} & \hat{D}_{h_{yx}} & \hat{D}_{h_{yy}} \end{bmatrix} = \left(\frac{1}{N} \sum \phi_y \cdot \phi_y^T \right)^{-1} \left(\frac{1}{N} \sum \phi \cdot F_{h_y} \right) \quad (3.4.3)$$

3.4.3 Results



(a) Force prediction



(b) Error estimation percentage error

Figure 3.4.2: Results

As can be seen on Figure 3.4.1a, the stiffness gets estimated well and the algorithm converges by the second sequence of perturbations. As can be seen

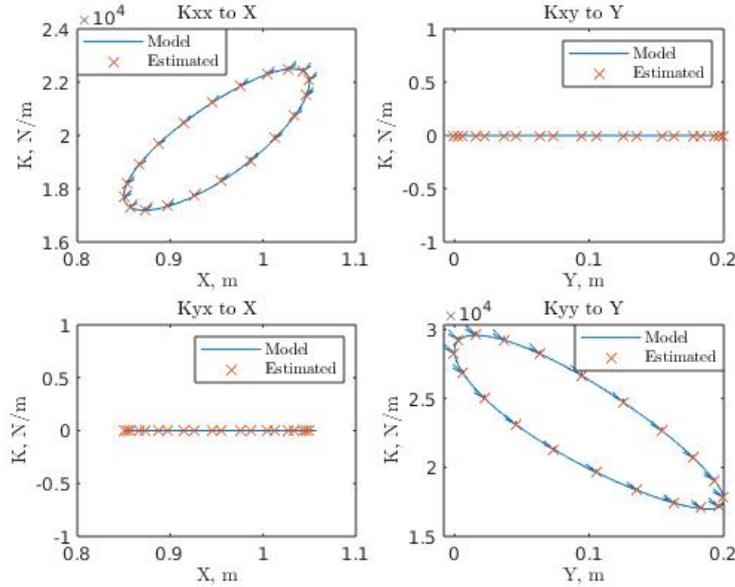


Figure 3.4.3: Stiffness to position relation

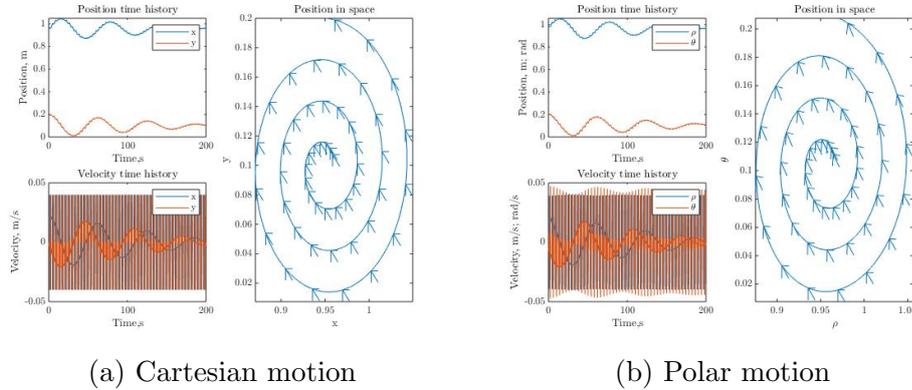
on Figure 3.4.1b, the damping values are not tracked well, but due to the quality of the stiffness estimation and the fact that the original model was deriving the damping values from the stiffness values, a computed version can be used as a substitute.

On Figure 3.4.2a forces time history is shown at the end of the simulation. The estimated force is indistinguishable from the forces generated by the used model. As can be seen from the Figure 3.4.2b, the error quickly drops to a very low level and then experiences fluctuations. Finally, stiffness-position relations can be seen on Figure 3.4.3.

3.5 Experiment 5: polar model exploration

3.5.1 Motion

The motion is equivalent to the one used in Subsection 3.2, with the only difference that the amplitude of the oscillation around the center of the circular trajectory is decaying linearly from 100% to 10%, as shown on Figure 3.5.1a. The perturbations were generated in the same amount and with the same frequency. The goal of picking a new motion is to map the original function in polar coordinates with higher precision. In a simulated experiment, a former model (described in Section 2.2) will be used to generate necessary



forces. Then the input and output data will be transformed into polar space, according to the equations provided in the Subsection 2.2.2. Acquired data will be used to estimate and map stiffness and damping parameters in said space. The identified parameters will be fit as polynomial functions in q_{pol} .

