POLITECNICO DI MILANO
Dipartimento di Elettronica, Informazione e Bioingegneria
Master of Science programme in Automation and Control Engineering

ONLINE STIFFNESS ESTIMATION OF HUMAN ARM IN PHYSICAL HUMAN ROBOT INTERACTION USING MACHINE LEARNING

Supervisor: Prof. Luca Bascetta
Co-supervisor: Prof. Elena De Momi

Master Thesis by
Ivan Vasileski
Matr. 878669

Academic year 2018/2019

Acknowledgments

I would like to thank to the supervisor Prof. Luca Bascetta and the co-supervisor Prof. Elena De Momi for their help and evaluation. I am also thankful to Mr. Jacopo Buzzi, and Mr. Andrea Mariani, for their guidance and collaboration. Thanks to the entire team that worked on this research project and the academic staff at NearLab laboratory.

Additionally, I would like to thank the complete university of Politecnico Di Milano for providing me with the great opportunity and inspiration to achieve Master of Science degree.

Abstract

As **H**uman **R**obot **I**nteraction (HRI) sees its advancements in medical and industrial use, shared control emerged as a new promising field of investigation. The class of impedance controllers plays a major role in enabling such continuous physical HRI. Variable impedance controllers have the potential to offer reliable accuracy, stability, and therefore safety during such kind of HRI. However, they depend on positional and force input by the user. To further unlock this potential, they can be provided with information on the dynamical parameters of the user's arm. Human arm stiffness, being the dominant dynamical parameter, had been widely investigated. Consequently, experimental offline methods to estimate it had been developed. Research suggests that humans extensively use arm configuration to control the direction of the stiffness, and muscular cocontractions to control the magnitude. Surface electromyography (sEMG) signals have been widely used as a reliable measure of muscular activity and therefore they have been utilized in estimating cocontraction of muscles. In this work, an online method to estimate human arm stiffness based on human arm configuration and sEMG signals is proposed. To further develop the cocontraction estimates based on sEMG, Muscular Jacobian matrix is used. The algorithm used to compute this Jacobian in real-time is derived too. Due to the ability to model complex non-linearities, an Artificial Neural Network (ANN) model is developed. Data used to train such model is acquired using experimental offline method in the frequency domain. The proposed model performs well compared to linear model proposed in the literature. Also, its ability to interpolate between samples presented in the training process is verified. The ANN model opens further possibility to be used in non-user specific strategy. Thus, sharing the common knowledge conveyed by the experimentally acquired data of multiple users.

Sintesi

Man mano che l'Interazione Uomo-Robot (Human Robot Interaction - HRI) vede i suoi progressi nell'uso medico e industriale, il controllo condiviso è emerso come un nuovo e promettente campo di ricerca. La classe di controllori di impedenza svolge un ruolo importante nel consentire tale HRI fisica continuamente. I controllori a impedenza variabile hanno il potenziale per offrire accuratezza, stabilità e quindi sicurezza durante questo tipo di HRI. Tuttavia, dipendono dall'input di posizione e forza dell'utente. Per sbloccare ulteriormente questo potenziale, possono essere fornite informazioni sui parametri dinamici del braccio dell'utente. La rigidità del braccio umano, essendo il parametro dinamico dominante, era stata ampiamente investigata. Di conseguenza, sono stati sviluppati metodi sperimentali offline per stimarla. La ricerca suggerisce che gli esseri umani utilizzano ampiamente la configurazione del braccio per controllare la direzione della rigidità e le cocontrazioni muscolari per controllare la sua grandezza. I segnali di elettromiografia di superficie (sEMG) sono stati ampiamente utilizzati come misura affidabile dell'attività muscolare e quindi sono stati utilizzati nella stima della cocontrazione dei muscoli. In questo lavoro, viene proposto un metodo online per stimare la rigidità del braccio umano in base alla configurazione del braccio e ai segnali sEMG. Per sviluppare ulteriormente le stime di cocontrazione basate su sEMG, viene utilizzata la matrice Jacobiana Muscolare. Viene anche derivato l'algoritmo per calcolare questo Jacobiano in tempo reale. A causa della capacità di modellare le non-linearità complesse, viene sviluppato un modello di rete neurale artificiale (ANN). I dati utilizzati per addestrare tale modello vengono acquisiti utilizzando un metodo sperimentale offline nel dominio della frequenza. Il modello proposto si comporta bene rispetto al modello lineare proposto in letteratura. Inoltre, viene verificata la sua capacità di interpolare tra i campioni presentati nel processo di addestramento. Il modello ANN apre ulteriori possibilità di essere utilizzato in strategie non specifiche a un utente. Quindi, condividendo la conoscenza comune trasmessa dai dati acquisiti sperimentalmente di più utenti.

Table of Contents

Index of Figures

Index of Tables

1. Introduction to Human-Robot Interaction

Robots can be described as machines working with constant interaction with their environment, performing certain task of convenience. Methods, tools, and concepts of standard robotics today enable only kinematic interaction between the environment and the machine. Such limited abilities for interaction require highly structured, organized, and controlled environments referred to as robotic cells. Often enough, these cells are furtherly adjusted to appeal to the abilities of the robot. It is not a rare case when even the actual product is adapted to make the operation of a robotic cell faster, and therefore more efficient. Examples of such tasks are: positioning (welding), following trajectory (CNC machining), and simple pick and place tasks (palletizing).

Introducing uncertainty to robot's environment produces need for handling the problem of identification, isolation, and appropriate reaction. Additionally, safety and productivity of the cell should be considered too. Uncertainties can be of a nature peculiar to robot's task e.g. agriculture robot has to deal with the event of rain which is not the case with indoor robots. But also mutual to all robots e.g. human behavior, mechanical obstacles, etc. If such approach is found, so that majority of the uncertainties are handled, then it softens the notion of "robotic cell" and improves robotic autonomy.

Handling uncertainties due to human behavior takes more significant position in the scientific development. It ranges from basic safety to putting such uncertainties into practical use by treating them as meaningful input information to guide the robot behavior.

Human-**R**obot **I**nteraction (HRI) is a field of study dedicated to understanding, designing, and evaluating robotic systems for use by or with humans. [1]

However, handling uncertainties comes with higher cost of equipment, engineering expertise, and sacrifice in productivity. Therefore, except for safety, it had long been considered economically unjustified

1.1 Contemporary use of HRI

Standard robot practices overwhelm the mass production industry. Their use is by far most widespread practice in manufacturing. However, recent socio-economical changes steered manufacturing practices from mass to flexible production.

1.1.2 HRI in manufacturing

To keep up with the growing customization ambitions of their customers, companies adopted flexible approaches throughout their departments. High degree of customization is achievable by adjusting the working strategy in the design stage, fabrication, assembly, marketing, and even through post-sale stage. However, modification of the assembly process is one of the most cost-effective approaches to high product variety [2].

Therefore, companies in collaboration with the scientific field searched for ways to increase the customization in the assembly process. Since robots were already the dominant machines used for assembly, naturally they have been one of the main objects to change. This has led to rediscovering the need for handling uncertainties produced by human behavior. However, this time not merely as safety requirement. But also, as new and hybrid manufacturing concept which utilizes human intelligence and creativity to create agile manufacturing paradigm capable of addressing markets' requirements. The advantages of human-robot cooperation have been evaluated in several studies [3, 4, 5].

Simple robotic tasks, without any form of computer intelligence, usually had been programmed by trained robotic technicians. The easiest way is to use the teach pendant of the robot-controller to position it manually into a set of poses in cartesian or joint space. During execution, the robot automatically replays the saved poses resulting in a continuous movement. Tasks which include intelligent control i.e. computer vision, usually take more time and often require an employee with engineering degree. The high demand for customization led to demand for minimization of the programming time. To satisfy that call, new and innovative method of "teaching-with-guiding" has emerged. Teaching-with-guiding is an HRI method that modulates the virtual impedance of the robot making it appear as object with mass, damping, and stiffness far smaller than the actual one. Therefore, operator doesn't need to manually position the robot through the controller's teach pendant. She/he can hold it by the end-effector and intuitively position it to the desired pose.

During replay of the saved poses, the robot is still unaware of human behavior in his environment. Therefore, human safety is not guaranteed. For this reason, conventional robots, as well as those programmed by teaching-with-guiding, are guarded by fences which limit human proximity to the cell. The safety mechanism in place makes sure that if such fence is opened, the movement of the robot comes to a complete halt.

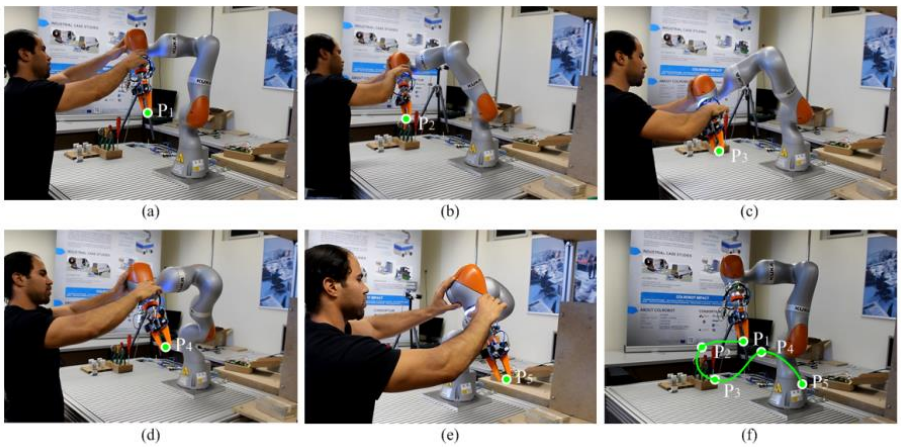


Figure 1. (a)-(e) Guiding the robot by holding its end-effector and positioning it in different poses (f) Resulting trajectory which is replayed

To achieve higher degree and diversity of HRI, the abandonment of fending fences is rational objective. Naturally, this objective calls for advancements in human safety mechanisms. ISO 10218 part 1 and part 2 defines safety requirements whose implementation may lead to abandonment of fences. Such implementations include [6]:

- *Safety Monitored Stop.* This approach monitors the robot's surroundings using laser or vision system and stops the execution of the robotic task if human (or other unexpected movement for that manner) is registered. This approach bears similarities with the conventional halting mechanism in case of opening the fence's door. However, it is important to notice that in this case the execution is only temporarily paused. Detecting human presence in the robot's surroundings is not accounted as dangerous breach of the standard operating protocol. It could be just operator continuously feeding more raw material to the cell or bringing out the finished product (or semi-product).
- *Speed and Separation Monitoring.* There is safety-zone defined around the robot. However, when human presence is detected, the robot doesn't come to a stop. It adjusts its velocity according to distance to the human. Linear relation might be one potential choice. As the human gets closer to the robot, the robot's velocity is lower. If the distance comes under pre-defined threshold, the execution comes to a complete stop.
- *Power and Force limiting.* Additional equipment in the mechanical structure of end-effector and joints allows robotic systems to take into consideration forces (and therefore, power) that they release to the environment. Such advantage makes it possible to pose limits on dynamical variables that will render the robot virtually uncapable of harming human being in its vicinity. As the robot moves through the environment, it can sense forces applied to his end-effector, and to his body. If such forces are not natural for the task it is performing, it may initiate safety reaction like

slowing down or coming to a complete stop. Additionally, it can choose to comply with the external forces by adjusting its virtual impedance, similarly like “teaching-with-guiding” method.

All these approaches have evident advantages and disadvantages. Out of common disadvantages, sacrifice of productivity can be identified as one of special interest. It is important to notice that combination and/or composition of these approaches may lead to compensating the decrease in productivity to some degree.

Once safety is guaranteed, the meaningfulness of the interaction is rational objective. Better HRI can be achieved if the robot is capable to grasp human intention and act accordingly to its functional position within the system. In mass customization, HRI is a way to utilize human intelligence and creativity. However, such human qualities can’t be transferred to the machine in their native, human form. Therefore, new forms of representation of human peculiarities must be proposed. Forms such that, are acceptable by humans and in the same time, a machine can draw proper conclusion for action. This objective is fertile ground for development and usage of Artificial Intelligence methods. They can provide the system with wider spectra of information upon which conclusions for action will be made. In this direction, developments in detecting human intention and enabling human-robot skill transfer can be observed.

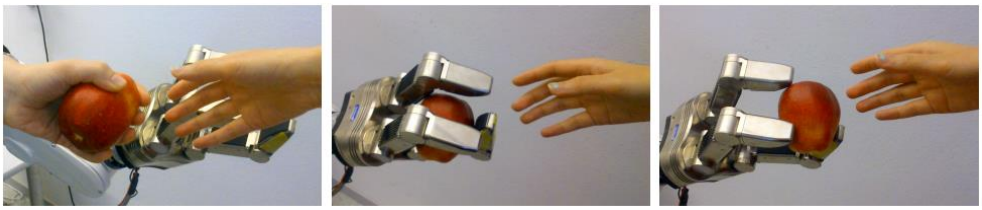


Figure 2. (a) Human demonstrates a grasp with an intention to hand-over an apple. (b) Robot imitates the power grasp configuration used by the human and fails to hand-over because there is not enough free space for regrasp. (c) Robot estimates that human intends to hand-over the apple. It also learns the task requires leaving enough free space on the object. It applies a precision grasp to achieve the task. [7]

1.1.3 HRI in medicine

Following the natural expansion of industrial robotics, medical robotics adopted the same premise of higher accuracy and higher speed of task execution as main economic incentive. After the initial attempts of robot usage in surgery, the premise for higher accuracy was confirmed. However, it was not the case with procedure times. Even though medical robotics managed to achieve similar time of the surgery workflow, the time required for setting up the machine resulted in overall number of procedures equal or lower than what an individual surgeon can manage. While reducing procedure time is still an important quest, other economic factors driving the medical robotics industry emerged: patient demand, reduction of surgical errors, augmenting surgical capabilities, and availability to perform Minimally Invasive Surgery.

Even though the term is self-explanatory, it is important to notice that MIS is defined as any surgery which is less invasive than open surgery of the same type. i.e. surgeries with small length of incisions (< 1cm), minimization of bone removal in orthopedic tasks, etc. MIS is the largest economical factor driving this industry recently. [8]

In a medical scenario, the human body represents the robotic environment, and it is characterized by two properties:

1. All bodies have the same internal anatomical structure,
2. bodies intend to have unique, peculiar manifestation of illnesses, malfunctioning, and conditions.

The first property suggests that due to the anatomical equality of all human beings, it is reasonable to consider certain repetitiveness in medical tasks. And repetitiveness is the natural justifier for robot usage. On the other side, the second property calls for an extremely high degree of customization. Such level of customization that conventional programming or AI still can't provide. Therefore, unlike robots used in manufacturing, even the earliest use of robots was implemented as a form of HRI.

Tele-operating approaches are one example of HRI in the medical robotics industry. Such systems implement slave-master architecture. An architecture that can be described as two robots working in collaboration. The slave mimics the movements of the master robot, which potentially can be remote or in close vicinity. The two sides don't have to adopt same mechanical and/or kinematical structure. The master robot is controlled by the surgeon, and the slave is in contact with the patient. Due to this setup of the system, the master robot usually adopts kinematical structure that allows intuitive following of surgeon's hand movements. On the other hand, the slave adopts kinematical structure that supports the specific purpose of the system. The kinematical difference is handled by appropriate transformations implemented in software.
Tele-operation implies two channels of HRI. The first one being the interaction between the surgeon and its master robot, and second, the remote way of interaction between the surgeon and the slave.

- *Master-surgeon Interaction.* This case has conceptual similarities as "teaching-with-guiding" programming method. The virtual impedance of the robot is adjusted so it appears as it has altered mass, damping, and stiffness. Combined with gravity compensation, this method allows to the surgeon to perform natural, intuitive hand movements which are recorded by the system. As a difference from "teaching-with-guiding", the system doesn't record poses which are later replayed, but surgeon's movements are sampled and immediately reproduced by the slave, filtering out natural human tremor.
- *Slave*-surgeon *Interaction.* This channel implies the feedback that the surgeon receives from the slave as a consequence of mimicking his actions. Visual imagery is usually captured by endoscopic or x-ray equipment and fed back to the surgeon.

Since it is dominant form of feedback, imagery is often in high-resolution and three dimensional. This allows for more meaningful feedback stream that can replace the lack of haptics.

Intuitive®'s (Sunnyvale, CA, USA) da Vinci system is prime example of tele-operating system for MIS. It consists of single or dual surgeon's console and a patient-side part with three or four robotic arms. In the surgeon's console, the master robot and the visual feedback are housed. On the patient-side, the robotic arms of the slave machine manipulate laparoscopic instruments, including an endoscope. Slave's arms have 7 DOFs to effectively mimic the dexterity of the human arm. This system provides the surgeon with intuitive control, range of motion, fine tissue manipulability, and visual feedback during the MIS.

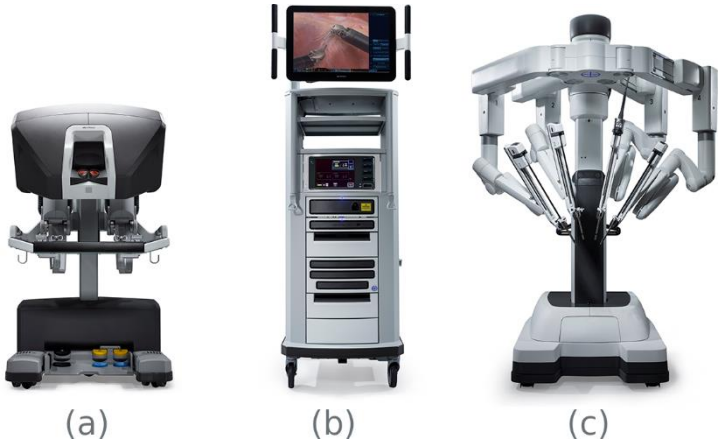


Figure 3. Intuitive®'s Da Vinci surgical system. (a) Slave console. (b) Supporting hardware. (c) Master console. [9]

Hansen®'s (Mountain View, CA, USA) Sensei robotic catheter system adopts the same tele-operating architecture. However, its slave console is not equipped for laparoscopic surgery like in the Da Vinci case. Sensei's slave robot has mechanical and kinematical structure that allows driving a catheter device at a desired point inside the heart. The visual feedback is provided to the surgeon at the master console through x-ray imaging device. The master is usually in near-vicinity to the slave, but still on a safe distance from the x-ray source.

Figure 4. Hansen®'s Sensei Robotic System. (a) Master console. (b) Slave console and the catheter. [10]

Aside from the dominance of the tele-operating systems, other robotic devices allow for HRI in the medical field. Their superiority in accuracy and their sturdy mechanical structure is exploited in neurosurgery and orthopedic tasks. Such example is the Acrobot Sculptor (Stanmore Implants Worldwide Ltd, UK) which is a robotic device that is controlled by "hands-

on" physical interaction with the surgeon. Such HRI is often referred as shared control. Acrobot Sculptor enables MIS in certain orthopaedical surgeries. This system will be discussed in detail later [8].

1.1.4 Exoskeletons

Exoskeletons are particular class of HRI driven machines. They are a kinematic chain of links and joints built to mimic the extremities of the human skeleton. The mechanical structure can be custom-made for specific set of skeleton parameters, or it can implement certain mechanism for adjustment to the user.

Such machine is attached to the human body so that the extremities of the user are attached to the appropriate link of the exoskeleton. This way, the joints of the exoskeleton mimic the movements of the human joints, in a simplified manner. Due to gravity compensation and adjusted virtual impedance, the impact on the user's intuitiveness is minimized.

Exoskeletons act as an amplifier to the motorial capabilities of the human body. It mimics the human movements but amplifies the forces and torques exerted by the user. Sensory equipment along the exoskeleton estimates the human intention and mimics it, modifying certain dynamical parameters according to a pre-defined rule.

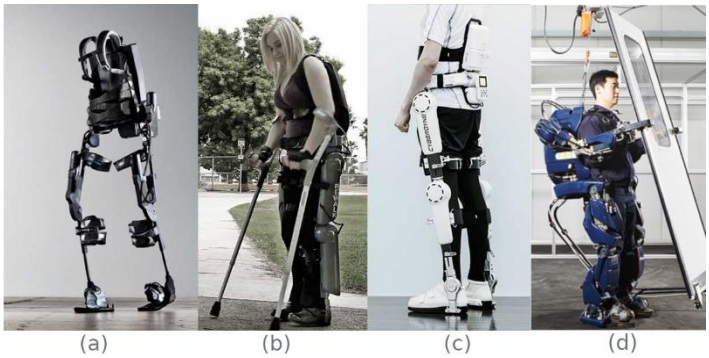


Figure 5. (a) Exoskeleton. (b) Exoskeleton in medical use. (c) Exoskeleton in commercial use. (d) Exoskeleton in manufacturing. [11] [12]

These types of robots only recently have been put in commercial use. Their capability to virtually amplify the muscular power of the human body is used in medicine, the military, and manufacturing.

1.2 Architectural concept of HRI

Foundation of HRI is establishing and maintaining channels of information flow between the human and the robotic system. Such channels define the type of data and the direction of flow e.g. positional information from the user to the robot, visual information from the robot to the user, etc. The technology used to capture and transfer the data depends on the required performance, which is specified according to the objective of the task. E.g. in certain

cases, tracking the human body with inertial measurements might be more appropriate then optical tracking.

Once the data is acquired, it is interpreted according to the overall objective of the system. Systems that have safety as unique objective may estimate the distance between the human and the robot in order to tune its velocity or initiate complete stop. Other systems may estimate the configuration of the human body joint angles in order to calculate the desired cartesian/joint space position of a slave device.

After the information is extracted from the data, the system's controller decides how to act to satisfy its goal. The difference between the actual state and the reference state is referred as the error signal. Even though every type of controller has its own algorithm for calculating decisions, the error is usually common variable used in the calculation. The robot, and potentially other actuators, execute the controller's decision.

The functioning concept behind a controller is general, but it may be specifically tuned to the task. Generally, all controllers allow for setting limits on their outputs. Even though in manufacturing such limits are usually constant, in other use cases like medicine, such limits may vary depending on pre-defined rule or on manual user's inputs. I.e. virtually adjusting the workspace of the robot in order to protect certain vital areas of the body. Such limits affect controller's decision.

Brief overview of common well-developed controllers, and their use in HRI is done next in order to point out the direction of improvement proposed with this work.

1.2.1 Mimetic position control

The goal of mimetic position control is to closely mimic the movements of the human body. This control structure can be described as a special case of conventional position control

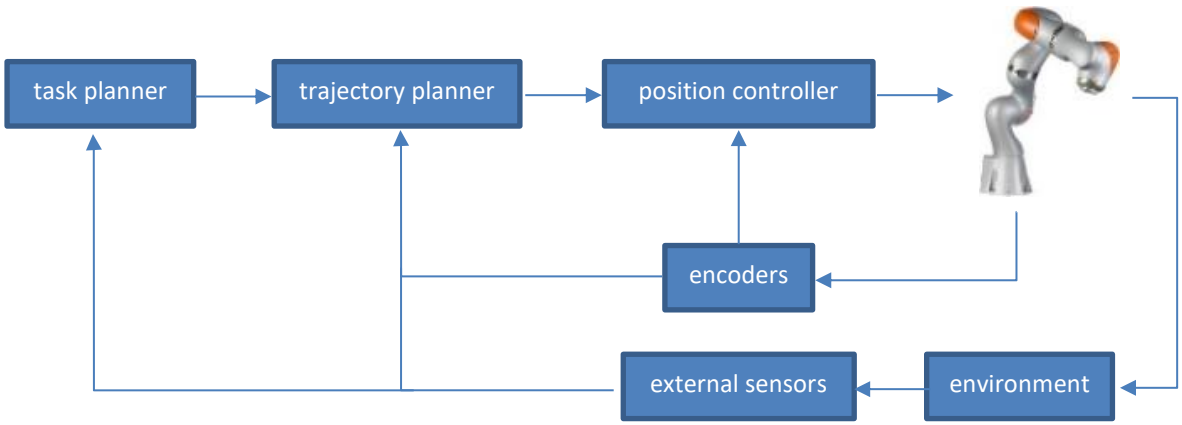


Figure 6. Conventional position control design.

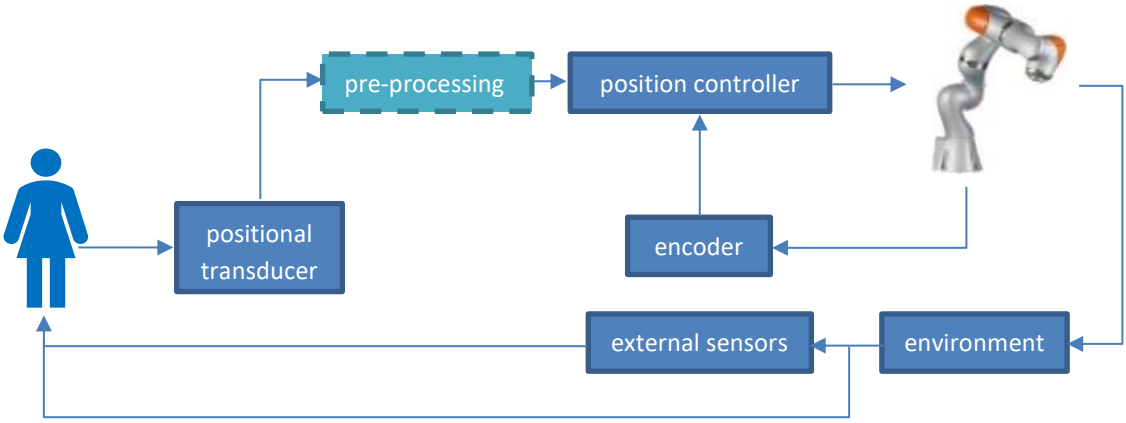


Figure 7. Mimetic position control design.

design. In this case, the functionalities of the task and trajectory planners are substituted by a human. This difference is shown on Figures 6 and 7.

Human intelligence does the function of the task planner, and the need for trajectory planning does not exist because the motion dynamics are implicitly specified in the movement of the human body. Depending on the context of use, the user may be able to directly see the changes of the environment done by the robot, or she/he can monitor it through external sensors.

In this control scheme, the channel that enables information flow from the human to the system is represented by a positional transducer. Therefore, the data type is positional, and the direction Is from the human towards the robot. Mimicking can be done in cartesian space or directly in joint space. When position control is applied to cartesian space, the problem of inverse kinematics must be taken into consideration. Due to the dexterity of human extremities (the arm for example), there might be multiple solutions of the inverse kinematics. This may lead to mimicking the position of the endpoint of the human extremity but maintaining difference in the joint positions.

In the case of cartesian space control, the position of certain points of interest on the human body is tracked. Usually these are the endpoints of the extremities i.e. hand or foot.

Joint angles of the human body are either calculated by inverse kinematic or directly monitored in the case of joint space control. Then, their values are pre-processed. This includes filtering out tremor, applying limits, buffering, etc. Processed data is taken as an input to a conventional position feedback control loop consisting of position controller, the robot as a controlled object, and feedback that provides the controller with the actual values of the robot's joint angles.

The position controller implements the actual control algorithm. It outputs the torques that shall be applied to the robot's joints in order to minimize the position error signal. If decentralized control is utilized, one example of control algorithm is PID. If the control is

centralized, model-based algorithms like PD + gravity compensation may be implemented. Also, various customizations and augmentations of such algorithms are applicable.

Purely position controllers monitor, compare, and control only the position of robot. Dynamical parameters of the robot's mechanical structure, force and torques exerted, are not part of the control loop. They may be kept constant or varying according to certain pre-defined rule.

1.2.2. Impedance control

Another type of control used in industry is the impedance control. The end-effector of any robot can be described as a composition of basic mass-spring-damper systems. Such mechanical systems are governed by the relation $M_d\ddot{x} + D\dot{x} + Kx = f$, where M_d is the mass of the body, D is the damping coefficient of the damper, K is the stiffness coefficient of the spring, and x is the position of the mass along the x-axis.

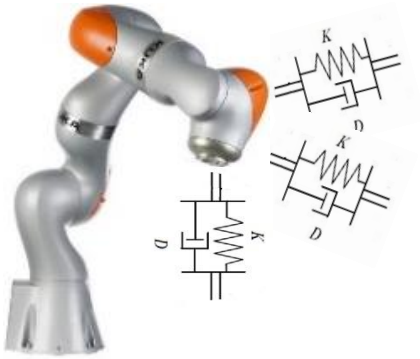


Figure 8. The end-effector can be observed as a composition of basic, one dimensional, mass-spring-damper systems.

The relation governing the end-effector in this case is:

$$\boldsymbol{M\ddot{x} + D\dot{x} + Kx = f} \quad [1]$$

Where:

- $\boldsymbol{M}$ is mass matrix,
- $\boldsymbol{D}$ is damping matrix,
- $\boldsymbol{K}$ is stiffness matrix,
- $\boldsymbol{x}$ is vector of cartesian position of the end-effector, and
- $\boldsymbol{h}$ is the vector of the resultant force applied at the end-effector.

$\boldsymbol{M}$, $\boldsymbol{D}$ and $\boldsymbol{K}$ matrices depend on the mechanical structure of the robot.

According to majority of textbooks on basic robotics, we can represent the dynamics of an anthropomorphic manipulator by using the following well known model:

$$\boldsymbol{B(q)\ddot{q} + C(q,\dot{q})\dot{q} + g(q) = \tau - J^T(q)f} \quad [2]$$

Where:

- $\boldsymbol{q}$ is the joint coordinate vector,
- $\boldsymbol{B}(\boldsymbol{q})$ is the inertia matrix of the manipulator,
- $\boldsymbol{C}(\boldsymbol{q}, \dot{\boldsymbol{q}})$ are the centrifugal and Coriolis terms,
- $\boldsymbol{g}(\boldsymbol{q})$ is the gravitational term,
- $\boldsymbol{\tau}$ is a vector representing the actuation torques in the joints,
- $\boldsymbol{J^T}(\boldsymbol{g})$ is the transpose of the Jacobian, and
- $\boldsymbol{f}$ is the vector of the contact forces and moments applied by the end-effector on the environment.

Classical impedance control assumes that by adopting appropriate form of inverse dynamics control, as:

$$\boldsymbol{\tau} = \boldsymbol{B}(\boldsymbol{q})\boldsymbol{y} + \boldsymbol{C}(\boldsymbol{q}, \dot{\boldsymbol{q}})\dot{\boldsymbol{q}} + \boldsymbol{g}(\boldsymbol{q}) \quad [3]$$

We can obtain completely decoupled control system governed by the relation [13]:

$$\boldsymbol{M}_d\ddot{\tilde{\boldsymbol{x}}} + \boldsymbol{D}_d\dot{\tilde{\boldsymbol{x}}} + \boldsymbol{K}_d\tilde{\boldsymbol{x}} = \boldsymbol{f}_A \quad [4]$$

Where:

- $\tilde{\boldsymbol{x}} = \boldsymbol{x}_d - \boldsymbol{x}$.
  $\boldsymbol{x}_d$ is the vector of desired cartesian position, and $\boldsymbol{x}$ is the actual cartesian position.
- $\boldsymbol{M}_d$ is virtual mass matrix,
- $\boldsymbol{D}_d$ is virtual damping matrix,
- $\boldsymbol{K}_d$ is virtual stiffness matrix, and
- $\boldsymbol{f}_A$ is a vector of the generalized forces.

This way, the dynamic model of the manipulator has been converted to a mechanical impedance system with the corresponding "virtual" parameters $\boldsymbol{M}_d$, $\boldsymbol{D}_d$, and $\boldsymbol{K}_d$ which can be arbitrarily chosen by the user.

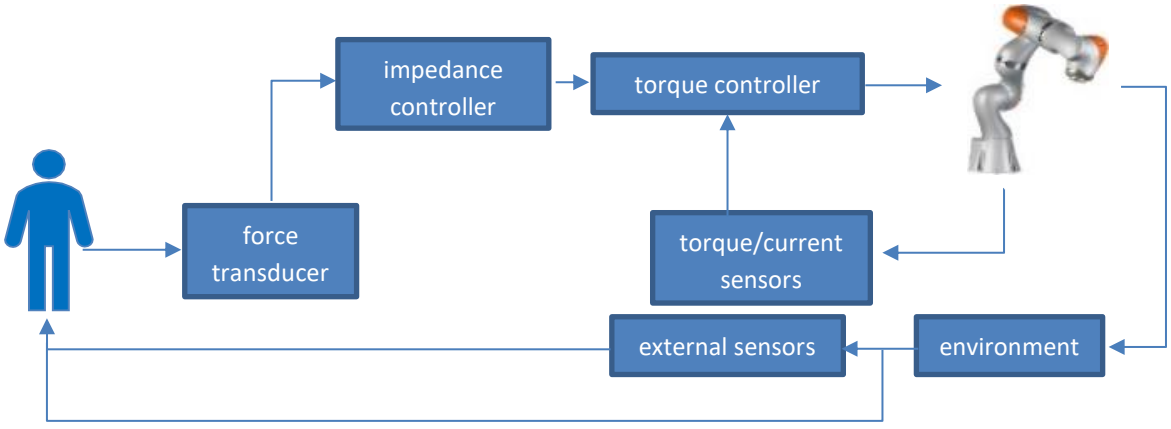


Figure 9. Impedance control design.

Selecting appropriate values for these three matrices allows to the user to transform the behavior of the end-effector to a mechanical system behaving like generalized mass-spring-damper system with known parameters. Therefore, one can use this approach to convert the end-effector behavior to a behavior of an object with relatively small mass, damping, and stiffness, easily bearable by a human operator. This allows for easy movement of the end-effector.

In this control scheme, the channel that enables information flow from the user to the robot is represented by force transducer. Therefore, the data type is force and the direction is from the user to the robot. Force information is processed by the impedance controller that calculates the required torques to be applied on the robot joint's angles. The torque control loop applies torque to the joints of the robot so that the new error signal is minimized. Torque control loop is basic control mechanism that exists in all robotic control schemes. However, usually it is nested inside the position or velocity control loop, and therefore it is omitted in the previous illustrations.

Impedance control can be divided in two categories:

1. *Constant impedance control*. In this case, the inverse dynamics equation to calculate the torques is chosen such that the final decoupled system is governed by the relation:

$$\overline{\boldsymbol{M}_d}\,\ddot{\tilde{\boldsymbol{x}}} + \overline{\boldsymbol{D}_d}\dot{\tilde{\boldsymbol{x}}} + \overline{\boldsymbol{K}_d}\tilde{\boldsymbol{x}} = \boldsymbol{f}_{A_{(t)}} \quad [5]$$

   Where $\overline{\boldsymbol{M}_d}, \overline{\boldsymbol{D}_d}$, and $\overline{\boldsymbol{K}_d}$ denote constant values for the inertia, damping, and stiffness matrix.

   Constant impedance control of this kind Is used in the aforementioned "teaching-with-guiding" programming method, as well as for master robot devices in tele-operation.

2. *Variable impedance control.* This control method assumes that $\boldsymbol{M}_d, \boldsymbol{D}_d$, and $\boldsymbol{K}_d$ matrices are not constant. Therefore, the torque setpoints are calculated such that the final decoupled system behavior can be described as:

$$\boldsymbol{M}_{d_{(t)}}\,\ddot{\tilde{\boldsymbol{x}}} + \boldsymbol{D}_{d_{(t)}}\dot{\tilde{\boldsymbol{x}}} + \boldsymbol{K}_{d_{(t)}}\tilde{\boldsymbol{x}} = \boldsymbol{f}_{A_{(t)}} \quad [6]$$

   $\boldsymbol{M}_{d_{(t)}}$, $\boldsymbol{D}_{d_{(t)}}$, and $\boldsymbol{K}_{d_{(t)}}$ vary with time according to pre-defined rule that is convenient for the overall task of the robotic system.

   Variable impedance control results in more complex interaction with the environment, which greatly expands the capabilities of the robot.

1.2.3. Direction of this work

In position control, the impedance of the end-effector is constant, and it has the natural values of the mechanical structure of the robot. Such control performs satisfactorily when the environment of the robot is structured, completely organized, and the task doesn't require significant interaction between the robot and the environment. However, with tasks that include interaction between the robot and its uncertain environment, simple position control may lead to development of undesirably high contact forces that may lead to instability.

To overcome this limitation, a composition of position and impedance control is performed. Constant impedance control protects the system against instabilities to some degree, but it also may limit the accuracy of the task. Therefore, to improve stability, accuracy, and safety, variable impedance control shall be implemented.

The law according to which the impedance parameters $\boldsymbol{M}_{d_{(t)}}$, $\boldsymbol{D}_{d_{(t)}}$, and $\boldsymbol{K}_{d_{(t)}}$ change, depends on the adoption of a specific strategy regarding the task. For example, if safety is the only concern, the impedance parameters may depend on the distance between the human and the robot. As the human approaches the robot, the robot adopts values for $\boldsymbol{M}_{d_{(t)}}$, $\boldsymbol{D}_{d_{(t)}}$, and $\boldsymbol{K}_{d_{(t)}}$ such that will minimize the force exerted from the robot to the user in the case of impact. However, if the task involves continuous physical interaction between the user and the robot, we can observe such situation as a complex system where the two entities are coupled. To make such HRI more human friendly, it is important to design the variable impedance controller considering the human characteristics. The impedance parameters of the robot are governed by the variable impedance controller, and the equivalent parameters of the human arm are controlled by the neuromuscular system of the user.

A real-time procedure to extract information for the impedance parameters of the human arm is presented. As a result, a new channel of information flow from the user to the data can be established. This new type of information is suitable to play major role in how the variable impedance controller calculates the virtual impedance parameters during physical HRI scenario. Such insight might potentially increase accuracy and stability of the interaction. Also, it might be used to couple robot dynamics to reflect human intention more closely and intuitively. Therefore, the aim of this work is to enable higher quality of variable impedance control by introducing human arm impedance estimations based on neuromuscular data.