3.5.2 Identification and results

The identification algorithm is identical to the one used in Section 3.2, with only difference that now q_{pol} is an input vector and F_{pol} is an output vector. When the stiffness and damping matrices are estimated, it is possible to fit them with a polynomial function, similarly to how the model from Section 2.2 was derived.

To pick the correct polynomial order, it is possible to fit the data to different polynomials and compare the quality of fit. Here, the quality of fit is represented by the degree-of-freedom adjusted coefficient of determination, R_{adj}^2 .

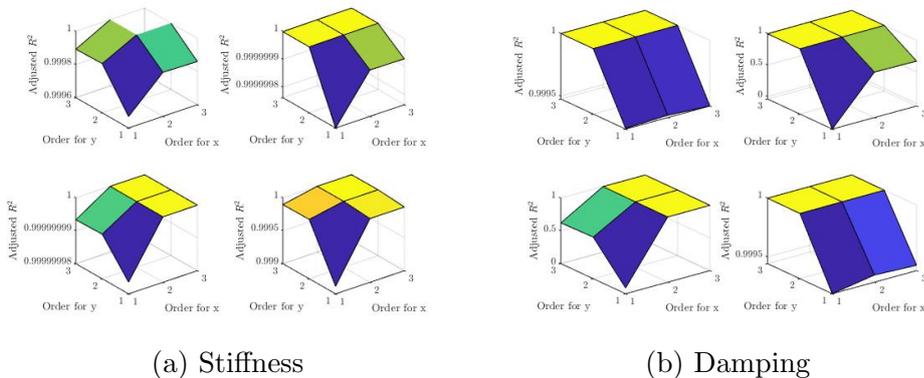


Figure 3.5.2: Cartesian fit quality

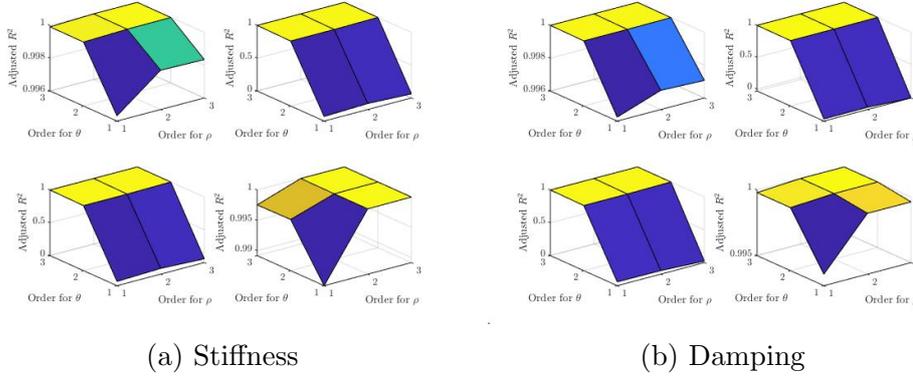


Figure 3.5.3: Polar fit quality

For reference and comparison, the order exploration of Cartesian model is provided on a Figure 3.5.2, while Figure 3.5.3 shows the results for the polar model. As can be seen, the original Cartesian stiffness is so steep that every coefficient of matrix K_h can be fit with a first-order polynomial in both x and y . The Cartesian damping, however, would require a higher order for some of it's coefficients: for d_{h12} it is first order in x and second order in y , while for d_{h21} it is second order in x and first order in y . In case of the polar model, both stiffness and damping required the same configuration: the diagonal elements were fit well with a first order in both ρ and θ , while the anti-diagonal elements required at least second order in θ for a good fit. The results of the fit are shown on Figures 3.5.4 and 3.5.5.

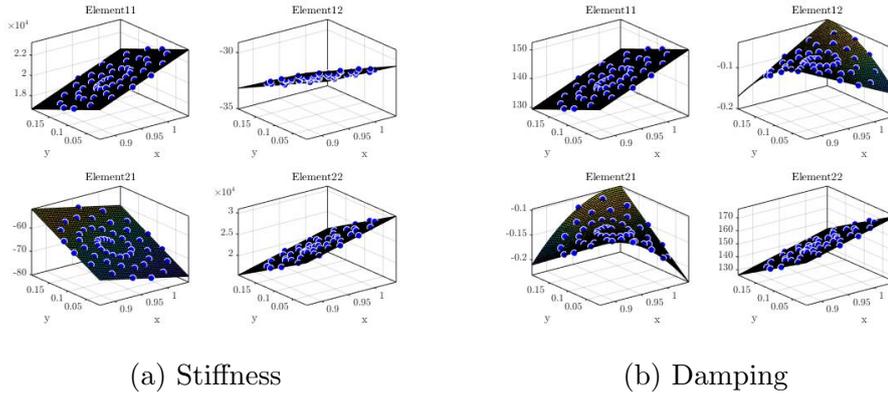


Figure 3.5.4: Cartesian fit quality

In both cases, using the described fit functions, the outputs were estimated well, as seen on Figures 3.5.6a and 3.5.6b. However, it is important to note that the acquired fitting functions will only be valid for the used input

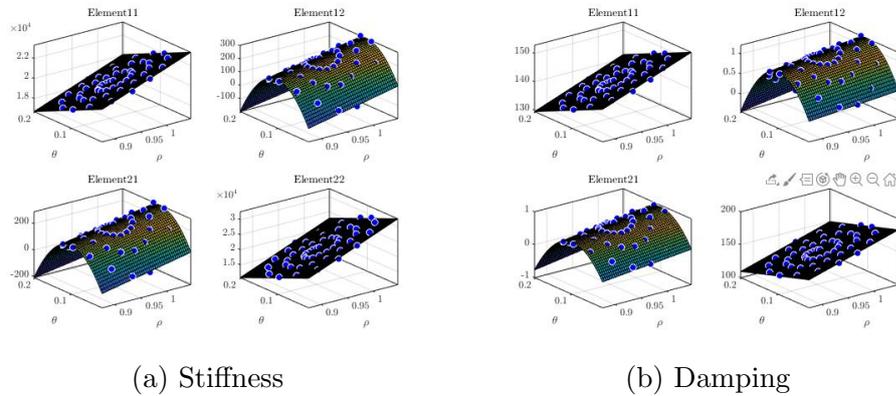


Figure 3.5.5: Polar fit quality

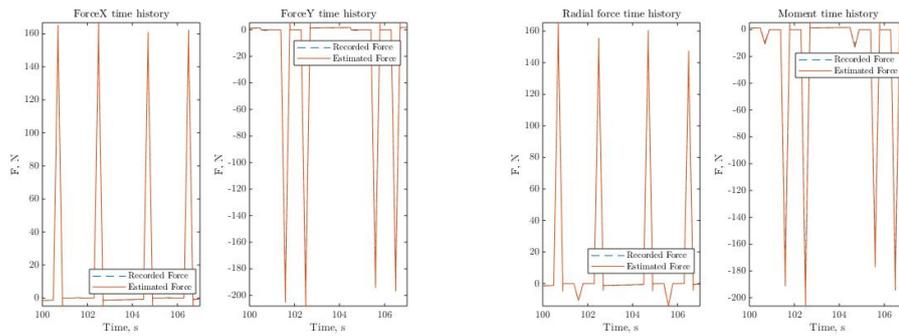
ranges, which incentivizes to cover as much of a working area of the human arm as possible. To compare the results, let's summarize the structure of the models.

The original stiffness model was formulated as a diagonal matrix, whose elements had quadratic relation with both x and y coordinates, requiring 6 parameters per coefficient, 12 in total. The original damping model was being computed directly from the stiffness, using only 1 scaling parameter per coefficient of the diagonal damping matrix.

In case of full matrix identification in Cartesian coordinates, the results were already good with linear relationship of the stiffness matrix coefficients in x and y , resulting in 3 parameters per coefficient, still 12 in total. The damping matrix was identified separately and resulted in 14 parameters in total - 3 per diagonal coefficient and 4 per antidiagonal coefficient, with 26 parameters for the whole model, excluding the modeling of co-contraction.

The polar model showed more consistency in the structure between stiffness and damping matrices that the Cartesian model, each requiring 14 parameters in the same way, 28 parameters in total.

All models estimated the original model well. Identification of the full matrix may provide more insight than the original model, but both Cartesian and polar models will have to be tested to see how they perform in a real-case application.