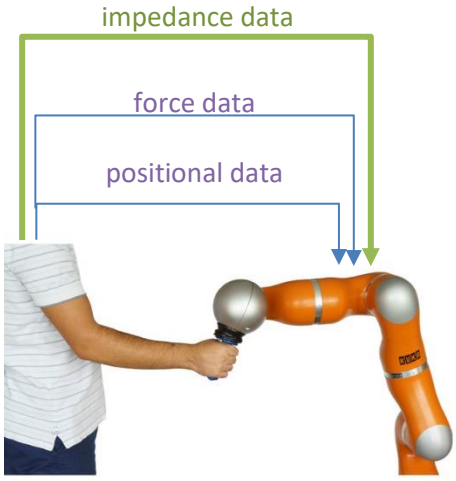


Figure 10. In this work we provide methods to estimate the impedance of the human arm to introduce new type of information transferred to the robot during HRI.

2. State of art

Variable impedance controllers based on intrinsic human dynamics, like arm impedance, haven't transitioned from research to practical use, yet. However, certain medical robots use models to assume the impedance of the human arm relative to the task. Analyzing such robots may give us an insight on the advantages that can potentially be brought by conveying the dynamics of the human arm as a new variable in variable impedance control design.

In the scientific field, few studies propose methods to estimate the impedance of the human arm during physical HRI. However, in this direction, conventional data of position and force is used. Low data diversity leads to significant approximations and limitations. Therefore, the applicableness of such methods in practice is still a challenge.

2.1. In medical use

Acrobot Sculptor is shared control robotic surgery system for Total Knee Replacement (TKR). Specialized in performing high-precision cuts in order to minimize the deviation of the prosthesis from the ideal alignment with the bone. It has spherical construction and relatively small reach of 30-50cm. Light mechanical structure actuated by low-power motor drives allows the surgeon to detect forces during hard bone cutting without force sensor attached to the cutting tool. Additionally, the robot is inherently safe because the potential damage, in case of instability, is limited both in terms of force and area.

The preoperative planning is done with dedicated software fed with appropriate CT scans of the patient's leg. The software generates model of the leg and assists the surgeon in creating model of the perfect prosthesis. Once the fit between the model of the leg and the one of the prosthesis reach certain correctness threshold, the planning software generates the constraint boundaries.

The robot is controlled by variable impedance controller and therefore 6 axes force sensor is attached on the handle at its tip. During surgery, the surgeon guides the robot by pushing on this handle. The law according to which the impedance of the robot is adjusted takes as an

input the distance between the actual position of the tip and the closest point of the boundary generated. This distance is defined as:

$$d = \|\boldsymbol{x} - \boldsymbol{x}_{np}\| = (\boldsymbol{x} - \boldsymbol{x}_{np})\boldsymbol{N}_{np} \quad [7]$$

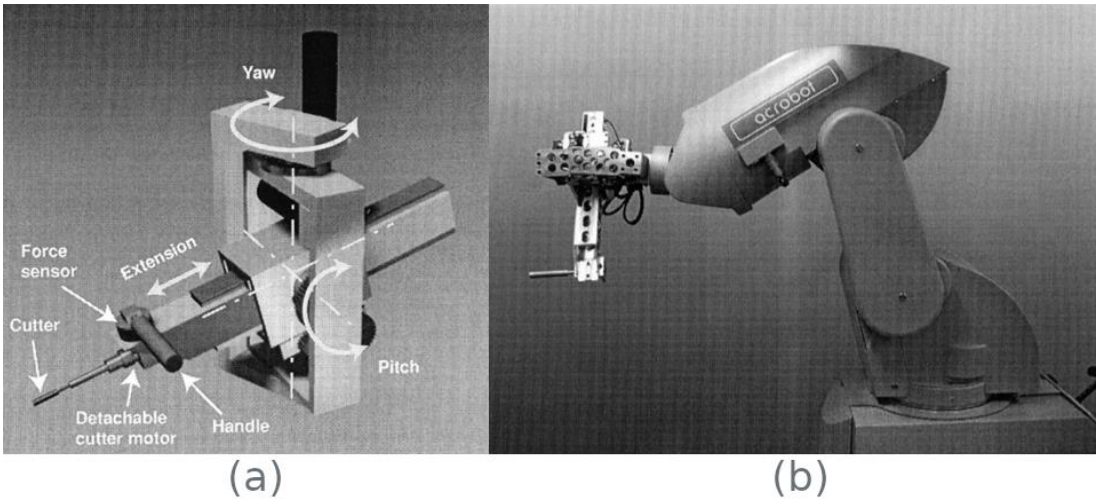


Figure 11. (a) Acrobot's kinematical structure. (b) Acrobot mounted on a gross positioning device. [14]

Where $\boldsymbol{x}$ is the actual robot position, $\boldsymbol{x}_{np}$ is the nearest point of the boundary, and $\boldsymbol{N}_{np}$ is the normal of the boundary pointing inside the region.

The inertia of such light system is negligible, and therefore the torque controller calculates the torques applied to the joints of the robot as:

$$\boldsymbol{\tau} = \boldsymbol{K}_P(\boldsymbol{q}_d - \boldsymbol{q}) + \boldsymbol{K}_D(\dot{\boldsymbol{q}}_d - \dot{\boldsymbol{q}}) + \boldsymbol{\tau}_C + \boldsymbol{f}^*(\boldsymbol{q}, \dot{\boldsymbol{q}}) + \boldsymbol{g}^*(\boldsymbol{q}) \quad [8]$$

Where:

- $\boldsymbol{q}_d$ and $\dot{\boldsymbol{q}}_d$ are the vectors of desired joint coordinates and velocities respectively,
- $\boldsymbol{K}_P$ and $\boldsymbol{K}_D$ are the proportional and derivative gains respectively,
- $\boldsymbol{\tau}_C$ is the guiding force compensation, and
- $\boldsymbol{f}^*(\boldsymbol{q}, \dot{\boldsymbol{q}})$, and $\boldsymbol{g}^*(\boldsymbol{q})$ are friction and gravity compensation respectively.

The variable impedance controller calculates $\boldsymbol{q}_d, \dot{\boldsymbol{q}}_d$, and $\boldsymbol{\tau}_C$ depending on $\boldsymbol{x}_d$,$\dot{\boldsymbol{x}}_d$, and $\boldsymbol{F}_C$:

$$\begin{aligned} \boldsymbol{q}_d &= \boldsymbol{K}^{-1}(\boldsymbol{x}_d) \quad [9] \\ \dot{\boldsymbol{q}}_d &= \boldsymbol{J}^{-1}\dot{\boldsymbol{x}}_d \\ \boldsymbol{\tau}_C &= \boldsymbol{J}^T\boldsymbol{F}_C \end{aligned}$$

Where:

- $\boldsymbol{K}^{-1}$ is the inverse kinematics function,
- $\boldsymbol{J}$ is the Jacobian matrix of the robot, and

- $\boldsymbol{F}_C$ is the compensation force.

The variable impedance controller manipulates with the mechanical impedance of the robot, dominantly stiffness, by adjusting $\boldsymbol{x}_d$, $\dot{\boldsymbol{x}}_d$, and $\boldsymbol{F}_C$**.** This control principle goes under the name of Active constraint control. Three cases are possible:

1. $d > D_1$
   Where D_1 is pre-defined constant describing distance from the boundary. When the distance of the robot d is higher than D_1, the robotic system assumes that the tip of the robot is inside the safe region and constant impedance control is applied. In this case:

$$\boldsymbol{x}_d = \boldsymbol{x} \quad [10]$$
$$\dot{\boldsymbol{x}}_d = A\boldsymbol{F}_G$$
$$\boldsymbol{F}_C = \boldsymbol{0}$$

Where A is the impedance and it is same in all directions, and $\boldsymbol{F}_G$ is the surgeon's guiding force.

2. $D_1 \geq d > 0$
   In this case it is assumed that the tip of the robot exited the safe region and started approaching the boundaries. Two sub-cases are possible:
   a. $\boldsymbol{F}_G \cdot \boldsymbol{N}_{np} \geq 0$, the guiding force points away from the boundary.
      In this subcase $\boldsymbol{x}_d$, $\dot{\boldsymbol{x}}_d$, and $\boldsymbol{F}_C$ stay regulated as in case 1.
   b. Otherwise, the guiding force $\boldsymbol{F}_G$ is split into normal ($\boldsymbol{F}_{GN}$), and tangential ($\boldsymbol{F}_{GT}$) components with regards to the boundary, and

$$\dot{\boldsymbol{x}}_d = A_N\boldsymbol{F}_{GN} + A\boldsymbol{F}_{GT} \quad [11]$$

   where A_N is the impedance in the normal direction and it increases with distance to the boundary:

$$A_N = \frac{D_1}{dA} \quad [12]$$

   Regarding force compensation, another pre-defined constant distance D_2 is defined such as:

$$\boldsymbol{F}_C(d) = \begin{cases} 0, & d > D_2 \\ -\frac{D_2 - d}{D_2}\,\boldsymbol{F}_{GN}, & D_2 > d > 0 \end{cases} \quad [13]$$

3. $d \leq 0$.
   In this case it is assumed that the tip of the robot is dangerously close to the boundary or even out of it, therefore the control gains are high in this region.
   $\boldsymbol{x}_d$ is set to the nearest point of the boundary $\boldsymbol{x}_{np}$ ($\boldsymbol{x}_d = \boldsymbol{x}_{np}$), and there are again two subcases regarding the guiding force:
   a. $\boldsymbol{F}_G \cdot \boldsymbol{N}_{np} \geq 0$, the guiding force points away from the boundary.
      In this subcase $\boldsymbol{x}_d$, $\dot{\boldsymbol{x}}_d$, and $\boldsymbol{F}_C$ stay regulated as in case 1.
   b. Otherwise:

$$\dot{\boldsymbol{x}}_d = A\boldsymbol{F}_{GT} \quad [14]$$
$$\boldsymbol{F}_C = -\boldsymbol{F}_{GN}$$

[14]

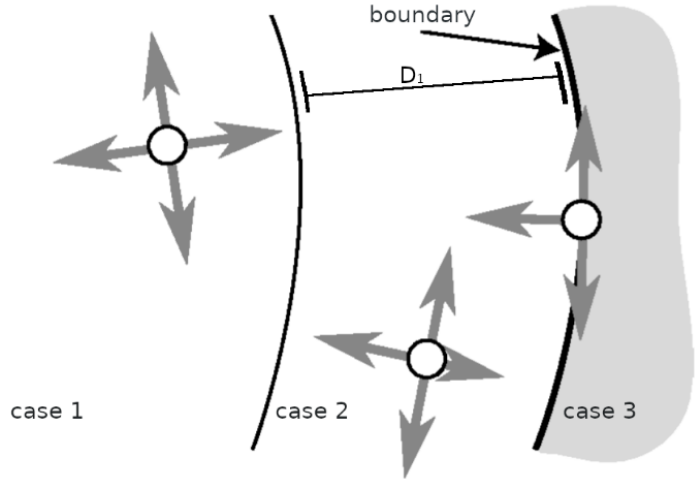


Figure 12. Three different impedance settings according to three different possible cases. [14]

Acrobot's variable impedance controller doesn't estimate the mechanical impedance of the surgeon's arm. However, in the second case its modelling the impedance of the robot similarly to what human would naturally do. Increasing the impedance when approaching the boundary in order to increase accuracy and robustness against undesired disturbances or tremor. However, Acrobot's controller is increasing the impedance only in the direction of the normal of the nearest point. With humans, this is the case only if the boundary Is a straight line (2D case) or plane (3D case). In case of curved boundary, a human would increase the impedance of hers/his arm in multiple directions, therefore:

$$\dot{\boldsymbol{x}}_d = \boldsymbol{A}_{human}\boldsymbol{F}_G \quad [15]$$

Where $\boldsymbol{A}_{human}$ is a diagonal matrix whose elements are in accordance with the actual mechanical impedance of the surgeon's arm. This is presented in Figure 13.

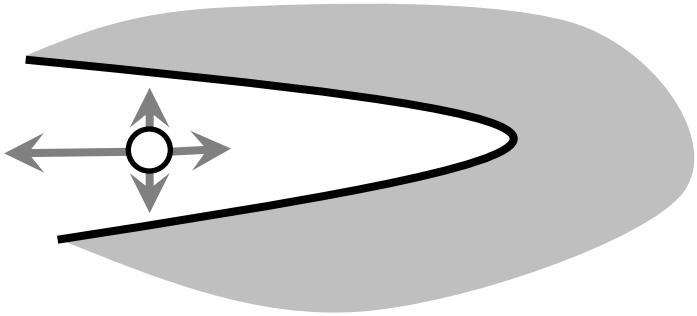


Figure 13. Natural human multidirectional impedance adjustment.

Augmenting variable impedance controllers with information about the dynamical characteristics of the human arm will increase intuitiveness, accuracy, and safety in use cases like Acrobot.

2.2 In research literature

Various offline experimental methods to estimate dynamical parameters of the human arm have been developed. [15] [16] However, majority of them rely on mechanical perturbation applied to certain point on the arm. Perturbations inhibit intuitiveness and may also interfere with the task performed. Therefore, such methods are inapplicable in online scenarios. Online estimation of dynamical characteristic of a human arm in physical HRI scenario has been performed in [17]. In this study, a fully decoupled 3-DOF parallel Tripetron planar robot is used. Additionally, multi-axes force sensor is mounted at the end-effector. During shared control, the user guides the robot by holding the end-effector. This setup is shown in Figure 14.

Figure 14. Tripetron 3 DOF planar robot with multi-axes force sensor attached to its end-effector. [17]

A variable impedance controller is implemented. It transforms the behavior of the robot to the following decoupled system:

$$\boldsymbol{f} = \boldsymbol{M}_d\ddot{\tilde{\boldsymbol{x}}} + \boldsymbol{C}_{d_{vi}}\dot{\tilde{\boldsymbol{x}}} \quad [16]$$

$$\boldsymbol{C}_{d_{vi}} = \boldsymbol{C} - \boldsymbol{S} \quad [17]$$

Where:

- $\boldsymbol{M}_d$ is the virtual inertia matrix,
- $\boldsymbol{C}$ is the constant part of the virtual damping parameter,
- $\boldsymbol{S}$ is a matrix whose elements $s_{ij} = \alpha\delta_{ij}sgn(\dot{x}_i)\dot{f}_i$. Where:
    - $\dot{f}_i$ is the time derivative of the ith component of the force acting on the end-effector,
    - $\dot{x}_i$ is the ith component of the velocity vector,
    - δ_{ij} is the Kronecker delta, and
    - α is a weighting factor.

The general idea behind such choice of the impedance parameters is utilizing the information conveyed by the rate of change of force. Similarly to the approximation of the future values of a function using Taylor expansion, $\dot{\boldsymbol{F}}$ conveys information on the future of $\boldsymbol{F}$ which might reveal the human intention. Therefore, sudden increase of force in the direction of actual velocity (or in case of zero velocity) is interpreted as a human intention to accelerate and the mechanical impedance of the structure is reduced. Vice-versa, sudden decrease of force is interpreted as human intention to decelerate, stop, or change direction, so the impedance is increased.

In order the system to be stable, an equation of the highest possible virtual impedance is found. Out of this impedance, the damping is utilized as a control parameter because it is related to the dissipation of energy. The critical damping is:

$$\boldsymbol{C}_{d_c} = -\frac{1}{2T}(\boldsymbol{M}_d + T\boldsymbol{C}_h) + \frac{1}{2T}\sqrt{(\boldsymbol{M}_d + T\boldsymbol{C}_h)^T(\boldsymbol{M}_d + T\boldsymbol{C}_h) + 4T^2\boldsymbol{K}_h\boldsymbol{M}_d} \quad [18]$$

Where $\boldsymbol{C}_h$ and $\boldsymbol{K}_h$ are the human nxn damping and stiffness matrices respectively, and T is the time delay. Therefore, in order to check for stability, the impedance of the human arm is estimated online.

Neglecting inertia and damping, the impedance of the human arm is modelled as:

$$\boldsymbol{K}_h\boldsymbol{x} + \boldsymbol{B} = \boldsymbol{f} \quad [19]$$

Where $\boldsymbol{B}$ is nxm offset matrix, which is generally not constant, and it depends on the human desired motion. Estimating $\boldsymbol{K}_h$ as a solution to this system might be made directly. However, in order to avoid obtaining singular system it is preferable to minimize the square of the errors given by the estimation of the matrix $\boldsymbol{K}_h$. This way, $\boldsymbol{K}_h$ is calculated as:

$$\boldsymbol{K}_h = (\boldsymbol{f} - \boldsymbol{B})\boldsymbol{x}^T(\boldsymbol{x}\boldsymbol{x}^T)^{-1} \quad [20]$$

Where $\boldsymbol{B}$ is approximated using linear regression as:

$$\boldsymbol{B}_{[nxm]} = \left(\bar{\boldsymbol{f}} - \boldsymbol{V}^{-1}\boldsymbol{\Sigma}\bar{\mathbf{x}}\right)_{[nx1]}\mathbf{1}_{[1xm]} \quad [21]$$

In this equation, $\bar{\boldsymbol{f}}$, and $\bar{\mathbf{x}}$ are the vectors of the n means value for the m past samples. $\boldsymbol{V}$ and $\boldsymbol{\Sigma}$ are nxn diagonal matrices of the variance of n position signals and the covariance between the latter and the n force components.

The robot is working in n dimensional space, however it is considered impossible to have significant measurements in all n dimensions. This may lead to singularity of $\boldsymbol{x}$ or $\boldsymbol{f}$ which poses difficulties in calculating $(\boldsymbol{x}\boldsymbol{x}^T)^{-1}$. E.g. if the motion is only translational (or only rotational) or it is constrained in a plane, then it is probable that $\boldsymbol{x}$ or $\boldsymbol{f}$ will be singular. In this study, this issue is solved by dimensionality reduction. All coordinates along which the variance of the measurements doesn't satisfy certain threshold are discarded. Which is to say, nxn diagonal matrix $\boldsymbol{H}$ is built according to the following rule:

$$h_i = \begin{cases} 0, & V_{i,i} - v_{thr} < 0 \\ 1, & V_{i,i} - v_{thr} \geq 0 \end{cases} \quad [22]$$

That results with a matrix that has value 1 only for the elements that represent coordinates along which the variance is greater than the threshold v_{thr}. For example, if the motion is in $x - y$ plane, and the vector $\boldsymbol{x}$ has the common order of elements $\boldsymbol{x} = [\mathrm{x\ y\ z}]^T$, then the matrix $\boldsymbol{H}$ would be:

$$\boldsymbol{H} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

If new matrix $\boldsymbol{G}$ is constructed by discarding the zero columns of $\boldsymbol{H}$, e. g $\boldsymbol{G} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}^T$ it can be used to reduce the dimensionality of $\boldsymbol{K}_h$ estimation and therefore avoiding the problems posed by eventual singularity. Therefore, the estimation equation becomes:

$$\boldsymbol{K}_h = \boldsymbol{G}^T[(\boldsymbol{f} - \boldsymbol{B})\boldsymbol{x}^T]\boldsymbol{G}(\boldsymbol{G}^T\boldsymbol{x}\boldsymbol{x}^T\boldsymbol{G})^{-1} \quad [23]$$

[17]

2.2.1. Limitations

This method models the human arm impedance as a simple sum of stiffness and a bias parameter neglecting the effects of inertia and damping. Such approximation inevitably leads to overestimation of the stiffness parameter. Additionally, since only force and positional data is used to estimate the impedance, the information of the human arm joint angles is not taken into consideration. It is shown in [18] that the configuration of the limbs of the human arm plays a role in the shape of the stiffness. The approach to deal with singularities in $\boldsymbol{x}$ and

$\boldsymbol{f}$ poses another major limitation to this method. First, no rule for calculation of the variance threshold is proposed and validated. This results with choosing v_{thr} by trial and error which can yield result peculiar to a specific task and specific hardware setup. Second, the dimensionality reduction may lead to reducing a coordinate along which potential disturbances are possible. Therefore, losing the advantage of robustness along that particular direction. For example, in case when the robot's workspace is three dimensional, but the task performed temporarily requires motion only along two coordinates. In such situation, the reduction of dimensionality will discard one coordinate, even though certain disturbances may appear along that very same discarded direction.