(a) Cartesian force estimation

(b) Polar force estimation

Discussion

Most of the current studies, some of which were described in Chapter 2, are still challenged by dealing with the neuromusculoskeletal system of a human. While materials of the human body can be tested mechanically in deceased specimens, the presence of the neural system provides enough uncertainty by means of adaptability to make it necessary to account for it. What complicates the problem is the inconvenience of taking the right measurements: the non-invasive methods, such as surface EMG recording, provide only the indirect information about the internal processes and the tracking of the human motion to gain feedback for control purposes can require the use of complex machine vision technology and processing algorithms.

Nevertheless, in a number of mentioned studies such difficulties were overcome with reasonably good results. All the models were using 'black box' and heuristic approaches to modeling, having to make assumptions to simplify the real system. In the meantime, models that start to resemble proper structural models of human body are slowly advancing, mainly within the scientific OpenSim community, allowing to test estimation and control algorithms in simulation and thus, making the research more convenient.

Future development

The most straightforward improvement to this project would be to support the provided suggestions and simulations with the experimental data. Considering that the most detailed suggestions were given to the study in the Section 2.2, a sufficient equipment for the experiments would be a set of a robot manipulator (such as KUKA LWR4+ [35]) with an equipped force sensor [36] and a sEMG reading device (such as Myo armbands [37]). During the experiments, it will be extremely important to cover as much of the working area of the human arm as possible, since the data will be used to build a position-to-parameters map. Considering that the estimation is best done point-by-point, the density of the points must be reasonable. Such experiments can be organized in a way when different areas are traversed at a

time and then the data is united into a single map.

To extend the usefulness of the experiments, it is possible to use more test subjects to do the experiments, since the estimated parameters vary from person to person. However, instead of accepting the variance of their parameters as a fact, it is possible to categorize these variations. For example, additional metrics can be used to map the variance in parameters to the: age, sex, physical fitness, height, arm length (related to height). For example, knowing the arm length, it would be possible to scale maps and compare the variance in estimated parameters (that depended on the spatial coordinates) in the context of the available working area of a specific test subject. With maps having been built and parameterized, it is also possible to move from the least squares estimation algorithm to the recursive least squares algorithm. While the recursive least squares algorithm is preferable for the real-time estimation purposes because of the greater performance, it also allows for the initialization of the parameters. In such a way it will be possible to use the data on the parameters that were estimated from the past test subjects to pick a good starting point for a new test subject, depending on his height, age, etc. This may allow for performing simpler tests that will establish a framework for simpler calibration phases, which would be a must in practical applications.

Finally, to further expand the only study that was suggestion rich in this thesis, an estimation algorithm can be fitted to work in the 3D space, to make the approach competitive with other methods, such as described in Section 2.1. The main difficulty to overcome in this case would be the understanding of how to do the additional measurements, mostly relevant to the arm configuration, such as shoulder position, to be able to provide the full information to the estimation algorithms in absence of the complicated tracking equipment.

Conclusions

In this project, a brief overview of the human modeling field was made. The overview described various application of human modeling, provided and explained a number of general modeling techniques and referenced some particular examples of such techniques being used in research.

Then, the project focused on exploring various researches, dedicated to the modeling of the human arm properties, such as impedance, stiffness in particular or EMG-force relationships. All referenced studies were recent, they were briefly explained and some remarks were given, which targeted the core

issues which were not addressed by the authors of the studies. One research in particular, described in Section 2.2, has received more detailed suggestions, some of which later were tested in simulations.

Among the simulations, an online least squares estimation was tested. It was shown that the least squares estimation is capable of estimating stiffness matrix coefficients as polynomial functions, given that in the model used for generation of the data they were computed as such. From that, it is clear that such methods can be used to track the human arm impedance in real-time mode, if necessary program is used with the equipment that is capable of recording the reaction forces of a perturbed human hand and the surface EMG signals from the human arm to compute the muscle activation levels. Lastly, a transformation to the polar coordinates space was explored within the simulation. It was shown that such models are also viable to estimate stiffness, with implications that such an approach can be a middle ground between methods that require tracking of the joint coordinates of the human arm and methods that track only the arm's endpoint coordinates.