This method may be used only in scenarios that satisfy the very particular setup assumed by the study, and tasks with relatively low safety requirements. Therefore, we find it useful to augment the estimation of dynamic properties of the human arm by introducing easily accessible biological signals of neuromuscular nature.

3. Human arm stiffness

We usually use our hands to interact with different environments by moving and manipulating objects while performing certain tasks. By doing so, we continuously create and abandon mechanical systems where the arm and the environment are coupled. Depending on the task, these mechanical systems might be inherently unstable, leading to destabilization of the arm posture. However, the central nervous system regulates the mechanical properties of the arm compensating for the instability of the coupled system and ensuring stable arm posture. [19] Therefore, the regulation of mechanical properties of a human arm improves the robustness characteristic of the interaction with the environment.

One such example is the task of drilling a hole into a material using hand drill, as shown on Figure 15 (a).

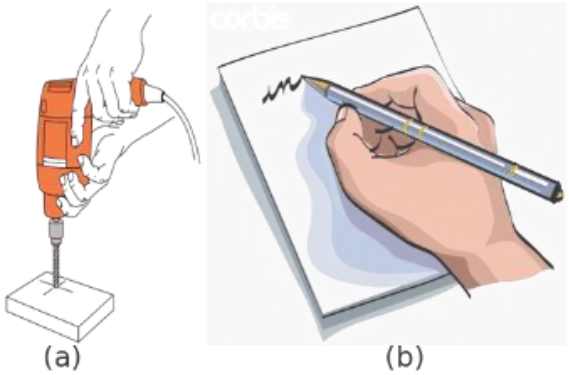


Figure 15. Examples where regulation of mechanical properties of the arm has significant role. (a) Drilling. (b) Handwriting.

In the case of drilling, the uncertainty of unknown or irregular density of the material might act as a disturbance with potential to destabilize the coupled system. Therefore, the central nervous system regulates the stiffness of the human arm in order to protect against such event. Even though the stiffness is highest along the direction of drilling, its magnitude is also increased in the other directions to compensate any disturbances that might emerge along the normal axis, such as natural tremor.

Another illustrative example of natural regulation of mechanical dynamics of the human arm Is handwriting. During handwriting, we guide the tip of the pen to follow imaginary path which represents the desired character. Leaving this path is considered instability of the coupled system. Therefore, the stiffness of the whole arm, including the hand, must be increased in a such a way to prevent deviation from the path. This is common case when we guide tools in pre-defined boundaries. Such is the case with the aforementioned Acrobot Sculptor TKR robotic system too. In these cases, the increased robustness due to regulation of dynamical properties of the arm results in increased accuracy of the task.

3.1. Definition of stiffness of the human arm

The human arm joint can be described as mechanical impedance system. Model of such system follows the form of basic mass-spring-damper system but is adopted to the rotational case.

Therefore, the mechanical impedance model of the human arm in joint space has the form:

$$\boldsymbol{M}_{j_{(t)}}\,\ddot{\boldsymbol{q}} + \boldsymbol{D}_{j_{(t)}}\dot{\boldsymbol{q}} + \boldsymbol{K}_{j_{(t)}}\boldsymbol{q} = \boldsymbol{\tau}_{(t)} \qquad [\,24\,]$$

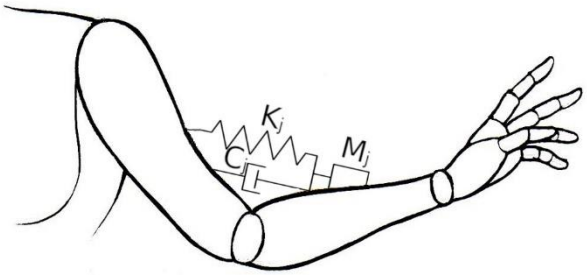


Figure 16. Human joint described as impedance system.

Where:

- $\boldsymbol{M}_{j_{(t)}}$ is joint inertia matrix,
- $\boldsymbol{D}_{j_{(t)}}$ is joint damping matrix,
- $\boldsymbol{K}_{j_{(t)}}$ is joint stiffness matrix, and
- $\boldsymbol{q}$ and $\boldsymbol{\tau}$ are the vectors of joint angles and torques, respectively.

From practical point of view, measuring human joint angles and/or joint torques is far more complicated then measuring the spatial position and forces applied on a point fixed to the arm. In this work we chose such a point to be at the end of the forearm, at the wrist. We also refer to this point as the endpoint. The mechanical impedance of the human arm in the

endpoint can be described as a composition of basic mass-spring-damper systems in cartesian space. Similar to the description of the robot's end-effector in Chapter 1.2.2.

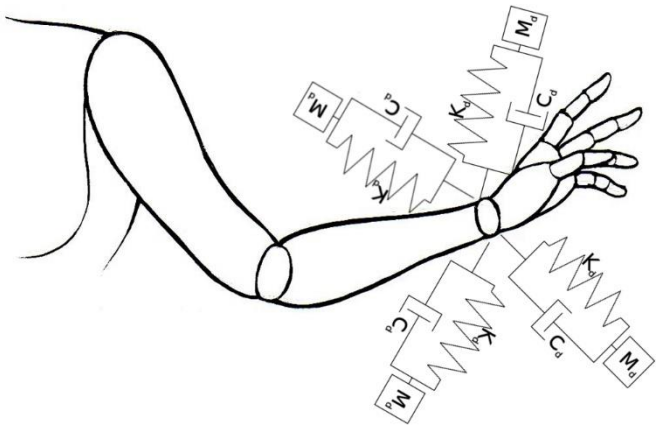


Figure 17. The wrist of the human arm observed as a composition of basic mass-spring-damper systems.4

In this case, the mechanical impedance of the arm in cartesian space is:

$$\boldsymbol{M}_{c_{(t)}}\,\ddot{\boldsymbol{x}} + \boldsymbol{D}_{c_{(t)}}\dot{\boldsymbol{x}} + \boldsymbol{K}_{c_{(t)}}\boldsymbol{x} = \boldsymbol{f}_{c\,(t)} \quad [\,25\,]$$

Where $\boldsymbol{M}_c$, $\boldsymbol{D}_c$, and $\boldsymbol{K}_c$ are the cartesian inertia, damping, and stiffness matrices, $\boldsymbol{x}$ is the vector of spatial position of the wrist, and $\boldsymbol{f}_c$ is the vector of spatial forces acting at the endpoint.

$\boldsymbol{M}$, $\boldsymbol{D}$, and $\boldsymbol{K}$ in both joint space and cartesian space are generally not constant as they are continuously changed by the central nervous system.

It has been shown that the effect of the cartesian inertia matrix $\boldsymbol{M}_c$ is negligible [20]. Additionally, the damping matrix $\boldsymbol{D}_c$ largely depends on the velocity of movement of the endpoint. Its manipulability by the central nervous system is limited and it can be safely neglected. On the other hand, the stiffness matrix $\boldsymbol{K}_c$ plays a major role in the mechanical impedance of the endpoint and it is predominantly controlled by the central nervous system. This makes the stiffness matrix $\boldsymbol{K}_c$ the obvious variable of choice to satisfactorily describe the human arm impedance.

If we assume two dimensional workspace, the cartesian stiffness $\boldsymbol{K}_c$ can be represented graphically by the following parametric equation of an ellipse:

$$\begin{bmatrix} f_x^K(t) \\ f_y^K(t) \end{bmatrix} = K_c \begin{bmatrix} \cos(t) \\ \sin(t) \end{bmatrix}; \quad 0 < t < 2\pi \quad [\,26\,]$$

Where $f_{x,y}^K(t_1)$ is the elastic component of the force in response to $[\cos(t_1)\, sin(t_1)]^T$ displacement along x or y axis.

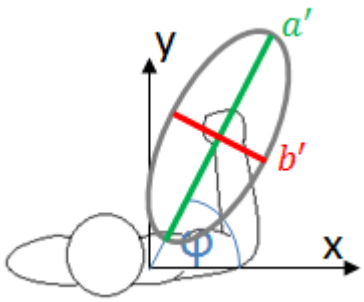


Figure 18. Graphical representation of the cartesian stiffness matrix as an ellipse.

Intuitively, the orientation (ϕ) of the major axis (a') denotes the direction along which the stiffness is highest, and the arm is most resistant to disturbances. And the orientation$(\frac{\pi}{2} + \phi)$ of the minor axis (b') denotes the direction along which the stiffness is lowest, and the arm is least resistant to disturbances. The area of the ellipse denotes the magnitude of the cartesian endpoint stiffness. [21]

This graphical representation can be extended to the three dimensional case too.

3.2. Central nervous system strategy to endpoint stiffness modulation

The central nervous system modulates the dynamical properties of the arm endpoint in different ways out of which most dominant strategies are:

- Cocontracting muscle groups acting on the limb [22],
- adaptation in the sensitivity of the reflex feedback [23], and
- selective control of the limb configuration [18].

Several factors of neuromotor control including:

1. The dominant contribution of the limb geometry to efficient modulation of the ellipse orientation ϕ,
2. the ergonomic efficiency of postural adjustments compared with cocontractions, and
3. the existence of cross-joint muscles in limbs,

lead to dominant use of the arm configuration to modulate the stiffness ellipse orientation ϕ.

On the other side, plethora of literature suggests the existence of synergistic relationship between the muscular activities of mono- and bi-articular muscular activities. These activities tend to produce coordinated stiffening profile across the all the arm joints [24] [25] [26]. Therefore, it can be safely assumed that humans tend to dominantly use cocontractions to modulate the area of the ellipse. [15]

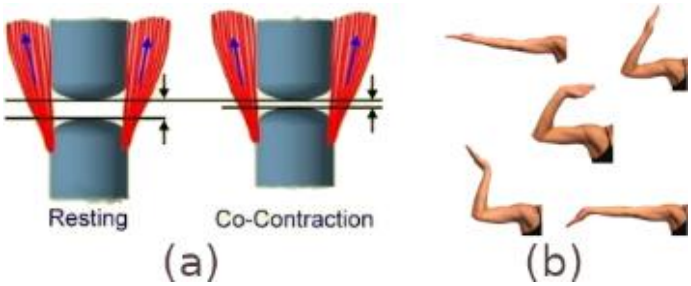


Figure 19. (a) Joint muscles resting and cocontracting to modulate stiffness ellipse area. (b) Different limb posture for different orientation of the stiffness ellipse.

3.3 Electromyography signals (EMG)

Muscular cocontraction is defined as the simultaneous contraction of agonist and antagonist muscles around a joint. [27] Therefore, it is justified to assume that a reliable measure of muscular activity with a proper method may convey information on the level of muscular cocontraction. Which in addition, may be related to the area of the stiffness ellipse. Such measure of muscular activity is the Electromyography signal (EMG).

EMGs are biological signals consisted of electrical currents generated in muscle tissue during its activity. Such currents are forced by electrical potentials called Action potentials. The nervous system always controls the muscle activity. Therefore, we can utilize EMGs to characterize neuromuscular activity in a specific muscle fiber.

However, measuring EMG signal of a specific muscle fiber needs intrusive probes in the form of a needle or wire. Such measuring equipment Is out of the scope of this work. Fortunately, we can acquire EMG signals by mounting electrodes on the surface of the skin, acquiring what is called, surface EMGs (sEMGs). In this case, the signal is a composition of all electrical potentials generated by a set of nearby muscle fibers. If this set consists of all muscle fibers of a single motor unit, then the composition of potentials is called Motor Unit Action Potential (MUAP). [28]

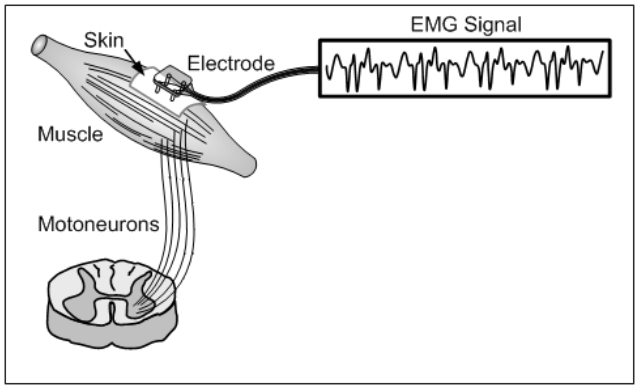


Figure 20.Conceptual diagram of sEMG acquisition. [29]

Due to simplicity, the terms sEMG and EMG will be used interchangeably throughout the text.

4. Materials and methods

4.1. Least squares identification method

A method to estimate the stiffness of the human arm in tele-impedance scenario is proposed in [15]. In this work, an effort has been done to adopt it and test it to the physical HRI scenario of shared control.

Biomechanical observations point to the conclusion that the human muscle-tendon can be modelled as a separate mechanical impedance system as shown in Figure 21. [30]

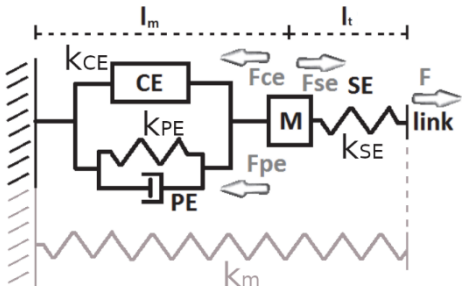


Figure 21. Hill's model of a muscle-tendon as mechanical impedance. [31]

Where:

- CE is the Contractile Element which generates force due to neuromuscular activation signal,
- PE is the Parallel Element which models the spring-damper properties originating from connective tissues within the muscle,
- SE is the Series Element which models the behavior of the tendon,
- M is muscular mass,
- l_m and l_t are the lengths of the muscle and the tendon respectively,
- $k_{SE}(\boldsymbol{q})$ and $k_{PE}(\boldsymbol{q})$ are the stiffness coefficients of the SE and PE and they depend on the joint angle vector **q**.
- $k_{CE}(p)$ is the stiffness coefficient of the elastic effect of the CE and it depends on the muscular activity p,

- F_{CE}, F_{SE}, and F_{PE} are the reaction forces of the CE, SE and PE respectively,
- F is the resultant external force acting on the muscle-tendon system, and
- $k_m(p, \boldsymbol{q})$ is a representation of the overall stiffness of the individual muscle.

According to this model, the overall stiffness of the muscle $k_m(p, \boldsymbol{q})$ can be calculated by superimposing the effects of $k_{SE}(\boldsymbol{q})$, $k_{PE}(\boldsymbol{q})$, and $k_{CE}(p)$. However, if the joint angle is not nearby its limits, then $k_{SE}(\boldsymbol{q}) \gg k_{CE}(p)$ so $k_{SE}(\boldsymbol{q})$ can be approximated as rigid link. In addition, under this condition $k_{PE}(\boldsymbol{q}) \ll k_{CE}(p)$ too, so $k_{PE}(\boldsymbol{q})$ can be neglected. [32]

Therefore, $k_{CE}(p)$ can be considered a reliable approximation of the overall muscle stiffness $k_m(p, \boldsymbol{q})$:

$$\hat{k}_m(p, \boldsymbol{q}) \approx \hat{k}_m(p) = k_{CE}(p) \quad [27]$$

By (i) implementing this approximation to the Hill's model of the muscle-tendon system, and (ii) observing the human arm as a chain of such dynamical systems, we could describe a mechanical impedance system in the muscle space. Such system would have stiffness matrix $\widehat{\boldsymbol{K}}_M(\boldsymbol{p})$. Where $\boldsymbol{p}$ is the vector of muscular activities of all considered muscles. The transformation between this muscle space and the joint space is performed by the Muscular Jacobian $\boldsymbol{J}_M(\boldsymbol{q})$. Therefore, we can relate the muscle stiffness matrix $\widehat{\boldsymbol{K}}_M(\boldsymbol{p})$ and the joint stiffness matrix $\widehat{\boldsymbol{K}}_j(\boldsymbol{p}, \boldsymbol{q})$ by:

$$\widehat{\boldsymbol{K}}_j(\boldsymbol{p}, \boldsymbol{q}) = \boldsymbol{J}_M^T(\boldsymbol{q})\widehat{\boldsymbol{K}}_M(\boldsymbol{p})\boldsymbol{J}_M(\boldsymbol{q}) \quad [28]$$

The Muscular Jacobian $\boldsymbol{J}_M(\boldsymbol{q})$ will be discussed in detail later.

On the other hand, the transformation between the joint space and the cartesian space is made by the Arm Jacobian $\boldsymbol{J}(\boldsymbol{q})$. This Jacobian is equivalent to the analytical Jacobian used in conventional robotics to map between the joint space and the cartesian space. If we model the human arm as a kinematic chain of bodies and joints, same way we model robots, the Jacobian of that model would be Arm Jacobian $\boldsymbol{J}(\boldsymbol{q})$. It relates the velocities from the joint space of the human arm, to the cartesian space by $\dot{\boldsymbol{x}} = \boldsymbol{J}\dot{\boldsymbol{q}}$ where $\boldsymbol{x}$ is the cartesian pose of the wrist, and $\boldsymbol{q}$ is the vector of joint angles of the human arm.

Using the Arm Jacobian $\boldsymbol{J}(\boldsymbol{q})$, we can relate the cartesian endpoint stiffness matrix and the joint stiffness matrix by:

$$\boldsymbol{K}_j(\boldsymbol{p}, \boldsymbol{q}) - \boldsymbol{G}_j(\boldsymbol{q}) = \boldsymbol{J}^T(\boldsymbol{q})\boldsymbol{K}_c(\boldsymbol{p}, \boldsymbol{q})\boldsymbol{J}(\boldsymbol{q}) \quad [29]$$

Where $\boldsymbol{G}_j(\boldsymbol{q})$ represents the effect of arm geometry due to the presence of external load and gravity:

$$\boldsymbol{G}_J(\boldsymbol{q}) = \frac{\delta \mathbf{J}^{\mathrm{T}}(\boldsymbol{q})\boldsymbol{f}_0}{\delta \boldsymbol{q}} + \frac{\delta \boldsymbol{\tau}_g(\boldsymbol{q})}{\delta \boldsymbol{q}} \quad [30]$$

$\boldsymbol{f}_0$ is the vector force of the external load, and $\boldsymbol{\tau}_g(\boldsymbol{q})$ is the effect of gravity. Due to assumption of tele-impedance scenario in [15], the effects of the external load and gravity are neglected. Therefore:

$$\boldsymbol{f}_0 = \mathbf{0} \text{ and } \boldsymbol{\tau}_g(\boldsymbol{q}) = \mathbf{0} => \boldsymbol{G}_J(\boldsymbol{q}) = 0 \quad [31]$$

$$\boldsymbol{K}_j(\boldsymbol{p},\boldsymbol{q}) = \boldsymbol{J}^T(\boldsymbol{q})\boldsymbol{K}_c(\boldsymbol{p},\boldsymbol{q})\boldsymbol{J}(\boldsymbol{q}) \quad [32]$$

The error between the estimated joint stiffness matrix $\widehat{\boldsymbol{K}}_j(\boldsymbol{p},\boldsymbol{q})$ and the actual stiffness matrix $\boldsymbol{K}_j(\boldsymbol{p},\boldsymbol{q})$ can be defined as the frobenius norm of their difference:

$$e = \left\|\widehat{\boldsymbol{K}}_j(\boldsymbol{p},\boldsymbol{q}) - \boldsymbol{K}_j(\boldsymbol{p},\boldsymbol{q})\right\| \quad [33]$$

if eq.28 and eq. 32 are substituted into eq.33 the error adopts the following form:

$$e = \left\|\boldsymbol{J}_M^T(\boldsymbol{q})\widehat{\boldsymbol{K}}_M(\boldsymbol{p})\boldsymbol{J}_M(\boldsymbol{q}) - \boldsymbol{J}^T(\boldsymbol{q})\boldsymbol{K}_c(\boldsymbol{p},\boldsymbol{q})\boldsymbol{J}(\boldsymbol{q})\right\| \quad [34]$$

Using the DH parameters of the human arm, the Arm Jacobian $\boldsymbol{J}(\boldsymbol{q})$ can be calculated analytically through standard procedures of robot kinematics. In addition, proper modelling of arm muscles allows to compute the Muscular Jacobian for a given vector of joint angles. The computation of $\boldsymbol{J}_M(\boldsymbol{q})$ is discussed in detail later. Therefore, remain two unknown variables, $\widehat{\boldsymbol{K}}_M(\boldsymbol{q})$, and $\boldsymbol{K}_c(\boldsymbol{p},\boldsymbol{q})$.

$\widehat{\boldsymbol{K}}_M(\boldsymbol{p})$ can be calculated through Hill's activation dynamics equations. This procedure takes as an input the level of muscular activity that is approximated by the EMG signals. However, it depends on complex modelling of the human musculoskeletal system. Therefore, it is considered as computationally expensive and not suitable for real-time tasks. On the other side, there exist observations suggesting that the Central Nervous System solves this task in rather elegant, simple and effective manner. Taking into consideration such strategy utilized by the Central Nervous System and the assumption that muscular cocontractions mainly contribute to the area of the stiffness ellipse, a linear model for $\widehat{\boldsymbol{K}}_M(\boldsymbol{p})$ is proposed.

The idea behind such model is to interpret the level of cocontraction as an arithmetic operation of EMG signals. And then, identify the contribution of each muscle to the active variations of the area of the endpoint stiffness ellipse. It is proposed that:

$$\widehat{\boldsymbol{K}}_M(\boldsymbol{p}) = a_{cc}(\boldsymbol{p})\boldsymbol{K}_s \quad [35]$$

$$a_{cc}(\boldsymbol{p}) = c_1 + c_2 I_{coc}(p) \quad [36]$$

$$\boldsymbol{K}_s = \begin{bmatrix} K_{s_1} & 0 & \cdots & 0 \\ 0 & K_{s_2} & \cdots & 0 \\ 0 & 0 & \ddots & 0 \\ 0 & 0 & 0 & K_{s_{n_m}} \end{bmatrix} \quad [37]$$

Where:

- $a_{cc}(\boldsymbol{p})$ models the muscular cocontraction taking into consideration the intrinsic muscle stiffness c_1 and the active gain c_2,
- $I_{coc}(\boldsymbol{p})$ is the cocontraction index. It depends on the number of muscles monitored and can adopt several forms. The form proposed in [15] is simple sum of the EMG signals of the long heads of biceps brachii and triceps brachii.
- $\boldsymbol{K}_s$ is a constant diagonal matrix implementing the contributions of each considered muscle to the active variation of the area of the stiffness ellipse.

Finally, the error can be represented as:

$$e = \left\| \boldsymbol{J}_M^T(\boldsymbol{q}) a_{cc}(\boldsymbol{p}) \boldsymbol{K}_s \boldsymbol{J}_M(\boldsymbol{q}) - \boldsymbol{J}^T(\boldsymbol{q}) \boldsymbol{K}_c(\boldsymbol{p}, \boldsymbol{q}) \boldsymbol{J}(\boldsymbol{q}) \right\| \quad [38]$$

Taking into consideration that c_1, c_2, and the elements of $\boldsymbol{K}_s$ are to be identified, $\boldsymbol{K}_c(\boldsymbol{p}, \boldsymbol{q})$ remains as the only unknown variable. This matrix is acquired experimentally as explained in the following Chapter. Therefore $n_m + 2$ parameters should be identified by minimizing the error using Least Squares method.

Once these parameters are identified, the joint stiffness matrix can be calculated as:

$$\boldsymbol{K}_j(\boldsymbol{p}, \boldsymbol{q}) = \boldsymbol{J}_M^T(\boldsymbol{q}) a_{cc}(\boldsymbol{p}) \boldsymbol{K}_s \boldsymbol{J}_M(\boldsymbol{q}) \quad [39]$$

The inputs and outputs of this method are summarized in Table 1.

The joint stiffness matrix can be transformed to the cartesian space, thus resulting with the cartesian endpoint stiffness matrix by:

$$\boldsymbol{K}_c(\boldsymbol{p}, \boldsymbol{q}) = \boldsymbol{J}^{+^T}(\boldsymbol{q}) \left[\boldsymbol{K}_j(\boldsymbol{p}, \boldsymbol{q}) - \boldsymbol{G}_J(\boldsymbol{q}) \right] \boldsymbol{J}^+(\boldsymbol{q}) \quad [40]$$

Where $\boldsymbol{J}^+(\boldsymbol{q}) = \boldsymbol{K}_j(\boldsymbol{p}, \boldsymbol{q})^{-1} \boldsymbol{J}(\boldsymbol{q}) \left(\boldsymbol{J}(\boldsymbol{q}) \boldsymbol{K}_j(\boldsymbol{p}, \boldsymbol{q})^{-1} \boldsymbol{J}(\boldsymbol{q})^T \right)^{-1}$ is the right inverse of the Arm Jacobian $\boldsymbol{J}(\boldsymbol{q})$ taking $\boldsymbol{K}_j(\boldsymbol{p}, \boldsymbol{q})$ as the metric tensor. [15]

Substituting eq.39 into eq.40 finalizes the expression of the endpoint cartesian stiffness matrix to:

$$\boldsymbol{K}_c(\boldsymbol{p}, \boldsymbol{q}) = \boldsymbol{J}^{+^T}(\boldsymbol{q}) \left[\boldsymbol{J}_M^T(\boldsymbol{q}) a_{cc}(\boldsymbol{p}) \boldsymbol{K}_s \boldsymbol{J}_M(\boldsymbol{q}) - \boldsymbol{G}_J(\boldsymbol{q}) \right] \boldsymbol{J}^+(\boldsymbol{q}) \quad [41]$$

Table 1. Inputs and outputs of the Least Squares identification method

Inputs			**Output**
$\boldsymbol{p}$	$\boldsymbol{q}$		
$\boldsymbol{I_{coc}(p)}$	$\boldsymbol{J(q)}$	$\boldsymbol{J_M(q)}$	$\boldsymbol{K_j(p,q)}$

4.1.1 Dimensionality of our case

The model of the human arm developed in [33] has been extensively used, and it is considered reliable by the research community. It is preferred due to its compatibility with the biomechanics simulation software library, OpenSim. This software platform is particularly useful since it provides tools for computation of different dynamical, kinematical, and biological variables. In this model, the human arm is modelled as a kinematic chain of bodies and joints with 5 DOF, excluding the movements of the wrist. The joints considered are:

1. Elevation angle of the arm,
2. shoulder elevation angle,
3. shoulder rotation angle,
4. elbow flexion angle, and
5. pro supination angle.

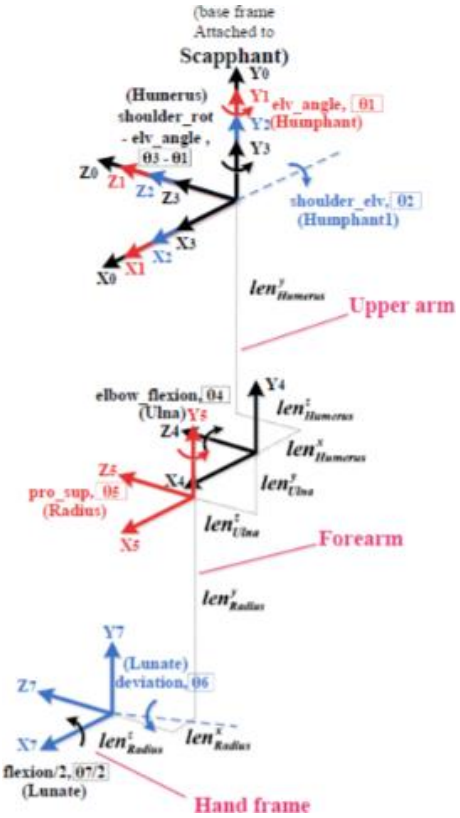


Figure 22. Kinematics of the human arm model developed in [33]. elv_angle, shoulder_elv, shoulder_rot, elbow_flexion, and pro_sup are the notions of the joints respectively.

Their position and direction in the kinematic chain can be observed in Figure 22. Therefore, the vector of joint angles $\boldsymbol{q} \in \mathbb{R}^{5x1}$ and the joint stiffness matrix $\boldsymbol{K}_j(\boldsymbol{p},\boldsymbol{q}) \in \mathbb{R}^{5x5}$. Furthermore, two dimensional motion is considered so the cartesian endpoint stiffness matrix $\boldsymbol{K}_c(\boldsymbol{p},\boldsymbol{q}) \in \mathbb{R}^{2x2}$, and the Arm Jacobian $\boldsymbol{J}(\boldsymbol{q}) \in \mathbb{R}^{2x5}$.

The contributions of the following 12 dominant muscles in the human arm are taken into consideration:

- Anterior, middle, and posterior deltoids,
- long and lateral triceps,
- long and short biceps,
- brachioradialis,
- extensors carpi radialis longus and carpi ulnaris,
- flexors carpi radialis and carpi ulnaris.

This makes $n_m = 12$, so the Muscular Jacobian $\boldsymbol{J}_M(\boldsymbol{q}) \in \mathbb{R}^{5x12}$ and $\boldsymbol{K}_s \in \mathbb{R}^{12x12}$.

Additionally, 8 EMG signals of the muscles on the upper arm and forearm are monitored, resulting in 16 different EMG signals meaning $\boldsymbol{p} \in \mathbb{R}^{1x16}$.

Adopting this method to our case results in inputs-output specification shown in Table 2.

Table 2. Inputs and outputs of the method adopted to the case of interest

Inputs			**Output**
$\boldsymbol{p_{1x16}}$	$\boldsymbol{q_{1x5}}$		
$\boldsymbol{I_{coc}(p)_{1x1}}$	$\boldsymbol{J(q)_{2x5}}$	$\boldsymbol{J_M(q)_{5x12}}$	$\boldsymbol{K_j(p,q)_{5x5}}$

4.2 Machine Learning method

According to the Hill's model, the force exerted by the CE in the muscle is [31]:

$$F_{CE} = F_{\max} a(t) f(l_m) g(v_m) \quad [42]$$

Where:

- $F_{\max}$ is the maximal force,
- $a(t)$ is the percentage of muscle activation,
- $f(l_m)$ is force-length function,
- v_m is the muscle shortening velocity, and
- $g(v_m)$ is force-velocity function.

The force-length function $f(l_m)$ and the force-velocity function $g(v_m)$ are both nonlinear components [34]. Nonlinearities in the force generating element of the muscle result with nonlinear behavior of the overall stiffness of $k_m(\boldsymbol{q}, p)$. The shape and degree of nonlinearity might change due to characteristics of the subject e.g. age. Therefore, it is justified to expect that integrating such nonlinearities in a model might yield more accurate estimation results.

Additionally, information integrated into a model is not completely related to the unique peculiar characteristics of the subject. It holds information on the basic phenomena that drive stiffness regulation by the Central Nervous System too. Therefore, in the process of modelling, it might be useful to exploit information common among the models of other users and adopt the peculiarities of the current user.

Due to the abilities to model complex nonlinearities and reuse common information, Artificial Neural Network (ANN) model is proposed in this work.

4.2.1 Architecture of Shallow Multilayer Artificial Neural Network model

The architecture of the shallow multilayer ANN used is seven-layer ANN composed of an input layer, an output layer and five hidden layers.

As **INPUT**, we use z-scores of:

- The vector of human arm joint angles $\boldsymbol{q}$,
- the corresponding Muscular Jacobian matrix $\boldsymbol{J}_M(\boldsymbol{q})$, and
- the cocontraction index $I_{coc}(\boldsymbol{p})$.

As **OUTPUT**, the joint stiffness matrix $\boldsymbol{K_j(p,q)}$ is used.

Because $\boldsymbol{q} \in \mathbb{R}^{5x1}$, $\boldsymbol{J_M(q)} \in \mathbb{R}^{5x12}$, and $I_{coc}(\boldsymbol{p})$ is a scalar, the size of the input layer would be 66. However, there are 26 elements of the Muscular Jacobian matrix which are constantly 0. Such inputs don't convey any information and they are discarded. Therefore, the input layer has 40 nodes. Since $\boldsymbol{K_j(p,q)} \in \mathbb{R}^{5x5}$, the output layer has 25 nodes. In between the input and the output layer there are five hidden layers. The first hidden layer consists of 30 nodes, and the next 4 layers have 20 nodes each. This results with 3605 learnable weights and 135 biases. Trial and error procedure suggest that this complexity of the network coupled with appropriate protection against overfitting produces satisfactory results. The first three hidden layers implement standard Tan-Sigmoid activation function, while the others use linear activation function. These functions take the expression:

$$Tan-Sigmoid: a = \frac{2}{1+e^{-2*n}} - 1 \quad Pure\ linear: a = n \qquad [43]$$

Where n is the input of the node, and a is the corresponding output. They are graphically shown on Figures 23.

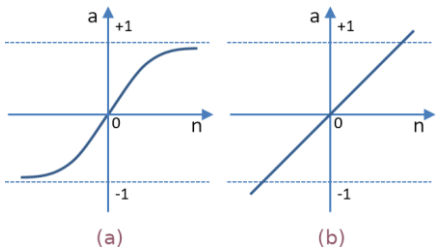


Figure 23. (a) Tan-Sigmoid, and (b) pure linear activation functions.

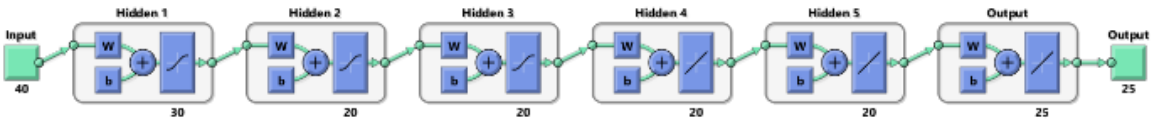


Figure 24. Architecture of the nominal ANN used.

Such linearity in the output enhances the abilities of the network to fit data instead of classifying it. The architecture of the ANN is shown in Figure 24 and the form of the input/output data is summarized in Table 3.

Table 3. ANN's inputs and outputs.

Inputs			Output
$\boldsymbol{q_{1x5}}$	$\boldsymbol{J_M(q)_{5x8}}$	$I_{coc}(\boldsymbol{p})_{1x1}$	$\boldsymbol{K_j(p,q)_{5x5}}$

4.2.2. Training options

Levenberg-Marquardt is used as optimization algorithm since it is considered to perform well with shallow multilayer ANN models. This optimization algorithm calculates the weights of the network according to the following equation:

$$\boldsymbol{w}_{k+1} = \boldsymbol{w}_k - (\boldsymbol{G}^T\boldsymbol{G} + \mu\boldsymbol{G})^{-1}\boldsymbol{G}^T\boldsymbol{e} \quad [44]$$

Where $\boldsymbol{w}$ is the vector of learnable elements (weights and biases), $\boldsymbol{e}$ is the vector of network output errors, and $\boldsymbol{G}$ is the gradient matrix that contains first derivatives of the network errors with respect to weights and biases. μ is a coefficient that transforms the behavior of the algorithm between behaving more like Newton's method or more like Gradient Descent method. It's changing according to the performance function which takes the vector $\boldsymbol{e}$ as an argument. [35] In this work, the mean squared normalized error is considered as a performance function. μ is decreased for 0.1 after each step that reduces this function, and it is increased for 10 after each step that increases the performance function.

Vector $\boldsymbol{w}$ is initialized using Nguyen-Widrow initialization algorithm. It generates initial weights and biases for every layer such that the active regions of the layer's neurons are distributed evenly over the input space. [36] Such initialization contains a degree of randomness, so the initial values are not same every time the initialization is performed. Thus, the training process has a different starting point with every execution leading to a different result every time. Therefore, five training processes with different initializations are done. The best performing result is considered.

The dataset provided to the algorithm is divided in two subsets: training, and testing. The ratio and strategy of the split depends on the specific application of the trained ANN. To protect against overfitting, Bayesian regularization takes place. The training procedure terminates if μ increases above certain threshold, or if the sum of squared errors is relatively constant over several epochs. Due to such regularization strategy, no validation subset is used.

4.3 Experimental offline stiffness estimation[1]

In both online stiffness estimation methods described here, the joint stiffness matrix $\boldsymbol{K}_j(\boldsymbol{p},\boldsymbol{q})$ is considered the output. After estimating $\boldsymbol{K}_j(\boldsymbol{p},\boldsymbol{q})$, the transformation to the cartesian endpoint stiffness matrix $\boldsymbol{K}_c(\boldsymbol{p},\boldsymbol{q})$ is straightforward by using the Arm Jacobian (eq.40).

However, obtaining the joint stiffness matrix directly requires monitoring of the human arm torques, which in addition requires complex hardware and is hard to perform. Therefore, common practice is to estimate the cartesian stiffness matrix $\boldsymbol{K}_c(\boldsymbol{p},\boldsymbol{q})$ and transform it to the joint space using eq. 32.