Appendix A

Proofs

Stiffness model in a different frame

The original stiffness model is represented by an equation

$$K_{h^{**}}(\mathbf{q}_h, t) = k_{*0} + [k_{*1} \quad k_{*2}] \mathbf{q}_h + \mathbf{q}_h^T \begin{bmatrix} k_{*3} & k_{*4} \\ k_{*4} & k_{*5} \end{bmatrix} \mathbf{q}_h,$$

where $\mathbf{q}_h = \begin{bmatrix} x_h \\ y_h \end{bmatrix}$. The change of frame would modify the \mathbf{q}_h vector to a new \mathbf{q}_h^* vector as follows:

$$\mathbf{q}_h^* = \mathbf{A} \mathbf{q}_h + \mathbf{b} = \begin{bmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} -O_x^* \\ -O_y^* \end{bmatrix},$$

where matrix \mathbf{A} describes the rotational motion of the frame and vector \mathbf{b} describes the translational motion of the frame. The new frame would be rotated by the θ angle relatively to the old frame and moved to the point $O^*\{O_x^*, O_y^*\}$. Let's see if the stiffness model equation in a new frame is different enough to require identification of a model with different structure.

$$\begin{aligned}
K_{h^{**}}^* (\mathbf{q}_h^*, t) &= k_{*0} + [k_{*1} \quad k_{*2}] \mathbf{q}_h^* + \mathbf{q}_h^{*T} \begin{bmatrix} k_{*3} & k_{*4} \\ k_{*4} & k_{*5} \end{bmatrix} \mathbf{q}_h^* \\
&= k_{*0} + [k_{*1} \quad k_{*2}] (\mathbf{A}\mathbf{q}_h + \mathbf{b}) + (\mathbf{A}\mathbf{q}_h + \mathbf{b})^T \begin{bmatrix} k_{*3} & k_{*4} \\ k_{*4} & k_{*5} \end{bmatrix} (\mathbf{A}\mathbf{q}_h + \mathbf{b}) \\
&= (k_{*0} + [k_{*1} \quad k_{*2}] \mathbf{b} + \mathbf{b}^T \begin{bmatrix} k_{*3} & k_{*4} \\ k_{*4} & k_{*5} \end{bmatrix} \mathbf{b}) + ([k_{*1} \quad k_{*2}] \mathbf{A}) \mathbf{q}_h \\
&\quad + \mathbf{q}_h^T (\mathbf{A}^T \begin{bmatrix} k_{*3} & k_{*4} \\ k_{*4} & k_{*5} \end{bmatrix} \mathbf{A}) \mathbf{q}_h \\
&= k_{*0}^* + [k_{*1}^* \quad k_{*2}^*] \mathbf{q}_h + \mathbf{q}_h^T \begin{bmatrix} k_{*3}^* & k_{*4}^* \\ k_{*4}^* & k_{*5}^* \end{bmatrix} \mathbf{q}_h,
\end{aligned}$$

where

$$\begin{aligned}
k_{*0}^* &= k_{*0} + [k_{*1} \quad k_{*2}] \mathbf{b} + \mathbf{b}^T \begin{bmatrix} k_{*3} & k_{*4} \\ k_{*4} & k_{*5} \end{bmatrix} \mathbf{b} \\
&= k_{*0} - k_{*1} O_x^* - k_{*2} O_y^* + k_{*3} O_x^{*2} + k_{*4} O_x^* O_y^* + k_{*5} O_y^{*2} \\
k_{*1}^* &= [k_{*1} \quad k_{*2}] \begin{bmatrix} \cos \theta \\ \sin \theta \end{bmatrix} \\
&= k_{*1} \cos \theta + k_{*2} \sin \theta \\
k_{*2}^* &= [k_{*1} \quad k_{*2}] \begin{bmatrix} -\sin \theta \\ \cos \theta \end{bmatrix} \\
&= -k_{*1} \sin \theta + k_{*2} \cos \theta \\
k_{*3}^* &= [\cos \theta \quad \sin \theta] \begin{bmatrix} k_{*3} & k_{*4} \\ k_{*4} & k_{*5} \end{bmatrix} \begin{bmatrix} \cos \theta \\ \sin \theta \end{bmatrix} \\
&= k_{*3} \cos^2 \theta + 2k_{*4} \sin \theta \cos \theta + k_{*5} \sin^2 \theta \\
k_{*4}^* &= [\cos \theta \quad \sin \theta] \begin{bmatrix} k_{*3} & k_{*4} \\ k_{*4} & k_{*5} \end{bmatrix} \begin{bmatrix} -\sin \theta \\ \cos \theta \end{bmatrix} \\
&= -k_{*3} \sin \theta \cos \theta - k_{*4} \sin^2 \theta + k_{*4} \cos^2 \theta + k_{*5} \sin \theta \cos \theta \\
k_{*5}^* &= [-\sin \theta \quad \cos \theta] \begin{bmatrix} k_{*3} & k_{*4} \\ k_{*4} & k_{*5} \end{bmatrix} \begin{bmatrix} -\sin \theta \\ \cos \theta \end{bmatrix} \\
&= k_{*3} \sin^2 \theta - 2k_{*4} \sin \theta \cos \theta + k_{*5} \cos^2 \theta
\end{aligned}$$