Cartesian endpoint mechanical impedance was already introduced as dynamical relation between force and displacement of the human arm endpoint. Therefore, cartesian stiffness is usually estimated by a family of mechanical perturbations methods. Common across all these procedures is the applying of known mechanical perturbations on the endpoint of the

[1] The experiments for offline generation of outputs were performed by a separate team working on this research project. Here considered for completeness.

human arm and monitoring the restoring forces. Such perturbations are performed when the arm endpoint is positioned in a certain cartesian point within the natural comfort zone of the user. This point serves as the nominal position around which the perturbations are performed. The union of such cartesian point and the corresponding human arm joint angles (arm configuration) is called a pose. Once the data of the displacements $\boldsymbol{x}$ and forces $\boldsymbol{f}$ is recorded, post-processing methods estimate the corresponding cartesian stiffness.

Number of such methods exist but we take closer look to two of them in this work. The time-domain displacement method [16], and the frequency domain method [15].

4.3.1. Hardware setup

Both experimental methods may be performed using very similar hardware setup described here.

Mechanical perturbations are applied to the subject's wrist by anthropomorphic robotic manipulator. The subject holds the end-effector of the robot while the wrist is immobilized. Therefore, the perturbations of the hand are transferred to the wrist, which is considered the endpoint of the arm. There is 6 axes force sensor attached to the end-effector to measure the reaction forces of the human arm. The subject is equipped with two armband devices recording EMG signals and joint angles. One of these devices is attached to the forearm and one to the upper arm. 16 EMG signals are recorded in total.

During experiments of the time-domain displacement method, the subject also has support for the elbow. It puts the axes of the forearm and the upper arm in same plane. This is the only setup difference between the two experimental methods.

The complete hardware setup is shown on Figure 25(a).

The mechanical perturbations are performed by KUKA® lwr 4+ (KUKA Robotics Corporation, Augsburg, Bavaria, Germany) robotic manipulator in position control (constant infinite stiffness in all directions). The reaction forces are recorded by M3815C force sensor provided by Sunrise Instruments (Nanning, China). Devices used to record EMGs and joint angels are Thalmic Labs®' Myo armbands (Kitchener, Canada).

The data is sampled with frequency of 250Hz, and every experiment lasts approximately 35 seconds.

4.3.2 Time-domain displacement method

In the time-domain displacement method, the assumption about the dominance of the stiffness in the impedance of the muscle is considered. Based on this assumption, it is further assumed that the complete neuromuscular system is predominantly spring-like. Therefore, the behavior of the endpoint cartesian impedance is also assumed to be purely elastic around the equilibrium position.

This method is performed solely in the planar case. Therefore, only the f_x and f_y components of the force and x and y components of the displacement are considered. The subject holds the robot's end-effector in one of a pre-defined set of cartesian positions. 8 perturbations are applied per position. The displacements are with magnitude of 5mm-8mm in a casual direction selected from a list of possible angles ranging from 0° to 360°, 45° apart. The displacement is done in a time-interval of about 120ms and is followed by holding phase in the displaced positions lasting 1.5s.

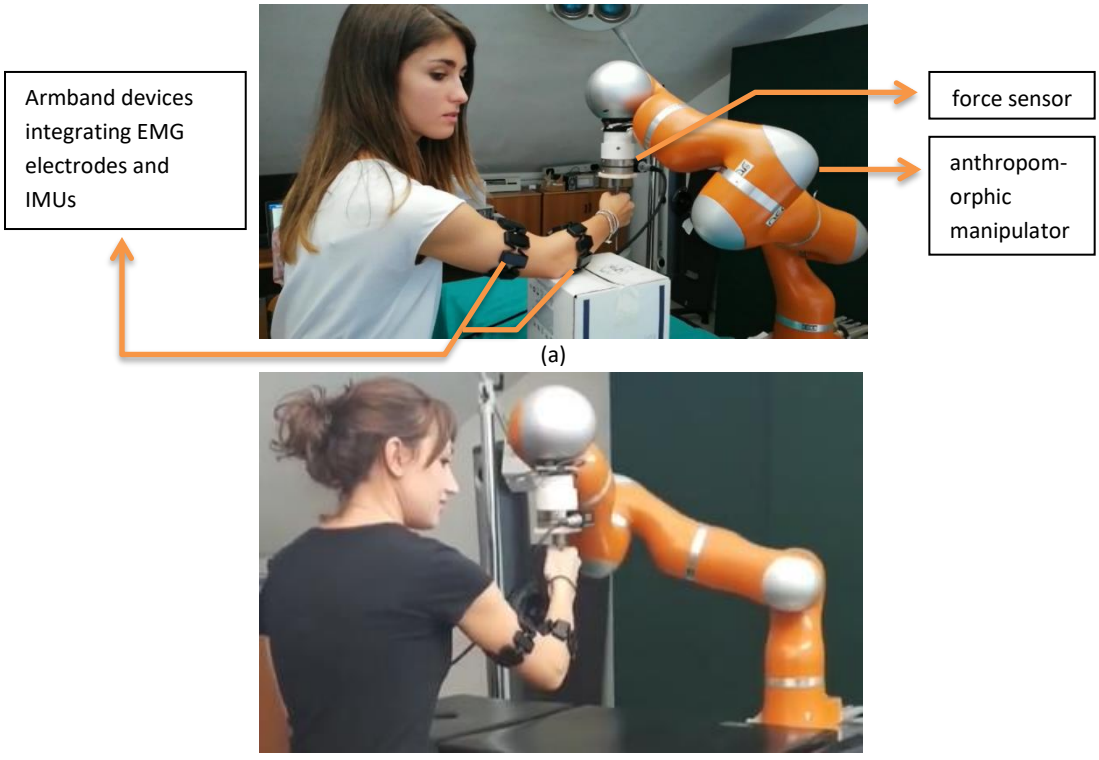


Figure 25. (a) Hardware setup. (b) Experiment performed in different pose in comparison with (a)

In the post-processing stage, due to the assumption of pure elasticity around an equilibrium position, the following relation between the recorded forces and displacements is considered:

$$\begin{bmatrix} f_x \\ f_y \end{bmatrix} = -\begin{bmatrix} K_{xx} & K_{xy} \\ K_{yx} & K_{yy} \end{bmatrix} \begin{bmatrix} dx \\ dy \end{bmatrix} \quad [45]$$

Where K_{xx},K_{yy}, K_{xy}, and K_{yx} are the elements of the cartesian stiffness matrix. Standard linear least squares identification procedure is used to determine these elements. [16]

4.3.2 Frequency domain method

In this work however, the frequency method is used. The idea behind this method is to identify the parameters of the human arm impedance in certain range of frequencies. Therefore, eliminating potential adverse effects of higher harmonics.

This method is performed in three dimensional workspace, and therefore all components of the force f_x, f_y, and f_z are recorded. All components of the displacement x, y, and z as well. The subject holds the robot's end-effector in one out of pre-defined set of cartesian positions. 4 perturbations per second are applied for 35 seconds. The displacement amplitude is 20mm and the orientation is completely casual. During the experiment, an index of muscular cocontraction is calculated from 16 EMG signals and is displayed to the subject in real-time. The subject is asked to keep the cocontraction index in one range out of the pre-defined set of ranges. Having the cocontraction index in several different ranges results with different stiffness levels of the human arm.

In the post-processing stage, the MIMO dynamics of the human arm endpoint are decomposed into linear systems associating each input with each output. Therefore, relating Fourier transformations of forces and displacements by:

$$\begin{bmatrix} f_x(\omega) \\ f_y(\omega) \\ f_z(\omega) \end{bmatrix} = \begin{bmatrix} \boldsymbol{G}_{xx}(\omega) & \boldsymbol{G}_{xy}(\omega) & \boldsymbol{G}_{xz}(\omega) \\ \boldsymbol{G}_{yx}(\omega) & \boldsymbol{G}_{yy}(\omega) & \boldsymbol{G}_{yz}(\omega) \\ \boldsymbol{G}_{zx}(\omega) & \boldsymbol{G}_{zy}(\omega) & \boldsymbol{G}_{zz}(\omega) \end{bmatrix} \begin{bmatrix} \mathrm{x}(\omega) \\ \mathrm{y}(\omega) \\ \mathrm{z}(\omega) \end{bmatrix} \quad [\,46\,]$$

Transfer functions $\boldsymbol{G}_{ij}(\omega)$ are estimated by a non-parametric identification algorithm fed with smoothed spectral estimates of force and displacement data. Consequently, for $\boldsymbol{G}_{ij}(\omega)$ a mechanical impedance system of second order is adopted:

$$\boldsymbol{G}_{ij}(\omega) = \boldsymbol{I}_{cij}s^2 + \boldsymbol{B}_{cij}s + \boldsymbol{K}_{cij} \qquad s = \omega\sqrt{-1} \quad [\,47\,]$$

Where $\boldsymbol{I}_{cij}$, $\boldsymbol{B}_{cij}$, and $\boldsymbol{K}_{cij}$ denote the endpoint inertia, damping, and stiffness matrices, respectively. These parameters are identified using least squares algorithm in the frequency from 0Hz to 10Hz. [15]

4.3.3 Performed experiments

Three healthy subjects of age 25, 24, and 23 took part in the experiments. User 1 performed experiments in 7 different poses, each with 5 different ranges of cocontraction index. Resulting in 35 successful trials. User 2 performed experiments in 8 different poses, each with 4 different levels of cocontraction index. Resulting in 32 successful trials. And User 3 performed experiments in 7 different poses and 4 different cocontraction levels, making for 28 successful trials. This user data is summarized in Table 4.

Table 4. Number of observations per user.

User	Poses	Cocontraction level	Number of observations
1	7	5	35
2	8	4	32
3	7	4	28
Total number of observations:			95

The estimated cartesian endpoint stiffness matrix $\boldsymbol{K}_c \in \mathbb{R}^{3x3}$ is transformed to planar format in order to fit the goal of this work. In accordance with the online stiffness estimation methods proposed in this work, each observation takes the form shown on the following Table 5.

Table 5. The set of data in one observation.

$\boldsymbol{q_1}$	$\boldsymbol{q_2}$	$\boldsymbol{q_3}$	$\boldsymbol{q_4}$	$\boldsymbol{q_5}$	$\boldsymbol{p_1}$	...	$\boldsymbol{p_{16}}$	$\boldsymbol{K_{c[2x2]}}$

4.4 Muscular Jacobian

4.4.1 Definition

The Muscular Jacobian $\boldsymbol{J}_M(\boldsymbol{q})$ represent a transformation between the muscular space and the joint space of the human arm. It relates the musculotendon length changes to the joint angle variations by $\boldsymbol{J}_M(\boldsymbol{q})\dot{\boldsymbol{q}} = \dot{\boldsymbol{l}}$. Where $\boldsymbol{l}$ is the vector of musculotendon lengths. For simplicity, by musculotendon length we assume the length of the path of muscle tension applied during the activity of muscle. Therefore, the Muscular Jacobian takes the form:

$$\boldsymbol{J}_M(\boldsymbol{q}) = \begin{bmatrix} \frac{\delta l_{1(\boldsymbol{q})}}{\delta q_1} & \frac{\delta l_{1(\boldsymbol{q})}}{\delta q_2} & \cdots & \frac{\delta l_{1(\boldsymbol{q})}}{\delta q_n} \\ \frac{\delta l_{2(\boldsymbol{q})}}{\delta q_1} & \frac{\delta l_{2(\boldsymbol{q})}}{\delta q_2} & \cdots & \frac{\delta l_{2(\boldsymbol{q})}}{\delta q_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\delta l_{n_m(\boldsymbol{q})}}{\delta q_1} & \frac{\delta l_{n_m(\boldsymbol{q})}}{\delta q_2} & \cdots & \frac{\delta l_{n_m(\boldsymbol{q})}}{\delta q_n} \end{bmatrix} \quad [48]$$

Where n_m is the number of observed muscles and n is the number of considered joint angles. It is assumed that no torques are present at the arm joints. In this work, tracking $\boldsymbol{J}_M(\boldsymbol{q})$ is vital since it's required as input in both online stiffness estimation methods discussed. The model of the human arm introduced in Chapter 4.1.1 and the software libraries associated are used for computation of the Muscular Jacobian.

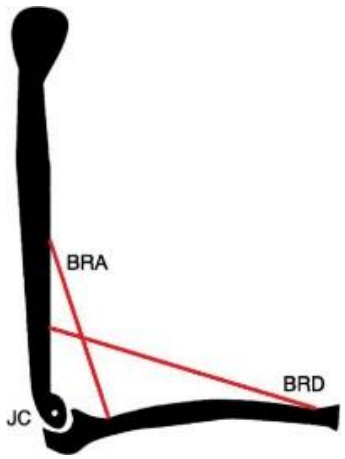


Figure 26. The brachialis (BRA) and brachioradialis (BRD) muscles are representatives of muscles that can be modelled with simple linear musculotendon paths (in red). JC is Joint Center of the elbow joint.

We can calculate $\delta l_i(\boldsymbol{q})/\delta q_j$ directly by making small change of Δq_j and measuring the resulting change Δl_i. This could be applicable for muscles with simple linear musculotendon paths. I.e. ones which occupy the shortest path between two points defining the direction of exerted tension. Such muscles are shown on Figure 26. However, muscles often have musculotendon paths that are wrapped around joints and they are highly nonlinear. The length of such musculotendon paths is usually calculated numerically and therefore approximate solution is acquired. This leads to numerical variability in the calculated length l_i which renders the

calculation of Δl_i problematic. The smaller the change of l_i, the more numerical variability in the result. Therefore, alternative approach is used.

The Muscular Jacobian element gives quantitative expression on the effectiveness of torque generation of a musculotendon tension into the considered joint at the specified configuration. If we assume that a human arm in certain static configuration is not performing any mechanical work, then we can use the work balance equation:

$$s_i dl_i(\boldsymbol{q}) = \tau_{q_j} dq_j \quad [49]$$

Where s_i is the tension of the musculotendon of the muscle i with length l_i, and τ_{q_j} is the torque at the considered human arm joint angle. From eq.49 we can represent the general element of the Muscular Jacobian matrix:

$$\frac{dl_i(\boldsymbol{q})}{dq_j} = \frac{\tau_{q_j}}{s_i} = r_{ij}(\boldsymbol{q}) \quad [50]$$

Where r_{ij} is the moment arm of the muscle i with regards to the joint q_j. The moment arm is mechanical concept defined as the distance from force line of action to the center of rotation. Examples of moment arms are shown on Figure 27. Therefore, there is an equality between the Muscular Jacobian and the matrix of moment arms:

$$\boldsymbol{J}_M(\boldsymbol{q}) = \begin{bmatrix} r_{11}(\boldsymbol{q}) & r_{12}(\boldsymbol{q}) & \cdots & r_{1n}(\boldsymbol{q}) \\ r_{21}(\boldsymbol{q}) & r_{22}(\boldsymbol{q}) & \cdots & r_{2n}(\boldsymbol{q}) \\ \vdots & \vdots & \ddots & \vdots \\ r_{n_m 1}(\boldsymbol{q}) & r_{n_m 2}(\boldsymbol{q}) & \cdots & r_{n_m n}(\boldsymbol{q}) \end{bmatrix} \quad [51]$$

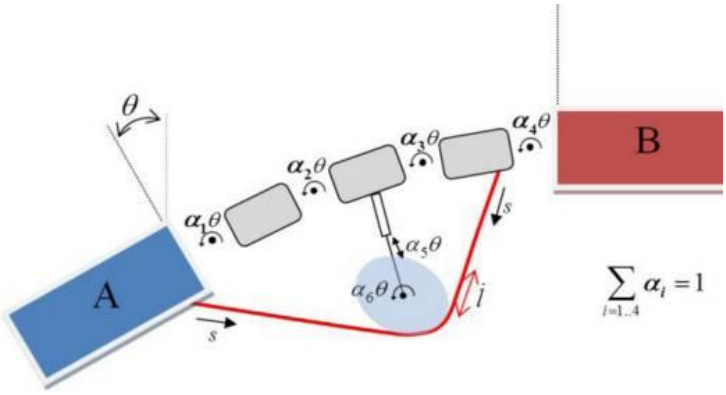


Figure 28. Graphical representation of a model of biomechanical joint θ between bodies A and B.

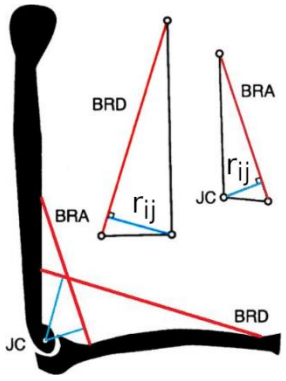


Figure 27. Example for moment arms (in light blue) for muscles that can be

4.4.2 Algorithm for calculation of the moment arm matrix

Unfortunately, not all joints systems on the human body are as simple and regular as the ones displayed in Figures 26 and 27. To perform reliable simulations, often the musculotendon path has to be modelled with higher complexity. Generally, musculotendon paths start from a starting point fixed to a body, they pass through several via points and

over frictionless wrapping surfaces before finishing at an insertion point fixed on another distal body. Additionally, what is considered biological human arm joint (also called generalized coordinate), on the mechanical level might be modelled with few standard joints that doesn't even have to be exclusively revolute. However, they can be coupled and/or constrained to realize a generalized coordinate. For example, the neck has few degrees of freedom (generalized coordinates), but it involves the coordinated motion of multiple mechanical joints in a model. Therefore, the general joint between two bodies A and B is denoted with θ. Its motion might be result of several coupled joints, but it also might be a primitive joint. Also, it might represent the motion of a generalized coordinate q, or it may be one of the coupled joints used to model the coordinate. Therefore:

$$\theta = \theta(\boldsymbol{\theta}) \quad q = q(\boldsymbol{\theta}) \quad [52]$$

If $\boldsymbol{\theta}$ is scalar, then the joint (or the generalized coordinate) is of the primitive type represented in Figure 26.

Such muscle-joint is presented in Figure 28 where:

- A and B are two bodies,
- $\theta = \theta(\boldsymbol{\theta}) = \theta_1 + \theta_2 + \theta_3 + \theta_4$ is the joint angle where
  - $\boldsymbol{\theta} = [\theta_1 \quad \theta_2 \quad \theta_3 \quad \theta_4]$ and $\theta_k = \alpha_k \theta$
- l is the musculotendon path upon which tension s is applied,
- $\theta_{5,6} = \alpha_{5,6}\theta$ are other coupled joints but they affect the musculotendon path, not the joint angle θ.

To calculate the moment arm of this system, few consequences of such modelling should be observed.

1. It should be stated that in OpenSim, the speeds of the generalized coordinates are not given directly by the time derivative of the coordinates, instead:

   $$\dot{\boldsymbol{q}} = \frac{d\boldsymbol{q}}{dt} = \boldsymbol{N}\boldsymbol{u} \quad [53]$$

   Where $\boldsymbol{u}$ is the vector of generalized speeds, and $\boldsymbol{N}$ is a block diagonal matrix.

2. Regarding the musculotendon path, uniform tension and linear tension-to-force transmission is assumed. This means that the spatial force applied by the muscle is completely described by a scalar tension $s > 0$. Therefore:

   $$\boldsymbol{F}(\boldsymbol{q}, s) = \boldsymbol{T}(\boldsymbol{q})s \quad [54]$$

   Where $\boldsymbol{F}(\boldsymbol{q}, s)$ is the matrix of spatial forces applied by the muscle with tension s, when the configuration of the arm is $\boldsymbol{q}$. $\boldsymbol{T}(\boldsymbol{q})$ is the force transmission matrix that linearly relates the muscle tension s and the spatial force matrix. OpenSim is always

capable of computing $\boldsymbol{F}(\boldsymbol{q}, s)$ for a given s and $\boldsymbol{q}$, but the matrix $\boldsymbol{T}(\boldsymbol{q})$ is not always explicitly calculated.

3. The matrix of spatial forces $\boldsymbol{F}(\boldsymbol{q}, s)$ can be mapped to the vector of forces/torques that act along the generalized coordinates (vector of generalized forces) by:

$$\boldsymbol{f}_q = \boldsymbol{J}_s^T \boldsymbol{F} \quad [55]$$

Where $\boldsymbol{f}_q$ is the vector of generalized forces, and $\boldsymbol{J}_s = \boldsymbol{J}_s(\boldsymbol{q})$ is the system Jacobian that maps generalized speeds to the body spatial velocities. Even though the term generalized forces is used, $\boldsymbol{f}_q$ also implies torques.

4. Since we assumed that in general case there is possible coupling between the generalized coordinates and $\boldsymbol{\theta}$, then:

$$\boldsymbol{u} = \boldsymbol{C}_{\boldsymbol{\theta}} \dot{\boldsymbol{\theta}} \quad [56]$$

Where $\boldsymbol{C}_{\boldsymbol{\theta}} \in \mathbb{R}^{nx1}$ is a coupling column matrix.

In order to calculate the muscle moment arm for the general case shown in Figure 28 without explicitly computing $\dot{l}$ as Δl, we can use the definition of moment arm introduced in eq.50 adapted for the general joint angle θ:

$$r(\boldsymbol{\theta}) = \frac{dl(\boldsymbol{\theta})}{d\theta} = \frac{\frac{dl(\boldsymbol{\theta})}{dt}}{\frac{d\theta}{dt}} = \frac{\dot{l}(\boldsymbol{\theta})}{\dot{\theta}} \quad [57]$$

Taking into consideration the necessary power equivalence $\boldsymbol{f}_q^T \boldsymbol{u} = s\dot{l}$, then:

$$r(\boldsymbol{\theta}) = \frac{\dot{l}}{\dot{\theta}} = \frac{\boldsymbol{f}_q^T \boldsymbol{u}}{s\dot{\theta}} \quad [58]$$

Combining equations eq.58, eq.56, eq.55, and eq.54, the final equation for calculating the moment arm is derived:

$$r(\boldsymbol{\theta}) = \frac{\boldsymbol{f}_q^T \boldsymbol{C}_{\boldsymbol{\theta}}}{s} \quad [59]$$

It is important to notice that in this equation, there is no need of explicit calculation of $\dot{l}$.

The algorithm to calculate the moment arm with this expression is:

1. Calculate $\mathbf{C}_{\boldsymbol{\theta}}$. It can be calculated by setting the rate of change of the joint angle of interest to a constant non-zero value $\left(\dot{\theta} = \dot{\theta}_0\right)$. Then OpenSim's solver can be used to find a vector of generalized speeds $\boldsymbol{u}$ such that all coupling constraints are satisfied (in both positional and velocity level) so $\dot{\theta}(\boldsymbol{u}) = \dot{\theta}_0$. Once the elements of this vector $\boldsymbol{u}$ are computed, we can calculate the elements of $\boldsymbol{C}_{\boldsymbol{\theta}}$ as: $c_i = \frac{u_i}{\dot{\theta}_0}$.

2. Using eq. 54 calculate the spatial force matrix $\boldsymbol{F}(\boldsymbol{q}, s)$ for a non-zero value of the tension $s = s_0$.
3. Using the transformation in eq.55 map the spatial forces to the generalized forces vector $\boldsymbol{f}_q^T$.
4. Use eq.59 to calculate the moment arm $r(\boldsymbol{\theta})$.

[37]

This algorithm is implemented with OpenSim's API library, and this software framework is used to calculate $\boldsymbol{J}_M(\boldsymbol{q}) \in \mathbb{R}^{12x\,5}$ for every human arm configuration $\boldsymbol{q}$.

5. Results

As a general metric to evaluate the results of both the online stiffness estimation methods, performance function proposed in [15] is used. This metric is defined as error by:

$$e_K = \frac{1}{n_T}\sum_{i=1}^{n_T}\frac{\left\|\boldsymbol{J}_{M_i}^T(\boldsymbol{q})a_{cc_i}(\boldsymbol{p})\boldsymbol{K}_S J_{M_i}(\boldsymbol{q}) - \boldsymbol{J}_i^T(\boldsymbol{q})\boldsymbol{K}_{c_i}(\boldsymbol{p},\boldsymbol{q})\boldsymbol{J}_i(\boldsymbol{q})\right\|}{\left\|\boldsymbol{K}_{c_i}(\boldsymbol{p},\boldsymbol{q})\right\|} \quad [60]$$

Where n_T is the number of samples considered in the test. Combining eq.39, eq.32, and eq.60, the more general form of the metric can be formed:

$$e_K = \frac{1}{n_T}\sum_{i=1}^{n_T}\frac{\left\|\widehat{\boldsymbol{K}}_{j_i}(\boldsymbol{p},\boldsymbol{q}) - \boldsymbol{K}_{j_i}(\boldsymbol{p},\boldsymbol{q})\right\|}{\left\|\boldsymbol{K}_{c_i}(\boldsymbol{p},\boldsymbol{q})\right\|} \quad [61]$$

To evaluate the performance of the Machine Learning method in different strategies of use, the 91 observations gathered from the experimental offline stiffness estimation are split into different training and test subsets. Each split is used to evaluate the performance of the method in a particular strategy of use. It should be noticed that taking into consideration the infinite number of possible arm configurations and stiffness levels, dataset of this size represents significant challenge. Even more, when the limited diversity of data is considered. Additionally, the calculation of the Muscular Jacobian was performed using generic model of the human arm.

5.1 User specific strategy

In order to make comparison between the LSQ identification method and the machine learning approach, a user specific strategy must be evaluated. This is to say, for the identification of the linear models and for the training of the ANN models, only the data of a specific user is used. The user specific data is casually split in two subsets. 80% is used for LSQ identification and training, and 20% is used for testing. Data separation is shown in Figure 29. Same procedure applies for all users. For such experiments, the evaluation of the performance function e_K is shown in Table 6. Additionally, actual and estimated values of the

joint stiffness in the elbow and in the shoulder for the test subset of User 1 are shown in Figure 31. User 2 and User 3's estimates maintain similar behavior.

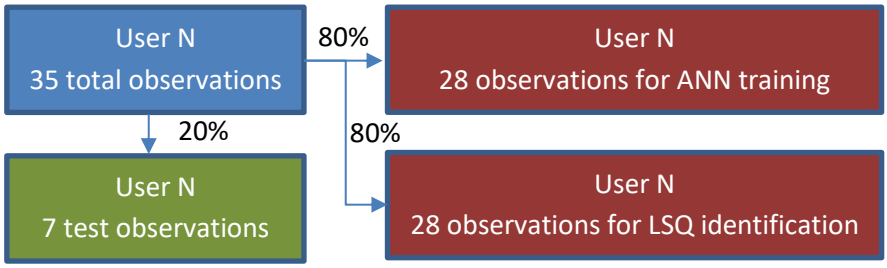


Figure 29. Data separation for user-specific strategy.

Table 6. e_K evaluation for the user specific strategy.

	LSQ identification method	Machine Learning method
User 1	6.8%	5.0%
User 2	9.3%	4%
User 3	11%	6.3%

Table 7. e_K evaluation for the non-user specific strategy.

	LSQ identification method	Machine Learning method
User 1	6.8%	7.6%
User 2	9.3%	3.5%
User 3	11%	8%

5.2 Non-user specific strategy

To evaluate how considering the data from multiple users affected the user-specific results, a non-user specific strategy was applied. The LSQ identification method is user-specific by its definition, therefore only the Machine Learning method is evaluated with this strategy. In the user-specific strategy, a separate ANN was trained for every user. However, in this case, one ANN is used to estimate the joint stiffness of all the users. The model is trained using observations of all users, but it is tested with a testing subset specific to each user. This dataset split is shown in Figure 30. The evaluation of the performance function e_K is shown in Table 7.

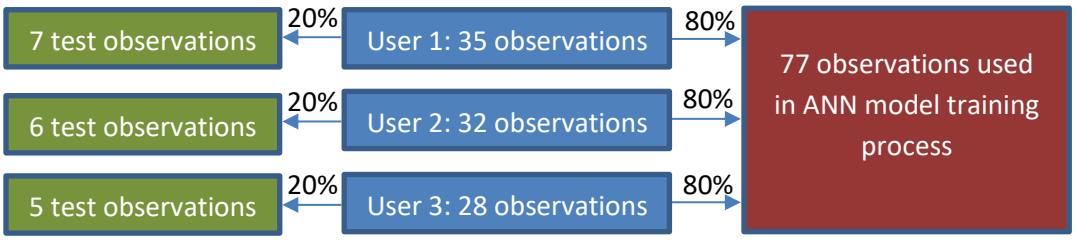


Figure 30. Data separation for non-user specific strategy.

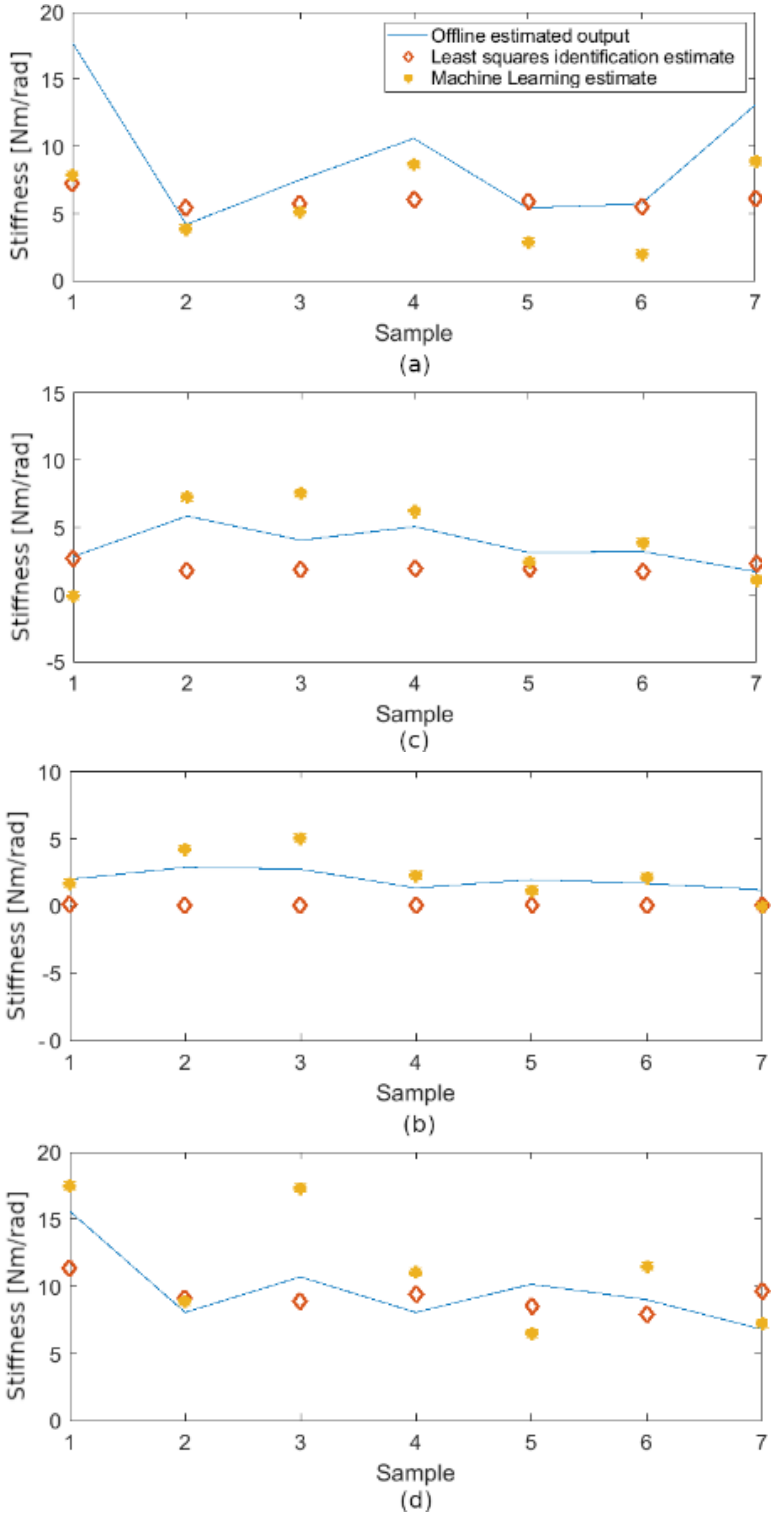


Figure 31. Joint stiffness estimates of the LSQ identification method and the Machine Learning method for the test subset of User 1. (a) Elevation plane joint of the shoulder. (b) Shoulder elevation joint. (c) Shoulder rotation joint. (d) Elbow flexion

5.3 Interpolation performance

In order to test the interpolation performance of the ANN model, two separate simulations were performed. In the first test, the set of observations of each user was split so that all observations for one specific pose were used as test subset. The rest of the data was used for training. This simulation is used to evaluate the performance of the ANN when the data for a specific arm configuration hasn't been presented during training. Therefore, the ability of the ANN model to interpolate from the data of the nearby arm configurations is tested. Data separation for User N for both cases is shown in Figure 32, the sample applies for the other users. The performance function e_K evaluation is shown in Table 8, and the joint stiffness estimates for the case of User 1 are presented in Figure 33. Other users maintain similar behavior. In the second test, the observations of each user were split so that all observations for one specific level of stiffness were used as test subset. The rest of the data was used for training. This simulation intends to test the capability of the ANN model to interpolate between levels of stiffness. The e_K evaluation is shown in Table 8, and the joint stiffness estimates for User 1 are shown in Figure 34. User 2 and User 3's estimates maintain similar behavior.

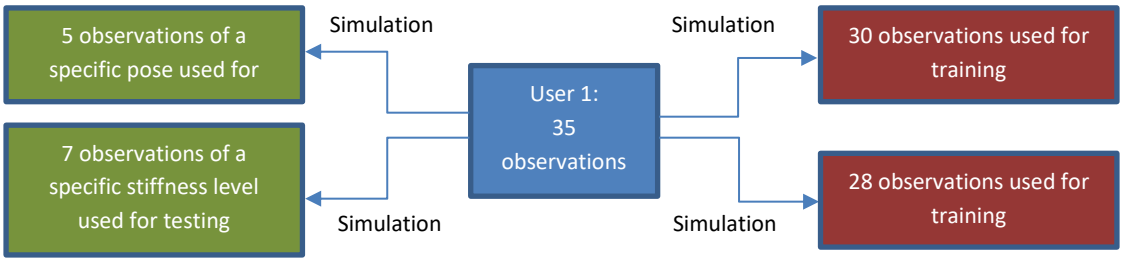
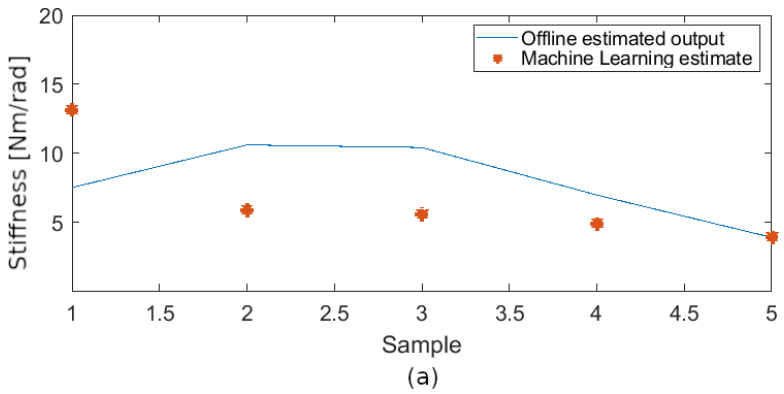


Figure 32. Data separation for interpolation performance evaluations.

Table 8. e_K evaluation of pose and stiffness interpolation performance

	Pose interpolation performance	**Stiffness interpolation performance**
User 1	6.4%	3.5%
User 2	8.1%	4.9%
User 3	9%	4.8%



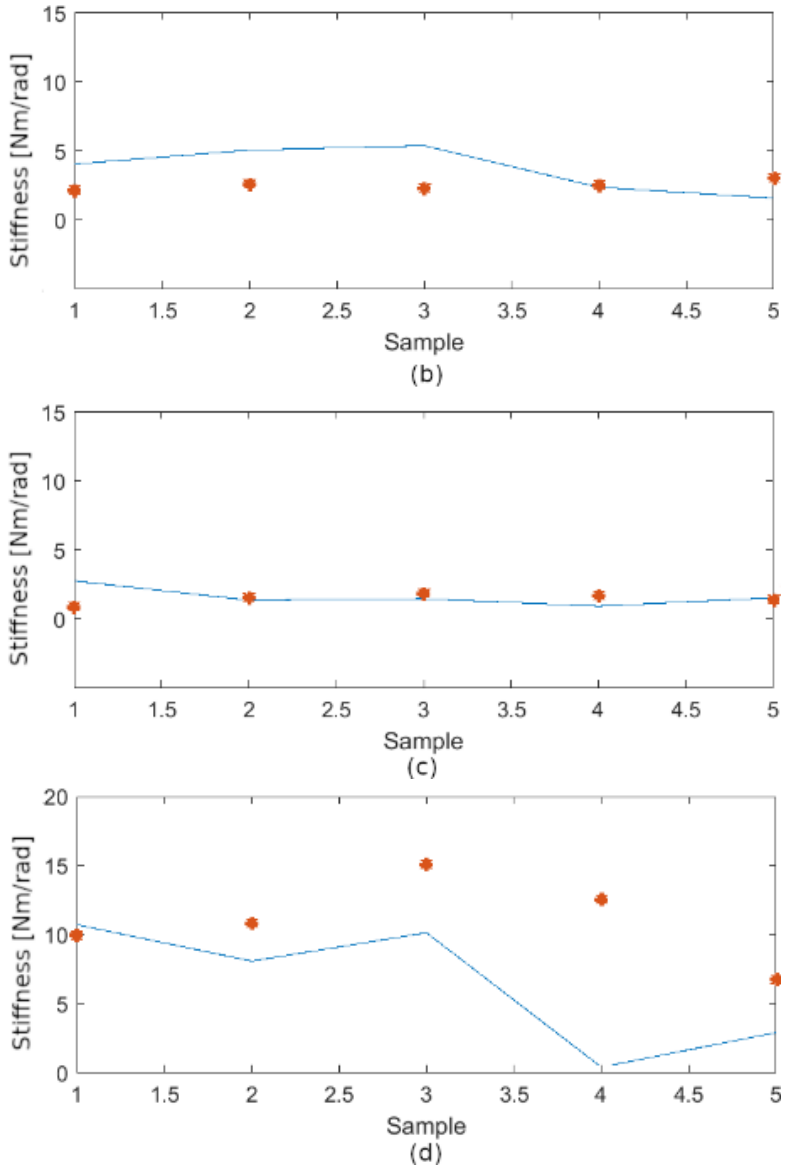
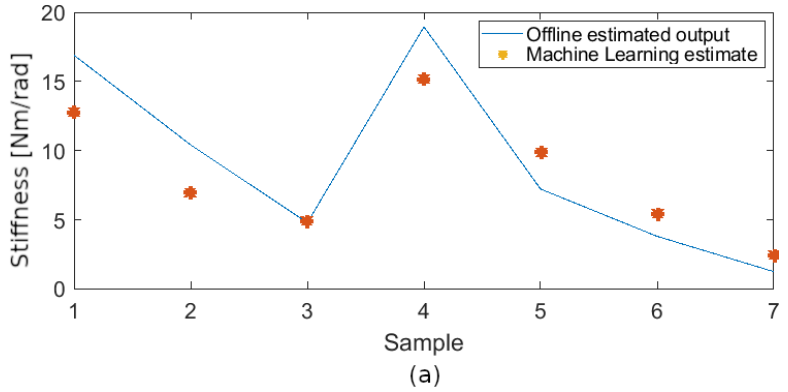