From this, it is possible to see that, even if the frame of the model changes, the structure does not.

Bibliography

- [1] G. E. Loeb and R. Davoodi. Musculoskeletal Mechanics and Modeling. *Scholarpedia*, 11(11):12389, 2016. revision #169693.
- [2] J. H. Lawrence and C. J. De Luca. Myoelectric signal versus force relationship in different human muscles. *Journal of Applied Physiology*, 54(6):1653–1659, 1983. PMID: 6874489.
- [3] Eleanor Criswell. *Cram’s introduction to surface electromyography*. Jones and Bartlett Publishers, Sudbury, Massachusetts, 2011.
- [4] Mohammad Ali Ahmadi-Pajouh, Farzad Towhidkhah, and Mohammad Moradi. Estimation of time-varying human arm stiffness using electromyogram signal. pages 26–30, 12 2012.
- [5] Heiner Bubb and Florian Fritzsche. A scientific perspective of digital human models: Past, present, and future. In Vincent G. Duffy, editor, *Handbook of Digital Human Modeling*, pages 3–1–3–26. CRC Press, Taylor & Francis Group, 2008.
- [6] Robert E. Kearney and Robert F. Kirsch. System identification and neuromuscular modeling. In Patrick E. Crago Jack M. Winters, editor, *Biomechanics and neural control of posture and movement*, pages 134–147. Springer-Verlag New York Inc, 2000.
- [7] Patrick E. Crago. Creating neuromusculoskeletal models. In Patrick E. Crago Jack M. Winters, editor, *Biomechanics and neural control of posture and movement*, pages 119–133. Springer-Verlag New York Inc, 2000.
- [8] Ian E. Brown and Gerald E. Loeb. A reductionist approach to creating and using neuromusculoskeletal models. In Patrick E. Crago Jack M. Winters, editor, *Biomechanics and neural control of posture and movement*, pages 148–163. Springer-Verlag New York Inc, 2000.

- [9] S. Chen and S. A. Billings. Representation of non-linear systems: the NARMAX model. *International Journal of Control*, 49(3):1012–1032, March 1989. Address: London.
- [10] M Raez, M Hussain, and Faisal Mohd-Yasin. Techniques of emg signal analysis: Detection, processing, classification and applications. *Biological procedures online*, 8:11–35, 02 2006.
- [11] Robert Kearney and I.W. Hunter. System identification of human joint dynamics. *Critical reviews in biomedical engineering*, 18:55–87, 02 1990.
- [12] Javad Hashemi, Evelyn Morin, Parvin Mousavi, Katie Mountjoy, and Keyvan Hashtrudi-Zaad. Emg-force modeling using parallel cascade identification. *Journal of electromyography and kinesiology : official journal of the International Society of Electrophysiological Kinesiology*, 22:469–77, 01 2012.
- [13] Katherine R. S. Holzbaur, Wendy M. Murray, and Scott L. Delp. A model of the upper extremity for simulating musculoskeletal surgery and analyzing neuromuscular control,” ann. *Biomed. Eng*, pages 829–840, 2005.
- [14] C. Fang, A. Ajoudani, A. Bicchi, and N. G. Tsagarakis. A real-time identification and tracking method for the musculoskeletal model of human arm. In *2018 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, pages 3472–3479, Oct 2018.
- [15] Chun Y. Seow. Hill’s equation of muscle performance and its hidden insight on molecular mechanisms. *The Journal of General Physiology*, 142(6):561–573, 2013.
- [16] Archibald Vivian Hill. The heat of shortening and the dynamic constants of muscle. *Proceedings of the Royal Society of London. Series B - Biological Sciences*, 126(843):136–195, 1938.
- [17] Huxley A.F. Muscle structure and theories of contraction. *Progress in Biophysics and Biophysical Chemistry*, 7:255–318, 1957.
- [18] Robert F. Kirsch and Richard B. Stein. Neural and muscular properties: Current views and controversies. In Patrick E. Crago Jack M. Winters, editor, *Biomechanics and neural control of posture and movement*, pages 39–57. Springer-Verlag New York Inc, 2000.