Figure 33. Pose interpolation performance. Joint stiffness estimates of the Machine Learning method for the pose interpolation test subset of User 1. (a) Elevation plane joint of the shoulder. (b) Shoulder elevation joint. (c) Shoulder rotation joint. (d) Elbow flexion



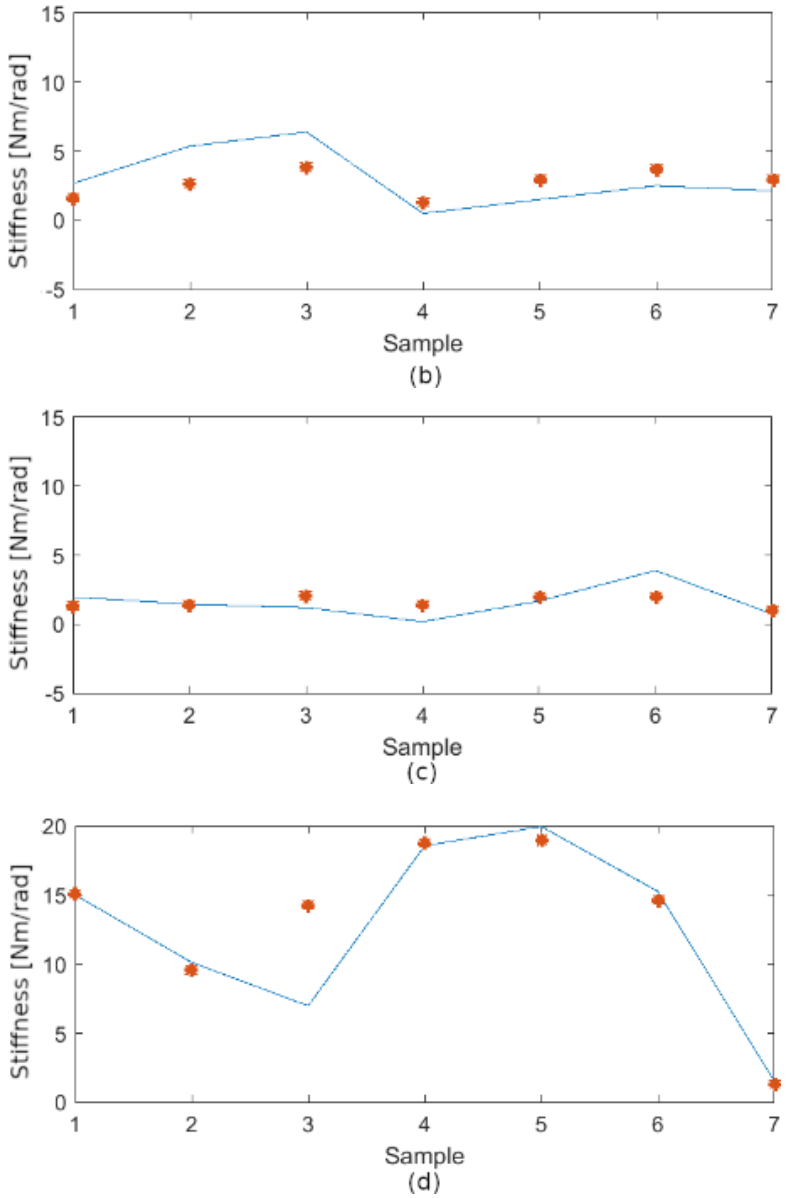


Figure 34. Stiffness level interpolation performance. Joint stiffness estimates of the Machine Learning method for the stiffness interpolation subset of User 1. (a) Elevation plane joint of the shoulder. (b) Shoulder elevation joint. (c) Shoulder rotation joint.

6. Discussion

Results presented in Table 6 indicate lower evaluation of the performance function e_K for the Machine Learning method in the user specific case. Due to adoption of non-linearities in the ANN model, such results are expected. The performance of the ANN model in non-user specific strategy indicates lower e_K evaluation in two cases compared to the linear model, and one case compared to the user specific strategy. However, observing the average error throughout all the users (Table 9), the user specific strategy with the Machine learning method performs slightly better compared to the non-user specific strategy. If the number of users considered in the training of the non-user specific ANN model increase, its average e_K throughout all the users is expected to decrease further. Possibly, lower than the average e_K of the user specific ANN.

Table 9. Average e_K evaluation of all users for both strategies.

LSQ identification method	**Machine Learning method**	
user specific	user specific	non-user specific
9.03%	5.1%	6.36%

The simulations performed to evaluate the interpolation performance suggest reliable behavior of the ANN model. This reliability comes from the comparability of the e_K evaluations of these simulations, shown in Table 8, to the general evaluation of both methods shown in Table 6 and Table 7. e_K evaluations are lower for the stiffness interpolation performance in comparison with the evaluations for the pose interpolation performance. This suggests that the ANN internally develops efficient rules for stiffness level interpolation in arm configurations presented during training. However, if the ANN is presented with an input with general arm configuration, it interpolates the parameters of such rule using knowledge of rules for nearby arm configurations. This shows that the ANN model has adopted proper fitting properties and it overcame the data diversity limitations of the offline experimental stiffness estimation method.

7. Future work

As a rule of thumb, the wider the information conveyed by the training subset, the better the performance of a Machine Learning method. Such rule applies to the method presented in this work too, taken into consideration that in the size of the training subset, the diversity of the data is considered too. However, due to the time and resource consuming nature of the offline experimental stiffness estimation method, several techniques to increase accuracy are proposed for further investigation:

1. *Transfer learning.* If offline experimental stiffness estimation is performed for sufficient number of subjects, a non-user specific ANN model could be trained. Such model would integrate the basic biological phenomenology of the Central Nervous System control over human arm stiffness. Therefore, this ANN might serve as a core model that describes the stiffness control behavior common among human beings. However, in order to achieve better performance, it could be adapted to a particular user of interest by Transfer Learning. Such adaption is performed by retraining only few of the last layers of the ANN model. Retraining might be done with small training subset containing low number of poses and possibly only the highest and the lowest stiffness levels. Thus, reducing the burden of the offline experimental method for new users. This technique results with hybrid ANN model that integrates the advantages of user specific and non-user specific strategies.
2. *ANN ensemble.* As pointed out previously, the training of each ANN model starts with different initialization values of the learnable elements. This leads to different performance of each ANN model. If small set of data is available, training several ANN models and averaging their output might result with increased performance.
3. *Ensemble of both methods proposed.* Another form of working ensemble might be achieved if both the LSQ identification method and the Machine Learning method are used together. The output of such approach might be aggregated by averaging the individual outputs of both methods. Additionally, an attempt to characterize the difference between the individual outputs might be useful. Even more educated guess might be made by utilizing such characterization with model predictive methods such as Kalman predictor.

During this work, it was observed that the interpolation characteristics of the ANN largely depend on the ANN architecture. Particularly on the number and arrangement of layers and the activation functions they adopt. Possible investigation in the characterization of the relation between the ANN's architecture and the stiffness and pose interpolation performance might be meaningful. It may provide guiding information useful in developing ANN model that utilizes a-priori information about the nature of the task and workspace.

8. Conclusion

The aim of this work is to propose and evaluate the performance of human arm stiffness estimation by a Machine Learning method. In the beginning, an introduction to HRI is first given. Contemporary use in manufacturing and medicine is also reviewed. Attention is payed to the special case of exoskeletons too. A review of state of the art places this work in proper position in the spectrum of bioengineering, robotics, and applied artificial intelligence. Following, definition of human arm stiffness is given with a review on the purpose and strategies of its regulation by the Central Nervous System. A Machine Learning method is proposed alongside LSQ identification method. Both methods utilize EMG signals along with human arm joint angles and the Muscular Jacobian in order to estimate the joint stiffness matrix. The LSQ identification method proposes linear model based on biological insight, and the Machine Learning method uses shallow multilayer ANN model. Then, two offline experimental methods for obtaining human arm stiffness are described. The data obtained with such methods is used for identification and training of the models proposed. Following, the Muscular Jacobian is defined and its equivalence with the moment arm matrix is shown. The method used to perform online calculation of the moment arms is derived. In the Results section, user specific and non-user specific strategies of the Machine Learning method are evaluated. Lower estimation error is shown by the user specific Machine Learning method compared to the linear model identified with LSQ procedure. The possibility of better performance of the non-user specific method is also argued. Additionally, the interpolation performance of the ANN model is evaluated both in the case of pose and stiffness interpolation with acceptable outcomes.

Potentially, both strategies of the Machine Learning method can be utilized as primary or advisory human arm joint stiffness transducers. They could augment the diversity of the input of variable impedance controllers. Therefore, enabling them to bring higher stability, accuracy, and intuitiveness in physical HRI scenarios. Tasks which require higher degree of safety and/or are more likely to involve small number of users might find the user specific strategy more appropriate. E.g. medical scenarios. On the other hand, scenarios of industrial collaboration, e.g. skill transfer, that don’t involve high degrees of safety might find the non-user specific strategy more favorable.

References

[1] M. A. Goodrich and A. C. Schultz, "Human-Robot Interaction: A Survey," *Foundations and Trends in Human-Computer Interaction,* pp. 23-275, 2007.

[2] S. J. Hu, J. Ko, L. Weyand, H. A. ElMaraghy, T. K. Lien, Y. Koren, H. Bley, G. Chryssolouris, N. Nasr and M. Shpitalni, "Assembly system design and operations for product variety," *CIRP Annals - Manufacturing Technology,* no. 60, p. 715–733, 2011.

[3] Y. Shen, J. Zastrow, G. Graf and Reinhart, "An uncertainty-based evaluation approach for human-robot cooperation within production systems," *Procedia CIRP,* no. 41, pp. 376-381, 2016.

[4] Cherubini, Passama, Crosnier, Lasnier and Fraisse, "Collaborative manufacturing with physical human-robot interaction," *Robot. Comput. Integr. Manuf.,* no. 40, pp. 1-13, 2016.

[5] Krüger, Lien and Verl, "Cooperation of human and machines in assembly lines," *Ann. Manuf. Technol CIRP,* no. 58, pp. 628-646, 2009.

[6] M. Bélanger-Barrette, "Robotiq Blog," Robotiq, 19 08 2015. [Online]. Available: https://blog.robotiq.com/what-does-collaborative-robot-mean. [Accessed 10 11 2018].

[7] D. Song, N. Kyriazis, I. Oikonomidis, C. Papazov, A. Argyros, D. Burschka and D. Kragic, "Predicting Human Intention in Visual Observations of Hand/Object Interactions," in *IEEE International Conference on Robotics and Automation (ICRA)*, Karlsruhe, 2013.

[8] P. Gomes, "Surgical robotics: Reviewing the past,analysing the present,imagining the future," *Robotics and Computer-Integrated Manufacturing,* no. 27, pp. 261-266, 2011.

[9] "Hackensack Meridian Health Raritan Bay Medical Center," Hackensack Meridian Health Raritan Bay Medical Center, [Online]. Available: https://www.rbmc.org/da-vinci-robotic-technology/da-vinci-xi-edit/. [Accessed 11 11 2018].

[10] "Hansen Medical," Hansen medical, 2016. [Online]. Available: http://hansenmedical.com/us/en/node/156. [Accessed 11 11 2018].

[11] D. Muoio, "Business Insider," Insider, 30 11 2016. [Online]. Available: https://www.businessinsider.com/exoskeletons-improve-strength-codebreaker-2016-11?IR=T. [Accessed 13 10 2018].

[12] F. Tobe, "The robot report," The robot report, 13 07 2014. [Online]. Available: https://www.therobotreport.com/3-exoskeleton-companies-go-public/. [Accessed 13 11 2018].

[13] P. Rocco, "Slides on Control of Industrial Robots - Robot Dynamics," Politecnico Di Milano - Dipartmento di Elettronica, Informazione e Bioingeneria, Milano, 2018.

[14] M. Jakopec, R. y. F. Baena, S. J. Harris, P. Gomes, J. Cobb and B. L. Davies, "The Hands-On Orthopaedic Robot “Acrobot”: Early Clinical Trials of Total Knee Replacement Surgery," *IEEE TRANSACTIONS ON ROBOTICS AND AUTOMATION,* vol. 19, no. 5, pp. 902-911, 2003.

[15] C. F. N. T. A. B. Arash Ajoudani, "Reduced-complexity representation of the human arm active endpoint stiffness for supervisory control of remote manipulation," *The International Journal of Robotics Research,* vol. 37, no. 1, pp. 155-167, 2018.

[16] N. H. A. E. B. F. A. Mussa-Ivaldi, "Neural, Mechanical, and Geometric Factors Subserving Arm Posture in Humans," *The Journal of Neuroscience,* vol. 5, no. 10, pp. 2732-2743, 1985.

[17] V. Duchaine and C. Gosselin, "Safe, Stable and Intuitive Control for Physical Human-Robot Interaction," in *IEEE International Conference on Robotics and Automation*, Kobe, Japan, 2009.

[18] R. D. Trumbower, M. A. Krutky and B. S. Yang, "Use of self-selected postures to regulate multi-joint stiffness during unconstrained tasks," *PLoS ONE,* vol. 4, no. 5, p. e5441, 2009.

[19] X. Hu, W. M. Murray and E. J. Perreault, "Muscle short-range stiffness can be used to estimate the endpoint stiffness of the human arm," *Journal of Neurophysiology ,* no. 105, pp. 1633-1641, 2011.

[20] M. Rahman, R. Ikeura and K. Mizutani, "Investigating the impedance characteristic of human arm for development of robots to co-operate with human operators," in *International Conference on Systems, Man, and Cybernetics*, 1993.

[21] R. F. K. P. E. C. Eric J. Perreault, "Voluntary Control of Static Endpoint Stiffness During Force Regulation Tasks," *Neurophysiol,* no. 87, p. 2808–2816, 2002.

[22] M. L. C. N. e. a. Gribble PL, "Role of cocontraction in arm movement accuracy," *Journal of Neurophysiology,* vol. 5, no. 89, pp. 2396-2405, 2003.

[23] M. T. a. S. R. Akazawa K, "Modulation of reflex EMG and stiffness in response to stretch of human finger muscle," *Journal of Neurophysiology,* vol. 1, no. 49, pp. 16-27, 1983.

[24] A. D. L. J. e. a. Buchanan TS, "Characteristics of synergic relations during isometric contractions of human elbow muscles," *Journal of Neurophysiology,* vol. 56, no. 5, pp. 1225-1241, 1986.

[25] I. M. a. A. P, "Proportional myoelectric control of robots: Muscle synergy development drives performance enhancement, retainment, and generalization.," *IEEE Transactions on Robotics,* vol. 31, no. 2, pp. 259-268, 2014.

[26] I. M. a. A. P, "The role of muscle synergies in myoelectric control: Trends and challenges for simultaneous multifunction control," *Journal of Neural Engineering,* vol. 11, no. 5, 2014.

[27] Medical Dictionary for the Health Professions and Nursing, 2012.

[28] M. H. F. M.-Y. M.B.I. Raez, "Techniques of EMG signal analysis: detection, processing, classification and applications," *Biol Proced Online,* no. 8, pp. 11-35, 2006.

[29] B. C. J. L. B. T. R. W. S. H. N. C. D. L. Shey-Sheen Chang, "BU College of Engineering Electrical & Computer Engineering," Boston University, [Online]. Available: http://www.bu.edu/eng/departments/ece/research/spotlight/research-spotlight/research-spotlight-archive-decomposition-of-highly-unpredictable-real-life-emg-signals/. [Accessed 19 11 2018].

[30] L. S. J. M. Winters, "Muscle Models: What is Gained and What is Lost by Varying Model Complexity," *Biological Cybernetic,* vol. 55, no. 6, pp. 403-420, 1987.

[31] J. V. N. M. Kosta Jovanović, "Hill's and Huxley's Muscle Models – Tools for Simulations in Biomechanics," *Serbian Journal Of Electrical Enginerring,* vol. 12, no. 1, pp. 53-67, 2015.

[32] L. M. a. Z. VM, "Joint stiffness: Myth or reality?," *Human Movement Science,* vol. 12, no. 6, pp. 653-692, 1993.

[33] K. R. Holzbaur, W. M. Murray and S. L. Delp, "A Model of the Upper Extremity for Simulating Musculoskeletal Surgery and Analyzing Neuromuscular Control," *Annals of Biomedical Engineering,* vol. 33, no. 6, pp. 829-840, 2005.

[34] D. W. F. H. K. H. G. K. D. T. Y. M. K. R. Osu, "Short- and long-term changes in joint co-contraction associated with motor learning as revealed from surface EMG," *Neurophysiol,* vol. 88, pp. 991-1004, 2002.

[35] M. a. M. M. Hagan, "Training feed-forward networks with the Marquardt algorithm," *IEEE Transactions on Neural Networks,* vol. 5, no. 6, pp. 989-993, 1999.

[36] "MathWorks Support," MathWorks, [Online]. Available: https://www.mathworks.com/help/deeplearning/ref/initnw.html. [Accessed 22 11 2018].

[37] A. S. a. S. L. D. Michael A. Sherman, "What is a moment arm? Calculating muscle effectiveness n biomechanical models using generalized coordinates," in *Proc ASME Des Eng Tech Conference*, 2015.

[38] N. Nikolakis, V. Maratos and S. Makris, "A cyber physical system (CPS) approach for safe human-robot collaboration in a shared workplace," *Robotics and Computer Integrated Manufacturing,* p. 11, 2018.

[39] P. Neto and M. Safeea, "KUKA Sunrise Toolbox: Interfacing Collaborative Robots with MATLAB".

[40] V. Duchaine and C. M. Gosselin, "General Model of Human-Robot Cooperation Using a Novel Velocity Based Variable Impedance Control," in *Second Joint EuroHaptics Conference and Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems*, 2007.

[41] a. H. G. R. Osu, "“Multijoint muscle regulation mechanisms examined by measured human arm stiffness and EMG," *Neurophysiol,* vol. 81, pp. 1458-1468, 1990.