- [19] George I. Zahalak and Shi-Ping Ma. Muscle Activation and Contraction: Constitutive Relations Based Directly on Cross-Bridge Kinetics. *Journal of Biomechanical Engineering*, 112(1):52–62, 02 1990.
- [20] Koen K. Lemaire, Guus C. Baan, Richard T. Jaspers, and A. J. ‘Knoek’ van Soest. Comparison of the validity of hill and huxley muscle–tendon complex models using experimental data obtained from rat m. soleus in situ. *Journal of Experimental Biology*, 219(7):977–987, 2016.
- [21] M. S. Erden and A. Billard. End-point impedance measurements at human hand during interactive manual welding with robot. In *2014 IEEE International Conference on Robotics and Automation (ICRA)*, pages 126–133, May 2014.
- [22] Satoko Abiko, Atsushi Konno, and Masaru Uchiyama. An asymmetric stiffness model of a human hand. In *IEEE/RSJ 2010 International Conference on Intelligent Robots and Systems, IROS 2010 - Conference Proceedings*, pages 5034–5041, 2010.
- [23] A. C. Schouten, E. de Vlugt, J. J. van Hilten, and F. C. T. van der Helm. Quantifying proprioceptive reflexes during position control of the human arm. *IEEE Transactions on Biomedical Engineering*, 55(1):311–321, Jan 2008.
- [24] Arash Ajoudani, Cheng Fang, Nikos Tsagarakis, and Antonio Bicchi. Reduced-complexity representation of the human arm active endpoint stiffness for supervisory control of remote manipulation. *The International Journal of Robotics Research*, 37(1):155–167, 2018.
- [25] Vladimir Zatsiorsky. *Kinetics of Human Motion*. 01 2002.
- [26] C. Fang, A. Ajoudani, A. Bicchi, and N. G. Tsagarakis. Online model based estimation of complete joint stiffness of human arm. *IEEE Robotics and Automation Letters*, 3(1):84–91, 2018.
- [27] L. Bascetta and G. Ferretti. Ensuring safety in hands-on control through stability analysis of the human-robot interaction. *Robotics and Computer Integrated Manufacturing*, 57:197–212, 2019.
- [28] Martina Cavenaghi. Design of an adaptive admittance control for physical human-robot interaction in the lead-through programming collaborative task. Master thesis, School of Industrial and Information Engineering, 2018.

- [29] Mattia Pesenti. Emg-driven human modelling to enhance human-robot interaction control. Master thesis, School of Industrial and Information Engineering, 2018.
- [30] Jan-Willem [van Wingerden] and Michel Verhaegen. Subspace identification of bilinear and lpv systems for open- and closed-loop data. *Automatica*, 45(2):372 – 381, 2009.
- [31] P.B. Cox. *Towards efficient identification of linear parameter-varying state-space models*. Phd thesis, Eindhoven University of Technology, 2018.
- [32] Estimate Hammerstein-Wiener model. MATLAB documentation. <https://uk.mathworks.com/help/ident/ref/nlhw.html>.
- [33] Jiang Zainan, Yang Fan, Li Chongyang, Liu Daxiang, Wang Chenliang, Li Tianhui, and Liu Hong. sEMG-based endpoint stiffness estimation of human arm using gene expression programming. *Journal of Physics: Conference Series*, 1267:012016, jul 2019.
- [34] Chenliang Wang, Li Jiang, Chuangqiang Guo, Qi Huang, Bin Yang, and Hong Liu. semg-based estimation of human arm force using regression model. pages 1044–1049, 12 2017.
- [35] KUKA LWR. User-friendly, sensitive and flexible. <https://www.coboticsworld.com/wp-content/uploads/2019/05/KUKA-LWR-Brochure.pdf>.
- [36] M38XX series product page. <https://srisensor.com/33.html>.
- [37] Myo Gesture Control Armband tech specs. <https://support.getmyo.com/hc/en-us/articles/202648103-Myo-Gesture-Control-Armband-tech-specs>.